

Analytics および ACLScript 16.x へ ヘルプ



目次

Analytics の使用	24
Analytics の使用	25
はじめに	26
はじめに	27
Analytics とは	28
ACL for Windows の概要	32
Analytics の基本操作 (非 Unicode 版)	37
Analytics(Unicode 版) の基本	71
Analytics のヘルプ	105
Analytics のユーザー インターフェイス	108
Analytics のユーザー インターフェイス	109
Analytics のユーザー インターフェイスの概要	110
Analytics テーブルの構造	118
Analytics のカスタマイズ	120
Analytics のオプションの設定	121
システム オプション	122
インターフェイス オプション	123
テーブル オプション	125
ビュー オプション	130
コマンド オプション	132
日付と時刻オプション	135
数値オプション	143
印刷オプション	146
アプリケーション フォント オプション	147
Analytics の初期設定ファイルの動作	148
ビューおよびレポートのフォント設定の変更	152
ビューのフォント サイズを変更する	153
Analytics ツールバーのカスタマイズ	154

Analytics のメインメニューへのカスタム項目の追加	155
Analytics コマンド ラインからのコマンドの実行	161
表示領域の情報の印刷	163
Analytics から電子メール通知を送信する	164
Analytics プロジェクト	169
Analytics プロジェクトを使った作業	173
Analytics コマンド ログの使用法	179
Analytics プロジェクトでのノートの使用法	183
Analytics プロジェクトの情報の印刷	190
予期せずに閉じられた Analytics プロジェクトの回復	192
一般的なデータ準備と分析タスク	194
一般的なデータ準備と分析タスク	195
結果の保存と出力フォルダーの指定	196
Analytics プロジェクト フォルダーと Windows フォルダーを一致させる	200
データの抽出	201
既存テーブルへの出力結果の追加	207
データ構造の比較	209
データのエクスポート	210
HighBond のリザルト アプリへの例外のエクスポート	216
HighBond のロボット アプリへのデータのエクスポート	224
キー フィールドについて	231
フィールドの連結	233
乱数の生成	234
ランダムに選択したレコードの生成	236
データの定義とインポート	238
データの定義とインポート	239
Analytics でアクセスできるデータソース	242
データ定義 ウィザードを使用したデータの定義とインポート	248
Microsoft Excel データのインポート	250
Microsoft Access データベース ファイルのインポート	263
区切り文字付きテキスト ファイルのインポート	265

印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート	279
クイックスタート:印刷イメージまたはPDFファイルを定義する方法	295
印刷イメージファイルの定義とインポート	300
PDFファイルの定義とインポート	308
フィールド定義での作業	317
レコード定義での作業	322
印刷イメージまたはPDFファイルで整列していないフィールドの定義	326
印刷イメージまたはPDFデータのサブセットの定義とインポート	330
複数行のレコードおよびフィールドの操作	332
ACCPAC マスターファイルのインポート	338
dBASE 互換ファイルのインポート	340
SAP 監査形式ファイルのインポート	342
XML ファイルのインポート	344
XML データ構造の選択	346
XML 要素の選択および構成	347
XML 列プロパティの修正	349
XML ファイルについて	351
XBRL ファイルのインポート	353
XBRL 要素の選択	355
XBRL コンテキストの選択	356
XBRL ファイルについて	357
Analytics サーバーのデータベースプロファイルデータの定義	358
外部定義ファイルの定義	360
Analytics テーブルの手動による定義	362
ソースデータに含まれる日付および時刻データの書式	366
データアクセス ウィンドウを使用したデータのインポート	372
データアクセス ウィンドウの作業	376
データアクセス ウィンドウのテーブルの結合	388
Active Directory への接続	392
ADP への接続	401
Airtable への接続	403

Amazon Athena への接続	405
Amazon DynamoDB への接続	407
Amazon Redshift への接続	411
Amazon S3 への接続	415
Apache Cassandra への接続	417
Apache Drill への接続	424
Apache HBase への接続	427
Apache Hive への接続	429
Apache Phoenix への接続	434
Apache Spark への接続	436
AWS Data Management への接続	441
Azure Analysis Services への接続	443
Azure Data Catalog への接続	445
Azure Data Lake Storage への接続	447
Azure Data Management への接続	449
Azure Table Storage への接続	451
Basecamp への接続	453
Box への接続	455
Cloudera Impala への接続	457
CockroachDB への接続	461
Concur への接続	463
SAP Concur への接続	468
Couchbase への接続	470
DigitalOcean への接続	477
DocuSign への接続	479
Dropbox への接続	481
Dynamics CRM への接続	483
Dynamics GP への接続	485
Dynamics NAV への接続	487
Dynamics 365 Business Central への接続	489
Dynamics 365 Finance and Operations への接続	491

Dynamics 365 Sales への接続	493
Edgar Online への接続	495
Elasticsearch への接続	497
電子メールへの接続	499
Epicor ERP への接続	508
Exact Online への接続	510
Excel Online への接続	512
Exchange への接続	514
Google Analytics への接続	524
Google BigQuery への接続	526
Google Cloud Storage への接続	530
Google Contacts への接続	532
Google Drive への接続	534
Google Sheets への接続	536
IBM Cloudant への接続	538
Jira への接続	540
JSON サービスへの接続	548
Kintone への接続	555
LDAP への接続	557
LinkedIn への接続	566
Marketo への接続	568
MarkLogic への接続	574
Microsoft Access への接続	576
Microsoft OneDrive への接続	578
Microsoft SQL Server への接続	580
Microsoft Teams への接続	583
MongoDB への接続	585
MySQL への接続	591
NetSuite への接続	594
OData への接続	596
Odoo への接続	598

Open Exchange Rates への接続	600
Oracle への接続	606
Oracle Eloqua への接続	609
Oracle HCM Cloud への接続	611
Oracle Sales Cloud への接続	613
Parquet への接続	615
Presto への接続	617
Qualys への接続	622
QuickBooks への接続	626
QuickBooks Online への接続	628
QuickBooks POS への接続	630
REST サービスへの接続	632
Rsam に接続する	644
RSS/ATOM への接続	648
Sage 50 UK への接続	652
Sage Cloud Accounting への接続	654
Sage Intacct への接続	656
Salesforce への接続	658
SAP への接続	662
ACLコネクタ(SAP) の設定	672
SAP ByDesign への接続	684
SAP Hybris Cloud for Customer への接続	686
SAP SuccessFactors への接続	688
ServiceNow に接続する	690
SFTP への接続	697
SharePoint への接続	700
ShipStation への接続	707
Slack への接続	709
Snowflake への接続	711
Splunk への接続	713
Square への接続	715

Stripe への接続	717
SugarCRM への接続	719
SuiteCRM への接続	721
SurveyMonkey への接続	723
Sybase への接続	725
Sybase IQ への接続	727
Tenable.sc への接続	729
Teradata への接続	734
Twitter への接続	740
UPS への接続	749
USPS への接続	751
Workday への接続	753
xBase への接続	755
ZenDesk への接続	757
HighBond プロジェクト データのインポート	759
HighBond リザルト データのインポート	764
テーブルレイアウトを使用したデータの構造化	768
テーブルレイアウトを使った作業	772
テーブルレイアウトのプロパティの設定	779
テーブルレイアウトのプロパティの表示	781
Analytics テーブルのデータの更新	782
Analytics テーブルのデータソースの変更	784
テーブルレイアウトのフィールドの定義	786
物理フィールドの定義	788
演算フィールドの定義	795
Analytics のデータ型	811
Custom データ型	819
テーブルレイアウトのフィールドの編集	821
テーブルレイアウトのフィールドの名前を変更する	822
テーブルレイアウトからのフィールドの削除	823
テーブルレイアウトにおけるフィールドのシフト	824

データのダンプ	827
テーブルの履歴の表示	829
ワークスペースを使用したフィールド定義の共有	830
データフィルターについて	835
テーブルビューでデータを表示する	838
ビューでの作業	841
ビュー内の列のカスタマイズ	849
ビューからのデータのコピー	857
ビューからのグラフの生成	858
ビューでの無効なデータの表示方法	859
複数のテーブルを開く	860
複数行にわたるレコードの書式設定	863
分析用のデータの準備	864
分析用のデータの準備	865
式の使用	866
式ビルダーの概要	870
式ビルダーによる式の作成	872
数式の端数処理と小数点精度の制御	874
財務関数での丸めの制御	880
数式でのオーバーフローエラーの防止	881
式の使用時の2つの一般的なエラー	882
式での日付時刻の使用	884
シリアル日付時刻	896
UTC オフセットが時刻日付式に与える影響	899
監査データの検証	901
データの検証	902
レコードのカウント	906
フィールドの合計	908
データの結合	910
データの結合の代替方法	917
データ構造およびデータ書式の要件	918

フィールドの一致	921
データ結合方法の比較	924
テーブルの追加	928
テーブルを追加する	938
データの抽出と追加	940
データを抽出して追加する	945
演算フィールドの抽出と追加	949
テーブルのマージ	951
テーブルのマージ	955
結合または関連付けの一般的な用途	959
テーブルの結合	960
テーブルの結合	970
結合タイプの例	976
曖昧結合	985
テーブル結合時の自動一致	997
テーブルの関連付け	999
テーブルの関連付け	1004
関係の編集	1008
テーブルの関連付けの構築方法	1009
複数のキーフィールドの使用	1013
キーフィールドの連結	1020
データのサンプリング	1021
サンプルの選択方法	1025
監査サンプリング用語	1030
レコード サンプリング(属性サンプリング)	1035
レコード サンプリングのチュートリアル	1037
レコード サンプルのサンプル サイズを計算	1045
レコード サンプリングの実行	1051
レコード サンプルの誤謬を評価する	1056
金額単位サンプリング	1060
金額単位サンプリングのチュートリアル	1063

金額単位サンプルのサンプルサイズを計算	1071
金額単位サンプリングの実行	1078
金額単位サンプルの誤謬を評価する	1087
従来の変数サンプリング	1093
従来の変数サンプリングのチュートリアル	1101
従来の変数サンプリングの準備	1121
従来の変数サンプリングの実行	1131
従来の変数サンプリングの誤謬の評価	1139
条件付きサンプリング	1148
データの分析	1150
データの分析	1151
データのプロファイル分析	1153
統計の生成	1155
異常値の特定	1159
並べ替え、フィルタリング、検索	1169
ビューのデータでクイックソートを使用する	1170
ビューのデータでクイックフィルターを使用する	1171
テーブルのクイック検索	1176
ソートとインデックス	1183
レコードの並べ替え	1188
レコードのインデックス付け	1196
演算キーフィールドを使った並べ替えとインデックスの作成	1204
データのフィルターリング	1209
グローバルフィルター(ビューフィルター)	1213
ビューへのグローバルフィルターの適用	1218
ローカルフィルター(コマンドフィルター)	1222
データの検索	1224
一致する最初のレコードの選択	1229
Analytics関数を使って検索とフィルタリングを行う	1235
順番検査	1249
ギャップの検査	1256

重複の検査	1265
あいまい重複分析	1276
あいまい重複の検査	1280
あいまい重複のヘルパー関数	1284
あいまい重複の出力結果を使った作業	1287
あいまい重複の結果のサイズ制御	1290
設定による動作の違い	1293
あいまい重複をグループ化する方法	1297
データのグループ化	1302
データの階層化	1304
データの年齢調べ	1312
分類化と要約の比較	1318
データの分類化	1319
データの要約	1326
データのクロス集計	1337
ヒストグラムを作成	1343
機械学習分析	1348
クラスと数値の予測	1350
データのクラスター	1364
ベンフォード分析の実行	1373
R スクリプトの実行	1379
調査結果のレポート作成	1384
調査結果のレポート作成	1385
Analytics レポートの書式設定および生成	1387
Analytics グラフを使った作業	1393
グラフの書式設定の変更	1394
グラフの元データへのドリルダウン	1400
グラフコマンドの編集	1401
グラフをクリップボードへコピー	1402
グラフをイメージとして保存	1403
グラフの印刷	1404

サードパーティレポート アプリケーションからの Analytics への接続	1405
参照情報	1412
参照情報	1413
Analytics における文字およびサイズの制限	1414
予約キーワード	1421
キーボード ショートカット	1423
Analytics のスクリプト	1426
Analytics のスクリプト	1427
スクリプトの基本	1430
スクリプトの基本	1431
初心者向けのスクリプト	1432
スクリプトとは何か	1433
あなたにとって最初の Analytics スクリプト	1437
テキスト データの比較	1440
空のデータ値のフィルタリング	1443
スクリプトでの決定	1446
Analytics でのスクリプト作成の基本事項	1450
コメント	1455
データ型	1457
式	1458
式を使用した演算フィールドの定義	1460
関数	1462
変数	1464
統制構造	1467
グループ化とループ処理	1470
関数を使用する方法	1477
関数とは何か	1479
さまざまな関数の詳細	1483
関数を使用してフィルターを作成する	1487
関数を使用してデータをクリーニングする	1491
データを同時にクリーニングしてフィルタリングする	1495

高度な関数の使用	1499
関数を使用して月別にレコードをグループ化する	1500
関数で変数を使用してユーザー入力を許可する	1507
すべてまとめる: スクリプトで関数を使用する	1512
最もよく使用される 30 個の Analytics 関数	1517
スクリプトの操作	1532
スクリプトの操作	1533
スクリプトの作成と編集	1536
スクリプトのテストおよびデバッグ	1543
スクリプトの実行	1549
ACLScript での変数の操作	1554
Analytics コマンドによって作成されたシステム変数	1582
インタラクティブ スクリプトの作成	1586
カスタム ダイアログ ボックスの作成	1588
スクリプト エディターのカスタマイズ	1602
スクリプトのコピー	1604
スクリプトのインポート	1605
ScriptHub からのインポート	1606
スクリプトのエクスポート	1608
テキストの検索と置換	1609
コマンド	1610
コマンドの概要	1611
ACCEPT コマンド	1625
ACCESSDATA コマンド	1631
ACTIVATE コマンド	1644
AGE コマンド	1646
APPEND コマンド	1651
ASSIGN コマンド	1660
BENFORD コマンド	1663
CALCULATE コマンド	1667
CLASSIFY コマンド	1670

CLOSE コマンド	1676
CLUSTER コマンド	1679
COMMENT コマンド	1682
COUNT コマンド	1684
CREATE LAYOUT コマンド	1687
CROSSTAB コマンド	1689
CVSEVALUATE コマンド	1694
CVSPREPARE コマンド	1698
CVSSAMPLE コマンド	1703
DEFINE COLUMN コマンド	1706
DEFINE FIELD コマンド	1709
DEFINE FIELD . . . COMPUTED コマンド	1716
DEFINE RELATION コマンド	1722
DEFINE REPORT コマンド	1725
DEFINE TABLE DB コマンド	1726
DEFINE VIEW コマンド	1730
DELETE コマンド	1732
DIALOG コマンド	1736
DIRECTORY コマンド	1744
DISPLAY コマンド	1750
DO REPORT コマンド	1755
DO SCRIPT コマンド	1757
DUMP コマンド	1760
DUPLICATES コマンド	1762
ESCAPE コマンド	1769
EVALUATE コマンド	1771
EXECUTE コマンド	1776
EXPORT コマンド	1784
EXTRACT コマンド	1801
FIELDSHIFT コマンド	1807
FIND コマンド	1809

FUZZYDUP コマンド	1811
FUZZYJOIN コマンド	1817
GAPS コマンド	1825
GROUP コマンド	1829
HB_API_DELETE コマンド	1836
HB_API_GET コマンド	1840
HB_API_PATCH コマンド	1846
HB_API_POST コマンド	1851
HB_API_PUT コマンド	1857
HELP コマンド	1862
HISTOGRAM コマンド	1863
IF コマンド	1868
IMPORT ACCESS コマンド	1870
IMPORT DELIMITED コマンド	1873
IMPORT EXCEL コマンド	1882
IMPORT GRCPROJECT コマンド	1892
IMPORT GRCRESULTS コマンド	1900
IMPORT LAYOUT コマンド	1910
IMPORT MULTIDELIMITED コマンド	1912
IMPORT MULTIEXCEL コマンド	1921
IMPORT ODBC コマンド	1929
IMPORT PDF コマンド	1933
IMPORT PRINT コマンド	1942
IMPORT SAP コマンド	1950
IMPORT XBRL コマンド	1957
IMPORT XML コマンド	1962
INDEX コマンド	1967
JOIN コマンド	1971
LIST コマンド	1979
LOCATE コマンド	1982
LOOP コマンド	1986

MERGE コマンド	1989
NOTES コマンド	1994
NOTIFY コマンド	1996
OPEN コマンド	2003
OUTLIERS コマンド	2006
PASSWORD コマンド	2015
PAUSE コマンド	2018
PREDICT コマンド	2020
PRINT コマンド	2023
PROFILE コマンド	2025
QUIT コマンド	2027
RANDOM コマンド	2029
RCOMMAND コマンド	2032
REFRESH コマンド	2040
RENAME コマンド	2044
REPORT コマンド	2046
RETRIEVE コマンド	2050
SAMPLE コマンド	2052
SAVE コマンド	2061
SAVE LAYOUT コマンド	2063
SAVE LOG コマンド	2068
SAVE TABLELIST コマンド	2070
SAVE WORKSPACE コマンド	2072
SEEK コマンド	2074
SEQUENCE コマンド	2077
SET コマンド	2081
SIZE コマンド	2097
SORT コマンド	2102
STATISTICS コマンド	2109
STRATIFY コマンド	2113
SUMMARIZE コマンド	2119

TOP コマンド	2127
TOTAL コマンド	2128
TRAIN コマンド	2130
VERIFY コマンド	2135
関数	2138
関数の概要	2139
ABS() 関数	2156
AGE() 関数	2157
ALLTRIM() 関数	2163
ASCII() 関数	2165
AT() 関数	2167
BETWEEN() 関数	2170
BINTOSTR() 関数	2179
BIT() 関数	2181
BLANKS() 関数	2183
BYTE() 関数	2185
CDOW() 関数	2187
CHR() 関数	2191
CLEAN() 関数	2193
CMOY() 関数	2195
COS() 関数	2198
CTOD() 関数	2200
CTODT() 関数	2205
CTOT() 関数	2210
CUMIPMT() 関数	2215
CUMPRINC() 関数	2217
DATE() 関数	2219
DATETIME() 関数	2223
DAY() 関数	2228
DBYTE() 関数	2231
DEC() 関数	2233

DHEX() 関数	2236
DICECOEFFICIENT() 関数	2238
DIGIT() 関数	2245
DOW() 関数	2247
DTOU() 関数	2250
EBCDIC() 関数	2253
EFFECTIVE() 関数	2255
EOMONTH() 関数	2257
EXCLUDE() 関数	2260
EXP() 関数	2263
FILESIZE() 関数	2265
FIND() 関数	2267
FINDMULTI() 関数	2272
FREQUENCY() 関数	2277
FTYPE() 関数	2279
FVANNUITY() 関数	2281
FVLUMPSUM() 関数	2285
FVSCHEDULE() 関数	2288
GETOPTIONS() 関数	2290
GOMONTH() 関数	2292
HASH() 関数	2295
HEX() 関数	2301
HOUR() 関数	2303
HTOU() 関数	2306
INCLUDE() 関数	2308
INSERT() 関数	2310
INT() 関数	2312
IPMT() 関数	2313
ISBLANK() 関数	2315
ISDEFINED() 関数	2317
ISFUZZYDUP() 関数	2319

LAST() 関数	2325
LEADING() 関数	2327
LEADINGZEROS() 関数	2329
LENGTH() 関数	2333
LEVDIST() 関数	2335
LOG() 関数	2339
LOWER() 関数	2341
LTRIM() 関数	2343
MAP() 関数	2345
MASK() 関数	2350
MATCH() 関数	2352
MAXIMUM() 関数	2360
MINIMUM() 関数	2363
MINUTE() 関数	2367
MOD() 関数	2370
MONTH() 関数	2372
NOMINAL() 関数	2375
NORMDIST() 関数	2377
NORMSINV() 関数	2379
NOW() 関数	2380
NPER() 関数	2382
OCCURS() 関数	2385
OFFSET() 関数	2388
OMIT() 関数	2390
PACKED() 関数	2394
PI() 関数	2397
PMT() 関数	2399
PPMT() 関数	2402
PROPER() 関数	2404
PROPERTIES() 関数	2406
PVANNUITY() 関数	2410

PVLUMPSUM() 関数	2414
PYDATE() 関数	2417
PYDATETIME() 関数	2419
PYLOGICAL() 関数	2422
PYNUMERIC() 関数	2424
PYSTRING() 関数	2426
PYTIME() 関数	2429
RAND() 関数	2431
RATE() 関数	2433
RDATE() 関数	2436
RDATETIME() 関数	2439
RECLLEN() 関数	2442
RECNO() 関数	2443
RECOFFSET() 関数	2445
REGEXFIND() 関数	2447
REGEXREPLACE() 関数	2455
REMOVE() 関数	2464
REPEAT() 関数	2467
REPLACE() 関数	2469
REVERSE() 関数	2473
RJUSTIFY() 関数	2474
RLOGICAL() 関数	2475
RNUMERIC() 関数	2478
ROOT() 関数	2481
ROUND() 関数	2483
RSTRING() 関数	2485
RTIME() 関数	2488
SECOND() 関数	2491
SHIFT() 関数	2494
SIN() 関数	2496
SORTWORDS() 関数	2498

SOUNDEX() 関数	2503
SOUNDSLIKE() 関数	2507
SPLIT() 関数	2510
STOD() 関数	2514
STODT() 関数	2518
STOT() 関数	2523
STRING() 関数	2527
SUBSTR() 関数	2531
TAN() 関数	2535
TEST() 関数	2537
TIME() 関数	2539
TODAY() 関数	2545
TRANSFORM() 関数	2547
TRIM() 関数	2549
UNSIGNED() 関数	2551
UPPER() 関数	2553
UTOD() 関数	2555
VALUE() 関数	2559
VERIFY() 関数	2562
WORKDAY() 関数	2564
YEAR() 関数	2568
ZONED() 関数	2571
ZSTAT() 関数	2575
アナリティクス スクリプト	2578
アナリティクス スクリプト の概要	2579
アナリティクス スクリプト の開発	2581
アナリティクス ヘッダーを使った作業	2588
アナリティクス開発のベスト プラクティス	2596
アナリティクス スクリプトのサンプル	2601
アナリティクス ヘッダーおよびアナリティクス タグ	2605
ANALYTIC タグ	2609

FILE タグ	2613
PARAM タグ	2616
PASSWORD タグ	2626
TABLE タグ	2629
FIELD タグ	2631
RESULT タグ	2634
DATA タグ	2641
アナリティクス スクリプトの Unicode への変換	2646
Unicode 互換性チェック	2649
スクリプト エンジンエラーコード	2651
ACL for Windows インストールおよびアクティベーション ガイド	2660
ACL for Windows インストールおよびアクティベーション ガイド	2661
ACL for Windows インストールおよびアクティブ化の概要	2666
Diligent Unicode 製品	2672
ACL for Windows をインストールする	2679
サイレント インストールによる ACL for Windows のインストール	2687
ACL for Windows のアンインストール	2697
Python の Analytics 連携用設定	2697
インストールのトラブルシューティングとアクティブ化	2700
プロキシ サーバーを介した HighBond への接続	2705
ACL for Windows のシステム要件	2707
接続要件	2715
重大なセキュリティ脆弱性 (CVE-2021-44832、-45105、-45046、-44228) のパッチ	2716
自動化と共有	2720
自動化と共有	2721
データをリザルトに公開する	2723
データをストーリーボードに公開する	2724
ロボットを使用した自動化	2725
Analytics およびロボットにおけるスクリプトの開発ワークフロー	2726
Analytics からロボットへのスクリプトのコミット (アップロード)	2731
ACL ロボットのテーブル、ファイル、ログの表示	2736

Analytics の使用

Analytics の使用

Analytics は、データを操作するためのさまざまなツールを提供します。Analytics には、データのインポートをはじめとして、データ分析サイクルを進めるためのさまざまなオプションがあります。

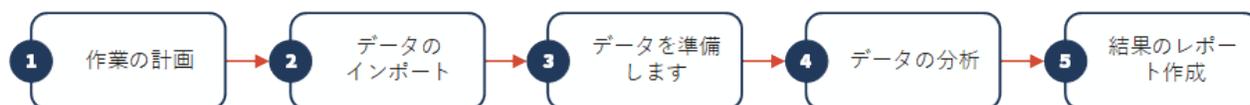
Analytics は、特定のデータ分析ワークフローを適用したり、必要とすることはありません。Analytics のコマンド、関数、および他のツールは、ワークフローの状態に合わせて組み立て、特定のデータセットを分析し、分析目標を達成することができます。

つまり、一般的なデータ分析サイクルを理解することで、Analytics での作業の階層化することができます。

データ分析のサイクル

データ分析には、5つの段階があります。これは略語の PIPAR で表されます。

Plan(計画)、Import(インポート)、Prepare(準備)、Analyze(分析)、Report(レポート)



ヒント

Analytics でのデータ分析サイクルの仕組みについては、導入チュートリアルを実行してください: "Analytics の基本操作(非 Unicode 版)" ページ 37

"作業の計画" ページ 38	データ分析作業を計画することは、Analytics で実際に分析を開始する前の前提として重要です。 必ず"作業の計画" ページ 38を確認してください。
データのインポート	分析する前に、データを Analytics にインポートする必要があります。
データを準備します	多くの場合、データの分析準備が整う前に、1つ以上のデータ準備タスクを実行する必要があります。
データの分析	コマンドおよび他のツールを使用して、Analytics で分析を実行し、調査しているデータに関する一般的な洞察を得て、特定の問題に対して回答します。
結果のレポート作成	データ分析が完了すると、Analytics は、複数の方法で、結果を報告または表示します。

はじめに

Analytics ヘルプのこのセクションでは、次のような、さまざまな導入および概要情報について説明します。

"Analytics とは" 見開きページ	Analytics 機能の概要と、Analytics を使用してデータを分析する際のエンドツーエンドプロセス。
"ACL for Windows の概要" ページ 32	ACL for Windows インストールパッケージと ACL for Windows のメイン画面。
"Analytics の基本操作 (非 Unicode 版)" ページ 37	この基本レベルの 1 時間のチュートリアルでは、Analytics を使用して、データを分析するエンドツーエンドのプロセスについて概要を説明します。 新しいすべての Analytics ユーザー向けです。
"Analytics のヘルプ" ページ 105	Analytics 使用時のヘルプ情報の参照箇所
"Analytics のユーザー インターフェイス" ページ 109	Analytics インターフェイスの概要。カスタマイズ可能な要素などが記載されています。
"Analytics プロジェクト" ページ 169	Analytics プロジェクトに関する情報。Analytics に作業を保存して整理するための情報です。

Analytics とは

Analytics は、データアクセス、データ分析、および統合レポートの強力な組み合わせを提供し、データの完全性を保証するデータ分析アプリケーションです。Analytics インターフェイスを使ってアドホックの分析を実行したり、統合 ACL スクリプティング言語でローカルで、またはクラウドで実行する分析を自動化したりします。

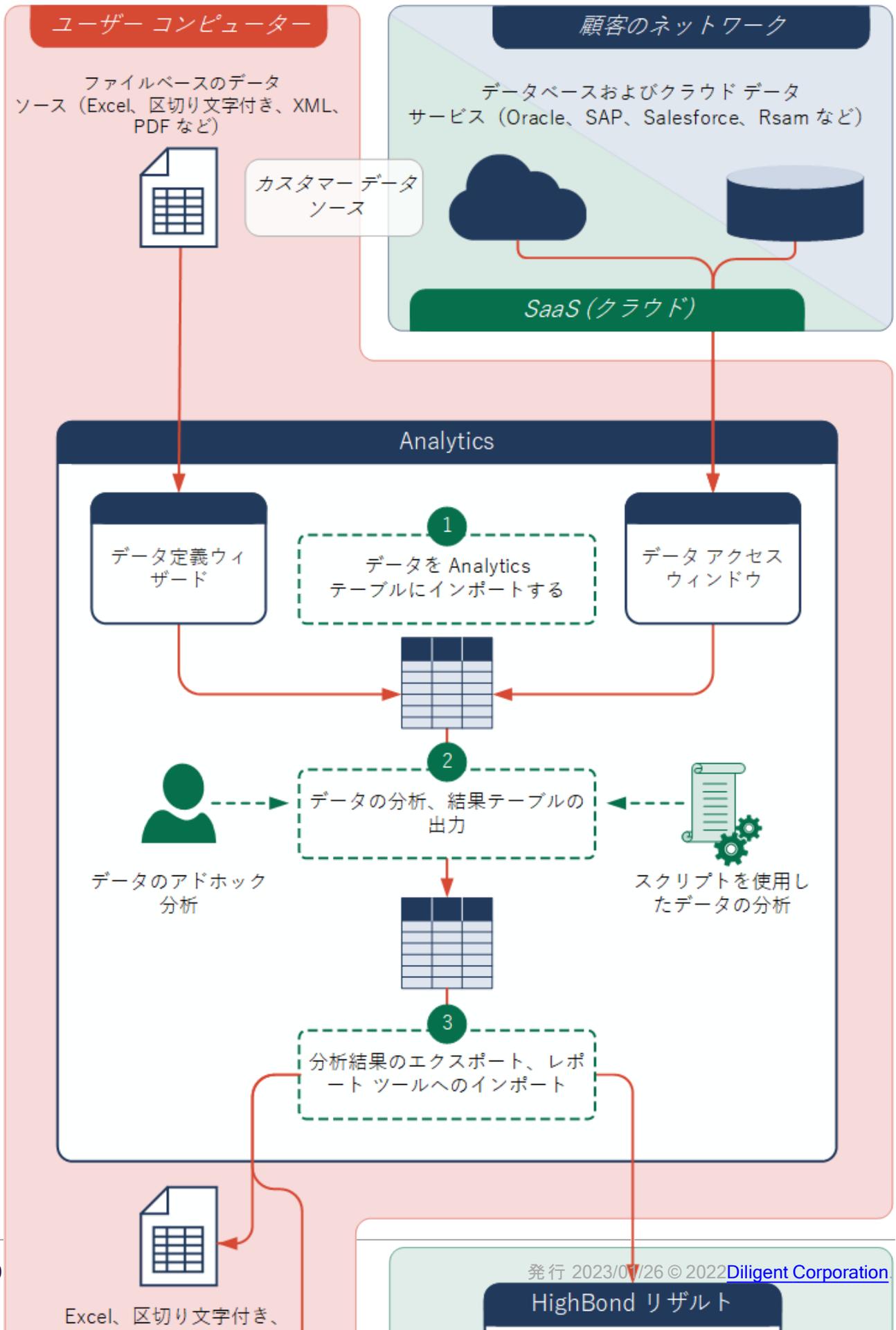
- **データアクセス** -ファイルベースのデータソース、データベース、またはクラウド データ サービスから、さまざまなタイプのデータをインポートします。
- **データ分析** -Analytics コマンド、関数、他のツールを使用して、調査しているデータに関する一般的な洞察を得て、特定の問題に対して回答します。ユーザー インターフェイスでアドホック データ分析を実行したり、Analytics の強力なスクリプト言語である ACLScript を使用して分析を自動化したりすることができます。
- **レポート** -Analytics のネイティブ レポート 機能を使用して調査結果を報告したり、Tableau などのサードパーティレポート ツールを使用して Analytics データをインポートしたりすることができます。
- **エクスポート機能** -調査結果または他のデータをエクスポートして、Excel または区切り文字付きテキストなどのファイルタイプを入力します。HighBond プラットフォームのリザルト アプリにレコードをアップロードして、ワークフロー自動化ツールを使った問題改善と処理、およびデータの視覚化を行うこともできます。

基本ワークフロー

以下の図は Analytics に関連付けられた基本ワークフローを示します。

1. データを Analytics テーブルにインポートする
2. データの分析、結果テーブルの出力
3. 分析結果のエクスポート、レポートツールへのインポート

3 番目の手順は任意です。Analytics でネイティブ レポート 機能を使用することもできます。



データによる作業

Analytics を使用すれば、組織にとって重要な取引データを即座に見ることができるようになります。データの読み込みや比較を行いますが、ソースデータが変更されることはなく、データの完全な品質と整合性が保持されることを保証します。

詳細については、「Analytics によるデータ アクセスは読み取り専用です」ページ 243を参照してください。

Analytics を使用すると、次のような方法によるデータの操作が可能になります。

- データ母集団全体、または母集団のサンプルを分析する
- 傾向と例外を特定し、考慮の対象となり得る領域を強調する
- 統制上の問題を特定し、貴組織の基準へのコンプライアンスを保証する
- 財務取引や時間依存の取引の年齢調べおよび分析を行う
- アナリティクステストを自動化し、結果の通知を即座に受け取る
- 分析手順の保存、再調査および結果の比較を可能にするために、実行した分析を記録する

Unicode および非 Unicode 版

Analytics には Unicode 版と非 Unicode 版があります。いずれのエディションも同じインストールパッケージに含まれ、インストール時にインストールするエディションを指定します。Analytics の製品およびサブスクリプション情報が記載されているダイアログボックス([ヘルプ > バージョン情報](#)]) で、バージョン番号の後ろに **Unicode** または **非 Unicode** が表示されます。

詳細については、「Diligent Unicode 製品」ページ 2672を参照してください。

製品名の変更

バージョン 11.4 以降、Analytics は ACL for Windows のコンポーネントになりました。ACL for Windows には、オフラインプロジェクトも搭載されています。

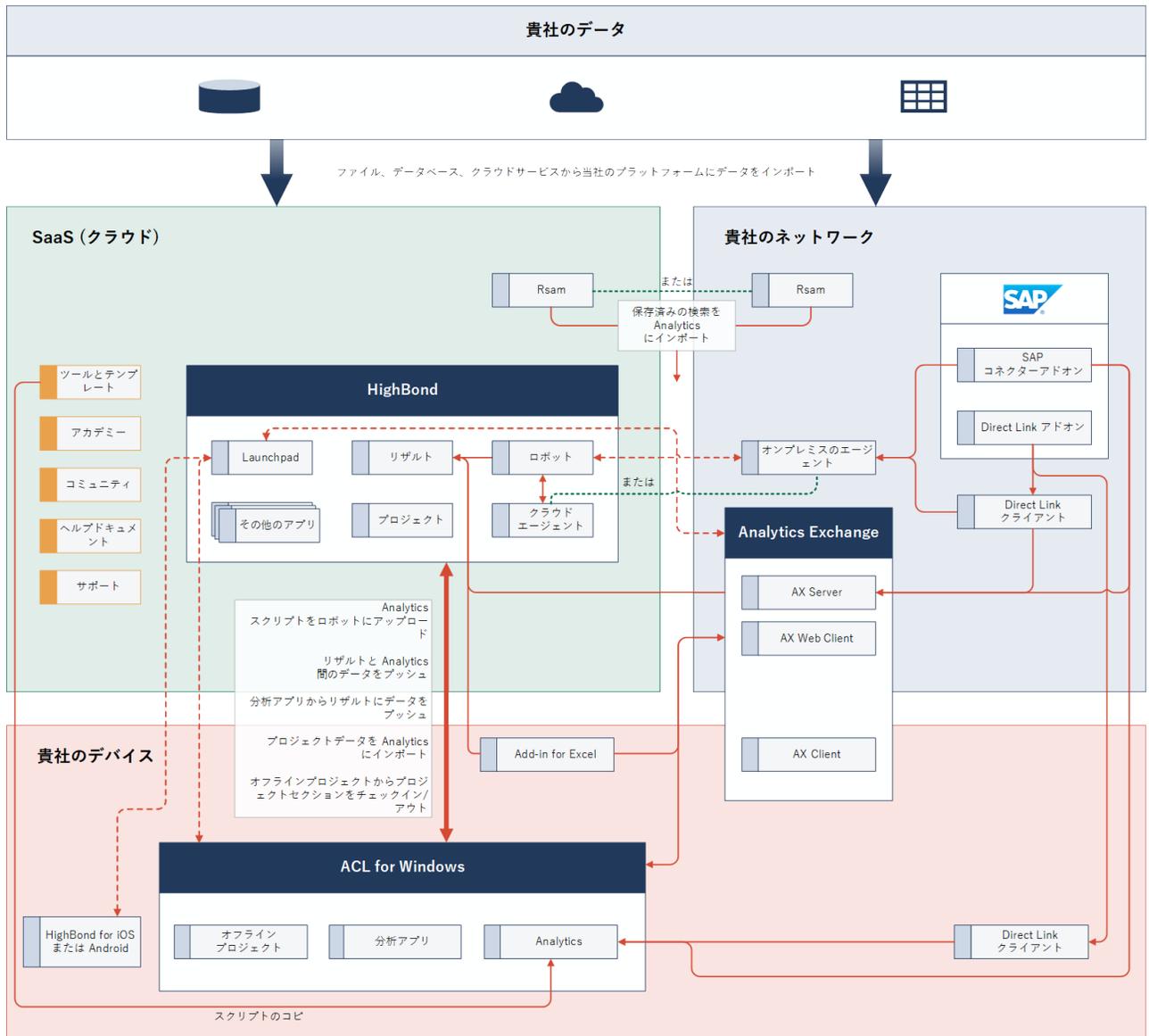
Analytics は、バージョン 10.0 より前では ACL Desktop と呼ばれていました。短縮名称は ACL でした。

Analytics が Diligent 製品のエコシステムにどう適合するか

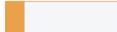
メモ

バージョン 16 以降の Analytics は Analytics Exchange(AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[ロボットにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。



凡例

-  リソース
-  アプリ/コンポーネント
-  データフロー
-  接続または通信
-  または 相互に排他的

セキュリティ

概念を表す目的でこの図が使用されています。特定のセキュリティの関係を示唆するものではありません。

ACL for Windows の概要

ACL for Windows は単一のダウンロード可能な製品であり、次へのアクセスを提供します。

- [Analytics](#)
- [ロボット](#)
- [リザルト](#)
- [オフラインプロジェクト](#)

メモ

各コンポーネントへのアクセスは、ご利用の Diligent のサブスクリプションによって決まります。

ACL for Windows のメイン画面を使用してアクセスできることは?

ACL for Windows を使って、以下のことが行えます。

- Analytics をアクティブ化して、製品の使用を開始する
- Analytics で Analytics プロジェクトを作成したり開いたりする
- HighBond リザルトに接続する。ここで、例外を整理、追跡、および修正するワークフローを構築できる
- Launchpad にアクセスする。ここで、HighBond ユーザーのすべての HighBond アプリへのリンク、および ScriptHub、インスピレーション、およびユーザー プロファイルへのリンクを提供する
- オフラインプロジェクトを使用して、HighBond プロジェクトからセクションをチェックアウトまたはチェックインし、作業をオフラインを実行する
- サポート、クイックスタートガイド、製品のフォーラム、オンラインヘルプ、およびトレーニングコースなどのリソースにアクセスする

ACL for Windows メイン画面の概要

The screenshot shows the ACL for Windows main interface. At the top, there is a dark blue header with the ACL logo on the left and a user profile icon with a red circle '2' next to it. Below the header is a search bar containing 'Vincicorp (US)' with a red circle '1' next to it. The main content area is divided into two columns. The left column has a section titled '最近のアナリティクス ファイル' with a refresh icon and a red circle '3'. Below this are two file entries: 'Sample Project.ACL' (Analytics プロジェクト, 既に開いています 7ヶ月前) and 'ACL_Cypress_PCards.acl' (Analytics プロジェクト, 既に開いています 7ヶ月前). Below these is a section titled 'サンプルファイル' with a red circle '6'. This section contains four file entries: 'ACL_Cypress_PCards.acl' (Analytics プロジェクト), 'ACL_DigiLink_Travel.acl' (Analytics プロジェクト), 'ACL_DigiLink_Travel.aclx' (分析アプリ), and 'ACL_Rockwood.acl' (Analytics プロジェクト). The right column has a section titled '開く' with a red circle '4'. Below this are three buttons: 'Analytics プロジェクト', '分析アプリ', and 'HighBond プロジェクト'. Below that is a section titled '作成' with a red circle '5'. Below this are two buttons: 'Analytics プロジェクト' and 'ワークフロー'. At the bottom of the right column is a section titled 'Galvanize に会う'. Below this is a paragraph of text: 'ACL は長い間、監査アナリティクスのリーダーとして知られてきました。もちろん、それは、ACL が非常に得意とする分野です。しかし、それだけではないことをご存知ですか。' Below the paragraph is a button labeled 'ブランド名称変更について'.

コンポーネント

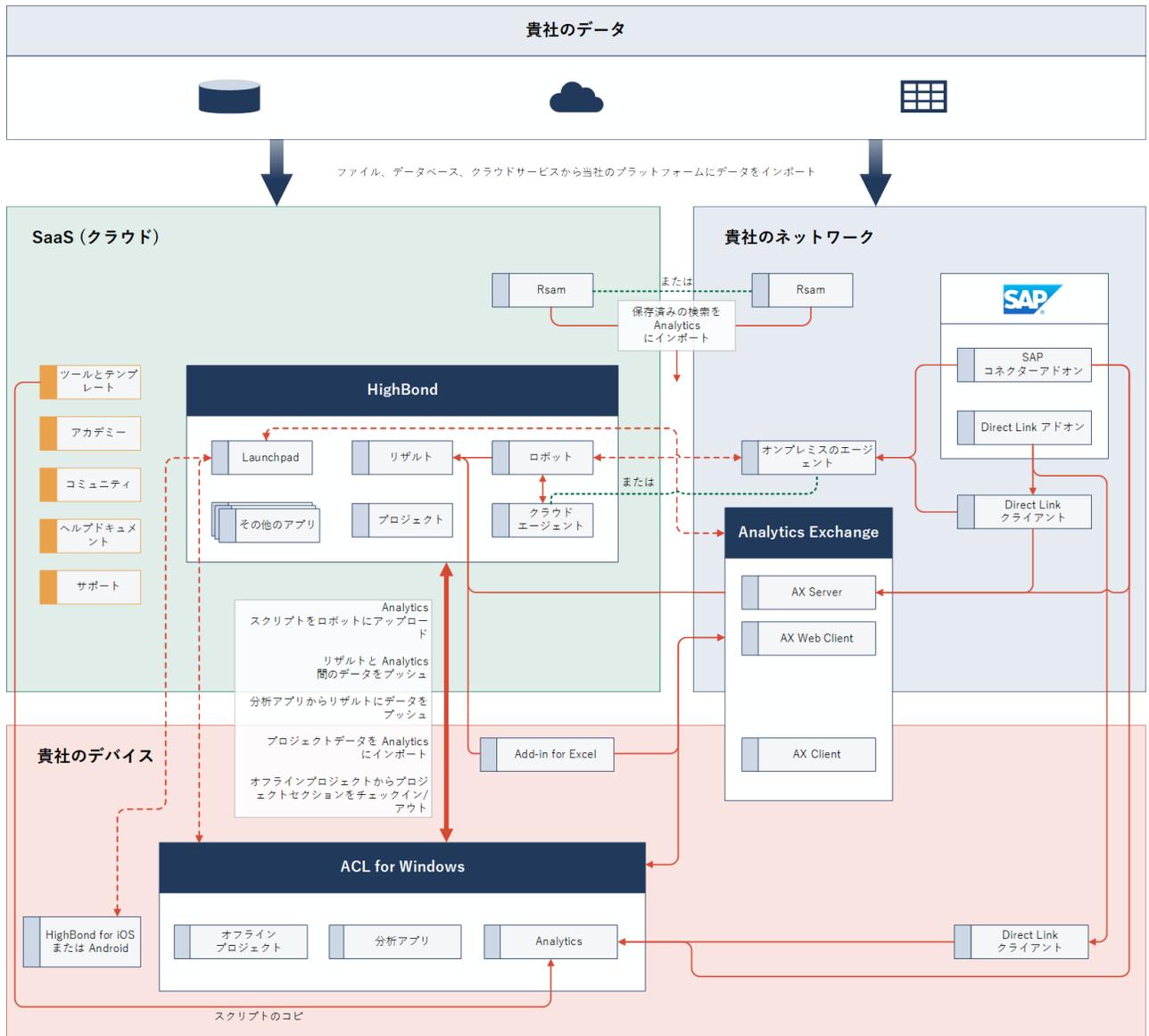
番号	コンポーネント	説明
1	組織セクター	このドロップダウンリストを使って、アクセス権のあるアカウント(組織)の間で切り替えを行います。
2	ツールバー	<p> プロフィール - プロフィールを更新したり、ACL for Windows からサインアウト することができます。</p> <p> 情報 - ヘルプまたは製品のドキュメントにアクセスしたりサポートに連絡することができます。</p>
3	最近のアナリティクスファイル	Analytics 内で最近アクセスしたファイルを表示します。 [リストの更新  <h2>ACL for Windows が Diligent 製品のエコシステムにどう適合するか</h2>

メモ

バージョン 16 以降の Analytics は Analytics Exchange(AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[サポートにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

はじめに



凡例

- リソース
- アプリ/コンポーネント
- データフロー
- 接続または通信
- または
相互に排他的

セキュリティ

概念を表す目的でこの図が使用されています。特定のセキュリティの関係を示唆するものではありません。

HighBond インスタンスの切り替え

所属の組織が複数のサブスクリプションを持っている、あなたがコンサルティングをしているのは複数の組織である、あるいはあなたがトレーニング インスタンスの一部である場合には、あなたは HighBond の複数のインスタンスに属している場合があります。複数の HighBond インスタンスに属している場合は、ACL for Windows を使用してインスタンス間を切り替えることができます。

HighBond インスタンスを切り替えると、別のサブスクリプションを使用して ACL for Windows をアクティブ化し、異なる組織や事業単位に属するデータにアクセスできます。

HighBond インスタンスを切り替えるには次を実行します。

1. ACL for Windows で、プロファイルドロップダウンリスト  から **サインアウトして閉じる** を選択します。
これで現在のインスタンスからサインアウトしました。
2. デスクトップにある ACL for Windows ショートカットをダブルクリックします。
Launchpad サイン イン画面が開きます。
3. HighBond アカウントを使用してサインインするには、ユーザー名 (電子メール) とパスワードを入力して、**サインイン** をクリックします。
4. ドロップダウン リストから該当するインスタンスを選択し、**アナリティクスのアクティブ化** をクリックします。

ACL for Windows が開きます。HighBond に関連して実行するすべての活動が、選択したインスタンスを対象に行われるようになります。

Analytics の基本操作 (非 Unicode 版)

この基本チュートリアルでは、Analytics を使用して、データを分析するエンドツーエンドのプロセスについて概要を説明します。

推定時間	60 分
要件	これまでに Analytics を使用したことがなくても構いません。一部の基本データ分析経験があることが前提ですが、絶対に必須であるわけではありません。
Analytics のバージョン	13.0 以降 (非 Unicode 版)
チュートリアルの正しいバージョンを実行する	非 Unicode 版の Analytics を使用している場合は、このバージョンのチュートリアルを実行してください。 Unicode 版を使用している場合は、"Analytics (Unicode 版) の基本" ページ 71 を実行してください。

ヒント

使用している Analytics のエディション(版)を確認するには、Analytics のメインページで、[\[ヘルプ > バージョン情報\]](#)に移動し、[\[アナリティクス\]](#)ダイアログボックスを開きます。バージョン番号の後に版の指定が表示されます。

メモ

中国語および日本語のユーザーインターフェイスは Unicode のみです。

シナリオ

法人クレジットカード取引の確認

2 か月間の法人クレジットカード取引を確認する必要があります。目標は、この期間中に従業員がどのようにカードを使用したのかについて概要を認識し、カードの悪用の可能性を特定することです。

取引データは 3 つの Excel ワークシートにあります。データを分析する前に、それを Analytics にインポートし、個別のデータセットを単一の Analytics テーブルに結合する必要があります。

データを分析した後、分析結果を視覚的に表示し、オーディエンスとの対話を効果的にすることができます。

任意のセクション

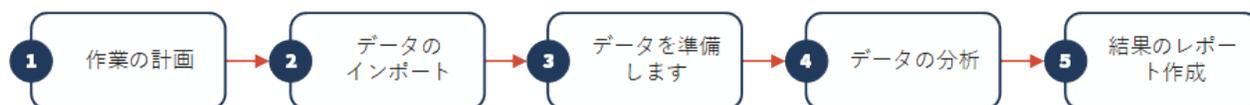
これから、法人クレジットカード取引の確認を繰り返し行うことが義務付けられました。

自分または他の誰かがすばやく正確に将来のレビューを実行できるようにするために、作業の一部を自動化するスクリプトを作成することにしました。

PIPAR - Analytics のデータ分析サイクル

Analytics のデータ分析には、5つの段階があります。これは略語のPIPARで表されます。

Plan(計画)、Import(インポート)、Prepare(準備)、Analyze(分析)、Report(レポート)



作業の計画

データ分析作業の計画は非常に重要です。計画ステージを省略し、データに対して分析コマンドを実行すると、問題の発生、追加作業、または重要な分析インサイトの喪失が生じるおそれがあります。

基本計画でも、計画がないよりはよいので、お勧めします。Analyticsに関するお客様の経験と知識が増えるとともに、計画をより詳細かつ正確に作成できるようになります。優れた計画は、スムーズかつ効率的に進むデータ分析プロジェクトにとって主要な要素です。

計画のガイドライン

明確で具体的な目標を策定する

分析で想定される最終製品は何ですか。

実現方法を計画するには、明確に定義された目標が必要です。たとえば、このチュートリアルでは、具体的な目標は次のとおりです。

- 各マーチャント カテゴリの法人クレジットカード取引のカウンとおよび合計金額を特定する
- 禁止されたカテゴリの取引を特定する

段階的なアプローチをマッピングする

目標を達成する方法

はじめに

多くの場合、目標を達成するには、複数の手順が必要になるため、目標に合わせて、詳細で段階的なアプローチをマッピングします。

たとえば、このチュートリアル計画の手順のうち2つは次のようになります。

- すべての個別の取引ファイルを単一のファイルに結合する
- 結合された取引データをマーチャントカテゴリにグループ化する

大きい目標を個別の手順に分解した後には、各手順を実行するために使用する Analytics の特徴および機能を検討することができます。

必要なデータを特定する

目標を達成するには、どのデータが必要ですか。

必要なソースデータを、特定のデータ要素またはフィールドのレベルに細分化します。適切な入力がない場合は、目的の出力を得ることはできません。

このチュートリアルには、主な取引ファイルがありますが、2番目の目標を達成するには、禁止されたマーチャントカテゴリコードのリストが必要になります。

技術要件を考慮する

考慮する必要がある技術的な考慮事項はありますか。

データ分析で使用しているツールに関係なく、制約内で作業する必要があります。ソースデータは、ツールがアクセスできる場所やシステムに格納され、読み取れる形式ですか。提案している分析はルールによってサポートされますか。

たとえば、Analytics で複数のテーブルを結合するためには、各テーブルの対応するフィールドのデータ型が同じである必要があります。Analytics は、フィールドのデータ型の変更をサポートしますが、計画で考慮する必要がある手順です。

繰り返す準備をする

進めるにつれ、計画を調整しなければならない場合があります。

分析の過程で、さらなる調査を保証する、予期しない事態に遭遇します。追加のデータと追加のアナリティクス手順が必要であることを認識します。

データの理解が深まるにつれ、計画も進化します。これにより、類似した本質となる将来の分析のために、より成熟した計画の基礎となることができます。

データのインポート

分析する前に、データを Analytics にインポートする必要があります。

データ定義ウィザードを使用して、3つの Excel ワークシートにインポートすることで、インポート処理に慣れていきます。Excel からのインポートは、Analytics で分析するためのデータを取得するための最も一般的な方法の1つです。ただし、Analytics は、さまざまなデータソースからのデータのインポートをサポートします。

Analyticsを開始し、"Sample Project.ACL"を開く

メモ

以下の手順では、Analyticsをアクティブ化済みであることが前提です。

手順

1. デスクトップにある **ACL for Windows** ショートカットをダブルクリックします。
2. **ACL for Windows** 画面の **開く**の下で、**Analytics プロジェクト**をクリックします。
3. **C:\Users\user_account_name\Documents\ACL Data\Sample Data Files**に移動し、**Sample Project.ACL**をダブルクリックします。Analytics内に
Sample Project.ACLが開かれます。

Analyticsをインストールしたときに**Sample Data Files**フォルダーをデフォルトの場所にインストールしていない場合は、インストールした場所に移動します。

2つのExcelワークシートのインポート

まず、2つのExcelワークシートを同時にインポートします。複数のExcelワークシートを同時にインポートすると、作業の手間を大幅に削減できます。

手順

1. Analyticsのメインメニューから、**インポート > ファイル**を選択します。
2. **定義するファイルの選択**ダイアログボックスで、**Trans_May.xls**を選択し、**開く**をクリックします。

Excelファイルは**Sample Project.ACL**と同じフォルダーにあります。

3. **ファイル形式**ページで、**Excel ファイル**オプションが選択されていることを確認し、**次へ**をクリックします。
4. **データソース**ページで、ファイルの両方のワークシートを選択します。
 - **Trans1_May\$**
 - **Trans2_May\$**
5. **先頭の行をフィールド名として使用する**が選択されていることを確認し、**次へ**をクリックしてから、**完了**をクリックします。

2つのExcelワークシートが2つの別のAnalyticsテーブルにインポートされます。

3番目のExcelワークシートのインポート

3番目のExcelワークシートをインポートします。単一のワークシートをインポートするときには、一部のメタデータ設定を、後からAnalyticsで調整するのではなく、インポート処理中に手動で調整するオプションがあります。

手順

1. 直前の手順にあるすべてのステップを繰り返すことで、**Trans_April.xls**を見つけて選択します。
2. **ファイル形式**ページで、**Excel ファイル**オプションが選択されていることを確認し、**次へ**をクリックします。

はじめに

3. **データソース]** ページで、**Trans_Apr\$** を選択します。
4. **先頭**の行をフィールド名として使用する] が選択されていることを確認し、**次へ]** をクリックします。
5. **Excel インポート]** ページで、ヘッダーをクリックして、**TRANS_DATE** 列を選択し、次の変更を行います。
 - **名前]** フィールドで、**TRANS_DATE** を **DATE** に変更します。
 - **長さ]** フィールドで、**19** を **10** に変更します。

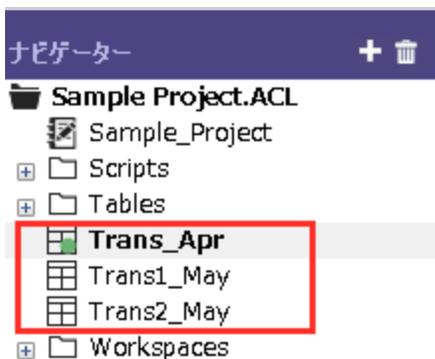
メモ

インポート処理中には、データ定義ウィザードでデータフィールドを調整します。データのインポートが完了した後に、後から調整することもできます。チュートリアル次のセクションでは、調整の理由について説明します。

6. **ファイル名]** フィールド型 **Trans_Apr** で **次へ]** をクリックし、**保存]** をクリックします。
7. **完了]** をクリックしてから、**OK]** をクリックします。

3番目のExcelワークシートはAnalyticsテーブルにインポートされます。

ナビゲーターの **概要]** タブに、新しい3つのAnalyticsテーブルが表示されます。これらのテーブルには、Excelデータの読み取り専用コピーが含まれます。Excelソースデータ自体は含まれません。



データの準備

多くの場合、データの分析準備が整う前に、1つ以上のデータ準備タスクを実行する必要があります。このチュートリアルでは、次の2つの準備タスクを実行します。

- 追加の調整を行い、データフィールドを合わせる
- 分析のために、3つの新しい Analytics テーブルを単一のテーブルに結合する

また、ベストプラクティスとして、分析作業を実行する前に、インポートされたデータの有効性を必ず確認することをお勧めします。テーブルに無効なデータが少しあるだけでも、すべての後続のデータ分析が無効になる可能性があります。

データの準備が必要な理由

データの分析を進めようとしても、適切なデータ準備を行わないと、分析を実行できない場合があります。あるいは、実行する分析に欠陥が生じる可能性があります。

何らかの初期準備がないと、さまざまな問題がソース データに影響し、分析に適さなくなる可能性があります。

例：

- ソース データは複数のファイルに分散しているため、単一のデータのセットとして分析できるように、結合する必要があります。
- 異なるファイルの対応するフィールドは、「調整」する必要があります。つまり、処理するための前提条件として、同じ構造と形式にします。
- 「汚いデータ」はクレンジングして正規化する必要があります。これは、Analytics 関数を使用して実行できます。

要点

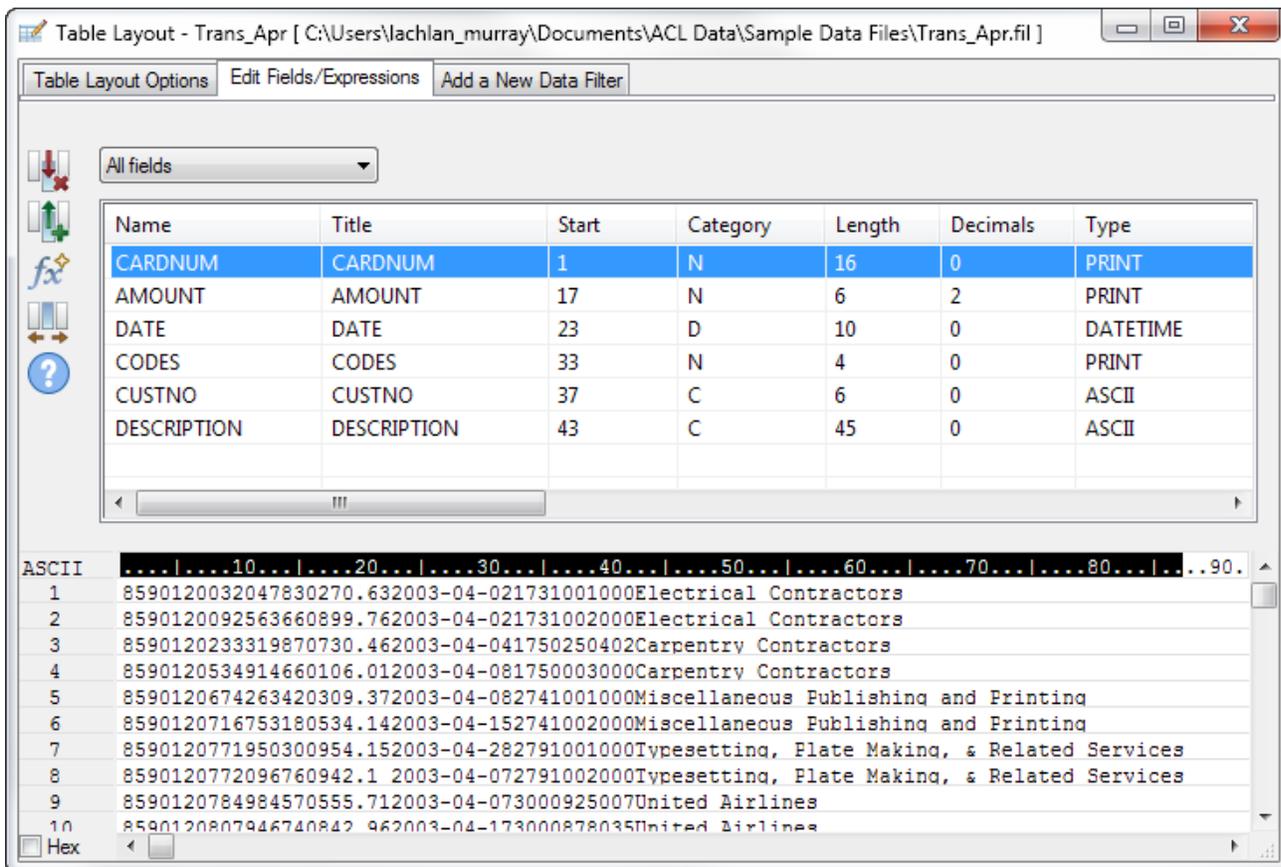
データのインポートと準備にかかる時間は、実際の分析にかかる時間を超えることがあります。ただし、これらは重要な初期段階であり、分析の基になる基盤を提供します。

テーブルレイアウトの調整

Analytics プロジェクトのすべてのテーブルには**テーブルレイアウト**があります。テーブルレイアウトは、フィールド名、フィールドの開始位置、フィールドの長さ、フィールドのデータ型などのメタデータを含んでいます。

3つの新しい Analytics テーブルを単一のテーブルに結合する前には、テーブルレイアウトの一部のメタデータを調整する必要があります。

ここでは、**Trans_Apr** テーブルレイアウトの内容を示します。Analytics に慣れるにつれ、テーブルレイアウトの操作方法をすぐに学ぶことができます。テーブルレイアウトには、多数の便利な点があります。



Trans_Apr テーブルレイアウトの調整

まず、Trans_Apr テーブルの2つのフィールドのデータ型を変更する必要があります。

手順

1. まだ開いていない場合は、Trans_Apr テーブルを開きます。
テーブルを開くには、ナビゲーターでテーブルをダブルクリックします。
2. テーブルビューの上で **テーブルレイアウトの編集**  を選択します。
3. **CARDNUM** フィールドをダブルクリックし、編集用にフィールド定義を開きます。
4. **有効なデータ型** の下で、**ASCII** をダブルクリックして、フィールドのデータ型を更新します。

他の2つのテーブルでは、**CARDNUM** フィールドのデータ型はASCIIです。ほとんどの部分では、データを結合するには、結合されるテーブルの対応するフィールドのデータ型が同じである必要があります。

5. **入力を受け入れる**  をクリックします。
プロンプトが表示されたら、**はい** をクリックして、変更を保存します。
6. **CODES** フィールドをダブルクリックしてデータ型を **ASCII** に変更します。

7. 入力を受け入れる  をクリックし、閉じる  をクリックして **テーブルレイアウト** ダイアログボックスを終了します。

Trans_May テーブルレイアウトの調整

調整を完了するには、Trans1_May および Trans2_May レイアウトの両方の2つのフィールドのデータ型を変更する必要があります。DATE フィールドの調整も必要な場合があります。

手順

上記のプロセスに従い、Trans1_May および Trans2_May レイアウトの両方で、次の変更を行います。

フィールド	次のデータ型に変更します。	追加の変更
CODES	ASCII	
AMOUNT	PRINT	小数点 フィールドに 2 を入力し、数値に2桁の小数点が表示されることを指定します。
DATE	変更なし	<p>メモ</p> <p>DATE フィールドの長さが既に10の場合は、調整は不要です。</p> <ul style="list-style-type: none"> ○ 長さフィールドで、19を10に変更します。この変更により、空の時刻データが省略されます。 ○ 書式設定ドロップダウンリストで、YYYY-MM-DDを選択します。

完了すると、May テーブルレイアウトが以下のレイアウトのようになります。

メモ

日付形式 (YYYY-MM-DD) はレイアウト サマリーに表示されません。DESCRIPTION フィールド長は、2つのMayレイアウトで異なります。

Table Layout - Trans2_May [C:\Users\lachlan_murray\Documents\ACL Data\Sample Data Files\Trans2_May.fil]

Table Layout Options Edit Fields/Expressions Add a New Data Filter

All fields

Name	Title	Start	Category	Length	Decimals	Type
CARDNUM	CARDNUM	1	C	19	0	ASCII
CODES	CODES	20	C	4	0	ASCII
DATE	DATE	24	D	10	0	DATETIME
CUSTNO	CUSTNO	43	C	6	0	ASCII
DESCRIPTION	DESCRIPTION	49	C	155	0	ASCII
AMOUNT	AMOUNT	204	N	9	2	PRINT

ASCII

```

1 8590-1224-9766-380727412003-05-04 00:00:00962353Miscellaneous Publishing and Printing
2 8590122281964011 50212003-05-01 00:00:00812465Office furniture
3 8590120784984566 30662003-05-02 00:00:00051593Southwest
4 8590-1242-5362-174479222003-05-03 00:00:00250402Theatrical Producers (except Motion Pictures)
5 8590125999743363 30072003-05-05 00:00:00778088Air France
6 8590120716753180 86992003-05-05 00:00:00778088Membership organization
7 8590128947747852 35432003-05-06 00:00:00250402Four Seasons Hotels
8 8590122720558982 35352003-05-07 00:00:00051593Hilton
9 8590128676326319 35352003-05-08 00:00:00778088Hilton
10 8590124781270125 85902003-05-08 00:00:00778088Membership organization
  
```

Hex

インポートされたデータの検証

では、3つのインポートされたテーブルのデータを検証し、追加のデータ準備、およびデータ分析を進めても問題がないことを確認しましょう。

メモ

データ型を更新した後に、データを検証しています。Analyticsでデータを検証するときには、フィールドのすべての値がフィールドのデータ型の要件に適合することを確認しています。このため、データ型が確定した後にのみデータを検証することに意味があります。

手順

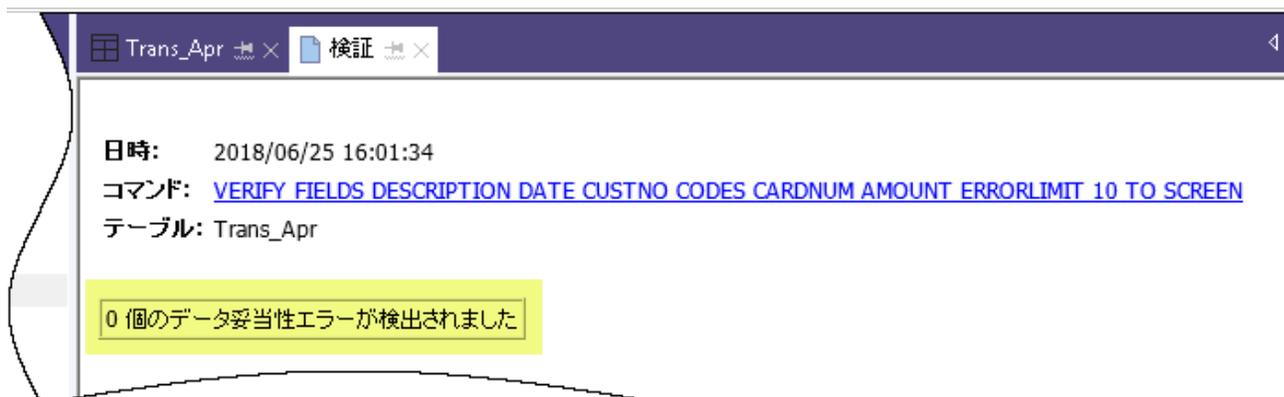
1. Trans_Apr テーブルを開きます。
2. Analytics のメインメニューから、**データ > 検証**]を選択します。
3. **検証**]ダイアログボックスで、フィールド リストのすべてのフィールドを選択します。

ヒント

Shift+click を使用して、複数の隣接するフィールドを選択します。

4. [OK]をクリックします。

結果は次のようになります。0個のデータ妥当性エラーが検出されました。



詳細

お客様がユーザーインターフェイスで実行したアクションが自動的にACLScript `VERIFY` コマンドに変換されたことに気が付きましたか。ユーザーインターフェイスで実行するすべてのコマンドレベルのアクションは、自動的に対応するACLScriptコマンドに変換され、各 Analytics プロジェクトに付属するコマンドログに取り込まれて格納されます。

このように、有効で実行可能なスクリプト構文の自動生成は、最も強力な Analytics 機能の1つです。チュートリアル最後のオプションのセクションで、スクリプトを確認します。

5. ナビゲーターで、Trans1_May テーブルをダブルクリックして開き、手順を繰り返して、データを検証します。
6. Trans2_May テーブルでも同じことを実行します。

両方のテーブルには、データ検証エラーは含まれません。

メモ

最大エラー上限に達しましたというエラーメッセージが表示される場合は、テーブルレイアウトで日付フィールドの形式を正しくYYYY-MM-DDに変更したことを確認します。

詳細表示

Analytics がデータ検証エラーを特定するときの処理を確認する場合は、Tables\Badfile を開き、検証プロセスを実行します。

3つのAnalytics テーブルを結合する

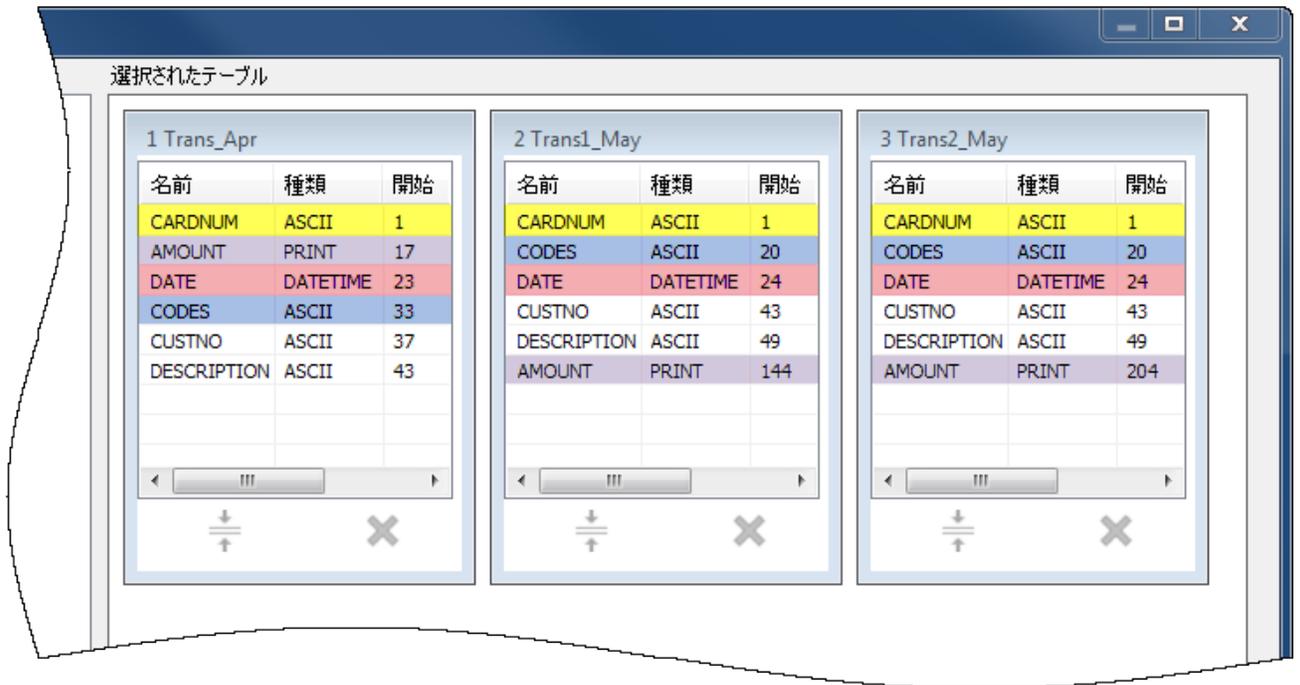
最終データ準備タスクでは、3つの新しいAnalytics テーブルを単一のテーブルに結合します。

簡単にするため、チュートリアルは3つのテーブルのみを結合します。ただし、同じプロセスを使用して、12個の月次テーブルを単一の年次テーブルに結合し、年度全体のデータに対して分析を実行することができます。

手順

1. Analytics のメインメニューから、**データ > 最後に追加]**を選択します。
2. **使用可能なテーブル]**の下で、各新しいテーブルをダブルクリックし、**選択したテーブル]**領域に追加します。
3. 3つのテーブルのフィールドを確認し、名前とデータ型が、データ定義ウィザードと**テーブルレイアウト]**ダイアログボックスで行った変更に基づいてどのように同じになっているのかを見てください。

対応するフィールドを後に追加するには、名前が同じであり、ほとんどの状況でデータ型も同じである必要があります。



4. **出力テーブルの使用]**を選択し、コマンドの実行後に、結合されたデータがあるシャットカテーブルが自動的に開くようにします。
5. **To]**フィールドで、**Trans_All**と入力し、**OK]**をクリックします。
6. ポップアップ通知で、**はい]**をクリックします。

メモ

通知については気にする必要はありません。追加コマンドは、数値フィールドの一部の自動調整を実行します。これにより、時間と作業工数を削減できます。

新しい **Trans_All** テーブルが作成され、3つの入力テーブルのすべてのレコードが含まれます。
Analytics のインターフェイスの下部にある状態バーのレコード数が、**レコード: 481** となります。

これで、実際のデータ分析に進むことができます。

データの分析

コマンドおよび他のツールを使用して、Analytics で分析を実行し、調査しているデータに関する一般的な洞察を得て、特定の問題に対して回答します。

メモ

分析段階では、前の計画の強度が明らかになります。調査に関する明確な目標を策定している場合は、実行する分析の種類について明確な考えを持つことになります。

データ分析

このチュートリアルでは、**Trans_All** テーブルのデータについて次の分析を実行します。

- 次の点を検出するために、マーチャント カテゴリコードによって、クレジットカード取引レコードをグループ化します。
 - 従業員が法人クレジットカードを使用する方法
 - 各カテゴリで費やされる金額
 - フィルターを作成し、禁止された取引を分離する

マーチャント カテゴリコードによって、クレジットカード取引レコードをグループ化する

データのセットをグループ化または要約すると、データの概要をすばやく確認することができます。

手順

1. まだ開いていない場合は、**Trans_All** テーブルを開きます。
2. Analytics のメインメニューから、**分析 > 要約** を選択します。
3. **要約** ダイアログボックスで、次のフィールドとオプションを選択します。

タブ	フィールドまたはオプション	選択または入力
メイン	要約オン	CODES を選択
	その他のフィールド	DESCRIPTION を選択
	小計フィールド	AMOUNT を選択
	平均、最小、最大	チェックボックスを選択

はじめに

タブ	フィールドまたはオプション	選択または入力
出力	To	ファイルを選択
	名前	タイプ <input type="text" value="Trans_All_Grouped"/>

4. [OK]をクリックします。

新しい **Trans_All_Grouped** テーブルが作成されます。テーブルに 110 レコードがあり、**Trans_All** テーブルの各一意のマーチャント カテゴリコードにつき 1 件のレコードがあります。**COUNT** フィールドは、各グループのソースレコード数を示します。

ヒント

テーブルビューを右クリックし、**すべての列のサイズを変更**を選択して、ビューをよりコンパクトにします。

調査のための簡易ツール

データの要約バージョンがあるため、一部の基本 Analytics ツールを使用して、法人クレジットカード利用についての一般的な洞察を得ることができます。

数回クリックするだけで、使用パターン、悪用の可能性について、多くの情報が得られます。

この洞察を得るには	Trans_All_Grouped テーブルでこれを実行します。
4 月および 5 月中の従業員の合計請求金額はいくらでしたか。	<ul style="list-style-type: none">◦ Total AMOUNT ヘッダーを選択します。◦ 分析 > 合計 の順に選択します。 支出合計は \$187,177.13 でした。
従業員が最もカードを使ったのはどこでしたか。	<ul style="list-style-type: none">◦ Total AMOUNT ヘッダーを右クリックして クイックソート 降順 を選択します。 DESCRIPTION フィールドは、次の項目に最も多い金額が費やされたことを示します。 <ul style="list-style-type: none">◦ Caterers◦ Eating places and Restaurants◦ Hilton International
最も大きい単一の支出は何か。	<ul style="list-style-type: none">◦ Maximum AMOUNT ヘッダーを右クリックして クイックソート 降順 を選択します。 DESCRIPTION および Maximum AMOUNT フィールドは、最も大きい単一の支出は Club Med の \$1999.06 であることを示しています。 Club Med は法人クレジットカードで許可されたマーチャント コードか。クレジットカード上限が \$2000 の場合、従業員の請求金額は上限以下か。
使用頻度が低いコードの検査で何が明らかになるか。	<ul style="list-style-type: none">◦ COUNT ヘッダーを右クリックして クイックソート 昇順 を選択します。 5 つのカテゴリにはそれぞれ 1 つの請求のみがありました。禁止されたカテゴリはあるか。おそらく、1 人または複数の従業員は、法人カードをごくまれに誤用しても、検出を逃れることができると考えているかもしれません。 <ul style="list-style-type: none">◦ Cigar Stores & Stands◦ Dating & Escort Svcs.◦ Babysitting services◦ Amusement Parks

この洞察を得るには	Trans_All_Grouped テーブルでこれを実行します。
	<ul style="list-style-type: none"> ○ Civic, Fraternal, and Social Associations
禁止されたカテゴリはあるか。	<ul style="list-style-type: none"> ○ DESCRIPTION]ヘッダーを右クリックし、クイックソート 昇順]を選択すると、簡単に精査できるようにフィールド値をアルファベット順に並べ替える ○ 不審なカテゴリを探すためにフィールドを精査する <p>1つ以上のカテゴリが禁止されている可能性があるか。</p> <ul style="list-style-type: none"> ○ Babysitting services ○ Betting (including Lottery Tickets, Casino) ○ Civic, Fraternal, and Social Associations ○ Dating & Escort Svcs. ○ Massage Parlors ○ Precious Stones and Metals, Watches and Jewel ○ Video Game Arcades/Establishments <p>メモ</p> <p>小さいデータセットを除き、手動精査は実践的ではありません。次に、より実践的で、より信頼できる方法を見ていきます。</p>

詳細表示

何らかの簡易分析のみを実行し、結果を新しいテーブルに出力したくない場合もあるでしょう。**要約**]ダイアログボックスで、**ファイル**]を選択する代わりに、**Trans_All** テーブルを要約すると、**画面**]を選択し、結果を Analytics 表示領域に出力できます。

日時: 2018/06/26 13:50:00
 コマンド: [SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO SCREEN PRESORT](#)
 テーブル: Trans_All

CODES	AMOUNT	カウント	DESCRIPTION
1731	1,486.30	3	Electrical Contractors
1750	1,095.57	3	Carpentry Contractors
2741	1,353.94	3	Miscellaneous Publishing and Printing
2791	2,141.29	3	Typesetting, Plate Making, & Related Services
3000	2,590.70	3	United Airlines
3001	1,276.42	3	American Airlines
3005	1,659.97	3	British Airways
3007	2,616.76	3	Air France
3008	3,497.90	4	Lufthansa
3009	2,680.30	4	Air Canada
3025	4,319.54	4	Air New Zealand

画面への出力は小さいデータセットでのみ実践的です。ただし、個別のグループで簡単にドリルダウンし、各グループのソースレコードのみを表示できるという利点があります。

(CODES = "3000")

CARDNUM	AMOUNT	DATE	CODES	CUSTNO	DESCRIPTION
9 8590120784984570	555.71	2003/04/07	3000	925007	United Airlines
10 8590120807946740	842.96	2003/04/17	3000	878035	United Airlines
389 8590122491455345	1192.03	2003/05/21	3000	503458	United

<< ファイルの終わり >>

フィルターを作成し、禁止された取引を分離する

フィルターを使用すると、特定の時点に関心があるレコードのみを分離できます。フィルターは、データに関する特定の問題に回答するための強力なツールです。

法人クレジットカード取引の一般レビューにより、一部の禁止された取引の可能性が通知されました。そこで、禁止されたマーチャント カテゴリコードの一覧をデータと照合し、禁止された取引があるかどうかを確認することに決めました。

手順

フィルター式を作成する

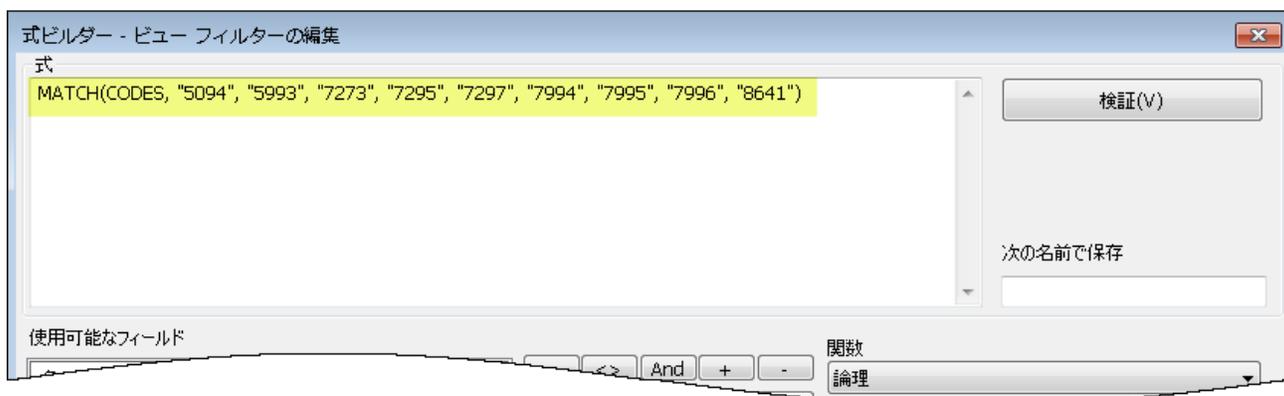
1. Trans_All テーブルを開きます。
2. テーブルビューの上部にある **ビューフィルターの編集**  をクリックして、**式ビルダー**を開きます。
式ビルダーは Analytics コンポーネントであり、式構文を手動で入力するのではなく、マウスを使用して式を作成できます。式は値と演算子を組み合わせただけのもので、これにより計算を実行したり結果を返したりすることができます。
3. **関数** ドロップダウンリストで、**論理** を選択してから、MATCH 関数をダブルクリックして、**式** テキストボックスに追加します。
MATCH を使用して、**CODES** フィールドの複数の禁止されたマーチャント カテゴリコードを分離します。
4. **式** テキストボックスで、**比較値** プレースホルダーをハイライト表示してから、**使用可能なフィールド** リストで、**CODES** をダブルクリックします。
CODES フィールドは **比較値** を置換します。
5. 以下の禁止されたコードの文字列をコピーし、`test1 , test2 <,test3...>` プレースホルダーを置換します。

"5094", "5993", "7273", "7295", "7297", "7994", "7995", "7996", "8641"

メモ

すべての引用符を含む文字列全体を必ずコピーします。

式は次のようになります。



式を検証し、フィルターを保存して適用します。

1. **検証]**をクリックし、式の構文が有効であることをテストします
作成してすぐに式を検証することをお勧めします。より時間がかかる後からのトラブルシューティングを回避するうえで役立つためです。
エラーメッセージが表示される場合は、式の構文が正確に上記の構文と一致することをもう一度確認します。
2. **名前を付けて保存]**フィールドでフィルター名 `f_Prohibited_codes` をコピーします。
Diligent は、保存されたフィルターの先頭に `f_` を付けることをお勧めします。
3. **OK]**をクリックします。
`f_Prohibited_codes` フィルターは `Trans_All` テーブルに適用されます。禁止されたマーチャント カテゴリコードを使用する取引が分離され、簡単に確認できます。数万件以上のレコードがあるテーブルを考えると、フィルターの価値はすぐにはっきりとわかります。

フィルターの削除または再適用

フィルターを削除して、再適用してみてください。

1. フィルターを削除するには、**フィルターの削除**  をクリックします。
2. フィルターを再適用するには、次のいずれかを実行します。
 - ビューの上部にある **フィルター履歴]** ドロップダウンリストから、フィルター名を選択します。
 - **ビューフィルターの編集**  をクリックして**式ビルダー**を開き、**フィルター]** リストでフィルター名をダブルクリックして **OK]** をクリックします。

ヒント

フィルター履歴] リストには、最大 10 個のフィルターがあるため、保存されたフィルターを再適用するために、**式ビルダー**を使用しなければならない場合があります。

詳細表示

フィルターよりも強力な機能

フィルターは、フィルターに含まれる基準または条件が管理可能である場合に、効果的に動作します。このチュートリアルで作成したフィルターには、9 コードしかありません。しかし、禁止されたマーチャント カテゴリコードのリストが何十個かそれ以上になった場合はどうなるでしょうか。

より効率的なアプローチは、禁止されたコードを含む Analytics テーブルを取引テーブルと結合することでしょう。結合された出力テーブルの一致でも、禁止された取引になります。

結合はこのチュートリアル範囲外ですが、Analytics では頻繁に使用される機能です。

結果のレポート作成

データ分析が完了すると、Analytics は、複数の方法で、結果を報告または表示します。

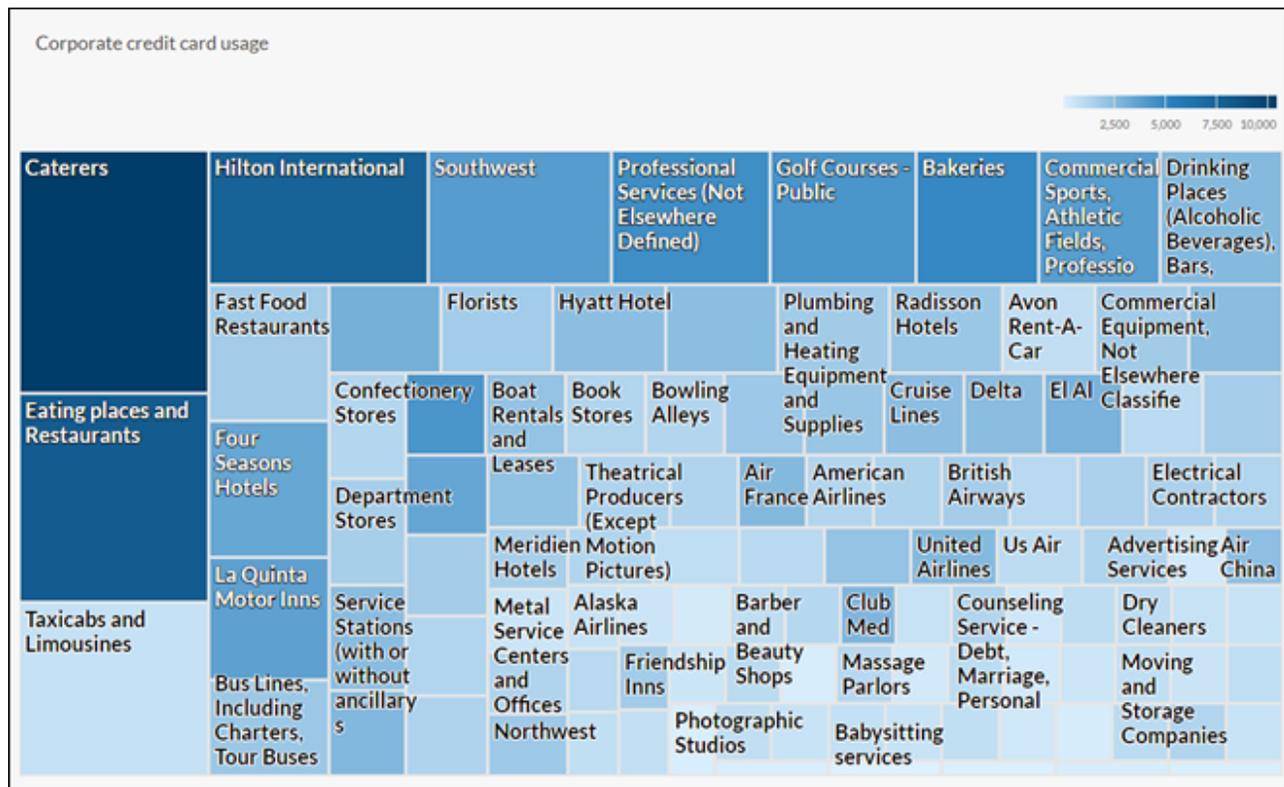
データ列を使用した従来のレポートがありますが、以下に説明する、よりわかりやすいデータ視覚化を使用して結果を伝える方法を確認していきます。

ツリーマップ視覚化

このツリーマップ視覚化は、**Trans_All_Grouped** テーブルで出力するグループ化されたクレジットカード取引を示します。グループ間の関係は、次の2つの方法で表されます。

- **ボックスのサイズ** -は、各グループの個別の取引のカウントを示します
ボックスが大きいほど、取引数が多くなります。ボックスは、左上から右下へ、サイズに配置されます。
- **ボックスの色強度** -は、各グループの合計数を示します
ボックスが濃いほど、合計数が多くなります。

このため、たとえば、右下の四分円にある **Club Med** ボックスのサイズは、取引数が少ないことを示しますが、色は合計の取引金額が大きいことを示します。



準備作業

リザルトでツリーマップ視覚化を作成しようとしています。リザルトは、クラウドベースの HighBond プラットフォームの問題修正アプリです。リザルトのライトバージョンへのアクセスは、ACL Robotics サブスクリプションに含まれます。

視覚化を作成するには、まず、視覚化を格納するための、シンプルな 2 レベルのデータコンテナを作成する必要があります。最初のレベルはコレクション、2 番目のレベルは分析と呼ばれます。すばやく簡単に作成できます。

Launchpad にサインインし、リザルトにアクセスします。

メモ

何らかの理由で Launchpad にサインインできないか、リザルトにアクセスできない場合は、"Analytics の他のレポート(作成)方法" ページ 59 の一覧にある代替レポート作成方法のいずれかを使用できます。

手順

1. Launchpad(www.highbond.com) に移動します。
2. HighBond アカウント資格情報(の電子メールとパスワード) を入力して、**[サインイン]** をクリックします。

Launchpad が開きます。

3. **[リザルト]** をクリックします。

リザルト ホームページが開きます。

メモ

リザルトにアクセスできない場合は、適切なサブスクリプションまたはリザルト ロールが割り当てられていない可能性があります。"Analytics の他のレポート(作成)方法" ページ 59 の一覧にある代替レポート作成方法のいずれかを使用します。

リザルトにアクセスする場合は、会社の Analytics アカウント 管理者に連絡してください。

コレクションの作成

手順

1. リザルト のホームページで、**新しいコレクション]** をクリックします。
2. **新しいコレクション]** ページの **名前]** フィールドで、**ACL チュートリアル** を入力またはコピーします。
3. ページの下部で **コレクションの作成]** をクリックします。

コレクションの設定] ページが表示されます。

分析を作成する

手順

1. **[コレクションの設定]** ページの最下部の **次の手順]** で、**初めてデータ分析を作成する]** をクリックします。

分析の詳細] ページが開きます。

2. **分析の詳細]** ページの **名前]** フィールドで、**サンプル レポート** を入力またはコピーします。
3. **分析の作成]** をクリックします。

新しい ACL チュートリアルコレクションが開き、作成した空の**サンプルレポート**分析が表示されます。

メモ

リザルトを開いたままにします。戻って、データ視覚化を作成します。

Analytics データのリザルト へのエクスポート

次の段階では、**Trans_All_Grouped** テーブルを Analytics からリザルトにエクスポートします。

手順

1. Analytics で、**Trans_All_Grouped** テーブルを開きます。
2. Analytics のメインメニューから、**データ > エクスポート]** を選択します。

はじめに

3. **【エクスポート】**ダイアログボックスで、次のオプションを選択します。

タブ	オプション	選択
メイン	ビュー	ビューを選択
	名前を指定してエクスポート	HighBondを選択

4. **【o】**をクリックし、**【送信先テストの選択】**ダイアログボックスで、作成した**サンプルレポート**分析コンテナに移動し、ダブルクリックして開きます。
5. **【新しいデータアナリティクス】**フィールドで `Trans_All_Grouped` を入力またはコピーし、**【作成】**をクリックします。

【エクスポート】ダイアログボックスに戻ります。ID番号とデータセンターコードは **【o】**テキストボックスに事前に入力されます。

6. **【OK】**をクリックします。
Trans_All_Grouped テーブルがリザルトにエクスポートされます。

視覚化の作成

リザルトで視覚化を作成する準備ができました。

手順

1. リザルトで **ACL チュートリアル**コレクションに戻り、**F5** を押して、ブラウザ ウィンドウを更新します。
Trans_All_Grouped テーブルが表示されます。
2. **【改善】**で **【レコードの表示】**をクリックします。
テーブルビューが開き、レコードが表示されます。
3. **【視覚化の追加】**をクリックし、**【ツリーマップ】**視覚化をクリックします。



ツリーマップ

4. **【視覚化の構成】**パネルで、以下に表示されるフィールドとオプションを選択します。

メモ

【視覚化の構成】パネルで、**【構成】**  をクリックします。



5. **適用**]をクリックします。

ツリーマップ視覚化が生成されます。

ツリーマップで個別のボックスの上にマウスカーソルを置き、埋め込まれたデータを表示できます。

ブラウザ ウィンドウのサイズを変更する場合は、ボックスを再配置し、関連付けられた説明の異なるセクションを表示または非表示にすると、動的に更新されます。

視覚化の保存

作成する視覚化を保持する場合は、保存する必要があります。各視覚化、および解釈と呼ばれる視覚化を格納するコンテナを個別に保存する必要があります。

手順

はじめに

1. ツリーマップ視覚化の左上端で **無題**] をクリックし、**取引ツリーマップ**などの視覚化のタイトルを入力し、Enter を押します。
2. **保存 > 名前を付けて保存**] をクリックします。
3. **タイトル** フィールドで、**チュートリアル視覚化**などの解釈の名前を入力し、**保存**] をクリックします。
解釈と視覚化はいずれも保存され、後からもう一度開くことができます。
4. タイトルバーに表示されているコレクション名、"**ACL チュートリアル**" をクリックすることで、分析コンテナ、**サンプルレポート**に戻ります。
5. **解釈列**の数字をクリックします。**解釈**] ダイアログボックスが表示されます。新しく作成された解釈、**チュートリアルの視覚化**が一覧にあるのが分かります。
各分析コンテナで複数の視覚化と解釈を作成できます。各視覚化は、**テーブルビュー**のデータに基づいています。

ストーリーボードへの公開

今作成した視覚化を表示するためのストーリーボードを作成します。ストーリーボードは、単一のプレゼンテーションで複数の視覚化とリッチテキストコンテンツを表示するコミュニケーションプラットフォームです。

手順

1. [ストーリーボードアプリを開きます](#)。
2. **ストーリーボードを追加**] をクリックします。
3. ストーリーボードのわかりやすいタイトルを入力します。ストーリーボードタイトルは最大 80 文字です。
4. **追加** ] をクリックし、**グラフの追加**] を選択します。 
5. 次のいずれかのオプションを選択します。
 - 解釈からテーブルビューを表示するには、親テーブルのエントリ 、**チュートリアルの視覚化**] を選択します。
 - 解釈から視覚化を表示するには、子のグラフエントリ 、**取引ツリーマップ**] を選択します。利用可能な可視化のリストをフィルター処理するには、検索フィールドにキーワードまたは語句を入力できます
6. 右上隅で **保存 > 保存**] をクリックします。

Analytics の他のレポート(作成)方法

リザルトで使用可能なデータ視覚化の他に、Analytics には、データ分析の結果を報告するために使用できる他の複数の方法があります。

レポート方法	説明
Analytics のレガシー グラフ	Analytics には、レガシー グラフおよびレガシー チャートの作成機能があり、基本的なビジュアルレポートを作成できます。 詳細については、"Analytics グラフを使った作業" ページ 1393を参照してください。
従来の列形式のレポート	一部の場合、データの行と列を使用した従来のテキストと数字に基づくレポート

レポート方法	説明
	<p>のみが必要です。</p> <p>詳細については、「Analytics レポートの書式設定および生成」 ページ 1387を参照してください。</p>
第三者レポート ツール	<p>Tableau または Microsoft BI などのサードパーティレポート ツールを使用して、Analytics から直接データをインポートできます。</p> <p>詳細については、「サードパーティレポート アプリケーションからの Analytics への接続」 ページ 1405を参照してください。</p>
データの Excel または CSV へのエクスポート	<p>データを Excel またはカンマ区切り値ファイルにエクスポートし、Excel または CSV ファイルの操作ができる任意のツールのレポート機能を使用できます。</p> <p>詳細については、「データのエクスポート」 ページ 210を参照してください。</p>

完了しました。

おめでとうございます。Analytics を使用したデータ分析に関する詳細な導入コースを完了しました。

次の学習ステップ

Analytics について学習し続けるための複数のオプションがあります。

アカデミー	<p>アカデミーは、さまざまな経験レベルに合わせて多様なコースを提供します。「Analytics でのデータ分析の基礎」プログラム(ACL 101)は一続きの6つのミニコースであり、新しいユーザー向けに Analytics の基本事項について説明します。</p> <p>アカデミーは、Diligent のオンライントレーニング リソース センターです。コース カタログに移動すると、利用可能なコースを確認できます。</p> <p>サブスクリプションのユーザーは、アカデミー コースを無償で受講できます。</p>
Analytics および ACLScript ヘルプ	<p>現在、Analytics および ACLScript ヘルプを表示しています。このヘルプは、Analytics のすべての側面に関する、リファレンスタイプの概念的な資料、段階的な手順、および ACLScript 構文を提供します。</p> <p>たとえば、完了したチュートリアルの一部である、追加処理に関するヘルプトピックがあります。</p> <ul style="list-style-type: none"> ◦ "テーブルの追加" ページ 928 (概念) ◦ "テーブルを追加する" ページ 938 (段階的な手順) ◦ "APPEND コマンド" ページ 1651 (ACLScript 構文)
コミュニティ	<p>コミュニティは、経験豊富な Analytics ユーザーが専門知識を共有し、質問に回答するカスタマーフォーラムを含む、さまざまなカスタマー リソースを提供する Web プラットフォームです。</p> <p>カスタマーフォーラムは、Analytics の実践的な使用方法と用途について学習するための最適な場所です。</p>

作業のスクリプト化(任意のセクション)

推定時間	20分
要件	以前のスクリプト経験が必要です。
Analytics のバージョン	13.0 以降(非 Unicode 版)

スクリプトを作成せずに、アドホックまたは手動によって、Analytics を使用して、さまざまな価値を得ることができます。ほとんどの場合、スクリプトで実行できる作業は、ユーザー インターフェイスでも実行できません。逆も同じです。ただし、Analytics の価値、能力、および効率を最大化するには、スクリプトを作成する必要があります。

ですが、幸いにも、Analytics には、初心者でもスクリプトを比較的容易に作成できるツールが用意されています。

スクリプトの事例

すべての現在の責任の他に、定期的に法人クレジットカードの使用状況を確認する責任が生じたとします。

時間の節約

基本的なレビュープロセスが標準化されます。各レビュー サイクルでは、時間をかけて手動で基本プロセスを繰り返すか、プロセスを自動化して時間を節約できます。

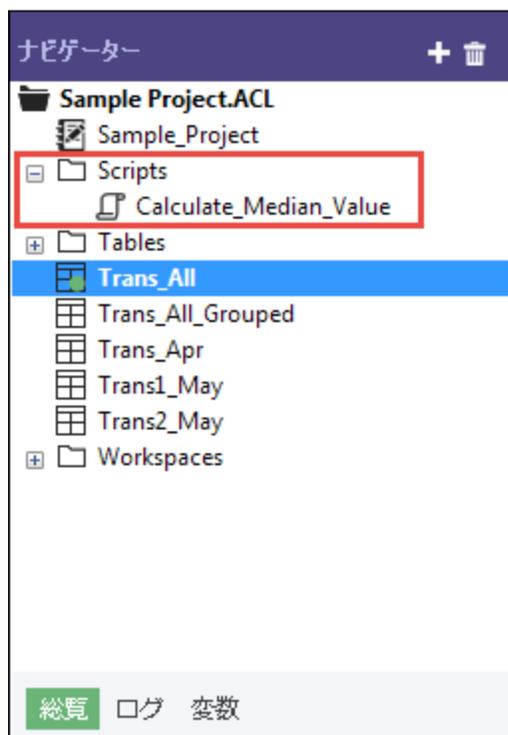
信頼できる委任

プロセスが自動化されると、タスクを部下に委任することができます。テストされたスクリプトを使用すると、経験が少ない従業員でも、作業負担を大幅に増やすことなく、タスクを一貫して正確に実行できることを確信できます。

スクリプトとは何か

Analytics スクリプトとは、特定のタスク、またはいくつかの関連するタスクを実行する、一連の ACLScript コマンドです。たとえば、このチュートリアル前半で手動で実行した作業は、スクリプトを使用して実行することもできます。

ACLScript は Analytics の基礎となるコマンド言語です。スクリプトは Analytics プロジェクトに保存されません。個別のスクリプトはナビゲーターに表示され、先頭にスクリプト アイコン  が表示されます。



Analytics コマンド ログの仕組み

ナビゲーターには、**ログ** タブがあります。スクリプト作成者は、Analytics コマンド ログを頻繁に使用することになります。

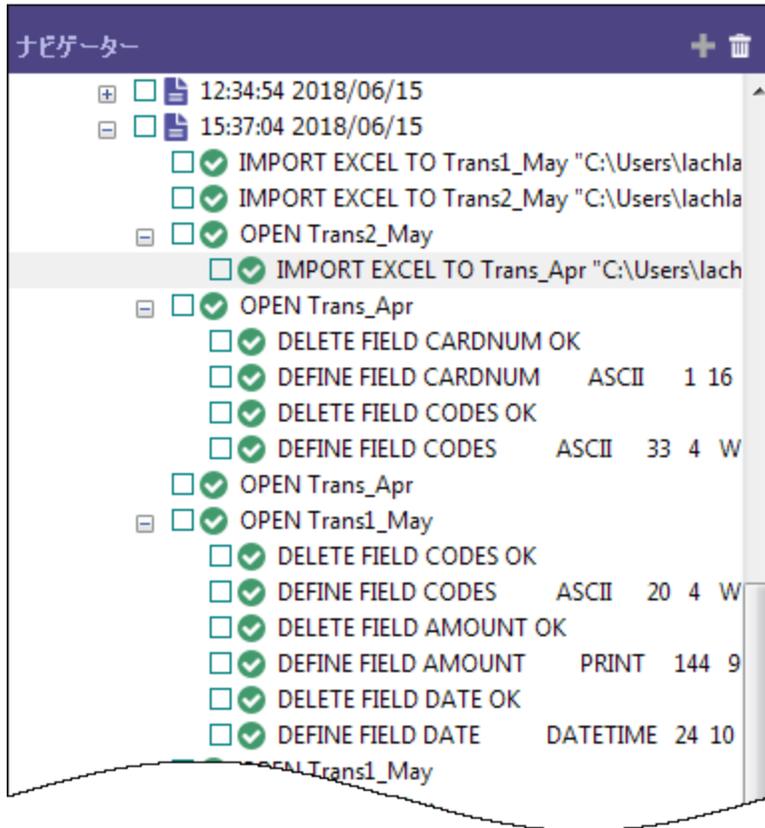
手順

1. **ログ** タブをクリックして開きます。

Analytics コマンド ログが表示されます。**ナビゲーター** パネルをドラッグして広げ、より多くのコンテンツを表示できます。

ログは各 Analytics セッション中に実行された正確な順番のコマンドを記録し、Analytics プロジェクトの一部として保存します。

このチュートリアル前半を完了したばかりの場合は、ユーザー インターフェイスで実行したすべてのアクションのリストがログに含まれています。



2. ログで、結果を新しいテーブルに出力する SUMMARIZE コマンドを検索してクリックします。

SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO "Trans_All_Grouped.FIL" OPEN PRESORT STATISTICS

当該のコマンドが、ツールバーの真下、Analytics のインターフェイスの上部近くにあるコマンドラインにあらかじめ入力されます。

メモ

コマンドラインが表示されない場合は、**[ウィンドウ>コマンドライン]**を Analytics メインメニューから選択します。

3. まだ開いていない場合は、**Trans_All** テーブルを開きます。
4. `f_Prohibited_codes` フィルターが適用される場合は、削除します。
5. **コマンドライン**をクリックし、`"Trans_All_Grouped.FIL"`を`"Trans_All_Grouped_2.FIL"`に変更して、Enter を押します。

要約コマンドが **Trans_All** テーブルで実行され、**Trans_All_Grouped_2** テーブルを出力します。これは、手動で作成した最初の出力テーブルを複製します。

最小限の努力で、**Trans_All** テーブルを要約するために必要なすべての以前の手動作業を再実行しました。コマンドラインからコマンドを実行することは、シンプルな 1 行のスクリプトの実行することに似ています。

ログからコマンドをコピーしてスクリプトを作成する

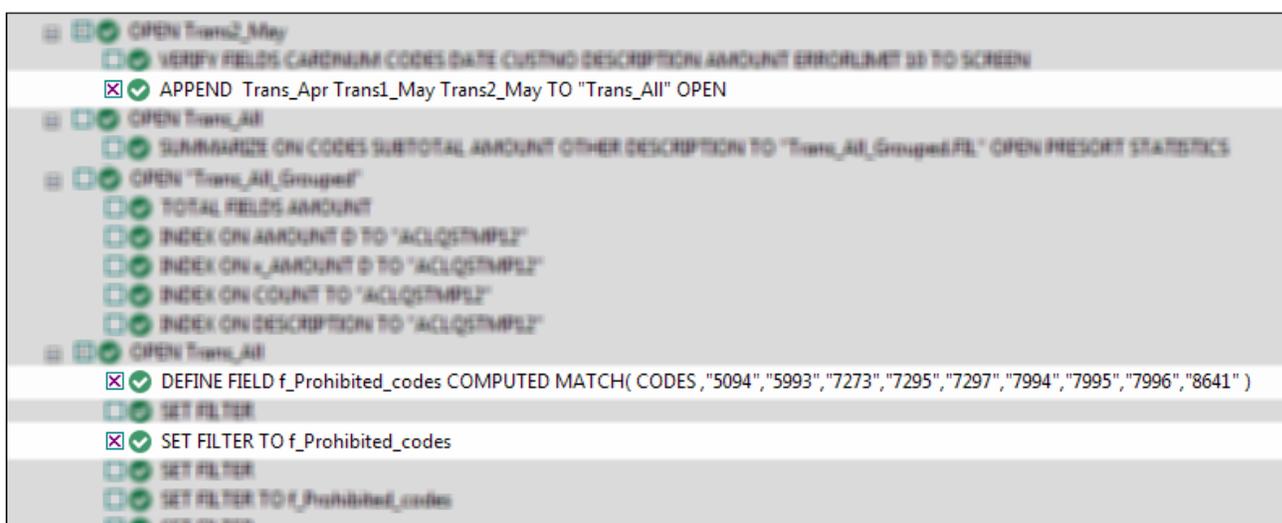
ここでも、ログから ACL スクリプト 構文を再利用できますが、今回は構文を Analytics スクリプトにコピーします。作業を迅速かつ簡単に行うために、チュートリアルで手動で実行した作業の一部のみをスクリプトにしますが、全部の作業をスクリプトにすることもできます。

メモ

このスクリプトの導入を簡潔にするために、一部のスクリプトのベストプラクティスは省略します。目標は、新しいユーザーでも Analytics でスクリプトを簡単に作成できることを示すことです。

手順

1. ログで、次のコマンドを見つけて、選択します。



2. ログを右クリックして **選択項目の保存 > スクリプト**]を選択します。
3. **スクリプトを別名で保存**]ダイアログ ボックスで、スクリプトの名前 `Append_and_filter` を入力して **OK**]をクリックします。
4. ナビゲーターの **概要**]タブで、新しく作成された **Append_and_filter** スクリプトをダブルクリックし、**スクリプト エディター** で開きます。

スクリプトが開き、ログで選択した3つのコマンドの構文全体が含まれています。

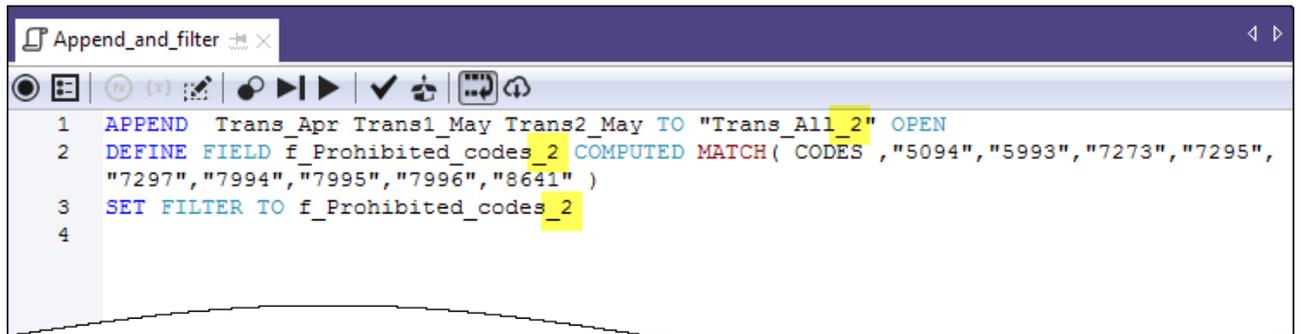
5. 時間を取って、各コマンドの構文を読みます。

ユーザー インターフェイスで前に実行したアクションがどのように ACLScript 構文の個別の部分に対応しているのかがわかりますか。たとえば、`APPEND` コマンドの後には、追加した3つのテーブルの名前があります。

```
Trans_Apr Trans1_May Trans2_May
```

ほとんどの部分では、ACLScript 構文とユーザー インターフェイスのアクション間の対応は比較的わかりやすくなっています。つまり、構文を理解することは、それほど難しくありません。

6. 次の場所に **2** を追加してスクリプトを修正します。



```
1 APPEND Trans_Apr Trans1_May Trans2_May TO "Trans_All_2" OPEN
2 DEFINE FIELD f_Prohibited_codes_2 COMPUTED MATCH( CODES , "5094", "5993", "7273", "7295",
"7297", "7994", "7995", "7996", "8641" )
3 SET FILTER TO f_Prohibited_codes_2
4
```

2 を追加して、既に手動で作成したテーブルとフィルターとの名前の競合を回避します。

7. [スクリプト エディター] ツールバーで、**実行**  をクリックして、スクリプトを実行します。

表示されるすべてのプロンプトで、**[はい]** をクリックします。

スクリプトが実行され、次のタスクを実行します。

- Excel からインポートした3つのテーブルを単一のテーブルの最後に追加し、新しいテーブルを開く
- 禁止されたコード フィルターを作成する
- 新しいテーブルにフィルターを適用する

わかるように、スクリプトの実行は、同じアクションを手動で実行するよりもかなり高速です。毎週や毎月、非常に複雑な分析が実行されるという現実の状況における、時間の節約と一貫性の改善を想像してみてください。

メモ

スクリプトを実行するには、**ナビゲーター**を右クリックし、**実行**]を選択することもできます。スクリプトを実行するには、開く必要はありません。

スクリプトのチュートリアル全体

手動で実行したチュートリアル全体は、以下のスクリプトに手動で表示されます(「手順」セクション)。このスクリプトの導入を完了するには、スクリプトを Analytics にコピーし、チュートリアル作業を元に戻しますが、今回はマウスを数回クリックするだけです。

メモ

スクリプトは、**Sample Data Files** フォルダがデフォルトの場所にインストールされていることを前提とします。フォルダが別の場所にインストールされている場合は、スクリプトのナビゲーションパスを修正し、正しい場所を参照するする必要があります。

スクリプトで作成されたテーブルの末尾には **s** が付くため、手動で作成したテーブルは上書きされません。

手順

新しい空のスクリプトを作成する

1. ナビゲーターの **概要** タブ内の **スクリプト** フォルダーを右クリックし、**新規作成 > スクリプト** の順に選択します。
2. **新しいスクリプト** を右クリックし、**名前の変更** を選択し、`Getting_Started_tutorial` を入力またはコピーして、**Enter** を押します。

チュートリアルスクリプトのコピーと貼り付け

1. 以下の **スクリプトを表示** をクリックします。
2. スクリプト全体をクリックしてドラッグしてから、**Ctrl+C** を押して、スクリプトをコピーします。

メモ

スクリプト全体を選択し、行を落とさないことが重要です。あるいは、[基本チュートリアル\(非 Unicode 版\)](#)

で、スクリプトが含まれたテキスト ファイルをダウンロードできます

3. スクリプト エディター ウィンドウをクリックし、**Ctrl+V** を押して、スクリプト構文を空の `Getting_Started_tutorial` スクリプトに貼り付けます。

スクリプトの更新と保存

1. スクリプトのナビゲーションパスの更新：
 - a. スクリプトの最初の行をクリックします。
 - b. 右クリックして **検索** を選択します。
 - c. **置換** ダイアログボックスで、次のエントリを入力します。
 - **検索対象**： `ユーザー アカウント名`
 - **置換後の文字**： `コンピューターの実際アカウント名`
 - d. すべての `ユーザー アカウント名` のインスタンスで検索と置換を実行します
2. **開いているプロジェクトの保存**  をクリックし、表示されるプロンプトで **はい** をクリックします。
保存アイコンが見つからない場合は、Analytics のメインメニューで **ウィンドウ > ツールバー** をクリックして、ツールバーを有効にします。

スクリプトを実行する

スクリプト エディター ツールバーで、**実行**  をクリックして、スクリプトを実行します。

スクリプトが実行され、すべてのチュートリアル作業を複製します。インタラクティブ通知は、スクリプトの実行時に重要な情報を提供します。

スクリプトを表示

メモ

以前にスクリプトで作業したことがない場合は、初めはスクリプト構文に躊躇することがあります。ほぼすべての構文が Analytics ログからコピーされました。

スクリプトのインタラクティブ通知の構文 (DIALOG コマンド) は、別の比較的シンプルな Analytics ツールによって自動生成されました。

緑色の COMMENT コマンドは、概要レベルで、スクリプトを説明します。前のチュートリアルで完了したタスクを認識します。

```
COMMENT
*** 非 Unicode 版 ***
このスクリプトは、「ACL Analytics の基本」チュートリアルで手動で実行したすべてのアクションを実行します。
END

COMMENT ユーザー確認なしでテーブルの上書きを許可します。
SET SAFETY OFF

COMMENT 3 つの Excel ワークシートをインポートします。

IMPORT EXCEL TO Trans1_May_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans1_May_s.fil" FROM "Trans_May.xls" TABLE "Trans1_
May$" KEEPTITLE FIELD "CARDNUM" C WID 19 AS "" FIELD "CODES" N WID 4 DEC 0
AS "" FIELD "DATE" D WID 19 PIC "YYYY-MM-DD hh:mm:ss" AS "" FIELD "CUSTNO" C
WID 6 AS "" FIELD "DESCRIPTION" C WID 95 AS "" FIELD "AMOUNT" C WID 9 AS ""

IMPORT EXCEL TO Trans2_May_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans2_May_s.fil" FROM "Trans_May.xls" TABLE "Trans2_
May$" KEEPTITLE FIELD "CARDNUM" C WID 19 AS "" FIELD "CODES" N WID 4 DEC 0
AS "" FIELD "DATE" D WID 19 PIC "YYYY-MM-DD hh:mm:ss" AS "" FIELD "CUSTNO" C
WID 6 AS "" FIELD "DESCRIPTION" C WID 155 AS "" FIELD "AMOUNT" C WID 9 AS ""

IMPORT EXCEL TO Trans_Apr_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans_Apr_s.fil" FROM "Trans_April.xls" TABLE "Trans_
Apr$" KEEPTITLE FIELD "CARDNUM" N WID 16 DEC 0 AS "" FIELD "AMOUNT" N WID 6
DEC 2 AS "" FIELD "DATE" D WID 10 PIC "YYYY-MM-DD" AS "" FIELD "CODES" N WID
4 DEC 0 AS "" FIELD "CUSTNO" C WID 6 AS "" FIELD "DESCRIPTION" C WID 45 AS
""

COMMENT 新しいアナリティクス テーブルのテーブル レイアウトを調整します。

OPEN Trans_Apr_s
DELETE FIELD CARDNUM OK
DEFINE FIELD CARDNUM ASCII 1 16 WIDTH 19
DELETE FIELD CODES OK
DEFINE FIELD CODES ASCII 33 4 WIDTH 7

OPEN Trans1_May_s
```

```
DELETE FIELD CODES OK
DEFINE FIELD CODES ASCII 20 4 WIDTH 7
DELETE FIELD AMOUNT OK
DEFINE FIELD AMOUNT PRINT 144 9 2 WIDTH 9
DELETE FIELD DATE OK
DEFINE FIELD DATE DATETIME 24 10 PICTURE "YYYY-MM-DD" WIDTH 27

OPEN Trans2_May_s
DELETE FIELD CODES OK
DEFINE FIELD CODES ASCII 20 4 WIDTH 7
DELETE FIELD AMOUNT OK
DEFINE FIELD AMOUNT PRINT 204 9 2 WIDTH 9
DELETE FIELD DATE OK
DEFINE FIELD DATE DATETIME 24 10 PICTURE "YYYY-MM-DD" WIDTH 27

COMMENT インポートされたデータを検証し、ユーザー通知を表示します。

OPEN Trans_Apr_s
VERIFY FIELDS CARDNUM AMOUNT DATE CODES CUSTNO DESCRIPTION ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans_Apr_
s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans_Apr_
s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

OPEN Trans1_May_s
VERIFY FIELDS CARDNUM CODES DATE CUSTNO DESCRIPTION AMOUNT ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans1_
May_s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans1_
May_s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

OPEN Trans2_May_s
VERIFY FIELDS CARDNUM CODES DATE CUSTNO DESCRIPTION AMOUNT ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans2_
May_s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans2_
May_s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

COMMENT Badfile テーブルを検証し、ユーザー通知を表示します。
OPEN Badfile
VERIFY FIELDS InvoiceNo Prodno Price OrderQty ShipQty Total ERRORLIMIT 10
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
```

```
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Badfile
テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Badfile
テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )
CLOSE

COMMENT 3 つの新しいアナリティクス テーブルを単一の結合されたテーブルの最後に追加します。
APPEND Trans_Apr_s Trans1_May_s Trans2_May_s TO "Trans_All_s" OPEN
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 630 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 500 12 DEFAULT 1 ) (TEXT TITLE "結合された取引テーブル
(Trans_All_s) には %WRITE1% レコードがあります" AT 12 28 )

COMMENT 結合されたテーブルをマーチャント カテゴリ コードでグループ化します。
SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO "Trans_All_Grouped_
s.FIL" OPEN PRESORT STATISTICS
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 700 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 570 12 DEFAULT 1 ) (TEXT TITLE "グループ化された取引テーブ
ル (Trans_All_Grouped_s) には %WRITE1% マーチャント カテゴリ コードがあります" AT 12 28
WIDTH 550 )

COMMENT 結合されたテーブルをフィルタリングし、禁止された取引のみを表示します。
OPEN Trans_All_s
DEFINE FIELD f_Prohibited_codes COMPUTED MATCH(CODES, "5094", "5993",
"7273", "7295", "7297", "7994", "7995", "7996", "8641")
SET FILTER TO f_Prohibited_codes

COMMENT 正常終了メッセージ。
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "スクリプトは正常に完了しま
した" AT 12 28 )

COMMENT テーブルを上書きする前に、ユーザー確認が必要です。
SET SAFETY ON
```

完了しました。

これで、スクリプトの簡単な導入が終わりました。スクリプトの価値を理解し、さらに学習する意欲が出てきたかと思います。

次の学習ステップ

Analytics でのスクリプト作成について学習できる選択肢がいくつか用意されています。

オプション	役立つ情報
チュートリアル	<p>Analytics ヘルプには、次の初級者向けチュートリアルがあります。</p> <ul style="list-style-type: none"> ◦ "初心者向けのスクリプト" ページ 1432 ◦ "Analytics でのスクリプト作成の基本事項" ページ 1450 ◦ "関数を使用する方法" ページ 1477 <p>ヘルプには、すべての ACLScript 言語リファレンスがあり、すべての Analytics コマンド および 関数 について詳細に説明しています。</p>
アカデミー	<p>アカデミーは、導入および高度なスクリプト コースの両方を提供します。</p> <ul style="list-style-type: none"> ◦ Analytics のスクリプトの基本 (ACL 106) ◦ Analytics のスクリプト (ACL 303) <p>アカデミーは、Diligent のオンライントレーニング リソース センターです。コース カタログに移動すると、利用可能なコースを確認できます。</p> <p>Analytics サブスクリプションのユーザーは、アカデミー コースを無償で受講できます。</p>
コミュニティ	<p>コミュニティは Web ベースのプラットフォームであり、Analytics スクリプトの作成について頻繁かつ詳細に議論されているカスタマー フォーラムなど、さまざまなカスタマー リソースがあります。</p>

Analytics(Unicode 版) の基本

この基本チュートリアルでは、Analytics を使用して、データを分析するエンドツーエンドのプロセスについて概要を説明します。

推定時間	60 分
要件	これまでに Analytics を使用したことがなくても構いません。一部の基本データ分析経験があることが前提ですが、絶対に必須であるわけではありません。
Analytics のバージョン	13.0 以降 (Unicode 版)
チュートリアルの正しいバージョンを実行する	Unicode 版の Analytics を使用している場合は、このバージョンのチュートリアルを実行してください。 非 Unicode 版を使用している場合は、"Analytics の基本操作 (非 Unicode 版)" ページ 37 を実行してください。

ヒント

使用している Analytics のエディション(版) を確認するには、Analytics のメイン ページで、[\[ヘルプ > バージョン情報\]](#) に移動し、[\[アナリティクス\]](#) ダイアログボックスを開きます。バージョン番号の後に版の指定が表示されます。

メモ

中国語および日本語のユーザー インターフェイスは Unicode のみです。

シナリオ

法人クレジットカード取引の確認

2 か月間の法人クレジットカード取引を確認する必要があります。目標は、この期間中に従業員がどのようにカードを使用したのかについて概要を認識し、カードの悪用の可能性を特定することです。

取引データは 3 つの Excel ワークシートにあります。データを分析する前に、それを Analytics にインポートし、個別のデータセットを単一の Analytics テーブルに結合する必要があります。

データを分析した後、分析結果を視覚的に表示し、オーディエンスとの対話を効果的にすることができます。

任意のセクション

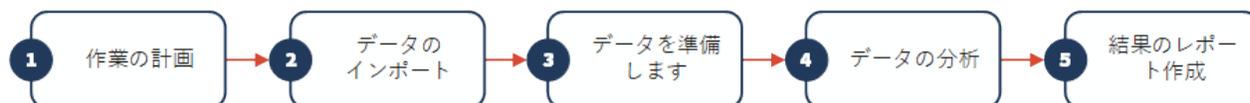
これから、法人クレジットカード取引の確認を繰り返し行うことが義務付けられました。

自分または他の誰かがすばやく正確に将来のレビューを実行できるようにするために、作業の一部を自動化するスクリプトを作成することにしました。

PIPAR - Analytics のデータ分析サイクル

Analytics のデータ分析には、5つの段階があります。これは略語のPIPARで表されます。

Plan(計画)、Import(インポート)、Prepare(準備)、Analyze(分析)、Report(レポート)



作業の計画

データ分析作業の計画は非常に重要です。計画ステージを省略し、データに対して分析コマンドを実行すると、問題の発生、追加作業、または重要な分析インサイトの喪失が生じるおそれがあります。

基本計画でも、計画がないよりはよいので、お勧めします。Analyticsに関するお客様の経験と知識が増えるとともに、計画をより詳細かつ正確に作成できるようになります。優れた計画は、スムーズかつ効率的に進むデータ分析プロジェクトにとって主要な要素です。

計画のガイドライン

明確で具体的な目標を策定する

分析で想定される最終製品は何ですか。

実現方法を計画するには、明確に定義された目標が必要です。たとえば、このチュートリアルでは、具体的な目標は次のとおりです。

- 各マーチャント カテゴリの法人クレジットカード取引のカウントおよび合計金額を特定する
- 禁止されたカテゴリの取引を特定する

段階的なアプローチをマッピングする

目標を達成する方法

多くの場合、目標を達成するには、複数の手順が必要になるため、目標に合わせて、詳細で段階的なアプローチをマッピングします。

たとえば、このチュートリアルの計画の手順のうちの2つは次のようになります。

はじめに

- すべての個別の取引ファイルを単一のファイルに結合する
- 結合された取引データをマーチャント カテゴリにグループ化する

大きい目標を個別の手順に分解した後には、各手順を実行するために使用する Analytics の特徴および機能を検討することができます。

必要なデータを特定する

目標を達成するには、どのデータが必要ですか。

必要なソース データを、特定のデータ要素またはフィールドのレベルに細分化します。適切な入力がないければ、目的の出力を得ることはできません。

このチュートリアルには、主な取引ファイルがありますが、2 番目の目標を達成するには、禁止されたマーチャント カテゴリコードのリストが必要になります。

技術要件を考慮する

考慮する必要がある技術的な考慮事項はありますか。

データ分析で使用しているツールに関係なく、制約内で作業する必要があります。ソース データは、ツールがアクセスできる場所やシステムに格納され、読み取れる形式ですか。提案している分析はルールによってサポートされますか。

たとえば、Analytics で複数のテーブルを結合するためには、各テーブルの対応するフィールドのデータ型が同じである必要があります。Analytics は、フィールドのデータ型の変更をサポートしますが、計画で考慮する必要がある手順です。

繰り返す準備をする

進めるにつれ、計画を調整しなければならない場合があります。

分析の過程で、さらなる調査を保証する、予期しない事態に遭遇します。追加のデータと追加のアナリティクス手順が必要であることを認識します。

データの理解が深まるにつれ、計画も進化します。これにより、類似した本質となる将来の分析のために、より成熟した計画の基礎となることができます。

データのインポート

分析する前に、データを Analytics にインポートする必要があります。

データ定義ウィザードを使用して、3 つの Excel ワークシートにインポートすることで、インポート処理に慣れていきます。Excel からのインポートは、Analytics で分析するためのデータを取得するための最も一般的な方法の 1 つです。ただし、Analytics は、さまざまなデータソースからのデータのインポートをサポートします。

Analytics を開始し、"Sample Project.ACL" を開く

メモ

以下の手順では、Analytics をアクティブ化済みであることが前提です。

手順

1. デスクトップにある **ACL for Windows** ショートカット をダブルクリックします。
2. **ACL for Windows** 画面の **開く**]の下で、**Analytics プロジェクト**]をクリックします。
3. **C:\Users\user_account_name\Documents\ACL Data\Sample Data Files** に移動し、**Sample Project.ACL** をダブルクリックします。Analytics 内に **Sample Project.ACL** が開かれます。

Analytics をインストールしたときに **Sample Data Files** フォルダをデフォルトの場所にインストールしていない場合は、インストールした場所に移動します。

2 つの Excel ワークシートのインポート

まず、2 つの Excel ワークシートを同時にインポートします。複数の Excel ワークシートを同時にインポートすると、作業の手間を大幅に削減できます。

手順

1. Analytics のメインメニューから、**インポート > ファイル**]を選択します。
2. **定義するファイルの選択**]ダイアログボックスで、**Trans_May.xls** を選択し、**開く**]をクリックします。

Excel ファイルは **Sample Project.ACL** と同じフォルダにあります。

3. **ファイル形式**]ページで、**Excel ファイル**]オプションが選択されていることを確認し、**次へ**]をクリックします。
4. **データソース**]ページで、ファイルの両方のワークシートを選択します。
 - **Trans1_May\$**
 - **Trans2_May\$**
5. **先頭の行をフィールド名として使用する**]が選択されていることを確認し、**次へ**]をクリックしてから、**完了**]をクリックします。

2 つの Excel ワークシートが 2 つの別の Analytics テーブルにインポートされます。

3 番目の Excel ワークシートのインポート

3 番目の Excel ワークシートをインポートします。単一のワークシートをインポートするときには、一部のメタデータ設定を、後から Analytics で調整するのではなく、インポート処理中に手動で調整するオプションがあります。

手順

1. 直前の手順にあるすべてのステップを繰り返すことで、**Trans_April.xls** を見つけて選択します。
2. **ファイル形式**]ページで、**Excel ファイル**]オプションが選択されていることを確認し、**次へ**]をクリックします。
3. **データソース**]ページで、**Trans_Apr\$** を選択します。
4. **先頭の行をフィールド名として使用する**]が選択されていることを確認し、**次へ**]をクリックします。
5. **Excel インポート**]ページで、ヘッダーをクリックして、**TRANS_DATE** 列を選択し、次の変更を行います。
 - **名前**]フィールドで、**TRANS_DATE** を **DATE** に変更します。
 - **長さ**]フィールドで、**19** を **10** に変更します。

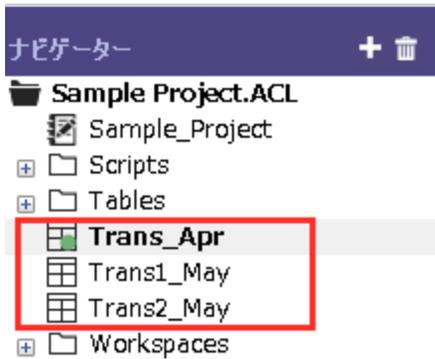
メモ

インポート処理中には、データ定義ウィザードでデータフィールドを調整します。データのインポートが完了した後に、後から調整することもできます。チュートリアル次のセクションでは、調整の理由について説明します。

6. [ファイル名]フィールド型 Trans_Apr で 次へ] をクリックし、保存] をクリックします。
7. 完了] をクリックしてから、OK] をクリックします。

3番目のExcelワークシートはAnalyticsテーブルにインポートされます。

ナビゲーターの 概要] タブに、新しい3つのAnalyticsテーブルが表示されます。これらのテーブルには、Excelデータの読み取り専用コピーが含まれます。Excelソースデータ自体は含まれません。



データの準備

多くの場合、データの分析準備が整う前に、1つ以上のデータ準備タスクを実行する必要があります。このチュートリアルでは、次の2つの準備タスクを実行します。

- 追加の調整を行い、データフィールドを合わせる
- 分析のために、3つの新しいAnalyticsテーブルを単一のテーブルに結合する

また、ベストプラクティスとして、分析作業を実行する前に、インポートされたデータの有効性を必ず確認することをお勧めします。テーブルに無効なデータが少しあるだけでも、すべての後続のデータ分析が無効になる可能性があります。

データの準備が必要な理由

データの分析を進めようとしても、適切なデータ準備を行わないと、分析を実行できない場合があります。あるいは、実行する分析に欠陥が生じる可能性があります。

何らかの初期準備がないと、さまざまな問題がソース データに影響し、分析に適さなくなる可能性があります。

例：

- ソース データは複数のファイルに分散しているため、単一のデータのセットとして分析できるように、結合する必要があります。
- 異なるファイルの対応するフィールドは、「調整」する必要があります。つまり、処理するための前提条件として、同じ構造と形式にします。
- 「汚いデータ」はクレンジングして正規化する必要があります。これは、Analytics 関数を使用して実行できます。

要点

データのインポートと準備にかかる時間は、実際の分析にかかる時間を超えることがあります。ただし、これらは重要な初期段階であり、分析の基になる基盤を提供します。

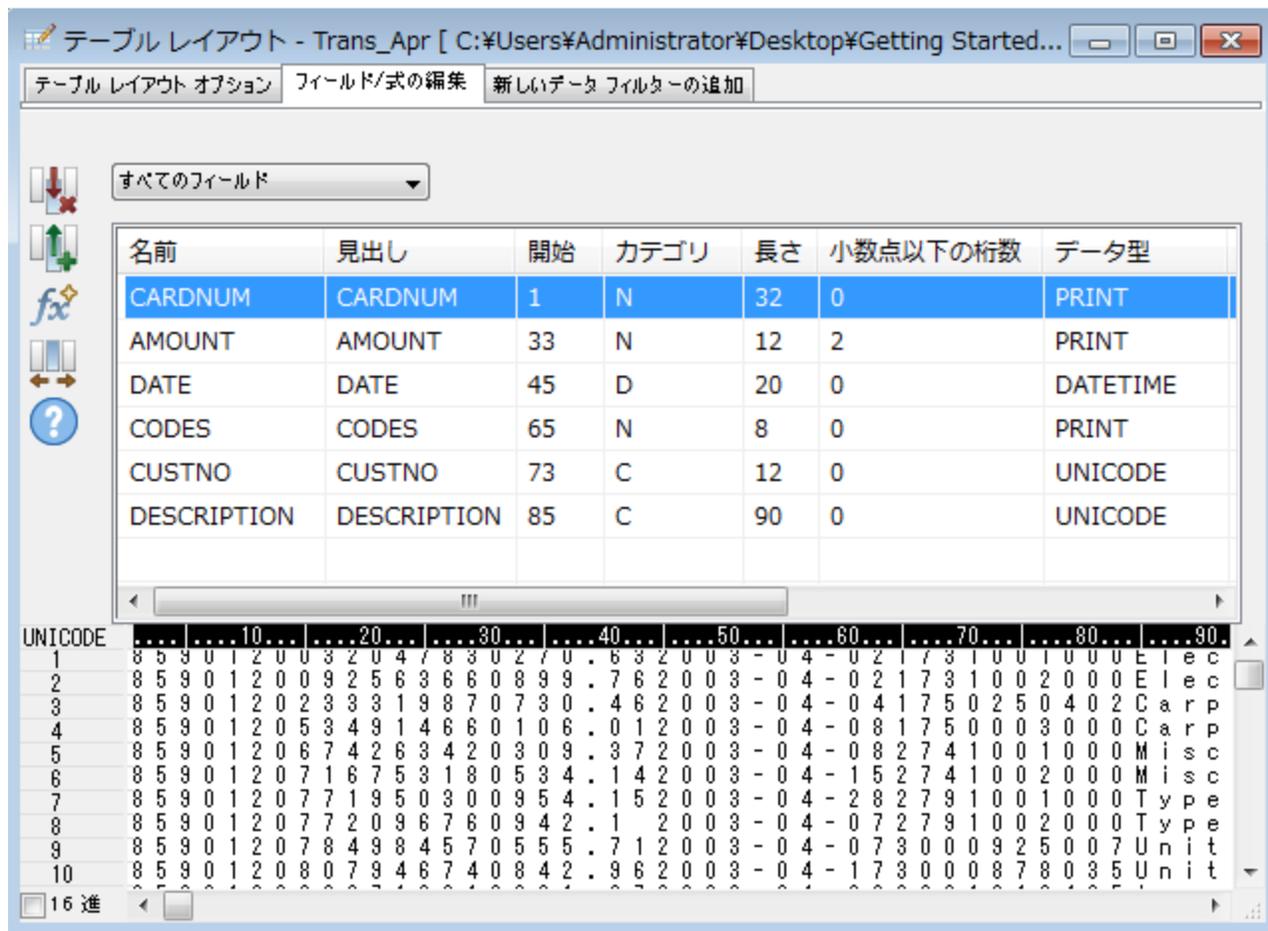
テーブルレイアウトの調整

Analytics プロジェクトのすべてのテーブルには**テーブルレイアウト**があります。テーブルレイアウトは、フィールド名、フィールドの開始位置、フィールドの長さ、フィールドのデータ型などのメタデータを含んでいます。

3つの新しい Analytics テーブルを単一のテーブルに結合する前には、テーブルレイアウトの一部のメタデータを調整する必要があります。

ここでは、**Trans_Apr** テーブルレイアウトの内容を示します。Analytics に慣れるにつれ、テーブルレイアウトの操作方法をすぐに学ぶことができます。テーブルレイアウトには、多数の便利な点があります。

はじめに



Trans_Apr テーブルレイアウトの調整

まず、Trans_Apr テーブルの2つのフィールドのデータ型を変更する必要があります。

手順

1. まだ開いていない場合は、Trans_Apr テーブルを開きます。
テーブルを開くには、ナビゲーターでテーブルをダブルクリックします。
2. テーブルビューの上で **テーブルレイアウトの編集**  を選択します。
3. **CARDNUM** フィールドをダブルクリックし、編集用にフィールド定義を開きます。
4. **有効なデータ型** の下で、**UNICODE** をダブルクリックして、フィールドのデータ型を更新します。
他の2つのテーブルでは、**CARDNUM** フィールドのデータ型はUNICODEです。ほとんどの部分では、データを結合するには、結合されるテーブルの対応するフィールドのデータ型が同じである必要があります。
5. **入力を受け入れる**  をクリックします。
プロンプトが表示されたら、**[はい]** をクリックして、変更を保存します。
6. **CODES** フィールドをダブルクリックしてデータ型を **UNICODE** に変更します。

7. 入力を受け入れる  をクリックし、閉じる  をクリックして **テーブルレイアウト** ダイアログボックスを終了します。

Trans_May テーブルレイアウトの調整

調整を完了するには、Trans1_May および Trans2_May レイアウトの両方の2つのフィールドのデータ型を変更する必要があります。DATE フィールドの調整も必要な場合があります。

手順

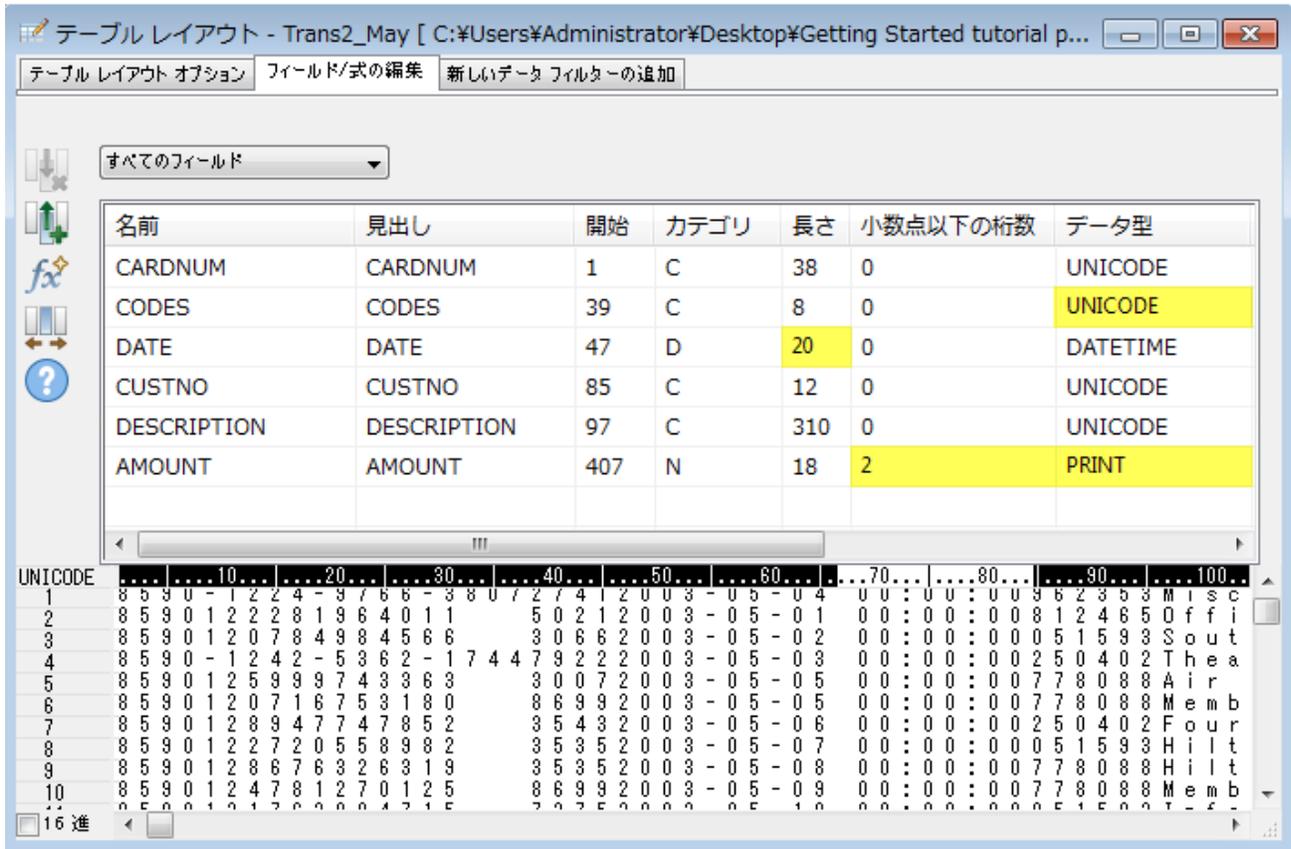
上記のプロセスに従い、Trans1_May および Trans2_May レイアウトの両方で、次の変更を行います。

フィールド	次のデータ型に変更します。	追加の変更
CODES	UNICODE	
AMOUNT	PRINT	小数点 フィールドに 2 を入力し、数値に2桁の小数点が表示されることを指定します。
DATE	変更なし	<p>メモ</p> <p>DATE フィールドの長さが既に20の場合は、調整は不要です。</p> <ul style="list-style-type: none"> ○ 長さフィールドで、38を20に変更します。この変更により、空の時刻データが省略されます。 ○ 書式設定ドロップダウンリストで、YYYY-MM-DDを選択します。

完了すると、May テーブルレイアウトが以下のレイアウトのようになります。

メモ

日付形式 (YYYY-MM-DD) はレイアウト サマリーに表示されません。DESCRIPTION フィールド長は、2つのMayレイアウトで異なります。



インポートされたデータの検証

では、3つのインポートされたテーブルのデータを検証し、追加のデータ準備、およびデータ分析を進めても問題がないことを確認しましょう。

メモ

データ型を更新した後に、データを検証しています。Analyticsでデータを検証するときには、フィールドのすべての値がフィールドのデータ型の要件に適合することを確認しています。このため、データ型が確定した後にのみデータを検証することに意味があります。

手順

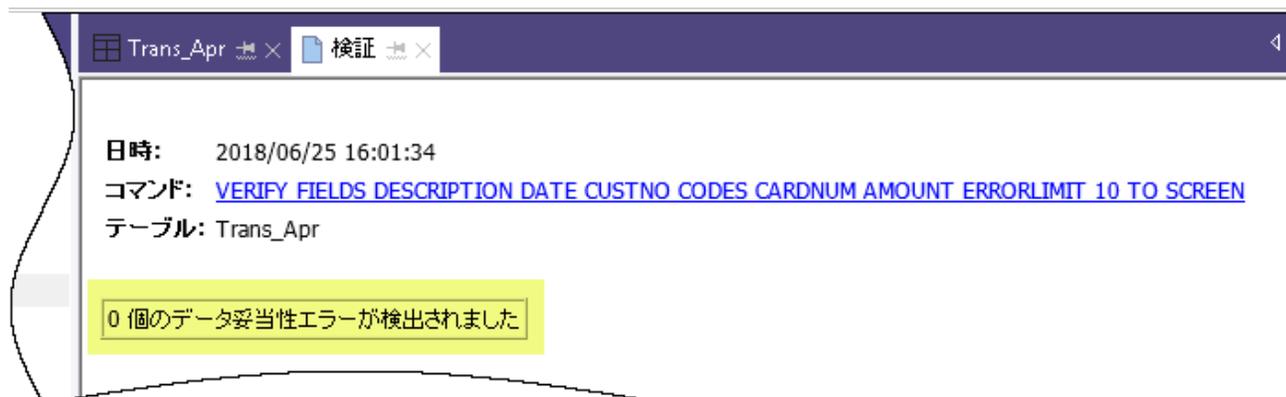
1. Trans_Apr テーブルを開きます。
2. Analytics のメインメニューから、**データ > 検証**] を選択します。
3. **検証**] ダイアログボックスで、フィールド リストのすべてのフィールドを選択します。

ヒント

Shift+click を使用して、複数の隣接するフィールドを選択します。

4. [OK]をクリックします。

結果は次のようになります。0個のデータ妥当性エラーが検出されました。



詳細

お客様がユーザーインターフェイスで実行したアクションが自動的にACLScript `VERIFY` コマンドに変換されたことに気が付きましたか。ユーザーインターフェイスで実行するすべてのコマンドレベルのアクションは、自動的に対応するACLScriptコマンドに変換され、各 Analytics プロジェクトに付属するコマンドログに取り込まれて格納されます。

このように、有効で実行可能なスクリプト構文の自動生成は、最も強力な Analytics 機能の1つです。チュートリアル最後のオプションのセクションで、スクリプトを確認します。

5. ナビゲーターで、Trans1_May テーブルをダブルクリックして開き、手順を繰り返して、データを検証します。
6. Trans2_May テーブルでも同じことを実行します。

両方のテーブルには、データ検証エラーは含まれません。

メモ

最大エラー上限に達しましたというエラーメッセージが表示される場合は、テーブルレイアウトで日付フィールドの形式を正しくYYYY-MM-DDに変更したことを確認します。

詳細表示

Analytics がデータ検証エラーを特定するときの処理を確認する場合は、Tables\Badfile を開き、検証プロセスを実行します。

3つのAnalytics テーブルを結合する

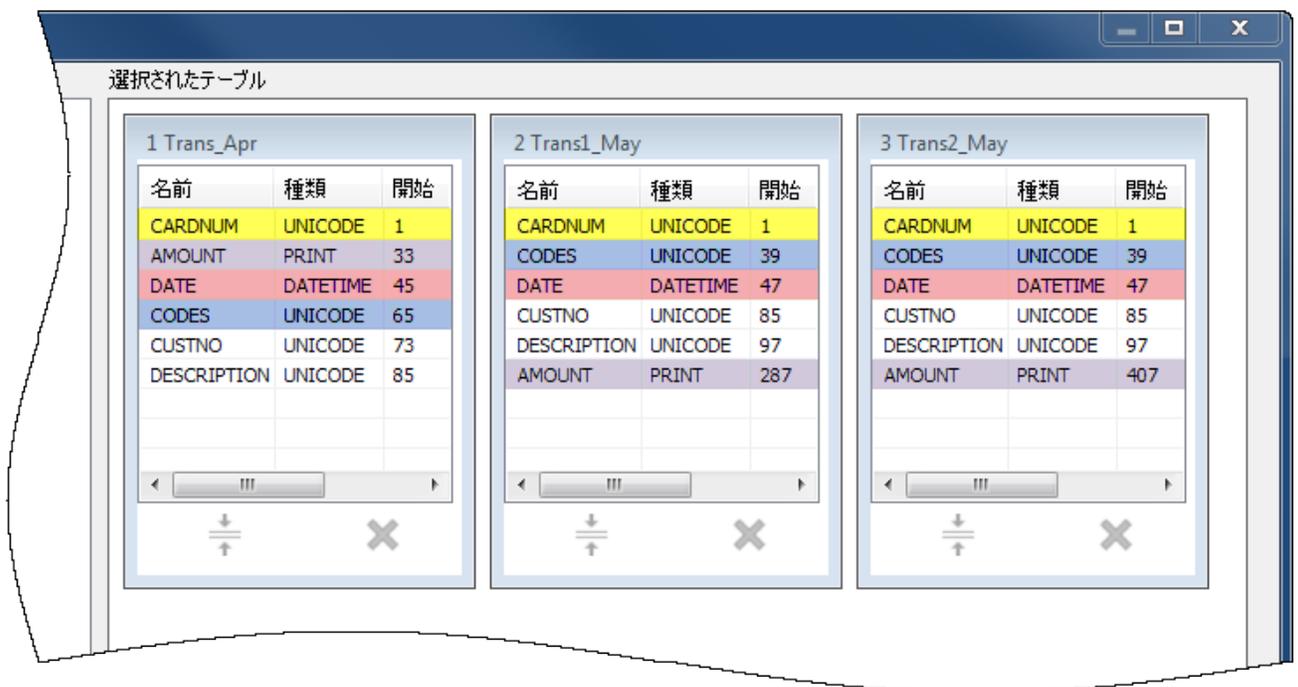
最終データ準備タスクでは、3つの新しいAnalytics テーブルを単一のテーブルに結合します。

簡単にするため、チュートリアルは3つのテーブルのみを結合します。ただし、同じプロセスを使用して、12個の月次テーブルを単一の年次テーブルに結合し、年度全体のデータに対して分析を実行することができます。

手順

1. Analytics のメインメニューから、**データ > 最後に追加**]を選択します。
2. **使用可能なテーブル**]の下で、各新しいテーブルをダブルクリックし、**選択したテーブル**]領域に追加します。
3. 3つのテーブルのフィールドを確認し、名前とデータ型が、データ定義ウィザードと **テーブルレイアウト**]ダイアログボックスで行った変更に基づいてどのように同じになっているのかを見てください。

対応するフィールドを後に追加するには、名前が同じであり、ほとんどの状況でデータ型も同じである必要があります。



4. **出力テーブルの使用**]を選択し、コマンドの実行後に、結合されたデータがあるシャットカテーブルが自動的に開くようにします。
5. **To**]フィールドで、**Trans_All**と入力し、**OK**]をクリックします。
6. ポップアップ通知で、**はい**]をクリックします。

メモ

通知については気にする必要はありません。追加コマンドは、数値フィールドの一部の自動調整を実行します。これにより、時間と作業工数を削減できます。

新しい **Trans_All** テーブルが作成され、3つの入力テーブルのすべてのレコードが含まれます。
Analytics のインターフェイスの下部にある状態バーのレコード数が、**レコード: 481** となります。

これで、実際のデータ分析に進むことができます。

データの分析

コマンドおよび他のツールを使用して、Analytics で分析を実行し、調査しているデータに関する一般的な洞察を得て、特定の問題に対して回答します。

メモ

分析段階では、前の計画の強度が明らかになります。調査に関する明確な目標を策定している場合は、実行する分析の種類について明確な考えを持つことになります。

データ分析

このチュートリアルでは、**Trans_All** テーブルのデータについて次の分析を実行します。

- 次の点を検出するために、マーチャント カテゴリコードによって、クレジットカード取引レコードをグループ化します。
 - 従業員が法人クレジットカードを使用する方法
 - 各カテゴリで費やされる金額
- フィルターを作成し、禁止された取引を分離する

マーチャント カテゴリコードによって、クレジットカード取引レコードをグループ化する

データのセットをグループ化または要約すると、データの概要をすばやく確認することができます。

手順

1. まだ開いていない場合は、**Trans_All** テーブルを開きます。
2. Analytics のメインメニューから、**分析 > 要約** を選択します。
3. **要約** ダイアログボックスで、次のフィールドとオプションを選択します。

タブ	フィールドまたはオプション	選択または入力
メイン	要約オン	CODES を選択
	その他のフィールド	DESCRIPTION を選択
	小計フィールド	AMOUNT を選択
	平均、最小、最大	チェックボックスを選択

タブ	フィールドまたはオプション	選択または入力
出力	To	ファイルを選択
	名前	タイプ <code>Trans_All_Grouped</code>

4. **OK]**をクリックします。

新しい **Trans_All_Grouped** テーブルが作成されます。テーブルに 110 レコードがあり、**Trans_All** テーブルの各一意のマーチャント カテゴリコードにつき 1 件のレコードがあります。**COUNT** フィールドは、各グループのソースレコード数を示します。

ヒント

テーブルビューを右クリックし、**すべての列のサイズを変更]**を選択して、ビューをよりコンパクトにします。

調査のための簡易ツール

データの要約バージョンがあるため、一部の基本 Analytics ツールを使用して、法人クレジットカード利用についての一般的な洞察を得ることができます。

数回クリックするだけで、使用パターン、悪用の可能性について、多くの情報が得られます。

この洞察を得るには	Trans_All_Grouped テーブルでこれを実行します。
4 月および 5 月中の従業員の合計請求金額はいくらでしたか。	<ul style="list-style-type: none"> ◦ Total AMOUNT ヘッダーを選択します。 ◦ 分析 > 合計]の順に選択します。 <p>支出合計は \$187,177.13 でした。</p>
従業員が最もカードを使ったのはどこでしたか。	<ul style="list-style-type: none"> ◦ [Total AMOUNT]ヘッダーを右クリックして クイックソート 降順]を選択します。 <p>DESCRIPTION]フィールドは、次の項目に最も多い金額が費やされたことを示します。</p> <ul style="list-style-type: none"> ◦ Caterers ◦ Eating places and Restaurants ◦ Hilton International
最も大きい単一の支出は何か。	<ul style="list-style-type: none"> ◦ Maximum AMOUNT]ヘッダーを右クリックして クイックソート 降順]を選択します。 <p>DESCRIPTION]および Maximum AMOUNT]フィールドは、最も大きい単一の支出は Club Med の \$1999.06 であることを示しています。</p> <p>Club Med は法人クレジットカードで許可されたマーチャント コードか。クレジットカード上限が \$2000 の場合、従業員の請求金額は上限以下か。</p>
使用頻度が低いコードの検査で何が明らかになるか。	<ul style="list-style-type: none"> ◦ COUNT]ヘッダーを右クリックして クイックソート 昇順]を選択します。 <p>5 つのカテゴリにはそれぞれ 1 つの請求のみがありました。禁止されたカテゴリはあるか。おそらく、1 人または複数の従業員は、法人カードをごくまれに誤用しても、検出を逃れることができると考えているかもしれません。</p> <ul style="list-style-type: none"> ◦ Cigar Stores & Stands ◦ Dating & Escort Svcs. ◦ Babysitting services ◦ Amusement Parks

この洞察を得るには	Trans_All_Grouped テーブルでこれを実行します。
	<ul style="list-style-type: none"> ○ Civic, Fraternal, and Social Associations
禁止されたカテゴリはあるか。	<ul style="list-style-type: none"> ○ DESCRIPTION] ヘッダーを右クリックし、クイックソート 昇順]を選択すると、簡単に精査できるようにフィールド値をアルファベット順に並べ替える ○ 不審なカテゴリを探すためにフィールドを精査する <p>1つ以上のカテゴリが禁止されている可能性があるか。</p> <ul style="list-style-type: none"> ○ Babysitting services ○ Betting (including Lottery Tickets, Casino) ○ Civic, Fraternal, and Social Associations ○ Dating & Escort Svcs. ○ Massage Parlors ○ Precious Stones and Metals, Watches and Jewel ○ Video Game Arcades/Establishments <p>メモ</p> <p>小さいデータセットを除き、手動精査は実践的ではありません。次に、より実践的で、より信頼できる方法を見ていきます。</p>

詳細表示

何らかの簡易分析のみを実行し、結果を新しいテーブルに出力したくない場合もあるでしょう。**要約**]ダイアログボックスで、**ファイル**]を選択する代わりに、**Trans_All** テーブルを要約すると、**画面**]を選択し、結果を Analytics 表示領域に出力できます。

日時: 2018/06/26 13:50:00
 コマンド: [SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO SCREEN PRESORT](#)
 テーブル: Trans_All

CODES	AMOUNT	カウント	DESCRIPTION
1731	1,486.30	3	Electrical Contractors
1750	1,095.57	3	Carpentry Contractors
2741	1,353.94	3	Miscellaneous Publishing and Printing
2791	2,141.29	3	Typesetting, Plate Making, & Related Services
3000	2,590.70	3	United Airlines
3001	1,276.42	3	American Airlines
3005	1,659.97	3	British Airways
3007	2,616.76	3	Air France
3008	3,497.90	4	Lufthansa
3009	2,680.30	4	Air Canada
3025	4,319.54	4	Air New Zealand

画面への出力は小さいデータセットでのみ実践的です。ただし、個別のグループで簡単にドリルダウンし、各グループのソースレコードのみを表示できるという利点があります。

(CODES = "3000")

CARDNUM	AMOUNT	DATE	CODES	CUSTNO	DESCRIPTION
9 8590120784984570	555.71	2003/04/07	3000	925007	United Airlines
10 8590120807946740	842.96	2003/04/17	3000	878035	United Airlines
389 8590122491455345	1192.03	2003/05/21	3000	503458	United

<< ファイルの終わり >>

フィルターを作成し、禁止された取引を分離する

フィルターを使用すると、特定の時点に関心があるレコードのみを分離できます。フィルターは、データに関する特定の問題に回答するための強力なツールです。

法人クレジットカード取引の一般レビューにより、一部の禁止された取引の可能性が通知されました。そこで、禁止されたマーチャント カテゴリコードの一覧をデータと照合し、禁止された取引があるかどうかを確認することに決めました。

手順

フィルター式を作成する

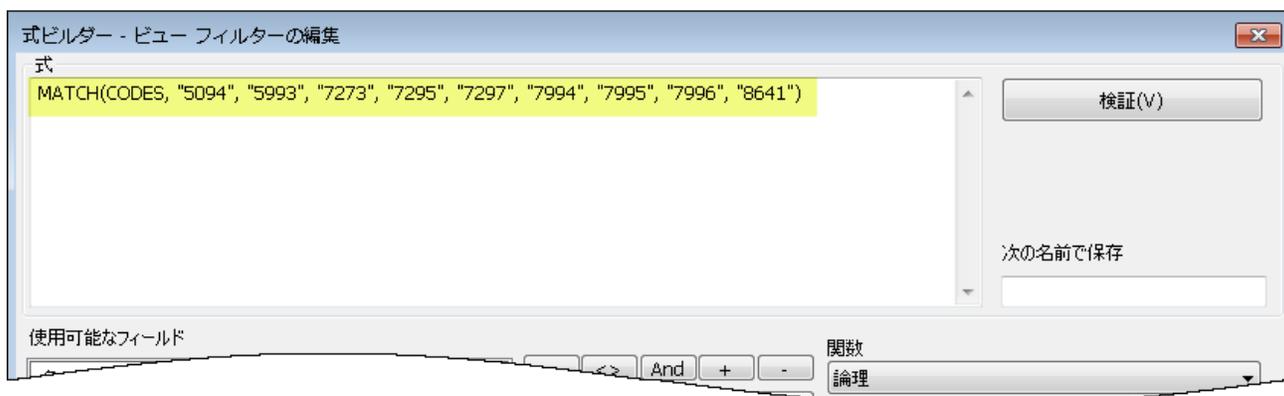
1. Trans_All テーブルを開きます。
2. テーブルビューの上部にある **ビューフィルターの編集**  をクリックして、**式ビルダー**を開きます。
式ビルダーは Analytics コンポーネントであり、式構文を手動で入力するのではなく、マウスを使用して式を作成できます。式は値と演算子を組み合わせただけのもので、これにより計算を実行したり結果を返したりすることができます。
3. **関数**] ドロップダウンリストで、**論理**] を選択してから、MATCH 関数をダブルクリックして、**式**] テキストボックスに追加します。
MATCH を使用して、**CODES** フィールドの複数の禁止されたマーチャント カテゴリコードを分離します。
4. **式**] テキストボックスで、**比較値** プレースホルダーをハイライト表示してから、**使用可能なフィールド**] リストで、**CODES** をダブルクリックします。
CODES フィールドは **比較値** を置換します。
5. 以下の禁止されたコードの文字列をコピーし、`test1 , test2 <,test3...>` プレースホルダーを置換します。

"5094", "5993", "7273", "7295", "7297", "7994", "7995", "7996", "8641"

メモ

すべての引用符を含む文字列全体を必ずコピーします。

式は次のようになります。



式を検証し、フィルターを保存して適用します。

1. **検証]**をクリックし、式の構文が有効であることをテストします
作成してすぐに式を検証することをお勧めします。より時間がかかる後からのトラブルシューティングを回避するうえで役立つためです。
エラーメッセージが表示される場合は、式の構文が正確に上記の構文と一致することをもう一度確認します。
2. **名前を付けて保存]**フィールドでフィルター名 `f_Prohibited_codes` をコピーします。
Diligent は、保存されたフィルターの先頭に `f_` を付けることをお勧めします。
3. **OK]**をクリックします。
`f_Prohibited_codes` フィルターは `Trans_All` テーブルに適用されます。禁止されたマーチャント カテゴリコードを使用する取引が分離され、簡単に確認できます。数万件以上のレコードがあるテーブルを考えると、フィルターの価値はすぐにはっきりとわかります。

フィルターの削除または再適用

フィルターを削除して、再適用してみてください。

1. フィルターを削除するには、**フィルターの削除**  をクリックします。
2. フィルターを再適用するには、次のいずれかを実行します。
 - ビューの上部にある **フィルター履歴]** ドロップダウンリストから、フィルター名を選択します。
 - **ビューフィルターの編集**  をクリックして**式ビルダー**を開き、**フィルター]** リストでフィルター名をダブルクリックして **OK]** をクリックします。

ヒント

フィルター履歴] リストには、最大 10 個のフィルターがあるため、保存されたフィルターを再適用するために、**式ビルダー**を使用しなければならない場合があります。

詳細表示

フィルターよりも強力な機能

フィルターは、フィルターに含まれる基準または条件が管理可能である場合に、効果的に動作します。このチュートリアルで作成したフィルターには、9 コードしかありません。しかし、禁止されたマーチャント カテゴリコードのリストが何十個かそれ以上になった場合はどうなるでしょうか。

より効率的なアプローチは、禁止されたコードを含む Analytics テーブルを取引テーブルと結合することでしょう。結合された出力テーブルの一致でも、禁止された取引になります。

結合はこのチュートリアル範囲外ですが、Analytics では頻繁に使用される機能です。

結果のレポート作成

データ分析が完了すると、Analytics は、複数の方法で、結果を報告または表示します。

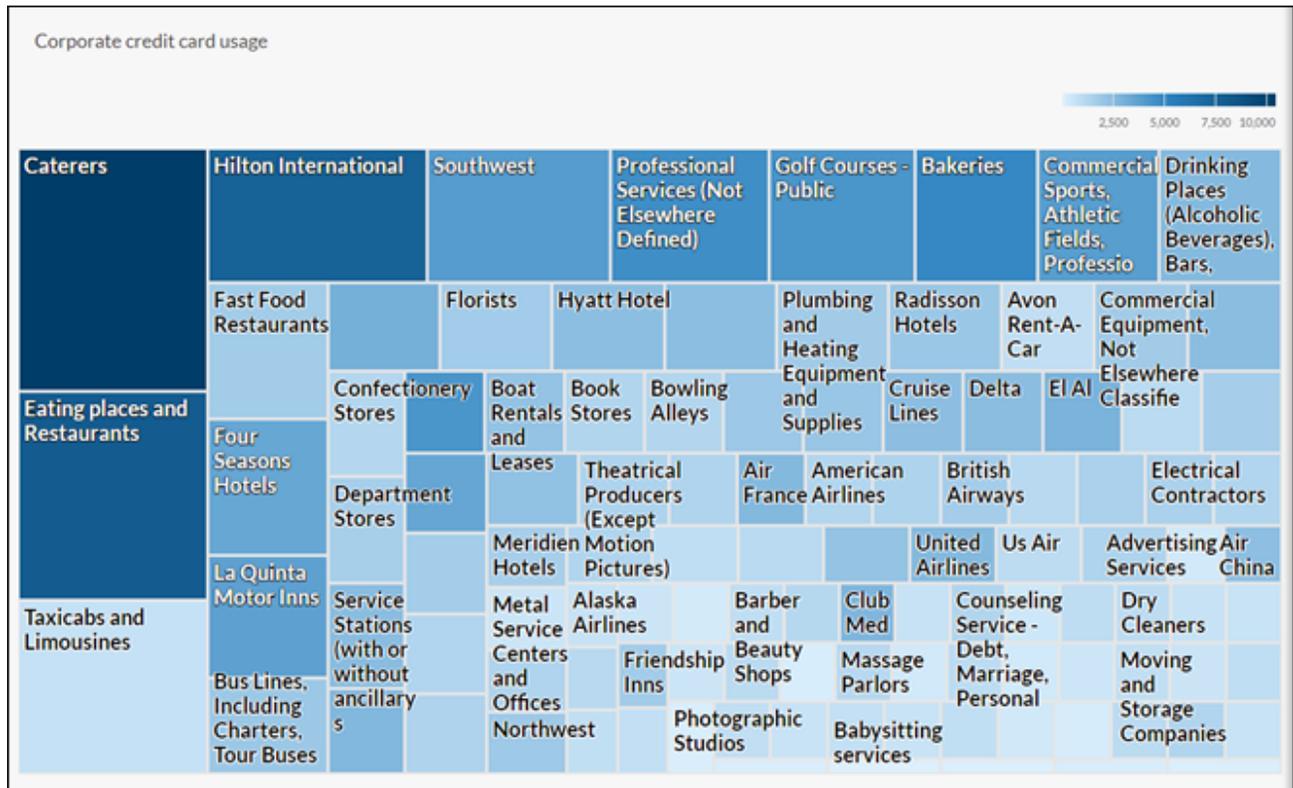
データ列を使用した従来のレポートがありますが、以下に説明する、よりわかりやすいデータ視覚化を使用して結果を伝える方法を確認していきます。

ツリーマップ視覚化

このツリーマップ視覚化は、**Trans_All_Grouped** テーブルで出力するグループ化されたクレジットカード取引を示します。グループ間の関係は、次の2つの方法で表されます。

- **ボックスのサイズ** -は、各グループの個別の取引のカウントを示します
ボックスが大きいほど、取引数が多くなります。ボックスは、左上から右下へ、サイズに配置されます。
- **ボックスの色強度** -は、各グループの合計数を示します
ボックスが濃いほど、合計数が多くなります。

このため、たとえば、右下の四分円にある **Club Med** ボックスのサイズは、取引数が少ないことを示しますが、色は合計の取引金額が大きいことを示します。



準備作業

リザルトでツリーマップ視覚化を作成しようとしています。リザルトは、クラウドベースの HighBond プラットフォームの問題修正アプリです。リザルトのライトバージョンへのアクセスは、ACL Robotics サブスクリプションに含まれます。

視覚化を作成するには、まず、視覚化を格納するための、シンプルな 2 レベルのデータコンテナを作成する必要があります。最初のレベルはコレクション、2 番目のレベルは分析と呼ばれます。すばやく簡単に作成できます。

Launchpad にサインインし、リザルトにアクセスします。

メモ

何らかの理由で Launchpad にサインインできないか、リザルトにアクセスできない場合は、"Analytics の他のレポート(作成)方法" ページ 93 の一覧にある代替レポート作成方法のいずれかを使用できます。

手順

1. Launchpad(www.highbond.com) に移動します。
2. HighBond アカウント資格情報(の電子メールとパスワード) を入力して、**[サインイン]** をクリックします。

Launchpad が開きます。

3. **[リザルト]** をクリックします。

リザルト ホームページが開きます。

メモ

リザルトにアクセスできない場合は、適切なサブスクリプションまたはリザルト ロールが割り当てられていない可能性があります。"Analytics の他のレポート(作成)方法" ページ 93 の一覧にある代替レポート作成方法のいずれかを使用します。

リザルトにアクセスする場合は、会社の Analytics アカウント 管理者に連絡してください。

コレクションの作成

手順

1. リザルト のホームページで、**新しいコレクション]** をクリックします。
2. **新しいコレクション]** ページの **名前]** フィールドで、**ACL チュートリアル** を入力またはコピーします。
3. ページの下部で **コレクションの作成]** をクリックします。

コレクションの設定] ページが表示されます。

分析を作成する

手順

1. **[コレクションの設定]** ページの最下部の **次の手順]** で、**初めてデータ分析を作成する]** をクリックします。

分析の詳細] ページが開きます。

2. **分析の詳細]** ページの **名前]** フィールドで、**サンプル レポート** を入力またはコピーします。
3. **分析の作成]** をクリックします。

新しい ACL チュートリアルコレクションが開き、作成した空の**サンプルレポート**分析が表示されます。

メモ

リザルトを開いたままにします。戻って、データ視覚化を作成します。

Analytics データのリザルト へのエクスポート

次の段階では、**Trans_All_Grouped** テーブルを Analytics からリザルトにエクスポートします。

手順

1. Analytics で、**Trans_All_Grouped** テーブルを開きます。
2. Analytics のメインメニューから、**データ > エクスポート]** を選択します。

はじめに

3. **【エクスポート】**ダイアログボックスで、次のオプションを選択します。

タブ	オプション	選択
メイン	ビュー	ビューを選択
	名前を指定してエクスポート	HighBondを選択

4. **【o】**をクリックし、**【送信先テストの選択】**ダイアログボックスで、作成した**サンプルレポート**分析コンテナに移動し、ダブルクリックして開きます。
5. **【新しいデータアナリティクス】**フィールドで `Trans_All_Grouped` を入力またはコピーし、**【作成】**をクリックします。

【エクスポート】ダイアログボックスに戻ります。ID番号とデータセンターコードは **【o】**テキストボックスに事前に入力されます。

6. **【OK】**をクリックします。
Trans_All_Grouped テーブルがリザルトにエクスポートされます。

視覚化の作成

リザルトで視覚化を作成する準備ができました。

手順

1. リザルトで **ACL チュートリアル**コレクションに戻り、**F5** を押して、ブラウザ ウィンドウを更新します。
Trans_All_Grouped テーブルが表示されます。
2. **【改善】**で **【レコードの表示】**をクリックします。
テーブルビューが開き、レコードが表示されます。
3. **【視覚化の追加】**をクリックし、**【ツリーマップ】**視覚化をクリックします。



ツリーマップ

4. **【視覚化の構成】**パネルで、以下に表示されるフィールドとオプションを選択します。

メモ

【視覚化の構成】パネルで、**【構成】**  をクリックします。



5. **適用**]をクリックします。

ツリーマップ視覚化が生成されます。

ツリーマップで個別のボックスの上にマウスカーソルを置き、埋め込まれたデータを表示できます。

ブラウザ ウィンドウのサイズを変更する場合は、ボックスを再配置し、関連付けられた説明の異なるセクションを表示または非表示にすると、動的に更新されます。

視覚化の保存

作成する視覚化を保持する場合は、保存する必要があります。各視覚化、および解釈と呼ばれる視覚化を格納するコンテナを個別に保存する必要があります。

手順

はじめに

1. ツリーマップ視覚化の左上端で **無題**] をクリックし、**取引ツリーマップ**などの視覚化のタイトルを入力し、Enter を押します。
2. **保存 > 名前を付けて保存**] をクリックします。
3. **タイトル** フィールドで、**チュートリアル視覚化**などの解釈の名前を入力し、**保存**] をクリックします。
解釈と視覚化はいずれも保存され、後からもう一度開くことができます。
4. タイトルバーに表示されているコレクション名、"**ACL チュートリアル**" をクリックすることで、分析コンテンツ、**サンプルレポート**に戻ります。
5. **解釈列**の数字をクリックします。**解釈**] ダイアログボックスが表示されます。新しく作成された解釈、**チュートリアルの視覚化**が一覧にあるのが分かります。
各分析コンテンツで複数の視覚化と解釈を作成できます。各視覚化は、**テーブルビュー**のデータに基づいています。

ストーリーボードへの公開

今作成した視覚化を表示するためのストーリーボードを作成します。ストーリーボードは、単一のプレゼンテーションで複数の視覚化とリッチテキストコンテンツを表示するコミュニケーションプラットフォームです。

手順

1. [ストーリーボードアプリを開きます](#)。
2. **ストーリーボードを追加**] をクリックします。
3. ストーリーボードのわかりやすいタイトルを入力します。ストーリーボードタイトルは最大 80 文字です。
4. **追加** ] をクリックし、**グラフの追加**] を選択します。 
5. 次のいずれかのオプションを選択します。
 - 解釈からテーブルビューを表示するには、親テーブルのエントリ 、**チュートリアルの視覚化**] を選択します。
 - 解釈から視覚化を表示するには、子のグラフエントリ 、**取引ツリーマップ**] を選択します。利用可能な可視化のリストをフィルター処理するには、検索フィールドにキーワードまたは語句を入力できます
6. 右上隅で **保存 > 保存**] をクリックします。

Analytics の他のレポート (作成) 方法

リザルトで使用可能なデータ視覚化の他に、Analytics には、データ分析の結果を報告するために使用できる他の複数の方法があります。

レポート方法	説明
Analytics のレガシー グラフ	Analytics には、レガシー グラフおよびレガシー チャートの作成機能があり、基本的なビジュアルレポートを作成できます。 詳細については、"Analytics グラフを使った作業" ページ 1393を参照してください。
従来の列形式のレポート	一部の場合、データの行と列を使用した従来のテキストと数字に基づくレポート

レポート方法	説明
	のみが必要です。 詳細については、"Analytics レポートの書式設定および生成" ページ 1387を参照してください。
第三者レポート ツール	Tableau または Microsoft BI などのサードパーティレポート ツールを使用して、Analytics から直接データをインポートできます。 詳細については、"サードパーティレポート アプリケーションからの Analytics への接続" ページ 1405を参照してください。
データの Excel または CSV へのエクスポート	データを Excel またはカンマ区切り値ファイルにエクスポートし、Excel または CSV ファイルの操作ができる任意のツールのレポート機能を使用できます。 詳細については、"データのエクスポート" ページ 210を参照してください。

完了しました。

おめでとうございます。Analytics を使用したデータ分析に関する詳細な導入コースを完了しました。

次の学習ステップ

Analytics について学習し続けるための複数のオプションがあります。

アカデミー	<p>アカデミーは、さまざまな経験レベルに合わせて多様なコースを提供します。「Analytics でのデータ分析の基礎」プログラム(ACL 101)は一続きの6つのミニコースであり、新しいユーザー向けに Analytics の基本事項について説明します。</p> <p>アカデミーは、Diligent のオンライントレーニング リソース センターです。コース カタログに移動すると、利用可能なコースを確認できます。</p> <p>サブスクリプションのユーザーは、アカデミー コースを無償で受講できます。</p>
Analytics および ACLScript ヘルプ	<p>現在、Analytics および ACLScript ヘルプを表示しています。このヘルプは、Analytics のすべての側面に関する、リファレンス タイプの概念的な資料、段階的な手順、および ACLScript 構文を提供します。</p> <p>たとえば、完了したチュートリアルの一部である、追加処理に関するヘルプトピックがあります。</p> <ul style="list-style-type: none"> ◦ "テーブルの追加" ページ 928 (概念) ◦ "テーブルを追加する" ページ 938 (段階的な手順) ◦ "APPEND コマンド" ページ 1651 (ACLScript 構文)
コミュニティ	<p>コミュニティは、経験豊富な Analytics ユーザーが専門知識を共有し、質問に回答するカスタマー フォーラムを含む、さまざまなカスタマー リソースを提供する Web プラットフォームです。</p> <p>カスタマー フォーラムは、Analytics の実践的な使用方法と用途について学習するための最適な場所です。</p>

作業のスクリプト化(任意のセクション)

推定時間	20分
要件	以前のスクリプト経験が必要です。
Analytics のバージョン	13.0 以降 (Unicode 版)

スクリプトを作成せずに、アドホックまたは手動によって、Analytics を使用して、さまざまな価値を得ることができます。ほとんどの場合、スクリプトで実行できる作業は、ユーザー インターフェイスでも実行できます。逆も同じです。ただし、Analytics の価値、能力、および効率を最大化するには、スクリプトを作成する必要があります。

ですが、幸いにも、Analytics には、初心者でもスクリプトを比較的容易に作成できるツールが用意されています。

スクリプトの事例

すべての現在の責任の他に、定期的に法人クレジットカードの使用状況を確認する責任が生じたとします。

時間の節約

基本的なレビュープロセスが標準化されます。各レビュー サイクルでは、時間をかけて手動で基本プロセスを繰り返すか、プロセスを自動化して時間を節約できます。

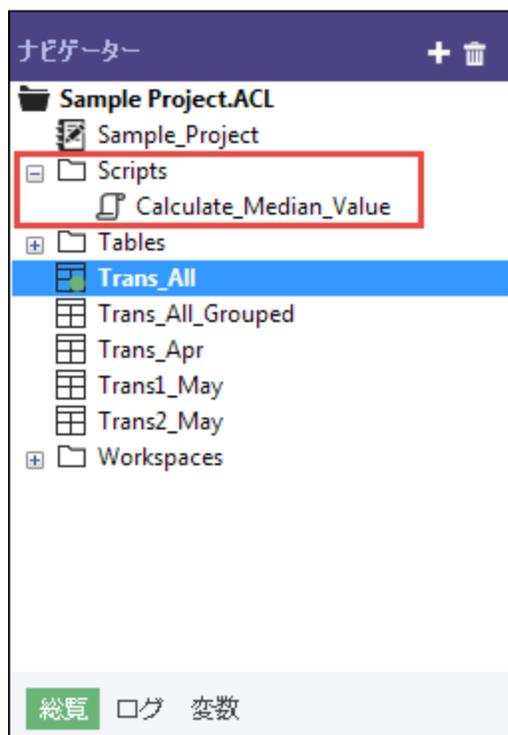
信頼できる委任

プロセスが自動化されると、タスクを部下に委任することができます。テストされたスクリプトを使用すると、経験が少ない従業員でも、作業負担を大幅に増やすことなく、タスクを一貫して正確に実行できることを確信できます。

スクリプトとは何か

Analytics スクリプトとは、特定のタスク、またはいくつかの関連するタスクを実行する、一連の ACLScript コマンドです。たとえば、このチュートリアル前半で手動で実行した作業は、スクリプトを使用して実行することもできます。

ACLScript は Analytics の基礎となるコマンド言語です。スクリプトは Analytics プロジェクトに保存されません。個別のスクリプトはナビゲーターに表示され、先頭にスクリプト アイコン  が表示されます。



Analytics コマンド ログの仕組み

ナビゲーターには、**ログ** タブがあります。スクリプト作成者は、Analytics コマンド ログを頻繁に使用することになります。

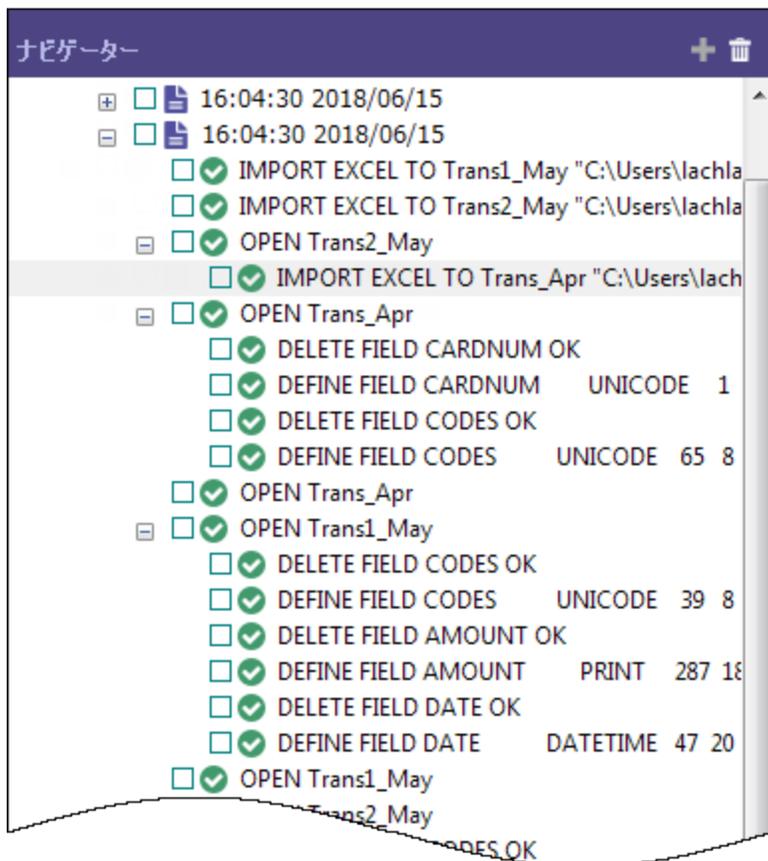
手順

1. **ログ** タブをクリックして開きます。

Analytics コマンド ログが表示されます。**ナビゲーター** パネルをドラッグして広げ、より多くのコンテンツを表示できます。

ログは各 Analytics セッション中に実行された正確な順番のコマンドを記録し、Analytics プロジェクトの一部として保存します。

このチュートリアル前半を完了したばかりの場合は、ユーザー インターフェイスで実行したすべてのアクションのリストがログに含まれています。



2. ログで、結果を新しいテーブルに出力する SUMMARIZE コマンドを検索してクリックします。

SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO "Trans_All_Grouped.FIL" OPEN PRESORT STATISTICS

当該のコマンドが、ツールバーの真下、Analytics のインターフェイスの上部近くにあるコマンドラインにあらかじめ入力されます。

メモ

コマンドラインが表示されない場合は、**[ウィンドウ> コマンドライン]**を Analytics メインメニューから選択します。

3. まだ開いていない場合は、**Trans_All** テーブルを開きます。
4. `f_Prohibited_codes` フィルターが適用される場合は、削除します。
5. **コマンドライン**をクリックし、`"Trans_All_Grouped.FIL"`を `"Trans_All_Grouped_2.FIL"` に変更して、Enter を押します。

要約コマンドが **Trans_All** テーブルで実行され、**Trans_All_Grouped_2** テーブルを出力します。これは、手動で作成した最初の出力テーブルを複製します。

最小限の努力で、**Trans_All** テーブルを要約するために必要なすべての以前の手動作業を再実行しました。コマンドラインからコマンドを実行することは、シンプルな 1 行のスクリプトの実行することに似ています。

ログからコマンドをコピーしてスクリプトを作成する

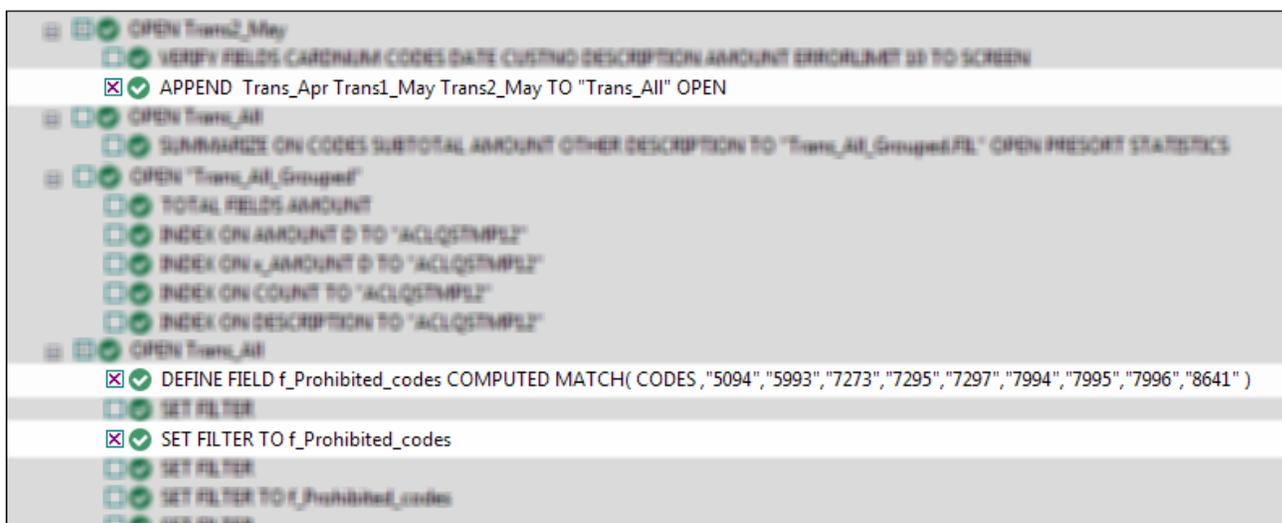
ここでも、ログからACLスクリプト構文を再利用できますが、今回は構文をAnalyticsスクリプトにコピーします。作業を迅速かつ簡単に行うために、チュートリアルで手動で実行した作業の一部のみをスクリプトにしますが、全部の作業をスクリプトにすることもできます。

メモ

このスクリプトの導入を簡潔にするために、一部のスクリプトのベストプラクティスは省略します。目標は、新しいユーザーでもAnalyticsでスクリプトを簡単に作成できることを示すことです。

手順

1. ログで、次のコマンドを見つけて、選択します。



2. ログを右クリックして **選択項目の保存 > スクリプト**]を選択します。
3. **スクリプトを別名で保存**]ダイアログボックスで、スクリプトの名前 `Append_and_filter` を入力して **OK**]をクリックします。
4. ナビゲーターの **概要**]タブで、新しく作成された **Append_and_filter** スクリプトをダブルクリックし、**スクリプト エディター** で開きます。

スクリプトが開き、ログで選択した3つのコマンドの構文全体が含まれています。

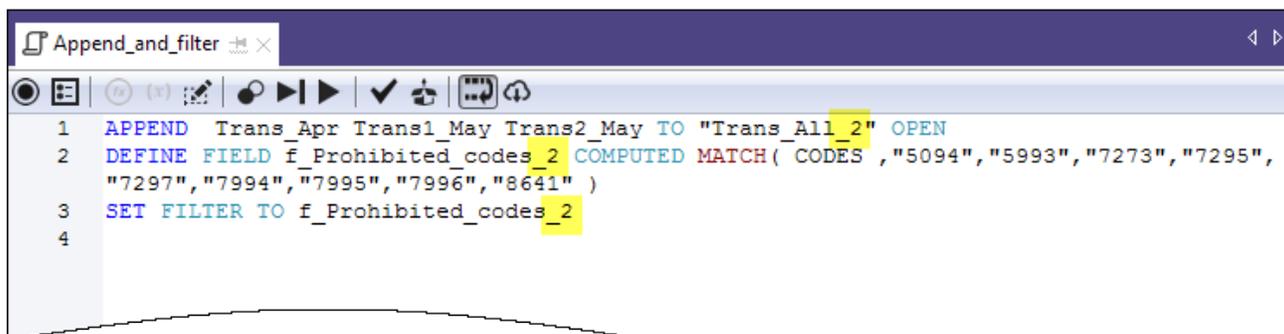
5. 時間を取って、各コマンドの構文を読みます。

ユーザーインターフェイスで前に実行したアクションがどのようにACLScript構文の個別の部分に対応しているのかがわかりますか。たとえば、`APPEND` コマンドの後には、追加した3つのテーブルの名前があります。

```
Trans_Apr Trans1_May Trans2_May
```

ほとんどの部分では、ACLScript構文とユーザーインターフェイスのアクション間の対応は比較的わかりやすくなっています。つまり、構文を理解することは、それほど難しくありません。

6. 次の場所に **2** を追加してスクリプトを修正します。



```
1 APPEND Trans_Apr Trans1_May Trans2_May TO "Trans_All_2" OPEN
2 DEFINE FIELD f_Prohibited_codes_2 COMPUTED MATCH( CODES , "5094", "5993", "7273", "7295",
"7297", "7994", "7995", "7996", "8641" )
3 SET FILTER TO f_Prohibited_codes_2
4
```

2 を追加して、既に手動で作成したテーブルとフィルターとの名前の競合を回避します。

7. [スクリプト エディター] ツールバーで、**実行**  をクリックして、スクリプトを実行します。

表示されるすべてのプロンプトで、**[はい]** をクリックします。

スクリプトが実行され、次のタスクを実行します。

- Excel からインポートした3つのテーブルを単一のテーブルの最後に追加し、新しいテーブルを開く
- 禁止されたコード フィルターを作成する
- 新しいテーブルにフィルターを適用する

わかるように、スクリプトの実行は、同じアクションを手動で実行するよりもかなり高速です。毎週や毎月、非常に複雑な分析が実行されるという現実の状況における、時間の節約と一貫性の改善を想像してみてください。

メモ

スクリプトを実行するには、**ナビゲーター**を右クリックし、**実行**]を選択することもできます。スクリプトを実行するには、開く必要はありません。

スクリプトのチュートリアル全体

手動で実行したチュートリアル全体は、以下のスクリプトに手動で表示されます(「手順」セクション)。このスクリプトの導入を完了するには、スクリプトを Analytics にコピーし、チュートリアル作業を元に戻しますが、今回はマウスを数回クリックするだけです。

メモ

スクリプトは、**Sample Data Files** フォルダがデフォルトの場所にインストールされていることを前提とします。フォルダが別の場所にインストールされている場合は、スクリプトのナビゲーションパスを修正し、正しい場所を参照するする必要があります。

スクリプトで作成されたテーブルの末尾には **s** が付くため、手動で作成したテーブルは上書きされません。

手順

新しい空のスクリプトを作成する

1. ナビゲーターの **概要** タブ内の **スクリプト** フォルダーを右クリックし、**新規作成 > スクリプト** の順に選択します。
2. **新しいスクリプト** を右クリックし、**名前の変更** を選択し、`Getting_Started_tutorial` を入力またはコピーして、**Enter** を押します。

チュートリアルスクリプトのコピーと貼り付け

1. 以下の **スクリプトを表示** をクリックします。
2. スクリプト全体をクリックしてドラッグしてから、**Ctrl+C** を押して、スクリプトをコピーします。

メモ

スクリプト全体を選択し、行を落とさないことが重要です。あるいは、[基本チュートリアル\(Unicode版\)](#)

で、スクリプトが含まれたテキスト ファイルをダウンロードできます

3. スクリプト エディター ウィンドウをクリックし、**Ctrl+V** を押して、スクリプト構文を空の `Getting_Started_tutorial` スクリプトに貼り付けます。

スクリプトの更新と保存

1. スクリプトのナビゲーションパスの更新：
 - a. スクリプトの最初の行をクリックします。
 - b. 右クリックして **検索** を選択します。
 - c. **置換** ダイアログボックスで、次のエントリを入力します。
 - **検索対象**: `ユーザー アカウント名`
 - **置換後の文字**: `コンピューターの実際アカウント名`
 - d. すべての `ユーザー アカウント名` のインスタンスで検索と置換を実行します
2. **開いているプロジェクトの保存**  をクリックし、表示されるプロンプトで **はい** をクリックします。

保存アイコンが見つからない場合は、Analytics のメインメニューで **ウィンドウ > ツールバー** をクリックして、ツールバーを有効にします。

スクリプトを実行する

スクリプト エディター ツールバーで、**実行**  をクリックして、スクリプトを実行します。

スクリプトが実行され、すべてのチュートリアル作業を複製します。インタラクティブ通知は、スクリプトの実行時に重要な情報を提供します。

スクリプトを表示

メモ

以前にスクリプトで作業したことがない場合は、初めはスクリプト構文に躊躇することがあります。ほぼすべての構文が Analytics ログからコピーされました。

スクリプトのインタラクティブ通知の構文 (DIALOG コマンド) は、別の比較的シンプルな Analytics ツールによって自動生成されました。

緑色の COMMENT コマンドは、概要レベルで、スクリプトを説明します。前のチュートリアルで完了したタスクを認識します。

```
COMMENT
*** Unicode 版 ***
このスクリプトは、「ACL Analytics の基本」チュートリアルで手動で実行したすべてのアクションを実行します。
END

COMMENT ユーザー確認なしでテーブルの上書きを許可します。
SET SAFETY OFF

COMMENT 3 つの Excel ワークシートをインポートします。

IMPORT EXCEL TO Trans1_May_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans1_May_s.fil" FROM "Trans_May.xls" TABLE "Trans1_
May$" KEEPTITLE FIELD "CARDNUM" C WID 19 AS "" FIELD "CODES" N WID 4 DEC 0
AS "" FIELD "DATE" D WID 19 PIC "YYYY-MM-DD hh:mm:ss" AS "" FIELD "CUSTNO" C
WID 6 AS "" FIELD "DESCRIPTION" C WID 95 AS "" FIELD "AMOUNT" C WID 9 AS ""

IMPORT EXCEL TO Trans2_May_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans2_May_s.fil" FROM "Trans_May.xls" TABLE "Trans2_
May$" KEEPTITLE FIELD "CARDNUM" C WID 19 AS "" FIELD "CODES" N WID 4 DEC 0
AS "" FIELD "DATE" D WID 19 PIC "YYYY-MM-DD hh:mm:ss" AS "" FIELD "CUSTNO" C
WID 6 AS "" FIELD "DESCRIPTION" C WID 155 AS "" FIELD "AMOUNT" C WID 9 AS ""

IMPORT EXCEL TO Trans_Apr_s "C:\Users\user_account_name\Documents\ACL
Data\Sample Data Files\Trans_Apr_s.fil" FROM "Trans_April.XLS" TABLE "Trans_
Apr$" KEEPTITLE FIELD "CARDNUM" N WID 16 DEC 0 AS "" FIELD "AMOUNT" N WID 6
DEC 2 AS "" FIELD "DATE" D WID 10 PIC "YYYY-MM-DD" AS "" FIELD "CODES" N WID
4 DEC 0 AS "" FIELD "CUSTNO" C WID 6 AS "" FIELD "DESCRIPTION" C WID 45 AS
""

COMMENT 新しいアナリティクス テーブルのテーブル レイアウトを調整します。

OPEN Trans_Apr_s
DELETE FIELD CARDNUM OK
DEFINE FIELD CARDNUM UNICODE 1 32 WIDTH 35
DELETE FIELD CODES OK
DEFINE FIELD CODES UNICODE 65 8 WIDTH 11

OPEN Trans1_May_s
```

```
DELETE FIELD CODES OK
DEFINE FIELD CODES UNICODE 39 8 WIDTH 11
DELETE FIELD AMOUNT OK
DEFINE FIELD AMOUNT PRINT 287 18 2 WIDTH 9
DELETE FIELD DATE OK
DEFINE FIELD DATE DATETIME 47 20 PICTURE "YYYY-MM-DD" WIDTH 27

OPEN Trans2_May_s
DELETE FIELD CODES OK
DEFINE FIELD CODES UNICODE 39 8 WIDTH 11
DELETE FIELD AMOUNT OK
DEFINE FIELD AMOUNT PRINT 407 18 2 WIDTH 9
DELETE FIELD DATE OK
DEFINE FIELD DATE DATETIME 47 20 PICTURE "YYYY-MM-DD" WIDTH 27

COMMENT インポートされたデータを検証し、ユーザー通知を表示します。

OPEN Trans_Apr_s
VERIFY FIELDS CARDNUM AMOUNT DATE CODES CUSTNO DESCRIPTION ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans_Apr_
s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans_Apr_
s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

OPEN Trans1_May_s
VERIFY FIELDS CARDNUM CODES DATE CUSTNO DESCRIPTION AMOUNT ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans1_
May_s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans1_
May_s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

OPEN Trans2_May_s
VERIFY FIELDS CARDNUM CODES DATE CUSTNO DESCRIPTION AMOUNT ERRORLIMIT 10
IF WRITE1=0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans2_
May_s テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Trans2_
May_s テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )

COMMENT Badfile テーブルを検証し、ユーザー通知を表示します。
OPEN Badfile
VERIFY FIELDS InvoiceNo Prodno Price OrderQty ShipQty Total ERRORLIMIT 10
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
```

```
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Badfile
テーブル: 0 データ検証エラーが検出されました" AT 12 28 )
IF WRITE1>0 DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "Badfile
テーブル: %WRITE1% データ検証エラーが検出されました" AT 12 28 )
CLOSE

COMMENT 3 つの新しいアナリティクス テーブルを単一の結合されたテーブルの最後に追加します。
APPEND Trans_Apr_s Trans1_May_s Trans2_May_s TO "Trans_All_s" OPEN
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 630 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 500 12 DEFAULT 1 ) (TEXT TITLE "結合された取引テーブル
(Trans_All_s) には %WRITE1% レコードがあります" AT 12 28 )

COMMENT 結合されたテーブルをマーチャント カテゴリ コードでグループ化します。
SUMMARIZE ON CODES SUBTOTAL AMOUNT OTHER DESCRIPTION TO "Trans_All_Grouped_
s.FIL" OPEN PRESORT STATISTICS
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 700 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 570 12 DEFAULT 1 ) (TEXT TITLE "グループ化された取引テーブ
ル (Trans_All_Grouped_s) には %WRITE1% マーチャント カテゴリ コードがあります" AT 12 28
WIDTH 550 )

COMMENT 結合されたテーブルをフィルタリングし、禁止された取引のみを表示します。
OPEN Trans_All_s
DEFINE FIELD f_Prohibited_codes COMPUTED MATCH(CODES, "5094", "5993",
"7273", "7295", "7297", "7994", "7995", "7996", "8641")
SET FILTER TO f_Prohibited_codes

COMMENT 正常終了メッセージ。
DIALOG (DIALOG TITLE "ユーザー ダイアログ" WIDTH 490 HEIGHT 100 ) (BUTTONSET
TITLE "&OK;&Cancel" AT 360 12 DEFAULT 1 ) (TEXT TITLE "スクリプトは正常に完了しま
した" AT 12 28 )

COMMENT テーブルを上書きする前に、ユーザー確認が必要です。
SET SAFETY ON
```

完了しました。

これで、スクリプトの簡単な導入が終わりました。スクリプトの価値を理解し、さらに学習する意欲が出てきたかと思います。

次の学習ステップ

Analytics でのスクリプト作成について学習できる選択肢がいくつか用意されています。

オプション	役立つ情報
チュートリアル	<p>Analytics ヘルプには、次の初級者向けチュートリアルがあります。</p> <ul style="list-style-type: none">◦ "初心者向けのスクリプト" ページ 1432◦ "Analytics でのスクリプト作成の基本事項" ページ 1450◦ "関数を使用する方法" ページ 1477 <p>ヘルプには、すべての ACLScript 言語リファレンスがあり、すべての Analytics コマンド および 関数 について詳細に説明しています。</p>
アカデミー	<p>アカデミーは、導入および高度なスクリプト コースの両方を提供します。</p> <ul style="list-style-type: none">◦ Analytics のスクリプトの基本 (ACL 106)◦ Analytics のスクリプト (ACL 303) <p>アカデミーは、Diligent のオンライントレーニング リソース センターです。コース カタログに移動すると、利用可能なコースを確認できます。</p> <p>Analytics サブスクリプションのユーザーは、アカデミー コースを無償で受講できます。</p>
コミュニティ	<p>コミュニティは Web ベースのプラットフォームであり、Analytics スクリプトの作成について頻繁かつ詳細に議論されているカスタマー フォーラムなど、さまざまなカスタマー リソースがあります。</p>

Analytics のヘルプ

Analytics を操作するときにヘルプを利用するには、複数の方法があります。

コンテキスト依存ヘルプ

Analytics 内の任意の場所から F1 キーを押すか、ほとんどの画面にある [ヘルプ] ボタン  をクリックすると、現在アクティブになっているウィンドウ、ダイアログボックス、タブ、あるいはウィザード画面を説明するヘルプトピックを開くことができます。

通常は、この最初のヘルプトピックからリンクをクリックすることにより、オンラインヘルプに含まれるさらに詳細な情報にアクセスできます。

オンラインヘルプとドキュメント

Analytics のメインメニューから、[ヘルプ > Analytics ヘルプ] を選択し、コンテキストに依存せずに、直接オンラインヘルプを表示できます。

一般に、古いバージョンの Analytics をお使いの場合でも、ヘルプは最新バージョンのものをお読みください。

ソフトウェアバージョンとサブスクリプション情報

Analytics のメインメニューから、[ヘルプ > バージョン情報] を選択し、次の情報のダイアログボックスを開きます。

- ソフトウェアバージョン番号
- エディションタイプ(Unicode または非 Unicode)
- サブスクリプションライセンス所有者の名前と会社
- サブスクリプションの有効期限

コミュニティ

[コミュニティ](#) に移動します。これは、Analytics の機能に関する質問を投稿できるカスタマーフォーラムを含む、多様なカスタマーリソースが登録された Web ベースのプラットフォームです。

サポート

(アカウントのサインインが必要です)

Analytics のメインメニューから、[ヘルプ > Galvanize への問い合わせ](#)]を選択し、Web ブラウザーを開き、サポートに接続します。

このページは意図的に空白のままにしています

Analytics のユーザー インターフェイス

このセクションでは、Analytics のユーザー インターフェイスに関する一般情報について説明します。

- ユーザー インターフェイス メニュー、タブ、ダイアログボックス、その他のユーザー インターフェイス要素の概要
- Analytics テーブルの構造
- カスタマイズ可能な Analytics 機能
- Analytics プロジェクトの概要
- 他の一般情報

Analytics を開始する

Analytics を開始するには、ACL for Windows デスクトップ ショートカットをダブルクリックし、ACL for Windows でオプションをクリックします。

- **新しいアナリティクス プロジェクト** - 新しい空の Analytics プロジェクトを作成します
- **アナリティクス プロジェクトを開く** - 既存の Analytics プロジェクトを開きます
- **最近のアナリティクスファイル**、または**サンプルファイル** - 最近開いた、またはサンプルの Analytics プロジェクト (.acl) を開きます

Analytics を終了する

Analytics を終了するには、**[ファイル > 終了]**を選択します。

保存されていない変更がプロジェクトで検出された場合は、終了する前にそれらの変更を保存するように求められます。変更を保存する場合は確認ダイアログボックスで **[はい]** をクリックして、終了します。

Analytics のユーザー インターフェイス の概要

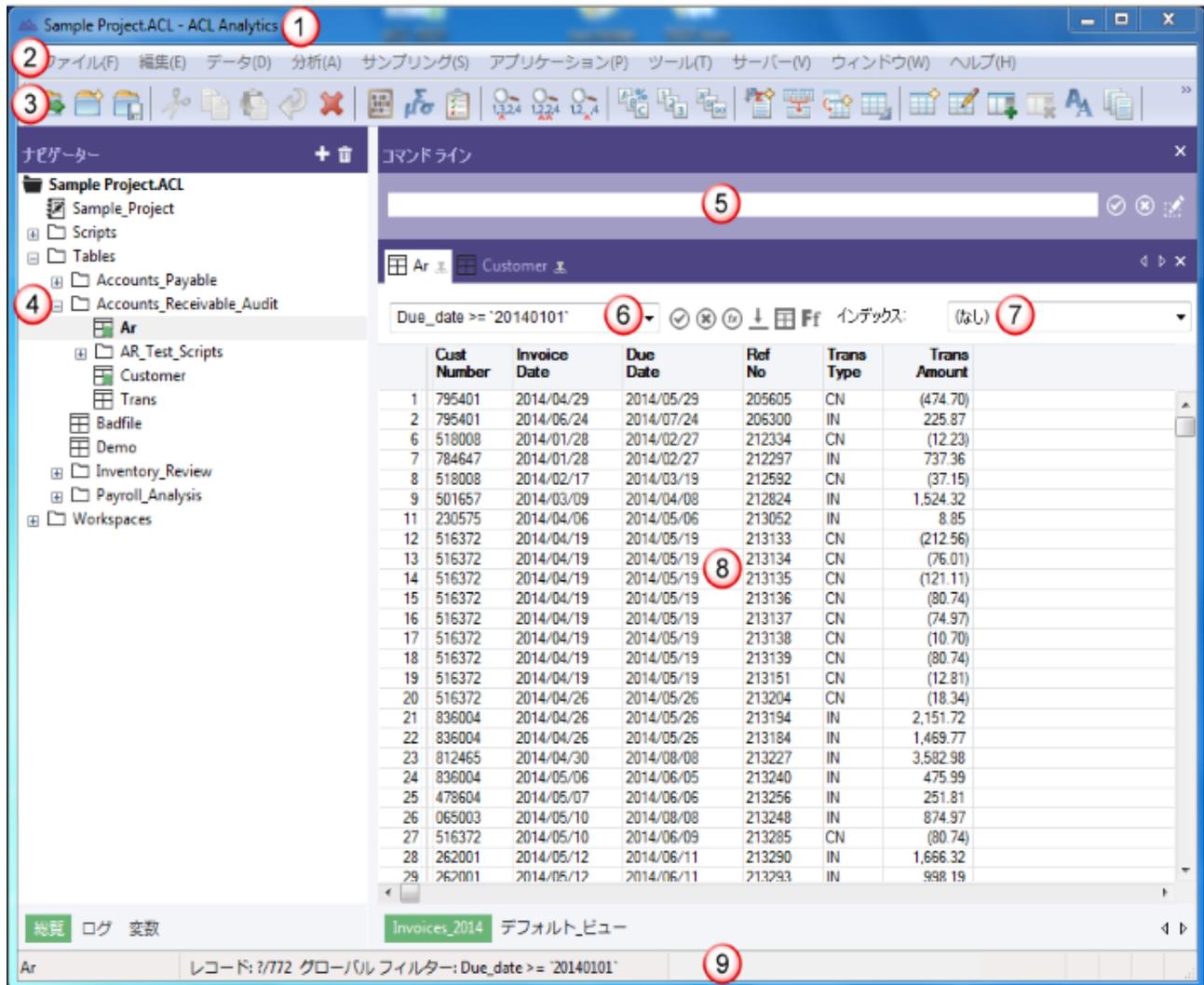
Analytics のユーザー インターフェイスには、特定の種類の情報が表示される多数の要素が含まれ、データを操作できます。

- "Analytics のメイン ユーザー インターフェイス" 下
- "コマンド ダイアログ ボックス" ページ 113
- "その他のユーザー インターフェイス要素" ページ 117
 - データ定義 ウィザード
 - テーブルレイアウトのダイアログ ボックス
 - [罫ビルダー] ダイアログ ボックス
 - ダイアログビルダー

ユーザー インターフェイス要素の構成と機能理解することにより、Analytics で効率的に作業できるようになります。

Analytics のメイン ユーザー インターフェイス

Analytics のメイン ユーザー インターフェイスは、Analytics を開始すると自動的に表示されます。



番号	要素	説明
1	タイトルバー	開いている Analytics プロジェクトの名前および ALC for Windows コンポーネント名が表示されます。
2	メインメニュー	次のメニューコマンドを含むほとんどの Analytics 機能を使用できます。 <ul style="list-style-type: none"> Analytics プロジェクトを使った作業 データ分析の実行 オプションと接続設定の構成
3	ツールバー	ツールバーのボタンは、よく使用するアクションへのショートカットになっています。Analytics は現在行っている作業に関連するボタンを有効にします。 ツールバーの表示を切り替えるには、 [ウィンドウツールバー] を選択します。

番号	要素	説明
		<p>メモ</p> <p>このツールバーに表示されるボタンをカスタマイズすることができます。詳細については、「Analytics ツールバーのカスタマイズ」ページ 154を参照してください。</p>
4	ナビゲーター	<p>開いている Analytics プロジェクトの情報を 3 つのタブに表示します。</p> <ul style="list-style-type: none"> ○ 総覧タブ - プロジェクトに属するすべての項目を表示します。 <p>任意のプロジェクト項目を右クリックして、アクションを実行することができます。 総覧の項目を整理するには、整理するプロジェクトのアイコンを右クリックして、新規作成フォルダーの順にクリックします。 総覧で作成したフォルダーには、任意のプロジェクト項目をドラッグすることができます。</p> <ul style="list-style-type: none"> ○ ログタブ - Analytics コマンド ログを表示します。 <p>このログには、プロジェクトに関連して実行するすべてのアクションが記録され、日時順に整列されています。ログ エントリを開くにはダブルクリックします。アクションを実行するには、ログ エントリを右クリックします。</p> <ul style="list-style-type: none"> ○ 変数タブ - プロジェクト内のすべての変数の名前、値、およびデータ カテゴリを表示します。 <p>タブの内容は、変数が作成、削除されたり、変数の値が変更されたりするにつれて、動的に更新されます。変数名はアルファベット順に表示されます。</p> <p>ヒント</p> <p>ナビゲーターのサイズを調整するには、ナビゲーターと表示領域の間の分割線をドラッグします。分割線をダブルクリックして、ナビゲーターを閉じるか開くこともできます。</p>
5	コマンド ライン	<p>ACLスクリプト コマンドを入力できます。</p> <p>コマンド ラインを表示または非表示にするには、ウィンドウコマンド ラインを選択します。</p>
6	フィルターとクイック検索	<p>2 つの異なるタスクを実行できるテキスト ボックスとドロップダウン リスト。</p> <ul style="list-style-type: none"> ○ ビュータブでフィルターをデータに適用します ○ 1 つ以上の検索語を入力して、ビュータブでデータのクイック検索を実行することもできます。
7	インデックス	<p>既存のインデックスをテーブルに適用したり、インデックスが現在適用されているかどうかを確認することができます。</p>
8	表示領域	<p>次のタブにさまざまな種類の情報を表示します。</p> <ul style="list-style-type: none"> ○ ビュータブ(上記に表示) - アクティブな Analytics テーブルを表示します。 <p>複数のテーブルを開く場合は、ビュー タブをピン留めすることができます。追加のテーブルを開くたびに、追加のビュー タブが開きます。個々のビュー タブを使用してテーブル間を切り替えることができます。</p> <p>ビュー タブの左端にある列はレコード番号列です。現在選択されているレコードの番号が緑色で強調表示されます。</p> <p>同一テーブルに複数のビューが存在する場合は、ビュータブの下部にあるボタンでビュー間を切り替えることができます。</p>

番号	要素	説明
		<ul style="list-style-type: none"> ○ 結果タブ - 分析操作の結果を画面またはグラフに出力したり、選択したコマンド ログのエントリを表示したりする場合、それらは結果タブに表示されます。 結果タブを固定して内容の表示が保持されるようにすると、以降の結果は追加の結果タブに表示されるようになります。 テキストとグラフを出力する操作では、タブの下部にあるボタンで2つの形式間を切り替えることができます。 ○ スクリプト エディター タブ - 新規または既存のスクリプトの内容を表示します。 スクリプトは手動で編集したり、Analytics で利用できる構文キャプチャなどのツールを使用して、またログからスクリプトへコマンドをコピーして編集することができます。 ○ ワークスペース エディター タブ - 新規または既存のワークスペースでフィールド定義を表示します。 ワークスペースを使用すると、Analytics テーブル間でフィールド定義を共有できます。 <p>ヒント 表示領域のサイズを調整するには、ナビゲーターと表示領域の間の分割線をドラッグします。分割線をダブルクリックして、ナビゲーターを閉じるか開くこともできます。</p>
9	ステータス バー	<p>次のような情報を表示します。</p> <ul style="list-style-type: none"> ○ アクティブな Analytics テーブルの名前 ○ レコード数 ○ 現在テーブルに適用されているフィルターの詳細 ○ 現在実行中のスクリプトの名前

コマンド ダイアログ ボックス

メインメニューから [要約] や [重複] などの Analytics コマンドを選択すると、コマンド ダイアログ ボックスが開きます。これらのダイアログ ボックスには、指定できるオプションがあります。

- 操作の入力フィールド
- 処理の動作に影響するさまざまなオプション
- 出力結果の種類

操作に応じて、ダイアログ ボックスのオプションは2つまたは3つのタブで編成されます。

- **メイン** タブ
- **詳細** タブ、または **出力** タブ、またはその両方

下の図は、3つのタブの例を示します。

メモ

ダイアログ ボックスのオプションは、選択する操作に基づきいらか変わります。以下の説明のとおり、ほとんどの操作に表示される標準のオプションは複数あります。

標準のオプション以外のオプションは、Analytics および ACLScript ヘルプで説明されています。

【メイン】タブ

【メイン】タブのオプション	できること...
フィールドの一覧	操作の入力フィールドを指定します
小計フィールド	任意で、操作の一環として小計を求めるために、1つ以上の数値フィールドを指定します
事前並べ替え	操作が実行される前に、自動的に並べ替えられる入力フィールドを指定します
もし次の	指定された条件を満たさないレコードを操作から除外するためのIFステートメントを指定します
変換先 (表示されません)	<p>出力結果を格納する Analytics テーブルの名前と場所を指定します</p> <p>メモ 【出力】タブがあるコマンド ダイアログ ボックスでは、【出力】タブに【名前】として現れます</p>

詳細]タブ

詳細]タブのオプション	できること...
範囲 すべて 第 1 次 少しいの間	操作の処理対象となる、テーブル内のレコード数を指定します <ul style="list-style-type: none"> すべて - すべてのレコードが処理されます 先頭 - テーブル内の先頭レコードから、指定した件数のレコードが処理されます 次 - テーブル内で現在選択されているレコードから、指定した件数のレコードが処理されます While - テーブル内の先頭レコードから、WHILE ステートメントが true と評価される限りは、レコードが処理されます
出力テーブルを開く	操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します [メイン]タブまたは 詳細]タブのいずれかに現れます
既存のファイルに追加する	Analytics テーブルまたはテキスト ファイルに格納された出力結果を、既存の Analytics テーブルまたはテキスト ファイルの末尾に追加するかどうかを指定します

出力]タブ

出力]タブのオプション	できること...
変換先 画面 印刷 グラフ ファイル	出力結果の種類を指定します <ul style="list-style-type: none"> 画面 - Analytics の表示領域に結果を表示します 印刷 - 通常使うプリンターに結果を送信します グラフ - 結果を基にグラフを作成し、それを Analytics 表示領域に表示します ファイル - Analytics テーブルまたはファイルに結果を保存または追加します 一部の操作は、4 種類すべての出力はサポートしていません
ファイルタイプ	出力結果をファイルに保存する場合は、Analytics テーブルまたはテキスト ファイルを指定します 操作に応じて、テーブルまたはテキスト ファイルのいずれかに保存できるか、あるいはどちらか一方にのみ保存できます
名前	出力結果を格納する Analytics テーブルまたはテキスト ファイルの名前と場所を指定します 出力]タブがないコマンド ダイアログ ボックスでは、 メイン]タブに 保存先] として現れます
ローカル	出力結果と Analytics テーブルをローカルに保存するか、またはサーバーに保存するかを指定します (サーバーテーブルに接続している場合にのみ有効になります) メイン]タブまたは 出力]タブのいずれかに現れます
ヘッダー	出力結果に加えるテキスト ヘッダーを指定します

出力]タブのオプション	できること...
	すべての出力の種類ではサポートされていません
フッター	出力結果に加えるテキスト フッターを指定します すべての出力の種類ではサポートされていません

その他のユーザー インターフェイス要素

Analytics には、以下のユーザーインターフェイス要素も含まれています。これらは、データの定義やインポート、テーブルでの作業、式の作成を行う場合、およびスクリプトへカスタムダイアログボックスを挿入する際に使用します。

- **データ定義ウィザード** - さまざまなデータソースにアクセスする標準の方法を提供するページベースのウィザード

データ定義ウィザードは、新しい Analytics プロジェクトを作成する際、および既存のプロジェクトに新しい Analytics テーブルを追加する際に自動的に表示されます。

詳細については、「データの定義とインポート」ページ 239を参照してください。

- **データアクセス ウィンドウ** - データベースまたはファイルのいずれかでソース データにアクセスするために使用できる複数のデータコネクターを含む視覚的なインターフェイス

詳細については、「データの定義とインポート」ページ 239を参照してください。

- **テーブルレイアウト]ダイアログ ボックス** - Analytics テーブルレイアウトを定義または変更するために使用されます。

テーブルレイアウトは、Analytics がデータソース内のレコードを識別し、個々のフィールドを読み取る方法を指定します。

詳細については、「テーブルレイアウトのダイアログ ボックス」ページ 771を参照してください。

- **式ビルダー]ダイアログ ボックス** - Analytics で式を入力するために使用されます。

式とは、データフィールド、演算子、関数、フィルター、および変数を組み合わせた、Analytics が値を評価して返すステートメントです。

詳細については、「式ビルダーの概要」ページ 870を参照してください。

- **ダイアログビルダー** - Analytics スクリプトでカスタムのダイアログ ボックスを作成するために使用されます。

カスタムダイアログボックスは、スクリプトの実行時にユーザー インターフェイスまたはフィードバックを提供します。

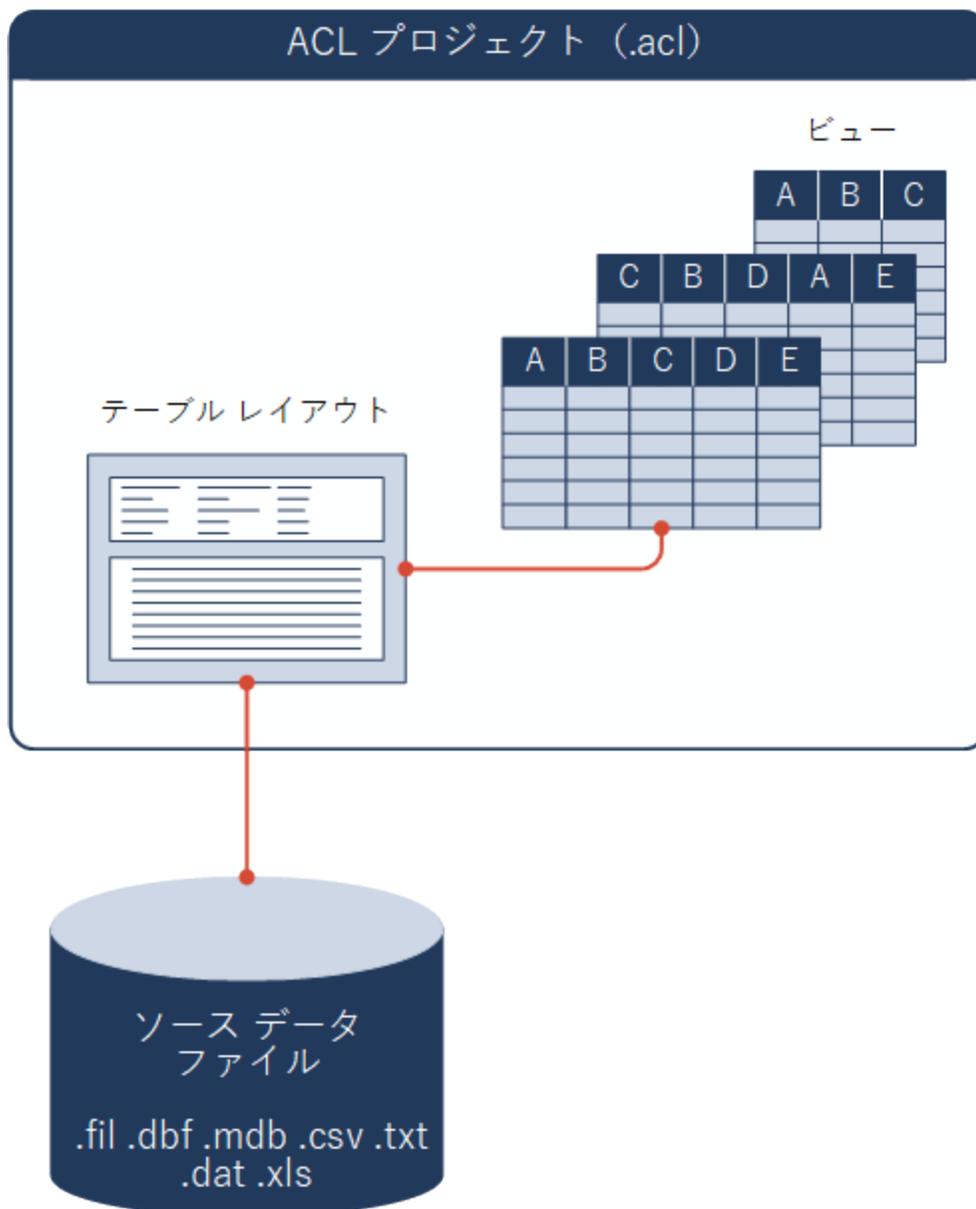
詳細については、「カスタムダイアログボックスの作成」ページ 1588を参照してください。

Analytics テーブルの構造

Analytics のデータはテーブルに格納されています。Analytics でデータを表示、分析しているときは、Analytics テーブルで作業を行っていることとなります。Analytics テーブルは、次の3つの部分から成る構造をしています。

- ソースデータファイル
- テーブルレイアウト
- 1つ以上のビュー

Analytics テーブルがどのように構造化されているか理解することは、Analytics で効率的に作業を行う基本です。以下の図は Analytics テーブルの構造を示します。



ソース データ ファイル

ソース データ ファイルには、フィールド名などの Analytics 書式情報を持たない、生データが入っています。生データは Analytics プロジェクトの外に存在します。Analytics プロジェクトと同じ Windows フォルダーに存在してもかまいませんし、同じコンピューター上やネットワークの場所、あるいはデータベース内のどこか別の場所にあってもかまいません。

Analytics ソース データ ファイル

ほとんどの場合、Analytics テーブルを新規作成するときに、Analytics が新しい Analytics ソース データ ファイルにソース データのコピーとインポートを行い、このファイルを拡張子 `.fil` のフラット ファイル形式で保存します。`.fil` ファイルが Analytics テーブルのベースとなります。

フラット ファイル形式は、固定長のフィールドと、固定長あるいはキャリッジ リターンおよびライン フィードのコードによって識別されるレコードを持っている、解釈が容易な形式です。

`.dbf` ファイルや `.txt` ファイルなどのいくつかの場合では、Analytics は既存のソース データ ファイルに直接接続できるため、`.fil` ファイルは必要なく、作成されません。

テーブルレイアウト

テーブルレイアウトはソース データ ファイルの生データの構造化された解釈です。テーブルレイアウトは、フィールド定義などのメタデータを含んでいます。フィールド定義では、フィールド名、フィールドの開始位置、フィールドの長さ、フィールドのデータ型などを指定します。

Analytics は生データの意味を理解するためにテーブルレイアウトを必要とします。テーブルレイアウトでは、ソース データ ファイルの場所も指定します。テーブルレイアウトは Analytics プロジェクト ファイル(`.acl`)の一部です。

詳細については、"テーブルレイアウトを使用したデータの構造化" ページ 768を参照してください。

1 つ以上のビュー

ビューは、名前が付けられた列と番号付きのレコードの配列で、テーブルレイアウトから構造化されたデータを表示するものです。Analytics のメイン ウィンドウでデータを確認するときには、ビューを見ていることとなります。

同じテーブルに対して、列の選択や配置が異なる複数のビューを作成することができます。ビューは Analytics プロジェクト ファイル(`.acl`)の一部です。

また、ビューは Analytics レポートのデータの書式設定も行います。

詳細については、"テーブルビューでデータを表示する" ページ 838を参照してください。

Analytics のカスタマイズ

Analytics は、アプリケーションのデフォルトの動作が定義される標準設定でインストールされます。[オプション]ダイアログボックス([ツール] > [オプション])でこれらの設定を変更し、次のようなアプリケーションの動作を修正できます。

- 機能をオンまたはオフにする
- データの表示方法を変更する
- コマンド出力の一部の要素を制御する

他のカスタマイズ可能なオプション

本ガイドのこのセクションでは、ビューとレポートのフォント設定の変更、Analytics ツールバーのカスタマイズ、スクリプトや頻繁に使用するコマンドなどのユーザー定義メニュー項目を表示するための[アプリケーション]メニューのカスタマイズについて説明します。

スクリプト エディター

スクリプト エディターのカスタマイズの詳細については、「スクリプト エディターのカスタマイズ」ページ 1602を参照してください。

Analytics のオプションの設定

オプション]ダイアログ ボックスを使用することで、Analytics の機能がどのように動作するかを制御するさまざまなオプションを設定できます。

必要であれば、Analytics プロジェクトごとに異なるオプションを設定することもできます。詳細については、「Analytics の初期設定ファイルの動作」 ページ 148を参照してください。

すべてのオプションをデフォルト設定に戻す

設定可能なすべてのオプションを Analytics のデフォルト設定に戻す場合は、**オプション]**ダイアログ ボックスの下部にある **既定値に戻す]** ボタンをクリックします。

注意

既定値に戻す] をクリックすると、現在開いているアクティブなタブだけでなく、**オプション]**ダイアログ内のすべてのタブの全オプションをデフォルトの設定に戻します。

各種設定は初期設定ファイルに保存される

設定可能なオプションの各種設定は初期設定ファイル(.prf ファイル)に保存されます。詳細については、「Analytics の初期設定ファイルの動作」 ページ 148を参照してください。

手順

1. Analytics のメイン メニューから **[ツール > オプション]** を選択します。
2. 変更したいオプションがあるタブをクリックして、そのオプションの設定値を変更します。

次のタブに、関連するオプションがグループ分けされています。

- [\[システム\] タブ](#)
- [\[インターフェイス\] タブ](#)
- [\[テーブル\] タブ](#)
- [\[ビュー\] タブ](#)
- [\[コマンド\] タブ](#)
- [\[日付と時刻\] タブ](#)
- [\[数値\] タブ](#)
- [\[印刷\] タブ](#)
- [\[アプリケーション フォント\] タブ](#)

3. **[OK]** をクリックして、変更を保存します。

システム オプション

[システム] タブのオプションを使用して、並べ替え処理およびインデックス処理でメモリがどのように使用されるかを制御することができます。

並べ替えとインデックスに追加システムリソースを使用する

このオプションが選択されると、並べ替え処理またはインデックス処理されるファイルのサイズ、およびコンピューターの空きメモリの量に応じて、並べ替えとインデックス処理が高速になります。

オプションを選択すると、Analytics は並べ替えとインデックス処理に、メモリマッピングされたファイルを使用できます。メモリマッピングされたファイルを使用すると、コンピューターで使用可能な RAM を超えているファイルの並べ替えとインデックス処理が高速化できます。例えば、500 MB のファイルの場合、オプションの設定に関係なく、並べ替え処理にかかる時間は変わらない可能性があります。3 GB のファイルの場合、オプションを有効にすると、並べ替え処理が大幅に高速化する可能性があります。

より小さいファイルを並べ替える場合のように、数ギガバイトの非常に大きいファイルを並べ替える場合は、オプションが有効でも速度の改善が見られない可能性があります。日常的に非常に大きいファイルを並べ替え、パフォーマンスの問題がある場合は、コンピューターの RAM を拡張し、Windows ページングファイルサイズを大きくすることを検討してください。

追加情報

- Analytics が並べ替えまたはインデックス処理の対象のファイルに格納されたレコード数を判定できない場合は、**並べ替えとインデックス処理で追加のシステムリソースを使用する**が選択されている場合でも、メモリマッピングされたファイルは処理で使用されません。
- 並べ替え処理やインデックス処理に追加のシステムリソースを使用すると、並べ替えまたはインデックス処理が実行中のときに、他のタスクの速度が低下する可能性があります。
- オプション** ダイアログ ボックスの他のすべてのオプションとは異なり、このオプションの設定は Analytics 初期設定ファイルに保存されません。このオプションは、設定されているコンピューターにのみ適用されます。

初期設定ファイルに保存されているオプションの詳細については、「Analytics の初期設定ファイルの動作」 ページ 148を参照してください。

インターフェイス オプション

[インターフェイス] タブのオプションを使用して、Analytics のいくつかの基本的な動作を指定します。

フィールド一覧にフィルターを含める

フィールド一覧に、フィルターを含む論理フィールドを表示させる場合は、このオプションを選択します。通常、論理フィールドはフィールド一覧から除外されます。フィルターは論理フィールドとして実装されるため、このオプションによってフィルターがフィールド一覧に表示される場合は、そのフィルターにコマンドを適用できます。

ファイルを上書きする前にメッセージを表示する

次のいずれかを上書きする前に、Analytics によって確認ダイアログボックスを表示させたい場合は、このオプションをオンにします。

- テーブルレイアウトにおけるフィールド
- Analytics テーブル
- Analytics データファイル(.fil) などのファイル

このオプションをオフにすると、Analytics は確認を行わずにフィールド、テーブル、およびファイルを上書きします。

このオプションのオン/オフは、スクリプトまたはコマンドラインで `SET SAFETY` を使用して切り替えることもできます。このオプションをオフにすると、既存のフィールド、テーブル、またはファイルを上書きするように設計されたスクリプトの実行が中断されるのを回避することができます。

[ファイルを上書きする前にメッセージを表示する] チェックボックスを使用して設定を変更した場合、その変更は、再度明確に変更されるまで有効です。`SET SAFETY` コマンドを使用して設定を変更した場合、その変更は、Analytics セッションの間のみ有効です。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET SAFETY {ON|OFF}
```

ACL サーバー統合を有効にする

AX Server に接続できるようにするための機能を有効にする場合は、このオプションを選択します。

スクリプトのオートコンプリートを無効にする

スクリプト エディターでキーワード オートコンプリートをオフにする場合は、このオプションを選択します。関数パラメーターのオンスクリーンヘルプを無効にすることはできません。

スクリプトをコミットする前に、スクリプト構文チェックを無効にする

このオプションを選択すると、スクリプトを HighBond のロボット アプリにコミット するときに、スクリプト 構文チェックをオフにします。

メモ

このオプションは、スクリプト 構文チェックのみを制御します。無効にできない個別のプロセスである、アナリティクス ヘッダー確認は制御しません。

作業完了時にビープ音を鳴らす回数

このオプションでは、Analytics が作業の完了時に鳴らすビープ音の数を指定します。0 から 255 までの数値を入力します。このオプションは、作業に時間がかかることが予想される場合に特に便利です。デフォルトは 0 です。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET BEEP 値
```

テーブル オプション

テーブル] タブのオプションを使用して、Analytics によるテーブルの処理方法を指定します。

開くときに自動的にプロファイルを実行する

開くときに自動的にプロファイルを実行する] をオンにすると、プロジェクトを開いたり、テーブルを変更したり、グローバルフィルターを変更したりする場合に、すべての数値フィールドに対して Profile コマンドが自動的に実行されるようになります。

Analytics は情報を保持しておき、その情報をヒストグラムや階層化の最小値および最大値、金額単位サンプリングの絶対値を提供するために使用します。

メモ

このオプションをオンにすると、数値フィールドのあるテーブルを開く速度が遅くなります。

テーブルと一緒にデータ ファイルを削除する

このオプションをオンにすると、テーブルを削除したときに、関連付けられているデータ ファイルが自動的に削除されるようになります。このオプションを使用すると、必要のないファイルをハード ディスクからすばやく削除できます。ただし、削除後はデータにアクセスできなくなります。

注意

このオプションをオンにするときには注意してください。テーブルと共に元のデータ ファイルが削除される場合もあります。

データ ファイルはすぐに削除されます。Windows のごみ箱には送られません。

このオプションのオン/オフは、スクリプトまたはコマンド ラインで `SET DELETE_FILE` を使用して切り替えることもできます。

データ ファイルとテーブルを表示する] チェック ボックスを使用して設定を変更した場合、その変更は、再度明確に変更されるまで有効です。`SET DELETE_FILE` コマンドを使用して設定を変更した場合、その変更は、Analytics セッションの間のみ有効です。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET DELETE_FILE {ON|OFF}
```

テーブルレイアウトを共有しない

メモ

フィールド定義を誤って削除しないように、**テーブルレイアウトを共有しない** チェックボックスはデフォルトでオンになっています。

このオプションをオフにすると、同じレコード構造を持つ複数のデータファイルやデータソースで、単一のテーブルレイアウトを共有することができます。この機能は、同じレコード構造を持つ Analytics テーブルに結果を出力できる Analytics 操作、つまり、抽出、並べ替え、サンプリング、およびマージと、テーブルレイアウトのコピーでのみ作用します。

テーブルレイアウトの共有が許可されている場合、同じレコード構造を持つ複数のソースデータファイル（たとえば Analytics データファイル(.fil)）やデータソースは、1つのフィールド定義セットを共有します。共有テーブルレイアウトに物理フィールドや演算フィールドを追加したり、関連するビューに列を追加したり、ビューを追加したりすると、その追加されたフィールド、列、またはビューは、共有テーブルレイアウトを使用しているすべての Analytics テーブルに自動的に追加されます。フィールド、列、またはビューを削除すると、共有テーブルレイアウトを使用している Analytics テーブルでそれを使えなくなります。

一般的には、データファイルごとに個別のテーブルレイアウトを保持する必要があります。しかし、単一のテーブルレイアウトを共有すると、同じレコード構造を持つ複数のデータファイルが同一のフィールド定義セットを必要とする場合には、手間を省くことができます。テーブルレイアウトへのあらゆる更新は、すべてのデータファイルに適用されます。たとえば、年間の取引テーブルから 12 箇の月次テーブルにレコードを抽出すると、同じレコード構造を持つテーブルが合計で 13 箇生成されます。**テーブルレイアウトを共有しない** チェックボックスがオンになっている場合は、テーブルごとに独自のレイアウトを持ちます。**テーブルレイアウトを共有しない** チェックボックスがオフになっている場合は、すべてのテーブルが元のテーブルのレイアウトを共有するため、レイアウトを一元管理することができます。

共有テーブルレイアウトは、それを使用しているテーブルの 1 つから削除しても、全体的な削除は行いません。共有テーブルレイアウトを使用している他のテーブルは、その後も使用することができます。

個別の Analytics プロジェクト以外は共有できません。テーブルを別のプロジェクトにコピーした場合は、**テーブルレイアウトを共有しない** の設定に関係なく、新しいテーブルレイアウトが作成されます。

正確な文字比較を行う

このオプションを使用すると、Analytics で文字型フィールド、識別子、リテラル値を比較する方法を制御できます。

メモ

スペースは文字のように扱われます。

このオプションをオフにした場合

このオプションをオフにした場合には、異なる長さの 2 つの文字列を比較する際に、短い方の文字列が採用されます。比較は左から右へ行われます。

正確な文字比較はオフになっている

True	False
<ul style="list-style-type: none"> ◦ "AB" = "AB" ◦ "AB" = "ABC" ◦ "AB" = "ABLMNOP" ◦ "AB " = "AB" 	<ul style="list-style-type: none"> ◦ "AB" = "ZZAB" ◦ "AB" = "ABC" ◦ " AB" = "AB"
<p>スペースのある例</p> <ul style="list-style-type: none"> ◦ "AB " = "AB" は True です。その理由は、3 カラム目のスペースが考慮されず、短い方の文字列 ("AB") が比較に使用されるためです。 ◦ "AB " = "ABC" は False です。その理由は、3 文字すべてが比較され、それぞれの 3 カラム目にあるスペースと "C" が異なるためです。 ◦ " AB" = "AB" は False です。その理由は、短い方の文字列 ("AB") が比較に使用され、1 カラム目のスペースと "A" が異なるためです。 	

このオプションをオンにした場合

このオプションをオンにした場合に一致を成すには、比較文字列同士が完全に同じでなければなりません。異なる長さの 2 つの文字列を比較する場合、Analytics により、長い方の文字列の長さに一致するよう、短い方の文字列の末尾にスペースが埋め込まれます。

正確な文字比較はオンになっている

True	False
<ul style="list-style-type: none"> ◦ "AB" = "AB" ◦ "AB " = "AB" 	<ul style="list-style-type: none"> ◦ "AB" = "ABC" ◦ "AB" = "ABLMNOP" ◦ "AB" = "ZZAB" ◦ "AB" = "ABC" ◦ " AB" = "AB"
<p>スペースのある例</p> <ul style="list-style-type: none"> ◦ "AB " = "AB" は True です。その理由は、長い方の文字列 ("AB ") に一致するよう、短い方の文字列 ("AB") の末尾にスペースが埋め込まれるため、"AB " と "AB " が一致するからです。 ◦ "AB " = "ABC" は False です。その理由は、3 文字すべてが比較され、それぞれの 3 カラム目にあるスペースと "C" が異なるためです。 ◦ " AB" = "AB" は False です。その理由は、長い方の文字列 (" AB") に一致するよう、短い方の文字列 ("AB") の末尾にスペースが埋め込まれるため、" AB " と "AB " が一致しなくなるからです。 	

スペースの削除

先頭および末尾のスペースを削除することで、文字列と文字列内部のスペースのみが比較されるようにするには、ALLTRIM() 関数を使用します。

たとえば、値に対して ALLTRIM() を使用した `ALLTRIM(" AB") = ALLTRIM("AB")` は True になりますが、この関数を使用しない場合の比較の結果は False になります。

正確な文字比較とフィルター

正確な文字比較を行う 設定は、フィルターがどのように働くかに影響を与えます。

- **このオプションがオフの場合** - `Address = "PO Box"` を指定すると、"PO Box" で始まるすべての住所が返されます。
- **このオプションがオンの場合** - `Address = "PO Box"` を指定すると、Address フィールドに "PO Box" という文字列以外何も入っていないレコードのみが返されます。

範囲

Analytics の操作や関数は、**正確な文字比較を行う** オプションの影響を受けるものもあれば、受けないものもあります。

影響を受ける	影響を受けない
<ul style="list-style-type: none"> ◦ 条件検索操作 ◦ MATCH() 関数 ◦ BETWEEN() 関数 	<ul style="list-style-type: none"> ◦ 結合操作 ◦ 関連付け操作 ◦ FIND() 関数 ◦ FINDMULTI() 関数

ログ エントリ

この設定への変更は、次の構文を使用してログに記録されます。

```
SET EXACT {ON|OFF}
```

開くときにフォーマットを表示する

このオプションをオンにすると、新しいテーブルを開いたときに、現在のテーブルレイアウトと演算フィールド定義が自動的に表示されるようになります。この表示内容はコマンド ログにも出力されます。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET FORMAT {ON|OFF}
```

フラット ファイルを手動で定義する

このオプションがオンになっていると、データ定義ウィザードでフラット ファイルからテーブルを作成するとき、一部の画面がスキップされます。データ定義は [テーブルレイアウト] ダイアログ ボックスで完成させます。

バッファ サイズ

このオプションでは、読み取るデータブロックのサイズを指定します。デフォルトは 33 キロバイトです。この値は大部分のアプリケーションで推奨されている値です。

使用できる値の範囲は 5 から 255 までです。バッファ サイズを変更すると、特定の環境においてパフォーマンスが若干改善される場合もあります。この設定の変更は、サポートからアドバイスされていない限り、行わないでください。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET READAHEAD 値
```

ソート メモリ

このオプションは、ソート処理とインデックス処理に割り当てるシステムリソースの最大量を指定します。ソート メモリは 0 から 2000 MB(メガバイト) で、20 MB 単位で入力します。ソートのパフォーマンスを最適化するには、システムで使用可能な物理メモリに基づいてソート メモリを設定します。これにより、Analytics は必要に応じて、テーブルのソートに必要なメモリ量をこの最大量まで使用できるようになります。

ソート メモリを 0 のままにしておくと、Analytics は現在使用可能なシステムリソースを使用します。

並び順

このオプションでは、文字フィールドのソート順を設定します。

ドロップダウン リストからロケールを選択します。非 Unicode 版の Analytics のデフォルトは "システム デフォルト" で、Unicode 版では "混合言語 (UCA)" です。デフォルトでは、Analytics は文字セット内の各文字のバイト順に基づく昇順でデータをソートします。並び替え順] オプションは、ソートしたりインデックスを作成したりするとき、クイックソートを実行するとき、また順番をテストするときのソート順に作用します。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET ORDER 値
```

ビュー オプション

[ビュー] タブのオプションを使用して、Analytics でビューを表示する方法を指定できます。

フィルターされたレコードを表示しない

[フィルターされたレコードを表示しない] オプションがオンのときには、フィルタリングされたテーブルには、フィルターによって含まれるレコードのみが表示されます。

このオプションがオフになる場合、フィルタリングされたテーブルは、すべてのレコードのみを表示し、フィルターで除外されるレコードはハイライト表示されます。

グリッド線を表示する

このオプションがオンの場合は、ビューにグリッド線が表示されます。

列幅をテストする

このオプションをオンにして列の幅を変更すると、Analytics ではビューを閉じる際に、変更作業を保存するか破棄するかを尋ねられます。

このオプションをオフにしてビューで列の幅のみを変更した場合、その変更は破棄されます。

ページの右端の境界線を表示する

このオプションがオンの場合は、Analytics ではビューに点線を表示することによって右余白が示されます。この余白は、レポートの印刷設定オプションに基づいています。

無効なデータを空白またはゼロで表示する

このオプションがオフの場合、Analytics はフィールド内の無効な文字をすべて受け入れます。

このオプションをオンにすると、Analytics によって、最初の無効な文字からフィールドの最後まで、無効な文字データは空白に、無効な数字データはゼロに置き換えられます。このオプションは、テキスト フィールドを除くすべてのフィールドに適用されます。Analytics では、テキスト フィールドの無効なデータは自動的に空白に置き換えられます。

無効なデータを空白またはゼロで表示する オプションは、可変長のデータ ファイルを扱う場合に非常に役立ちます。このオプションにより、Analytics ではデータを正確に表示することが可能になります。

この設定への変更は、次の構文を使用してログに記録されます。

SET CLEAN {ON|OFF}

再描画する秒数

このオプションは、Analytics でビューの再描画にかかる最大時間を秒数で示します。再描画が指定した時間よりも長くなる場合は、Analytics は処理を中断してメッセージを表示します。指定できる最大秒数は 100 秒です。デフォルトの時間は 10 秒です。

ごく一部のレコードしか選択しない制限的なフィルターを使用する場合、特に、非常に大きなファイルで作業をする場合には、再描画時間を長くする必要があるかもしれません。再描画時間を長くすると、データが表示されるまでの時間が長くなる可能性があります。待ち時間を短くするには、**フィルターされたレコードを表示しない**オプションをオフにします。

グローバルページのタイトル

このテキスト ボックスに入力した内容が、印刷される各ページの上部にあるページ番号の下に左揃えで表示されます。このボックスを空白にしておくこともできます。

Analytics の認証されたグループライセンス バージョンでは、**グローバルページのタイトル**]に顧客の指定テキストが入ります。

この設定 への変更は、次の構文を使用してログに記録されます。

SET DESIGNATION 値

コマンド オプション

[コマンド] タブのオプションを使用して、Analytics のコマンドの実行方法を指定します。

コマンドを自動実行する

このオプションをオンにすると、Analytics は、入力としてビューの選択したフィールドを使用し、だちに特定のコマンドを実行します。コマンドの編集やローカルフィルターの適用は行うことができません。このオプションは一部のコマンドにのみ適用されます。選択した入力フィールドは、コマンドに必要なデータ型です。

出力ファイル名を自動的に付ける

このオプションをオンにすると、コマンド出力ファイルの名前は Analytics によって自動生成されるようになります。この自動生成された名前には、コマンド名と 01 から始まる増分数が含まれます。この名前を受け入れても、もっと意味のある名前を指定してもかまいません。

出力テーブルを開く

このオプションをオンにする場合は、テーブルを作成するコマンド ダイアログボックスにある **出力テーブルを使用する** チェックボックスがデフォルトで選択されています。チェックボックスがオンになり、コマンドを実行すると、Analytics は自動的にアクティブなテーブルを閉じて、コマンドによって作成されたテーブルを開きません。出力テーブルを開かない場合は、コマンドを実行する前にチェックボックスをオフにすることができます。

ログにグループテストを表示する

このオプションは、スクリプトを実行したとき、コマンド ログに表示されるグループ結果の表示に影響しません。Analytics は、適用するコマンドに加えて、If、While、For、Next の各検証のグループ結果を表示します。グループ内の複数の検証が各コマンドに適用されるので、構文が非常に長くなる場合があります。

このオプションをオフにすると、Analytics はスクリプト実行時に検証を表示しなくなります。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET TEST {ON|OFF}
```

コマンド結果の XML 出力を生成しない

このオプションがオンの場合、画面へのコマンド出力を書式付きのテキストではなく標準のテキストとして生成します。

この設定 への変更は、次の構文を使用してログに記録されます。

```
SET SUPPRESSXML {ON|OFF}
```

ヌルフィールドに適合する値を返す

このオプションは、NULL 文字値の解釈方法を制御します。文字値は、空の場合、つまりデータが入っていない場合は "NULL" になります。たとえば、`NAME=""` は、NULL 文字値を含む式となります。2 つの引用符の間には何も挿入されません。

このオプションがオンの場合、Analytics は NULL をワイルドカードと解釈します。たとえば、ビューフィルターを `NAME = ""` とすると、NAME フィールドに何か値が入っているレコードが表示されます。

このオプションがオフの場合、Analytics は NULL を文字どおりに NULL と解釈します。たとえば、ビューフィルターを `NAME = ""` とすると、NAME フィールドにデータが入っていないレコードのみが表示されます。

階層数(間隔数)

このオプションでは、階層化またはヒストグラムにおいて、デフォルトで選択される階層数(間隔数) を指定します。1 から 255 までの数値を入力します。デフォルトは 10 です。

エラー制限

このオプションは、Sequence コマンドまたは Verify コマンドの処理を停止するデフォルトのエラー数を設定します。1 から 255 までの数値を入力します。デフォルトは 10 です。

コマンド再試行回数

メモ

Analytics スクリプトにのみ適用されます。Analytics のユーザー インターフェイスには適用されません。

最初の試行が失敗した場合、Analytics がデータのインポートまたはエクスポートを試行する回数を指定します。0 から 255 までの数値を入力します。0 を入力すると、最初の失敗後に追加の試行は行われません。デフォルトは 0 です。

再試行の間の待ち時間はありません。各連続した試行は、直前の失敗直後に行われます。

再試行を指定する機能は、データベースまたはクラウド データ サービスに接続する場合に便利ですが、一時的に利用できない可能性があります。

この設定 への変更は、次の構文を使用してログに記録されます。

```
SET RETRY 数
```

以下のコマンドに適用されます:

インポート	<ul style="list-style-type: none"> ◦ ACCESSDATA ◦ IMPORT GRCPROJECT ◦ IMPORT GRCRESULTS ◦ REFRESH <p>(ACCESSDATA または IMPORT SAP を使って最初に作成されたテーブルの場合のみ)</p>
SAP (Direct Link)	<ul style="list-style-type: none"> ◦ IMPORT SAP ◦ RETRIEVE
エクスポート	<ul style="list-style-type: none"> ◦ EXPORT . . . ACLGRC <p>(HighBond のリザルト へのエクスポート)</p>

最大カテゴリ

このオプションは、学習コマンドの入力として使用される文字キーフィールドで発生する一意の値の最大数を指定します。1 から 255 までの数値を入力します。

通知設定

再試行回数

このオプションは、通知操作が最初の試行に失敗した後、電子メールの送信を試行する回数を指定します。0 から 255 までの数値を入力します。0 を入力すると、最初の失敗後に追加の試行は行われません。デフォルトは 5 です。

通知操作が電子メールの送信に失敗することについて考えられる理由の 1 つは、電子メールサーバーを利用できないということです。

再試行間隔 (秒)

このオプションは、電子メールの送信を最初に失敗した後、追加で試行する間隔を秒単位で指定します。1 から 255 までの数値を入力します。デフォルトは 10 秒です。

通知に失敗したらスクリプトを停止する

このオプションをオンにした場合、Analytics は通知操作が失敗するとスクリプトの処理を停止します。スクリプトは、最初の失敗後に停止するか、または **再試行回数** に指定された試行回数後、どの試行も成功しなかった場合に停止します。デフォルトの設定はオフで、この場合は、通知操作が失敗してもスクリプトを続行することができます。

無効な電子メール受信者は、通信操作の失敗と見なされないため、**通知に失敗したらスクリプトを停止する** の設定に関係なく、これによりスクリプトが停止されることはありません。

日付と時刻オプション

日付と時刻 タブのオプションを使用すると、Analytics で日付、日付時刻、および時刻を表示する方法の指定や、日付と時刻に関するその他のオプションの設定を行います。

日付の設定

日、月、年

日]、**月]**、**年]** テキスト ボックスを使用して、日付および日付時刻書式の年月日の構成要素を表す文字を指定します。デフォルトの値は、日は "D"、月は "M"、年は "Y" ですが、英語以外の言語の場合は別の文字を指定できます。指定する文字は、大文字でそれぞれ異なる文字でなければなりません。また、"D"、"M" および "Y" を使用できるのはそれぞれのデフォルトの位置だけです。

日付の表示書式

このオプションでは、Analytics でビュー、レポート、およびエクスポート ファイルにおける日付と、日付時刻の日付部分を表示する方法を指定できます。これはログ エントリの日付の書式も制御します。**日付の表示書式** ドロップダウン リストから書式を選択する、または独自の日付表示書式を作成することもできます。

メモ

このオプションは、Analytics がデータソースから日付を読み取る方法には影響しません。Analytics による日付の読み取り方法を指定するには、**データ定義ウィザード**、または **テーブルレイアウト** ダイアログ ボックスの **書式** フィールドを使用します。詳細については、「ソース データに含まれる日付および時刻データの書式」 ページ 366を参照してください。

日付の表示書式]ドロップダウン リストに含まれる書式

選択する表示書式	表示内容
DD/MM/YY	31/12/14
DD/MM/YYYY	31/12/2014
MM/DD/YY	12/31/14
MM/DD/YYYY	12/31/2014
MMM DD, YYYY	Dec 31, 2014

選択する表示書式	表示内容
YYYYDDD	2014365
YYYY-MM-DD	2014-12-31

独自の日付表示書式の作成

独自の日付表示書式を作成するには、「日付の表示書式の文字」下内の文字(日、月、年がデフォルトの書式文字列であることが前提)を組み合わせて作成した書式を「日付の表示書式」テキストボックスに入力します。

以下のガイドラインが適用されます。

- 日、月、年の各構成要素は、スペースを含めずに表示することも、スペースや句読点を使用して区切ることもできます。
- 各構成要素はどのような順序でも表示できます。
- 3つの構成要素のうち、1つまたは2つを省くことができます。
- 構成要素は2文字未満にすることはできません。
- 書式の指定には大文字を使用する必要があります。
- 日付の表示書式は全体で、スペースや句読点を含め12文字以内で指定する必要があります。

日付の表示書式の文字

指定する表示書式	表示内容
DD	日(1 ~ 31)
DDD	日(1 ~ 366)
MM	月(1 ~ 12)
MMM	月(Jan ~ Dec)
YY	年(00 ~ 99)
YYYY	年(1900 ~ 9999)

メモ

全部が表示されない日付表示書式を指定した場合、日付または日付時刻によるクリックフィルターは機能しません。たとえば、日、月、年のデータを持つ日付に対して **MMM YYYY** という書式を指定すると、ビュー内で日付、または日付時刻に対してクリックフィルターを実行しても結果には何も返りません。

日付の表示書式への変更は、次の構文を使用してログに記録されます。

SET DATE 値

世紀を解釈する開始年

データファイルの多くは、日付の年を表すのに2桁の数字のみを使用しています。これは、年に対応する世紀が未指定であることを意味します。20世紀に割り当てられている最も古い年を示す2桁の年が、データファイルのセットによっては異なることがあります。この年は、多くの場合、「世紀を解釈する開始年」または「ピボット年」と呼ばれます。

ピボット年は、2桁の年だけに適用され、4桁で年を表すデータへの影響はありません。Analytics では、1900 から 9999 までの4桁の年を読み取ることができます。

世紀を解釈する開始年のデフォルト設定は40です。この設定の場合、Analytics は、2桁の年40から99までを1940年から1999年と解釈し、2桁の年00から39までを2000年から2039年と解釈します。

このピボット年を変更するには、0から99までの数字を入力します。たとえば、ピボット年として1950年を設定する場合は、**世紀を解釈する開始年**テキストボックスに「50」と入力します。次の表では異なるピボット年の例を示します。

世紀を解釈する開始年の設定	ソースデータ内の年	解釈
00	00 ~ 99	1900 ~ 1999
40	40 ~ 99、00 ~ 39	1940 ~ 1999、2000 ~ 2039
50	50 ~ 99、00 ~ 49	1950 ~ 1999、2000 ~ 2049
99	99、00 ~ 98	1999、2000 ~ 2098

世紀を解釈する開始年の年とは異なるピボット年を使用しているデータファイルで作業する場合は、式を使用して、2桁の年を正確に解釈したり、4桁の年に変換したりする演算フィールドを作成することができます。

この **世紀を解釈する開始年** 設定への変更は、次の構文を使用してログに記録されます。

```
SET CENTURY 値
```

年齢調べ間隔

このオプションは、**年齢調べ**ダイアログボックスのデフォルトの年齢調べ間隔を設定します。特定の年齢調べ間隔のセットを頻繁に使用する場合は、そのセットを **年齢調べ間隔**テキストボックスに入力しておくことができます。そうすると、Analytics は **年齢調べ**ダイアログボックスのデフォルトの年齢調べ間隔として、この設定を使用します。必要であれば、**年齢調べ**ダイアログボックスの年齢調べ間隔を無効にすることができます。

期間は日付単位で、スペースを入れずにカンマで区切って入力します。年齢調べ間隔は必要な数だけ設定できます。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET PERIODS 値
```

月の名前の省略形

このオプションでは、MMM 形式の月名のデフォルトの省略形を設定します。月名の省略形は3文字で、1月から正しい順番に、スペースを入れずにカンマで区切って設定する必要があります。

このオプションは、Analytics がテーブルから日付を読み取る方法に作用するので、これによってビュー、レポート、および出力ファイルでの日付の表示方法を変えることができます。たとえば、入力ファイルの MMM 形式の日付がフランス語である場合は、次のようにフランス語の月の省略形を指定する必要があります。

`Jan,Fév,Mar,Avr,Mai,...`

Analytics は、文字列 `Fév` は2月、`Avr` は4月と正しく解釈します。MMM を使って月を表示する日付の表示書式を選択すると、Analytics は提供された省略形を使用して、ビュー、レポート、出力ファイルで省略形の月名を表示します。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET MONTHS 値
```

時刻の設定

時、分、秒

`時]`、`分]`、`秒]`テキスト ボックスを使用して、時刻および日付時刻書式の時、分、秒の構成要素を表す文字を指定します。デフォルトの値は、`時`は"h"、`分`は"m"、`秒`は"s"ですが、英語以外の言語の場合は別の文字を指定できます。指定する文字は、小文字でそれぞれ異なる文字でなければなりません。また、"h"、"m" および "s" を使用できるのはそれぞれのデフォルトの位置だけです。

時刻の表示形式

このオプションでは、Analytics での、ビュー、レポート、およびエクスポート ファイルにおける時刻、日付時刻の時刻部分の表示方法を指定できます。`時刻の表示書式]`ドロップダウン リストから書式を選択する、または独自の時刻表示書式を作成することもできます。

メモ

このオプションは、Analytics がデータソースから時刻を読み取る方法には影響しません。Analytics による時刻の読み取り方法を指定するには、`データ定義ウィザード`、または `テーブルレイアウト` ダイアログ ボックスの `書式]`フィールドを使用します。詳細については、「ソース データに含まれる日付および時刻データの書式」 ページ 366を参照してください。

時刻の表示書式]ドロップダウンリストに含まれる書式

選択する表示書式	表示内容	表示内容
hh:mm	23:59	11:59
hh:mm P	11:59 P	11:59 A
hh:mm PM	11:59 PM	11:59 AM
hh:mm:ss	23:59:59	11:59:59
hh:mm:ss P	11:59:59 P	11:59:59 A
hh:mm:ss PM	11:59:59 PM	11:59:59 AM
hh:mm:ss±hh:mm	23:59:59-05:00	11:59:59-05:00

独自の時刻表示書式の作成

独自の時刻表示書式を作成するには、"時刻の表示書式の文字" 下内の文字(時、分、秒がデフォルトの書式文字列であることが前提)を組み合わせて作成した書式を **時刻の表示書式]**テキストボックスに入力します。

以下のガイドラインが適用されます。

- 時、分、秒の各構成要素は、スペースを含めずに表示することも、スペースや句読点を使用して区切ることもできます。
- 各構成要素は、時、分、秒の順序で表示する必要があります。
- 構成要素の"秒"は省くことができます。構成要素の"時"と"分"は省くことができません。
- 構成要素は2文字でなければなりません。
- 書式の指定には小文字を使用する必要があります。
- オプションのAM/PMインジケータを指定すると、時刻表示は24時間制から12時間制に切り替わります。AM/PMインジケータは、時刻の構成要素の後ろであればどこでも置くことができ、必要であれば、インジケータの前にスペースを1つ置くこともできます。
- UTCオフセットは、前にプラス記号(+)またはマイナス記号(-)を付ける必要があります。
- 分の要素はUTCオフセットから省くことができます。表示する時刻データのいずれかにUTCオフセットが含まれており、データが整数時間でない場合は、分の要素を省かないでください。
- 時刻の表示書式は全体で、スペース、句読点、プラスやマイナス記号を含め14文字以内で指定する必要があります。

時刻の表示書式の文字

指定する表示書式	表示内容
hh	時(00 ~ 23)
mm	分(00 ~ 59)

指定する表示書式	表示内容
ss	秒(00 ~ 59)
A または P	AM/PM インジケータ- (A および P)
AM または PM	AM/PM インジケータ- (AM および PM)
+ または -	UTC オフセット インジケータ- (+ および -)

メモ

全部が表示されない時刻表示書式を指定した場合、日付時刻または時刻によるクイックフィルターは機能しません。たとえば、時、分、秒のデータを持つ時刻に対して `hh:mm` という書式を指定すると、ビュー内で日付時刻または時刻に対してクリックフィルターを実行しても結果には何も返りません。

時刻の表示書式への変更は、次の構文を使用してログに記録されます。

SET TIME 値

UTC オフセット付きの時刻を UTC として表示

Analytics では、UTC オフセット (後で説明があります) を含むローカル時刻データを受け付けます。たとえば、`10:30:15-05:00` (`-05:00` が UTC オフセット) などのデータです。UTC (協定世界時) は経度 0 度地点における時刻で、UTC オフセットはタイムゾーン インジケータ-です。

[UTC オフセット付きの時刻を UTC として表示] オプションを使用すると、Analytics によって、ローカル時刻を UTC オフセットなしの UTC に変換するか (デフォルト設定)、あるいは、UTC ベースの時刻データを UTC オフセット付きのローカル時刻として表示するかを指定できます。たとえば、同じ UTC ベースのデータに対して、次のように 2 つの異なる表示方法を用いることができます。

- `31/12/2014 15:30:15`
(**[UTC オフセット付きの時刻を UTC として表示]** オプションをオン - デフォルトの設定)
- `31/12/2014 10:30:15-05:00`
(**[UTC オフセット付きの時刻を UTC として表示]** オプションをオフ)

[UTC オフセット付きの時刻を UTC として表示] オプションがオンの場合、Analytics では時刻データに UTC オフセットを反映させ、時刻データを適切な時間数で調整します。上の例では、UTC への変換によってローカル時刻データを 5 時間進めました。

ローカル時間の UTC への変換は、表示目的のためだけであり、ソースデータには影響しません。変換後も UTC オフセットは含まれます。この 2 つの異なる表示モードは、いつでも変更して切り替えることができます。

Analytics が UTC オフセット付きのローカル時刻に対して計算を実行する場合は、その UTC オフセットが自動的に反映され、ローカル時刻に相当する UTC に対して計算が実行されます。**[UTC オフセット付きの時刻を UTC として表示]** オプションがオンの場合は、計算で使用される実際の時刻データが見られ、

結果がわかりやすくなります。詳細については、"UTC オフセットが時刻日付式に与える影響" ページ 899 を参照してください。

UTC について

UTC は、グリニッジ標準時 (GMT) に取って代わるグローバルな時間標準です。これら 2 つの標準は、ほぼ同じものであると言えます。UTC ベースの時刻データの最後の部分 (`-05:00` または `+01:00` など) は **UTC オフセット** で、ローカル時刻が UTC に対して何時間進んでいる、または遅れているかを示しています。例:

- `31/12/2014 10:30:15-05:00` は、2014 年 12 月 31 日 10:30:15 AM、東部標準時 (北米) を表しています。
- `31/12/2014 15:30:15` (UTC) は、経度 0 度地点における時刻で同じ時点を表しています。

UTC ベースの日付時刻データで、UTC への変換によって午前零時の境界をまたいで進むまたは戻るような場合は、日付が 1 日調整されます。

メモ

UTC オフセットは**タイムゾーンオフセット**とも呼ばれますが、これらは完全に同じものではありません。複数のタイムゾーンが同じ UTC オフセットを持つ可能性があります。

Analytics における UTC ベースおよび非 UTC の時刻データの表示方法

UTC ベースの時刻データ

ソースデータ	[UTC オフセット付きの時刻を UTC として表示] オプションをオン (デフォルトの設定)	[UTC オフセット付きの時刻を UTC として表示] オプションをオフ
時刻の表示書式 = hh:mm:ss		
31/12/2014 10:30:15-05:00	31/12/2014 15:30:15	31/12/2014 10:30:15-05:00
01/01/2015 00:30:15+01:00	31/12/2014 23:30:15	01/01/2015 00:30:15+01:00
時刻の表示書式 = hh:mm:ss+hh:mm		
31/12/2014 10:30:15-05:00	31/12/2014 15:30:15+00:00	31/12/2014 10:30:15-05:00
01/01/2015 00:30:15+01:00	31/12/2014 23:30:15+00:00	01/01/2015 00:30:15+01:00

非 UTC の時刻データ

ソース データ	[UTC オフセット 付きの時刻を UTC と して表示] オプションをオン(デフォルトの 設定)	[UTC オフセット 付きの時刻を UTC と して表示] オプションをオフ
時刻の表示書式 = hh:mm:ss		
31/12/2014 10:30:15	31/12/2014 10:30:15	31/12/2014 10:30:15
時刻の表示書式 = hh:mm:ss+hh:mm		
31/12/2014 10:30:15	31/12/2014 10:30:15+00:00	31/12/2014 10:30:15+00:00

数値オプション

数値] タブのオプションを使用して、Analytics による数値データの処理方法および表示方法を指定します。

数値のオーバーフローで停止する

このオプションがオンの場合、Analytics は数値のオーバーフローが発生すると処理を停止します。数学演算では、中間の計算も含めて、結果が 22 桁を超える場合には、Analytics が処理を停止することがあります。ビューでは、影響を受けたフィールドに `### ERR ###` と表示されます。

このオプションをオフにすると、Analytics は処理を続行しますが、超過した桁を左端から切り捨てるため、計算結果は不正確なものとなります。ユーザーがゼロで除算しようとした場合、Analytics はその結果に非常に大きな数値を代入します。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET OVERFLOW {ON|OFF}
```

データを検証する

このオプションをオンにすると、テーブルを開いている間フィールドを処理するたびに、データフィールドの内容がテーブルレイアウトのフィールドのデータ型(文字、数値、日付時刻、論理など)に対応しているかどうか自動的に検証されるようになります。**無効なデータを空白にする]** オプションもオンにしない限り、エラーが発生すると処理は停止します。ビューでは、影響を受けたフィールドに `### ERR ###` と表示されません。

このオプションをオフにすると、Analytics はデータの妥当性を検証しなくなるので、処理速度が改善されます。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET VERIFY {ON|OFF|BLANK}
```

無効なデータを空白にする

データを検証する] オプションをオンにした場合のみ、このオプションを使用できます。**無効なデータを空白にする]** をオンにすると、Analytics は自動的に無効な文字データを空白に、無効な数値データをゼロに置き換えます。

このオプションがオンの場合に、無効なデータを含んでいるフィールドを処理すると、Analytics は Analytics プロジェクトファイル (`*.acl`) と同じフォルダーにエラーログを作成し、"ファイル内に不正なデータがあります。ERROR ログを参照してください。" というメッセージを表示します。ログを表示または印刷する場合は、ワードプロセッサやテキストエディターを使用して、作業ファイルを保存しているディレクトリにあるログを開きます。Analytics で "DUMP コマンド" ページ 1760 を使用して、エラーログにアクセスできます。

この設定への変更は、次の構文を使用してログに記録されます。ここで、BLANK は、このオプションが選択されていること、ON は、**データを検証する**]オプションは選択されているが、**無効なデータを空白にする**]オプションは選択されていないことを意味します。

```
SET VERIFY (BLANK|ON)
```

式フィールドの幅

このオプションには、数値演算フィールドや一時的に作成した数式の表示幅のデフォルト値(文字)を指定します。Analytics は最大幅を指定することはできません。デフォルトは 12 文字で、これは Analytics のデフォルトのアプリケーション フォントを基準とします。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET WIDTH 文字
```

小数点の記号

Analytics では、デフォルトの小数点の文字としてピリオドを使用します。このデフォルトの設定は、カンマまたはスペースに変更することができます。テキスト ボックスに、変更する文字を入力してください。3 つの区切り文字(小数点、桁区切り、リスト)のうち、小数点の記号は一意である必要があります。

この設定への変更は、**デフォルトの数値書式**]への変更としてログに記録されます。

デフォルトの数値書式

デフォルトでは、Analytics は数値を表示する際、桁区切り記号を使用せず、先頭にマイナス記号を付けて負の数値を示す数値書式を使用します。

デフォルトの数値書式]オプションを変更すると、別の書式を使って数値を表示することができます。ドロップダウン リストからあらかじめ定義されている書式のうちの 1 つを選択することも、独自の書式を作成することもできます。たとえば、桁区切り記号を指定する書式や、負の数値に別のインジケータを使用する書式を選択してもかまいません。

指定した書式は、全アプリケーションのデフォルトの書式になります。フィールド レベルまたは列レベルの書式が指定されていない、すべての数値フィールドおよび列に適用されます。通貨記号の入った書式を指定すると、通貨でないフィールドも含め、すべての数値フィールドが通貨記号を持つようになってしまうので、そのような書式は指定しないでください。必要に応じて、フィールド レベルまたは列レベルで通貨フィールドを書式設定することができます。フィールド レベルおよび列レベルにおける数値の書式設定の詳細については、"ビューにおける数値の書式" ページ 852を参照してください。

この設定への変更は、次の構文を使用してログに記録されます。

```
SET PICTURE 書式
```

桁区切り記号

Analytics では、数値出力のデフォルトの3桁の区切り記号としてカンマを使用します。このデフォルトの設定は、ピリオドまたはスペースに変更することができます。テキスト ボックスに、変更する文字を入力してください。桁区切り記号と小数点の記号を同じものにすることはできません。

リストの区切り文字

Analytics では、デフォルトのリスト区切り文字としてカンマを使用します。主に関数パラメーターを区切るのに使用されます。このデフォルトの設定は、セミコロン (;) またはスペースに変更することができます。テキスト ボックスに、変更する文字を入力してください。リストの区切り文字と小数点の記号を同じものにすることはできません。

印刷オプション

[印刷] タブのオプションを使用して、レポートのデフォルトの印刷設定や、印刷結果の余白設定を指定します。

レポートにレポート履歴を含める

このオプションがオンの場合は、レポートの最終ページにレポート履歴が追加されます。レポート履歴には、Analytics プロジェクト名、テーブル名やデータファイル名と、レポートの生成に使用された REPORT コマンド、テーブルレイアウト ノート、およびテーブル履歴が含まれています。

テーブル履歴にフィールド定義を含める

このオプションがオンの場合は、テーブルレイアウト内の各物理データフィールドおよび演算フィールドのフィールド定義がレポートに追加されます。フィールド定義にはフィールドノートが含まれます。このオプションは、[レポートにレポート履歴を含める] オプションも選択されていない限り、効果はありません。

レポート履歴にビューノートを含める

このオプションがオンの場合は、アクティブなビューに関連付けられているすべてのビューノートがレポートに追加されます。このオプションは、[レポートにレポート履歴を含める] オプションも選択されていない限り、効果はありません。

余白

[左余白]、[止余白]、[右余白]、[下余白] テキストボックスを使用して、すべての印刷結果で使用される余白を指定できます。値を変更するには、テキストボックスに新しい値を入力するか、またはテキストボックスの横にある上下矢印をクリックして、値を増減させます。

プリンターの印刷可能領域よりも大きい余白を指定すると、プリンターの印刷可能領域が余白として使用されます。

それぞれ個々の余白設定への変更は、次の構文を使用してログに記録されます。

```
SET MARGIN {LEFT|RIGHT|TOP|BOTTOM} 値
```

アプリケーション フォント オプション

[アプリケーション フォント] タブのオプションを使用して、表示領域のビュー タブを除く、すべてのウィンドウでデータの表示に使用するフォントを指定します。

固定幅フォント

Analytics では、[テーブルレイアウト]、[スクリプト]、[ワークスペース] ウィンドウで表示される情報には固定幅フォントを使用します。固定幅フォントのデフォルト設定は "MS ゴシック" です。リスト ボックスから別のフォントを選択できます。

プロポーショナルフォント

Analytics では、ビューやレポートでプロポーショナルフォントを使用します。プロポーショナルフォントは、ステータス バーに表示されるプロジェクト ファイル名、テーブル、レコード数などの情報を表示するのにも使用されています。プロポーショナルフォントのデフォルト設定は "MS Shell Dlg" です。リスト ボックスから別のフォントを選択できます。

言語バージョン

Analytics では、フィールド名に文字、数字、アンダースコア文字を使用することができます。フィールド名のデフォルトの標準言語バージョンは、西ヨーロッパ言語の文字に対応しています。設定を "タイ語" にすると、英語に加えてタイ語の文字も使用できます。

スクリプト エディターの設定

Analytics ではスクリプト エディターの背景色およびフォント スタイルをカスタマイズすることができます。スクリプト エディターの背景色、デフォルトのテキスト、コメント、コマンド、パラメーターおよび関数キーワードの色とスタイルを選択できます。スクリプト エディターのカスタマイズの詳細については、"スクリプト エディターのカスタマイズ" ページ 1602を参照してください。

Analytics の初期設定ファイルの動作

メモ

初期設定ファイルの動作は Analytics バージョン 10.0 で変更されました。このトピックでは、バージョン 10.0 以降で初期設定ファイルがどのように動作するかを説明します。

Analytics における構成可能なオプションの設定 (**オプション**] ダイアログボックスの各種設定) は、**ac1win16.prf** (非 Unicode 版) または **ac116.prf** (Unicode 版) という初期設定ファイル (.prf ファイル) に保存されます。

オプション] ダイアログ ボックスで行った変更は、自動的に .prf ファイルへ保存されます。それらの変更は、再度明確に変更しない限り有効です。

グローバルとプロジェクト固有の初期設定ファイル

1 つのグローバル .prf ファイルで、Analytics とすべての Analytics プロジェクトの動作を管理できます。また、プロジェクト単位で設定をカスタマイズする方法として、個々の Analytics プロジェクトに別々の .prf ファイルを関連付けることもできます。たとえば、プロジェクトごとに異なる日付の表示書式を指定する、あるいは、あるプロジェクトではテーブルレイアウトを削除したときにソース データファイルを削除するが、別のプロジェクトでは削除しない、と指定することができます。

Analytics スクリプト作成者向けの警告

Analytics スクリプトの作成者が、自分のスクリプトを他の Analytics ユーザーに提供し、そのユーザーのコンピュータ上で Analytics 初期設定ファイルを制御する必要がある場合には、初期設定ファイルについて理解しておく必要があります。

グローバルな初期設定ファイル

Analytics をインストールすると、次の場所にデフォルトの設定 (出荷時の設定) で .prf ファイルが作成されます。

- アプリケーション データ フォルダー - **C:\ユーザー\< ユーザー アカウント名 >\AppData\Local\ACL** (Windows 10 の場合の場所)

メモ

アプリケーション データ フォルダーは、Windows オペレーティングシステムによって非表示になっている場合もあります。必要であれば、Windows のフォルダー オプションで、非表示になっているファイルとフォルダーを表示させるオプションを有効にしてください。

アプリケーション データ フォルダーにある .prf ファイルには、Analytics のグローバルな初期設定が含まれています。[オプション] ダイアログ ボックスで行った変更は、プロジェクト固有の .prf ファイルを使用していない限り、このグローバルな .prf ファイルに保存されます。

グローバル .prf ファイルは次のときに使用されます。

- Analytics プロジェクトを開かずに Analytics を開始するとき
- プロジェクト固有の .prf ファイルがないプロジェクトを開くとき
- Analytics を終了せずにプロジェクトを閉じるとき

ユーザー単位のグローバル初期設定

複数のユーザー アカウントが、同一コンピューター上の Analytics にアクセスする場合は、ユーザー アカウントごとのアプリケーション データ フォルダーにそれぞれ .prf ファイルが存在し、ユーザーがそれぞれ独自のグローバル初期設定を持ちます。

グローバル .prf ファイルの自動再生成

アプリケーション データ フォルダーにあるグローバル .prf ファイルを削除したり、名前を変更したりした場合、あるいは、このファイルが何らかの理由で使用できなくなった場合は、Analytics を開始するときに、自動的にデフォルト設定の新しい .prf ファイルがアプリケーション データ フォルダーに作成されます。

プロジェクト固有の初期設定ファイル

1 つ以上の Analytics プロジェクトの初期設定ファイルをカスタマイズする場合は、グローバルな .prf ファイルを、アプリケーション データ フォルダーから個々の Analytics プロジェクトが含まれるフォルダーへ手動でコピーすることができます。コピーされた .prf ファイルはプロジェクト固有の .prf ファイルになります。

注意

グローバルな .prf ファイルをコピーするときに、誤って移動(切り取り)してしまわないよう注意してください。グローバルな .prf ファイルを移動してしまうと、指定していたグローバルな初期設定が失われ、デフォルト設定のものに置き換わります。

メモ

別のバージョンの Analytics が一緒にインストールされている場合は、正しいバージョンの .prf ファイルをコピーすることを確認します。

Analytics プロジェクト ファイル(拡張子が .acl) とプロジェクト固有の .prf ファイルを関連付けるには、それら 2 つのファイルが同じフォルダー内に存在する必要があります。プロジェクトが開いているときは、プロ

プロジェクト固有の .prf ファイルで指定された初期設定が使用されます。[オプション] ダイアログ ボックスで行った変更は、グローバルな .prf ファイルではなく、プロジェクト固有の .prf ファイルに保存されます。

プロジェクト固有の .prf ファイルの利点

プロジェクト固有の .prf ファイルを持つ利点は、特定のプロジェクトやプロジェクト内のスクリプトの要件に基づいて設定をカスタマイズできることです。

Analytics プロジェクトを別のユーザーへ送るときにプロジェクト固有の .prf ファイルも含めれば、そのユーザーがプロジェクト内のスクリプトを実行したときに、結果が自分のコンピューターでの実行結果と同じであることを保証できます。

同時に、その .prf ファイルはプロジェクト固有のものなので、ほかのユーザーのグローバルな初期設定の影響を受けません。

グローバル .prf ファイルに戻す

プロジェクト固有の .prf ファイルを削除または名前変更すれば、いつでも再びプロジェクト用のグローバルな .prf ファイルを使用できるようになります。

使用されている初期設定ファイルの特定

Analytics では、[オプション] ダイアログ ボックスの下部で、現在アクティブな .prf ファイルのパスが表示されます。アプリケーション データ フォルダーへのパスが表示される場合は、グローバルな .prf ファイルが使用されています。Analytics プロジェクト フォルダーへのパスが表示される場合は、プロジェクト固有の .prf ファイルが使用されています。

デフォルトの初期設定に戻す

[オプション] ダイアログ ボックスの下部にある **既定値に戻す** ボタンをクリックすると、デフォルトの初期設定に戻すことができます。**既定値に戻す** をクリックすると、現在開いているアクティブなタブだけでなく、[オプション] ダイアログ内のすべてのタブの全オプションをデフォルトの設定に戻します。このデフォルト設定への復帰が適用されるのは、[オプション] ダイアログ ボックスの下部に示される現在アクティブな .prf ファイルに対してのみです。

初期設定ファイルの優先度

Analytics プロジェクトを開く

Analytics プロジェクトを開く場合は、以下の優先順位で .prf ファイルが開かれます。

1. Analytics プロジェクトが含まれるフォルダーにあるプロジェクト固有の .prf ファイル
2. プロジェクト固有の .prf ファイルがない場合は、アプリケーション データ フォルダーに .prf ファイルを読み込みます。
3. グローバル .prf ファイルがない場合は、デフォルトの設定 (出荷時の設定) を使用して、アプリケーション データ フォルダーにグローバル .prf ファイルを自動的に再作成し、そのファイルを読み込みます。

Analytics プロジェクトを開かない

プロジェクトを開かないで Analytics を開始する場合は、以下の優先順位で .prf ファイルが読み込まれます。

1. アプリケーション データ フォルダーのグローバル .prf ファイル
2. グローバル .prf ファイルがない場合は、デフォルトの設定 (出荷時の設定) を使用して、アプリケーション データ フォルダーにグローバル .prf ファイルを自動的に再作成し、そのファイルを読み込みます。

Analytics セッション中に設定を作成する

SET コマンドを使用すると、Analytics セッション中にのみ有効な一時設定を作成できます。たとえば、`SET DATE "DD MMM YYYY"` では、一時的に日時表示形式が変更されます。

この動作は、SET コマンドの使用場所が Analytics コマンド ラインであろうと Analytics スクリプトであろうと適用されます。

Analytics を閉じるとすぐに、該当する .prf ファイルに保存されている設定に戻ります。SET コマンドで .prf ファイルの内容が変更されることはありません。

詳細については、"SET コマンド" ページ 2081を参照してください。

ビューおよびレポートのフォント設定の変更

ビューおよびレポートでのテキスト表示に使用されるフォント設定をカスタマイズするには、**ビューフォントの選択** ダイアログボックスを使用します。ビューやレポートの各領域で使用されるフォントを個別にカスタマイズするか、あるいは一度にすべての領域のフォント設定を変更することができます。

フォントの設定を変更した場合、その変更は現在のビューとそのビューに基づくレポートにのみ作用します。この設定は Analytics プロジェクトに追加され、Analytics セッションが変わっても存続します。新しいビューを作成すると、現在開かれているビューがどのようなものであっても、そのビューのフォントの設定を複製します。必要であれば、新規ビューの作成直後にフォントの設定を変更することができます。

1. 次のいずれかを実行します。
 - 表示領域にある **フォントの変更**  をクリックします。
 - ビュー内で右クリックしてから **ビューフォントの選択** を選択します。
2. **表示フォントを選択** をダイアログボックスで、次のボタンのいずれかをクリックします。
 - **すべて** - 個々の領域に対して限定的に設定が適用されない限り、ここで選択したフォント設定が、ビューおよびレポートのすべての領域に適用されます。
 - **ヘッダー** - レポート ヘッダーのフォント設定。
 - **タイトル** - 列の見出しのフォント設定。
 - **データ** - ビューおよびレポート データのフォント設定。
 - **合計** - レポートの小計のフォント設定。
 - **フッター** - レポート フッターのフォント設定。
3. **フォント** ダイアログボックスで、フォント設定に必要な変更を加えて **OK** をクリックします。
4. さらに領域のフォント設定を変更したい場合は、手順 2 と 3 を繰り返します。
5. **OK** をクリックします。

ビューのフォント サイズを変更する

Analytics プロジェクトのビュー内のフォント サイズは一時的に大きくまたは小さくすることができます。このフォント サイズの変更は、プロジェクト内のすべてのビューに適用され、サイズをリセットするか、Analytics を終了するまで維持されます。

1. 開いているビュー内でクリックします。
2. 次のいずれかを実行します。
 - **Ctrl** キーを押したまま **+** キー(テンキーのプラスキー)を繰り返し押すと、フォント サイズが大きくなります。
 - **Ctrl** キーを押したまま **-** キー(テンキーのマイナスキー)を繰り返し押すと、フォント サイズが小さくなります。
 - **Ctrl** キーを押したまま **0** キー(テンキーの0キー)を押すと、フォント サイズがデフォルトのサイズにリセットされます。

メモ

メインキーボードではなく、テンキーのプラスキー、マイナスキーおよび0キーを使用してください。ノート型パソコンでは、**Fn+Ctrl+** にテンキー上の該当するキーを押してください。

Analytics ツールバーのカスタマイズ

よく使う機能のボタンを追加したり、ほとんど使わない機能のボタンを削除したりして、Analytics ツールバーをカスタマイズすることができます。また、ツールバー内のボタンの順序を変えたり、関連するボタンをグループ化するために区切りを追加したりできます。

1. 必要に応じて、**[ウィンドウツールバー]**を選択してツールバーを表示します。
2. ツールバー上の何もない場所をダブルクリックすると、**[ツールバーの変更]**ダイアログボックスが開きます。
3. 次の手順を完了します。
 - **利用できるツールバー ボタン]**リストでボタンを選択し、**[追加]**をクリックしてツールバーに追加します。
 - **現在のツールバー ボタン]**リストでボタンを選択し、**[削除]**をクリックしてツールバーから削除します。
 - **利用できるツールバー ボタン]**リストにある **[区切り]**ボタンを選択して **[追加]**をクリックし、関連するボタンを視覚的にグループ化するための縦棒を挿入します。縦棒は必要だけ追加できます。
 - **現在のツールバー ボタン]**リストでボタンを選択し、**[上へ]**または **[下へ]**をクリックして、ボタンの位置を変更します。上から下へのボタンの順序は、ツールバーの左から右への位置に対応しています。

ツールバーは、変更を加えるに従って動的に更新されます。

4. 必要に応じて、**[リセット]**をクリックし、ツールバーに行ったすべての変更を元に戻せます。

メモ

一度 **[閉じる]**をクリックすると、変更が保存されて、**[リセット]**をクリックしても変更が破棄されなくなります。**[ツールオプション既定値に戻す]**を選択すれば、デフォルトのツールバー設定に戻すことができます。

5. **[閉じる]**をクリックして、変更を保存します。

Analytics のメインメニューへのカスタム項目の追加

独自の Analytics スクリプトや頻繁に使用されるコマンドなどのカスタム項目を、Analytics のメインメニューに追加できます。カスタム項目は、**アプリケーション**メニューの下に表示されます。この機能は、他のユーザーが使用するスクリプトを作成し、スクリプトのアクセスするための単一の使いやすい場所が必要な場合に、特に有用です。

プロジェクト レベルまたはグローバル アクセス

個別の Analytics プロジェクトに制限されるカスタムメニュー項目を作成するか、Analytics が開くたびにグローバルで項目を使用可能にすることができます。

- **プロジェクト レベル アクセス**-Analytics プロジェクト (*.acl) と同じフォルダーのカスタムメニュー項目を含むテキスト ファイル (*.mnu) を見つけます
- **グローバル アクセス**-Analytics の実行ファイル (ACLWin.exe) と同じフォルダーのカスタムメニュー項目を含むテキスト ファイル (*.mnu) を見つけます

また、アプローチを結合し、プロジェクト レベルとグローバルカスタムメニュー項目の両方を作成することもできます。

メモ

他の Analytics ユーザーがカスタムメニュー項目を使用する場合は、*.mnu ファイルに、ファイルを検索する場所に関する説明を付加します。

構成可能なメニューファイル (*.mnu)

.mnu ファイル拡張子の 1 つ以上のテキスト ファイルを使用し、カスタムメニュー項目を作成します。

サブメニュー エントリ

各 *.mnu ファイルは、**アプリケーション**メニューの下に、個別のサブメニュー エントリを作成します。たとえば、**Account scripts.mnu** は、**アカウント スクリプト** サブメニュー エントリと次のメニュー構造を作成します。**アプリケーション > アカウント スクリプト**。

サブメニュー エントリは、**アプリケーション**メニューにアルファベット順で表示されます。

カスタム メニュー項目

個別のカスタムメニュー項目は、各サブメニュー エントリ内に含まれます。たとえば、次のスクリプトを実行する2つの項目を作成できます。

- 買掛金分析
(アプリケーション > アカウント スクリプト > 買掛金分析)
- 売掛金分析
(アプリケーション > アカウント スクリプト > 売掛金分析)

カスタムメニュー項目は、項目が *.mnu ファイルにリストされている順序でサブメニューに表示されます。要件に合わせて、複数の *.mnu ファイルを使用し、複数レベルのカスケード サブメニューを作成できます。

ヒント

サブメニューレベルが多すぎると、ユーザーが混乱する可能性があります。サブメニュー項目を3つに制限することをお勧めします。

カスタムメニュー項目の最大数

アプリケーションメニューには、最大 179 個のカスタムメニュー項目を含めることができます。最大数は、すべてのサブメニューレベルとすべての *.mnu ファイルにわたるカスタムメニュー項目の合計数に適用されます。

メニュー ファイル構文

メモ

メニューファイルの必要な構文は、正確に指定する必要があります。余分なスペースが1つあるだけでも、Analytics はメニューファイルを無視し、カスタムメニュー項目が表示されないことがあります。

以下の要件に正確に従ってください。

ヒント

メニューファイルを、すべての出力不能な文字が表示される Notepad++ などのテキストエディターで作成または編集し、ファイルに含まれる文字が正確に表示されるようにします。

モノスペースまたは固定幅フォントを使用すると、個別の文字が垂直に配置されます。

同じメニューファイルの `Template.mnu` は、Analytics と一緒にインストールされる、`Sample Data Files` フォルダーに含まれます。

- `Template.mnu` は、`Sample Project.acf` および `Sample Data Files` フォルダーに含まれる他のサンプル Analytics プロジェクトで、`[アプリケーション]` メニューのサブメニュー エントリ `テンプレート`

ト]を作成します。

- **テンプレート]** サブメニュー エントリには、最初のレベルに 6 つのカスタムメニュー項目が含まれます。
- 最初のレベルのカスタムメニュー項目の 1 つである **Margins Analysis** には、2 番目のレベルに 4 つのカスタムメニュー項目があります。
- **Template.mnu** のカスタムメニュー項目のほとんどは、メニューファイルの概念を示すプレースホルダーにすぎません。

Template.mnu の内容は、付随する構文要件とともに、以下に示します。

Template.mnu の内容

```

MAIN MENU                                6                                .
Margins Analysis                          8 menu_def                          .
Inventory Analysis                        PAUSE 'SAMPLE INVENTORY ANALYSIS BATCH' .
Accounts Payable Analysis                 PAUSE 'LAUNCH YOUR A/P BATCH(ES)'     .
Accounts Receivable Analysis             PAUSE 'DO A/R BATCH(ES) HERE'        .
Internal Audit Functions                  PAUSE 'SAMPLE INTERNAL AUDIT PROCESSES' .
Quit ACL                                  QUIT                                  .

MARGINS ANALYSIS                          4                                .
Exception Listing                        PAUSE 'DO Batch where margin<=0'     .
High Margin Products                     PAUSE 'Sample Batch top 5 margins'    .
Low Margin Products                       PAUSE 'Calculate lowest 5 margins'    .
Margin Statistics                         STATISTICS                            .
    
```

メニューファイル構文要件

プロパティ	要件
行の長さ	メニューファイルの各行の長さはすべて同じでなければなりません。 必須ではありませんが、ピリオド(.)を使用して、改行の直前で、各行の最後を視覚的に示すことをお勧めします。
行番号	メニューファイル内の行数はゼロから数えます。 メニューファイル構文で行番号参照を指定するときにはいつでも、この番号体系を考慮してください。使用しているテキストエディターが 1 から行番号を表示してカウントする場合は、メニューファイルの行番号参照を指定するときに、テキストエディターの行番号を 1 減らす必要があります。 上記の例では、 Margins Analysis メニュー項目は行 1 に表示され、 [MARGINS ANALYSIS] サブメニュー構文ブロックは行 8 ~ 12 に表示されます。
空白行	空白行は、構文ブロック間に表示できますが、構文ブロック内には表示できません。 スペース文字から成る空白行は、メニューファイルの他の行と同じ長さである必要があります。 必須ではありませんが、構文ブロック間の 1 つ以上の空白行は、メニューファイルを視覚的に分離します。

プロパティ	要件
構文ブロック	<p>構文ブロックは、カスタムメニュー項目の各グループを定義します。複数の構文ブロックを使用して、複数のメニューレベルを作成できます。</p> <ul style="list-style-type: none"> ブロックの左側には、メニュー項目の名前が各行に1つずつ含まれます。これらは Analytics で表示される名前です。 名前の長さは最大 35 文字です。 ブロックの右側には、ACLScript コマンド、または構文の下位ブロックへの行参照が含まれます。 ブロックの右側の行は、すべての文字位置 37 から開始する必要があります。 テキスト要素の配置には、スペース文字のみを使用してください。タブ文字は使用しないでください。 <p>メモ メニューファイルにタブ文字が1つでもあると、ファイルが無視されます。タブ文字を表示できるテキスト エディターを使用し、タブ文字の有無を確認してください。</p>
ブロック見出し	<p>各構文ブロックは単一の見出し行で開始します。</p> <p>行の左側にはブロック識別子を大文字で記述し、右側にはブロック内の行数を格納します。</p> <p>上の例では、0 行にはブロック識別子 MAIN MENU が含まれており、そのブロック内に 6 行あることが示されています。見出し行はカウントされません。</p> <p>ブロック識別子は任意です。メニューファイル内で構文ブロックが整理されます。Analytics アプリケーション メニュー構造には表示されません。</p> <p>ブロック識別子を省略する場合、ブロックの指定された行数は、文字位置 37 から開始する必要があります。</p>
構文の下位ブロックへの参照	<p>構文の下位ブロックへのメニュー項目からの参照は、<code>num menu_def</code> の形式です。<code>num</code> 構文の下位ブロックが開始する行番号を指定します。つまり、下位レベルの構文ブロックの見出し行です。</p> <p>上記の例では、行 1 には Margins Analysis メニュー項目が含まれます。これは、MARGINS ANALYSIS 下位構文ブロックが開始する行を参照します (<code>8 menu_def</code>)。</p>
カスタムメニュー項目	<p>カスタムメニュー項目を指定できます。</p> <ul style="list-style-type: none"> 有効な ACLScript コマンド 構文の下位ブロックへの行参照 <p>Analytics スクリプトを実行するカスタムメニュー項目を作成するには、<code>DO SCRIPT スクリプト名</code> を指定します。例：</p> <pre>中央値の計算スクリプト DO SCRIPT Calculate_Median_Value</pre> <p>メモ スクリプトは、カスタムメニュー項目が表示される Analytics プロジェクトに含まれる必要があります。</p> <p>短いコマンドは、直接 .mnu ファイルに入力できます。複数のパラメーターを使用する長いコマンドはスクリプトに保存してください。これは、<code>DO SCRIPT</code> コマンドを使用して参照できます。</p>

メニュー ファイルの作成または編集

メニューファイルを作成する最も簡単な方法は、Analytics サンプルメニューファイル(`Template.mnu`)をコピーして修正することです。

既に使用中のメニューファイルを編集するときは、以下の点に留意する必要があります。

- メニューファイルを編集する前に、バックアップ用のコピーを作成してください。
 - 行を追加または削除する場合は、必ず適切に行番号参照を調整してください。
 - 可能であれば、項目はメニューファイルの最後に追加してください。これにより既存の行参照を保持できます。
1. `Template.mnu` を、Analytics サンプル データ ファイル フォルダーから作業場所にコピーします。

注意

元のテンプレート ファイルは編集しないでください。問題が発生した場合に、元のファイルを再コピーして、再開できます。

2. コピーしたファイルを適切な名前に変更します。

ファイルに指定した名前は、Analytics の [アプリケーション] メニューのサブメニュー エントリの名前になります。

メモ

メニューファイルを最初から作成している場合は、ファイル拡張子を `.mnu` に変更します。

3. 名前を変更したファイルを Notepad++ などのテキスト エディターで開き、編集して、サブメニューとカスタムメニュー項目を作成します。

上記の "メニューファイル構文要件" ページ 157 に正確に従います。

4. 次のいずれかを実行します。

- カスタムメニュー項目を表示する Analytics プロジェクトを含むフォルダーにファイルを保存します。
- Analytics 実行フォルダーにファイルを保存し、コンピューターで開いているすべての Analytics プロジェクトでカスタムメニュー項目を使用できるようにします。

ヒント

必要に応じて、プロジェクト レベルおよびグローバルメニュー ファイルの両方を作成できます。

5. Analytics を閉じてからもう一度開き、[アプリケーション] メニューを更新します。

サブメニュー エントリとカスタムメニュー項目は、[アプリケーション] メニューで使用できます。

サブメニューとカスタムメニュー項目が表示されない場合は、上記の構文要件に照らして、メニュー項目の内容を注意して確認してください。使用しているテキスト エディターで、すべての出力不能

な文字を表示します。行末の追加のスペースまたはメニュー ファイルのタブ文字は、ファイルが無視されます。

Analytics コマンド ラインからのコマンドの実行

メニューからオプションを選択し、ダイアログ ボックスで必要な情報を入力することで達成する Analytics 内のほとんどの機能は、コマンド ラインからコマンドを実行することによっても達成できます。



Analytics には、アプリケーション全体を通じてコマンドの処理や分析手順の記録に使用される、ACLScript という言語が含まれています。たとえば、メインメニューから **分析 > レコードのカウント**]を選択して、**カウント**]ダイアログ ボックスで必要な情報を入力し終わると、Analytics は自動的に、入力された情報をコマンドの実行に使用されるコマンド ステートメントに変換し、ログに操作を記録します。

コマンドラインを使用するためのガイドライン

- 一度に1つのコマンドしか入力できません。
- コマンドは大文字と小文字が区別されません。大文字または小文字を使用できます。
- コマンドには、一連の必須パラメーターやオプション パラメーターを含めることができます。
- コマンドのテキストが長い場合、Analytics は **コマンド ライン**]テキスト ボックスを広げて行を折り返し、コマンド全体が表示されるようにします。テキスト ボックスの外側をクリックするとテキスト ボックスが縮められ、内側をクリックするとコマンド全体が再度表示されます。
- ACLScript のコマンドや関数、キーワードは、省略形にすることができます。省略形には、コマンド、関数またはキーワードの先頭文字が含まれている必要があります。項目を一意に特定できさえすれば、省略形はいくらでも短縮できます。通常、少なくとも最初の3文字を入力する必要があります。

各 ACLScript コマンドに必要な構文の詳細については、「コマンドの概要」 ページ 1611を参照してください。

手順

1. **コマンド ライン]テキスト ボックスが表示されていない場合は、[ウィンドウコマンド ライン]の順にクリックします。**
2. 次の方法のいずれかを使用して、コマンド テキストを入力します。
 - 必要な構文を使用してコマンドを入力する。
 - **ナビゲーターの [ログ] タブでエントリをクリックして、コマンドをコマンド ラインに追加する。** コマンドはそのまま実行しても、実行する前に編集してもかまいません。
 - 既存の Analytics スクリプトまたはその他のテキスト ファイルからコマンド構文をコピーして、**コマンド ライン]テキスト ボックスに貼り付ける。** コマンドはそのまま実行しても、実行する前に編集してもかまいません。
3. 省略可能。Analytics のユーザー インターフェイスに、そのコマンドに関連付けられたダイアログ ボックスがある場合は、**コマンドの編集**  をクリックすると、関連するダイアログ ボックスが表示されます。このダイアログ ボックスを使用して、コマンドのパラメーター設定を変更できます。
4. **実行**  をクリックするか **Enter** キーを押して、コマンドを実行します。
コマンドの実行]、**[入力をクリア]**および **コマンドの編集]**オプションは、**コマンド ライン]テキスト ボックス内で右クリックして表示されるメニューからも利用できます。**

表示領域の情報の印刷

アクティブなビュータブや、スクリプト エディタータブ、結果タブ、またはワークスペース エディタータブの表示領域に表示される情報を印刷することができます。ビュータブの情報を印刷する場合は、ビューのデフォルトのレポート設定が使用されます。

表示領域内の情報を印刷するには

1. 印刷するビュー、ワークスペース、またはスクリプトを開きます。
2. **[ファイル > 印刷]**の順にクリックします。
3. 表示領域内の情報が2 ページ以上の長さになる場合は、ページ範囲を指定するオプションがあります。範囲を指定するには、**[ページ指定]**を選択して、印刷する開始 ページ番号と終了 ページ番号を入力します。
4. プリンターやプリンター プロパティ(用紙サイズ、印刷の向きなど)を変更する場合は、**[設定]**をクリックして **[ページ設定]**ダイアログ ボックスで必要な変更を加え、**[OK]**をクリックします。
5. **[印刷]**をクリックします。

Analytics から電子メール通知を送信する

Analytics から 1 人以上の受信者に電子メール通知メッセージを送信できます。メッセージには、データファイルや Analytics プロジェクトを添付して含めることができます。

電子メール通知の一般的な用途は、スクリプトが予期せず失敗したときに適切な担当者に警告することです。

電子メール通知の仕組み

Analytics の通知機能は、SMTP メールサーバー(Simple Mail Transfer Protocol サーバー) に送信電子メールメッセージを送信します。次にメールサーバーは、受信者にメッセージを中継します。Analytics とメールサーバー間の接続は、認証済みまたは非認証にすることができます。非認証接続は、最新の電子メールシステムではもはや一般的ではありません。

互換性のある SMTP サーバー

Analytics の電子メール通知機能は、認証接続を必要とするさまざまな SMTP サーバーで動作します。ただし、通知はすべての SMTP サーバーまたはサービスで動作するわけではありません。たとえば、通知機能は Gmail では動作しません。サードパーティアプリを Google アカウントに接続するには、Analytics がサポートするユーザー名とパスワード以上のセキュリティ構成が必要です。

一般に、Analytics を使用して SMTP サーバーに電子メールを正常に送信するには、Analytics 側だけでなく SMTP サーバー側にも互換性のある構成が必要であることを注意してください。Analytics が接続に失敗し、電子メールを正常に送信できない場合は、IT 部門と連携して、Analytics とサーバーの間に互換性のある接続構成があることを確認してください。

認証済みの SMTP 接続を使用して電子メールメッセージを送信する

メモ

ユーザーインターフェイスから認証済み SMTP 接続を設定するには、Analytics バージョン 16.1 以降を使用する必要があります。バージョン 16.0 を使用している場合、認証済み接続の設定はスクリプトでのみサポートされます。詳細については、「NOTIFY コマンド」ページ 1996を参照してください。

Analytics の 16.0 より前のバージョンでは、SMTP メールサーバーとの認証済み接続はサポートされません。

Analytics のユーザーインターフェイスを使用して、SMTP メールサーバーとの認証済み接続を使用した電子メール通知を設定します。最近の電子メールシステムのほとんどは、認証済み接続を必要とします。

ユーザーインターフェイスで接続を設定する利点は、Analytics により ACLScript 構文が作成されることです。動作する接続を設定したら、コマンドログからスクリプトに構文をコピーできます。

1. Analytics のメインメニューから **ツール > 電子メールによる通知**] を選択します。

通知] ダイアログ ボックスが表示されます。

2. **\$SMTP**] を選択し、次の情報を入力します。

- **SMTP ユーザー(オプション)** -SMTP サーバーに対して認証し、アクセスするためのユーザーアカウントの名前を入力します。

このフィールドを空白にすると、**送信者**] で指定された電子メールアドレスが SMTP サーバーに対する認証とアクセスに使用されます。

- **ポート** -SMTP サーバーへのアクセスに使用するポート番号を入力します。

使用するポートについては、IT 部門に問い合わせることが必要になる場合があります。SMTP サーバーの一般的なポート番号は、次のとおりです。

- 587
- 2525
- 465

- **SSL** -オプション。SMTP サーバーが安全な SMTP/SSL 接続を必要とする場合、**SSL** を選択します。

SMTP サーバーが SSL 接続を必要とするかどうかは、IT 部門に問い合わせることが必要になる場合があります。または、**SSL** を選択した場合と選択していない場合のテストメールを送信して見る必要がある場合があります。

3. 次の情報を記入します。

- **送信者** - SMTP サーバー経由で送信される電子メールメッセージの送信者電子メールアドレスを入力します。

- **パスワード** - 送信者電子メールアカウントのパスワードを入力します。
パスワードは Analytics によって自動的に暗号化され、暗号化されたバージョンがログに表示されます。詳細については、「暗号化されたパスワードの生成」ページ 2000を参照してください。
- **メールボックスパス** - 電子メールメッセージを送信するために使用する SMTP サーバーのドメイン名を入力します。
例: `smtp.example.com`
- **送信先** - 1人以上の受信者の電子メールアドレスを入力します。複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。

メモ

最大 1020 文字を入力します。

- **Cc** - 省略可能。1人以上のカーボンコピー受信者の電子メールアドレスを入力します。複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。

メモ

最大 1000 文字を入力します。

- **Bcc** - 省略可能。1人以上のブラインドカーボンコピー受信者の電子メールアドレスを入力し、複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。
- **件名** - 電子メールメッセージの件名を入力します。
- **本文** - メール本文を入力します。
メッセージはプレーンテキストのため、HTML はサポートされていません。メッセージに改行を挿入したい場合は、2つのカラット文字 (^ ^) を使用します。
- **添付ファイル** - オプション。1つ以上の添付ファイルのパスとファイル名を指定するか、**参照** をクリックして **ファイルを選択** ダイアログボックスを開きます。

ファイルが Analytics プロジェクトと同じフォルダーにある場合は、パスを指定する必要はありません。

カンマで区切られたファイルのリストを入力して、複数の添付ファイルを指定します。

例: `result1.csv,result2.csv`

メモ

カンマ区切りのリストでは、カンマの後にスペースを挿入しないでください。

4. **OK** をクリックします。

Analytics は、SMTP メールサーバーに電子メールを送信し、受信者に中継します。

Analytics とメールサーバー間の接続試行に失敗した場合、Analytics は指定した回数だけ自動的に接続を再試行します。詳細については、「接続の試行に失敗した場合の Analytics の応答の仕方」ページ 2001を参照してください。

未認証の接続を使用して電子メールメッセージを送信する

Analytics ユーザーインターフェイスを使用して、SMTP メールサーバーまたはローカルメールシステムとの非認証接続を使用した電子メール通知を設定します。非認証接続は、最新の電子メールシステムでもはや一般的ではありません。

Analytics からオープン SMTP サーバーへの非認証接続では、ポート 25 が使用されます(接続を成功させるにはこのポートがサーバー上で開いていることが必須です)。

ユーザーインターフェイスで接続を設定する利点は、Analytics により ACLScript 構文が作成されることです。動作する接続を設定したら、コマンドログからスクリプトに構文をコピーできます。

1. Analytics のメインメニューから **ツール > 電子メールによる通知** を選択します。

通知 ダイアログ ボックスが表示されます。

2. **SMTP** が選択されていないことを確認します。
3. 次の情報を記入します。

- **送信者** - SMTP サーバーまたはローカルメールシステム経由で送信される電子メールメッセージの送信者電子メールアドレスを入力します。

- **パスワード** - 送信者電子メールアカウントのパスワードが必要な場合、パスワードを入力します。

パスワードは Analytics によって自動的に暗号化され、暗号化されたバージョンがログに表示されます。詳細については、「暗号化されたパスワードの生成」ページ 2000を参照してください。

- **メールボックス パス** - 電子メールメッセージを送信するために使用する SMTP サーバーのドメイン名を入力します。

例:

ローカルのメールシステムを使用している場合は、ローカルのメールボックスへのパスを入力するか、または **参照** をクリックして、**フォルダーの参照** ダイアログ ボックスを開きます。

- **送信先** - 1 人以上の受信者の電子メールアドレスを入力します。複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。

メモ

最大 1020 文字を入力します。

- **Cc** - 省略可能。1人以上のカーボンコピー受信者の電子メールアドレスを入力します。複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。

メモ

最大 1000 文字を入力します。

- **Bcc** - 省略可能。1人以上のブラインドカーボンコピー受信者の電子メールアドレスを入力し、複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。
- **件名** - 電子メールメッセージの件名を入力します。

- **本文** - メール本文を入力します。

メッセージはプレーンテキストのため、HTML はサポートされていません。メッセージに改行を挿入したい場合は、2つのキャレット文字 (^ ^) を使用します。

- **添付ファイル** - オプション。1つ以上の添付ファイルのパスとファイル名を指定するか、**参照**] をクリックして **ファイルを選択**] ダイアログ ボックスを開きます。

ファイルが Analytics プロジェクトと同じフォルダーにある場合は、パスを指定する必要はありません。

カンマで区切られたファイルのリストを入力して、複数の添付ファイルを指定します。

例: `result1.csv,result2.csv`

メモ

カンマ区切りのリストでは、カンマの後にスペースを挿入しないでください。

4. **OK**] をクリックします。

Analytics は、SMTP メールサーバーまたはローカルメールシステムに電子メールを送信し、受信者に中継します。

Analytics とメールサーバーまたはメールシステムとの接続に失敗した場合、Analytics は指定された回数だけ接続を自動的に再試行します。詳細については、"接続の試行に失敗した場合の Analytics の応答の仕方" ページ 2001 を参照してください。

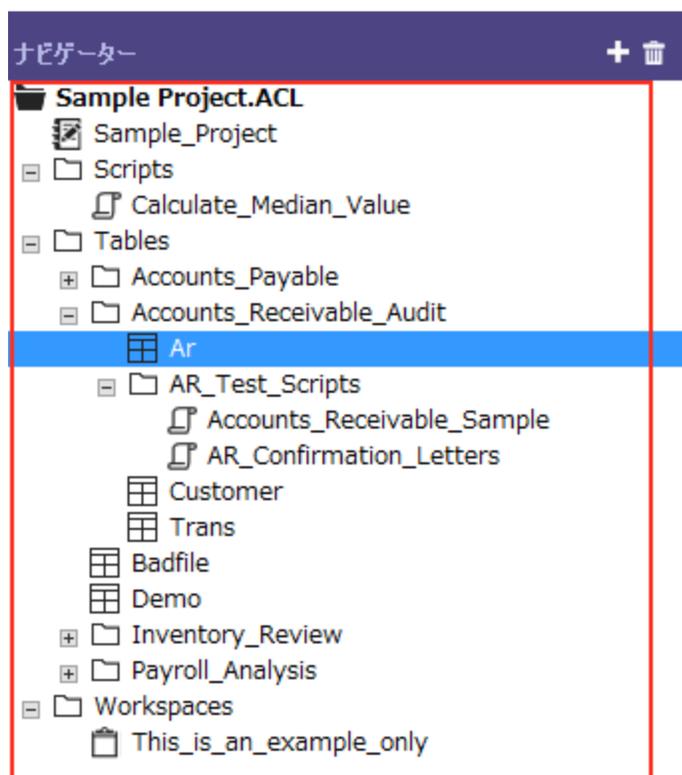
Analytics プロジェクト

Analytics プロジェクトは、Analytics における最上位の階層であり、データ分析プロジェクトに関連付けられているすべての情報が格納されます。

メインの Analytics プロジェクト ファイル (.ACL) には、ほとんどのプロジェクト情報が格納されます。一連の追加ファイルには、ログやインデックスなどのプロジェクトに関する特定の種類の情報が格納されます。データは、プロジェクトの外部、つまり、ネイティブ Analytics データファイルまたはデータベースに保存されます。

現在操作している Analytics プロジェクトは、**ナビゲーターの 概要** タブに表示されます。ログの内容は、**ログ** タブに表示されます。一度に開けるプロジェクトは 1 つだけです。

サンプル Project.ACL は、**ナビゲーター**で下に表示されます。



総覧 ログ 変数

ナビゲーター: [総覧] タブ

Analytics プロジェクトには、複数の異なる項目タイプが含まれます。ナビゲーターでこれらの項目を表示および操作できます。

以下の表は、ナビゲーター ツリービューに表示される項目タイプの一覧を示します。

アイコン	項目の種類	説明
	テーブル	Analytics テーブル。このテーブルは、テーブルレイアウトとそれに関連付けられたデータソースという2つのパートで構成されます。 テーブルレイアウトには、レコード長やフィールド名など、データを表示する方法に関する情報が含まれています。データソースとはファイルやデータセット(データベーステーブルなど)で、テーブルの内容が含まれています。データソースは Analytics プロジェクトの外部に存在します。
	サーバーテーブル	テーブルレイアウトがローカルの Analytics に存在し、かつ関連するデータソースがサーバーに存在するテーブル。テーブルレイアウトは、データベース プロファイルまたはサーバー プロファイルを使用してデータソースに接続します。
	スクリプト	Analytics プロジェクト内から実行できる一連の ACLScript コマンド。
	サーバー スクリプト	サーバーに置かれている ACLScript ファイル(.aclscript、以前は .bat)。
	ワークスペース	ほかのテーブルで再利用するために保存されたフィールド定義を1つ以上含んでいる、Analytics プロジェクト項目。
	プロジェクト	ツリービューの最上位のエントリは、Analytics プロジェクトです。プロジェクトは、ファイル拡張子 .acl を持つ物理ファイルに保存されます。
	ログ	Analytics プロジェクトで作業しているときに実行されたコマンドの記録。
	フォルダー	Analytics プロジェクト内のフォルダー。これらのフォルダーは、Analytics プロジェクト ファイル(.ACL) 内のみ存在します。Windows フォルダーとして物理的に作成されません。

ナビゲーター: [ログ] タブ

以下の表は、ナビゲーターの [ログ] タブに表示されるエントリのタイプの一覧を示します。

アイコン	エントリタイプ	説明
	グループ	特定の日付範囲内のログセッションのグループ。
	セッション エントリ	日付時刻別に示される個々のセッション。セッションは、プロジェクトを開くたび、またセッションを手動で作成するときに作成されます。

アイコン	エントリタイプ	説明
	コマンド成功	正常に完了したコマンドを特定します。
	コマンド失敗	失敗したコマンドを示します。

Analytics プロジェクト ファイル

次に挙げるファイルの種類を使用して、Analytics プロジェクトの情報が記録されます。プロジェクトをバックアップまたはアーカイブするときは、完全なプロジェクトを復元するために、確実にすべてのファイルをコピーする必要があります。

ファイルタイプ (拡張子)	説明
Analytics プロジェクト ファイル (.ACL/.acl)	Analytics プロジェクト ファイルには、データ分析の重要な情報のすべてが保存されます。 <ul style="list-style-type: none"> テーブルレイアウトとビュー定義 スクリプト プロジェクト フォルダー ソースから更新コマンドを使用してテーブルを更新するコマンド構文。 テーブル履歴 ワークスペース
Analytics プロジェクト 自動保存ファイル (.ac)	プロジェクトを開くたびに、一時的な自動保存ファイルが作成されます。 このファイルの目的は、Analytics プロジェクトに保存されていないすべての変更を記録しておく、Analytics が予期せず閉じた場合に変更を回復できるようにすることです。 プロジェクトが保存され、正常に閉じられた場合には、.ac ファイルは削除されます。そうでない場合は、このファイルからプロジェクトを復元するよう求めるメッセージが表示されます。
Analytics データ ファイル (.fil)	多くの場合、データソースから Analytics テーブルを定義するときに、データソースから新しい Analytics データ ファイル(ファイル拡張子 .fil) にデータがコピーされます。 .fil ファイルにデータをコピーするデータソースの種類の一覧については、"Analytics でアクセスできるデータソース" ページ 242を参照してください。
ログ ファイル (.log)	ログファイルは、プロジェクトが開いている間に、Analytics によって実行されたすべてのコマンドを記録します。 デフォルトのログは Analytics プロジェクト ファイルと同名で、拡張子は .log です。必要に応じて、カスタム ログファイル名を指定できます。
ログ インデックス ファイル (.lix)	インデックス ファイルは、ログ エントリを特定のセッションに関連付けるために使用されます。セッションは、プロジェクトを開くたびに作成されます。また、いつでも手動で作成することができます。
インデックス ファイル (.inx)	Analytics テーブルにインデックスを作成するときに、インデックス ファイルが作成されます。ファイル名は、Analytics のインデックス名と同じで、.inx 拡張子です。

ファイルタイプ (拡張子)	説明
	また、インデックスファイルは、テーブルで クイックソート 昇順]または クイックソート 降順]コマンドを使用するときにも作成されます。クイックソート コマンドで作成されるインデックスのファイル名は、ACLQSTMP.inx になります。

追加の Analytics ファイルタイプ

追加の4つのファイルの種類は、Analytics プロジェクトから作成したり、Analytics プロジェクトへインポートしたりすることができます。

これらのファイルの種類は、プロジェクトにとって必要なものではありませんが、存在する場合、任意のバックアップ処理に含めてください。

ファイルタイプ (拡張子)	説明
テーブルレイアウト ファイル (.layout)	Analytics テーブルレイアウトの外部コピー。
ビューファイル (.rpt)	Analytics ビューの外部コピー。
Analytics スクリプト ファイル (.aclscript)	Analytics スクリプトまたは分析の外部コピー。
ワークスペース ファイル (.wsp)	Analytics ワークスペースの外部コピー。

Analytics プロジェクトを使った作業

Analytics プロジェクトは、データ分析プロジェクトに関連付けられているすべてのテーブルと処理をグループ化および組織化する方法を提供します。Analytics プロジェクトは何個でも作成できます。プロジェクトが作成された瞬間から、Analytics で完了したデータ分析は Analytics プロジェクトのコマンド ログに記録されます。

新しい Analytics プロジェクトを作成する

新しいプロジェクトを作成するときは、通常、そのプロジェクト用の新しい Windows フォルダーを作成し、Analytics プロジェクトと Windows フォルダー間で一対一の関係を保持することをお勧めします。Windows フォルダーとサブフォルダーを必要に応じて使用し、次の項目を保存します。

- Analytics プロジェクト ファイル (.acl) とインデックス ファイル (.inx) などのすべての関連付けられたファイル
- ソース データ ファイル
- Analytics データ ファイル (*.FIL)
- Analytics で実行された分析から生成された結果ファイル

Analytics 内または ACL for Windows メイン画面から、新しいプロジェクトを作成できます。

方法の説明

メモ

ファイル拡張子 (.acl) を含む Analytics プロジェクト パスとプロジェクト名を組み合わせた長さを 259 文字以下です。

Analytics を使って新しいプロジェクトを作成する

1. Analytics のメインメニューから、**ファイル > 新規 > プロジェクト**] を選択します。
2. **新規プロジェクトを保存する名前**] ダイアログボックスで、プロジェクトを保存するフォルダーを選択してファイル名を入力したら、**保存**] ボタンをクリックします。

新しいプロジェクトが作成され、データ定義ウィザードが開きます。ウィザードの手順を続行して、新しいプロジェクトに新しい Analytics テーブルを作成することができます。テーブルを作成しない場合は **キャンセル**] をクリックします。

ACL for Windows から新しいプロジェクトを作成する

1. **ACL for Windows** のメイン画面で、**新しいアナリティクスプロジェクト**] をクリックします。
2. **名前を付けて保存**] ダイアログボックスで、プロジェクトを保存するフォルダーを選択してファイル名を入力したら、**保存**] ボタンをクリックします。

新しいプロジェクトが Analytics 内で作成され、開かれます。

既存の Analytics プロジェクトを開く

既存の Analytics プロジェクトを開くと、新しいセッションが作成され、Analytics によって処理されたすべてのコマンドがログに記録されます。

既存のプロジェクトは、Analytics 内または ACL for Windows メイン画面から開くことができます。

方法の説明

Analytics から既存のプロジェクトを開く

1. Analytics のメインメニューから、**ファイル > プロジェクトを開く**]を選択します。
2. **プロジェクト**]ダイアログボックスで、Analytics プロジェクト ファイル(.acl)に移動して、**開く**]をクリックします。

プロジェクトが Analytics によって開かれます。Analytics で既に別のプロジェクトが開かれていた場合は、選択したプロジェクトを開く前に、そのプロジェクトへの変更を保存するよう求めるメッセージが表示され、保存後にプロジェクトが閉じます。

ACL for Windows から既存のプロジェクトを開く

1. ACL for Windows メイン画面で、**アナリティクス プロジェクトを開く**]をクリックするか、または **最近のアナリティクス ファイル**]の下から Analytics プロジェクト(.acl)を選択します。
2. **アナリティクス プロジェクトを開く**]をクリックした場合は、Analytics プロジェクト ファイル(.acl)に移動して、ファイルを選択し、**開く**]をクリックします。

プロジェクトが Analytics によって開かれます。

Analytics プロジェクトを保存する

初めて Analytics プロジェクトを保存するときは、ファイル名と場所を指定する必要があります。**プロジェクトを上書き保存**]メニューコマンドを使用すると、同じ場所に最新バージョンのプロジェクトを保存できます。プロジェクトのコピーを保存したい場合は、**プロジェクトを別名で保存**]メニューコマンドを使用して、別のファイル名と場所を選択します。

方法の説明

1. 次のいずれかを実行します。
 - 現在のバージョンのプロジェクトを保存する場合は、**ファイルプロジェクトを上書き保存**]を選択します。
 - 現在のバージョンのプロジェクトを別の名前で保存する場合は、**ファイルプロジェクトを別名で保存**]を選択します。**プロジェクトを保存する名前**]ダイアログボックスで場所を選択して新しい名前を入力したら、**保存**]ボタンをクリックします。

このオプションを選択した場合、プロジェクトは新しい名前でも保存されますが、その新しいプロジェクトは Analytics によって開かれません。

2. プロジェクトが最後に保存された後に変更されたプロジェクト項目がある場合は、それらを保存するように求められます。確認ダイアログボックスで **OK** をクリックします。

Analytics プロジェクト プロパティを表示する

Analytics プロジェクトに関連付けられた多数のプロパティを表示し、プロジェクトを含むフォルダーに迅速に移動できます。

方法の説明

1. **ナビゲーターの [総覧]** タブで、Analytics プロジェクトを右クリックします。
Analytics プロジェクトが、ツリービューの最上位のエントリとして表示されます。
2. **プロパティ** を選択します。
3. **プロジェクト プロパティ** ダイアログボックスで、次のタブをクリックして、プロジェクト プロパティを表示または修正します。
 - **全般** - プロジェクト ファイルの基本プロパティを表示します。ファイル名、ファイルの場所、最終更新日時、およびプロジェクト ファイルの物理サイズが表示されます。
ファイルの場所を開く をクリックすると、Analytics プロジェクト ファイル(.acl) が入っているフォルダーに直接移動することができます。
 - **ノート** - プロジェクトに関連付けられているすべての注釈を表示します。既存のノートを変更したり、新しいノートを追加したりすることができます。詳細については、"Analytics プロジェクトのノートの追加と編集" ページ 183を参照してください。
 - **ビュー** - プロジェクト内のすべてのビューを、"ビュー名[テーブルレイアウト名]" 形式を使用して表示します。このタブからプロジェクト内のビューを管理できます。詳細については、"ビューでの作業" ページ 841を参照してください。
4. 行った変更を保存してダイアログボックスを閉じるには、**OK** をクリックします。

別の Analytics プロジェクトからプロジェクト項目をコピーする

Analytics プロジェクト項目を、お使いのコンピューター上またはアクセス可能なネットワークドライブ上の任意の Analytics プロジェクトから、現在のプロジェクトにコピーすることができます。

同じ種類であれば、同時に複数のプロジェクト項目をコピーできます。たとえば、複数のスクリプトを同時にコピーすることができます。複数の種類の項目、たとえば、スクリプトとテーブルレイアウトをコピーしたい場合は、項目の種類ごとに以下の手順を繰り返す必要があります。

Analytics プロジェクトの外部に個別のファイルとして存在するプロジェクト項目をインポートしたい場合は、"プロジェクト項目のインポート" 見開きページを参照してください。

方法の説明

1. コピーされた項目を格納するプロジェクトを開きます。
2. ナビゲーターの **総覧**] タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**別のプロジェクトからコピー < プロジェクト項目の種類 >]** を選択します。 < プロジェクト項目の種類 > は次のオプションのいずれかです。
 - テーブル
 - ビュー
 - スクリプト
 - ワークスペース

メモ

テーブルをコピーする場合は、ソース データ ファイル (.fil) でなくテーブルレイアウトのみをコピーすることになります。

3. **プロジェクト ファイルの場所**] ダイアログ ボックスで、プロジェクト項目のコピー元とする Analytics プロジェクトを探して選択し、**開く**] をクリックします。
4. **インポート**] ダイアログ ボックスで、次のいずれかの作業を行って、1 つ以上のプロジェクト項目を **インポート先 < プロジェクト名 >]** リストに追加します。
 - 個別のプロジェクト項目をダブルクリックします。複数のプロジェクト項目を
 - **Ctrl キー**を押しながらクリックし、右矢印ボタンをクリックする。
 - **すべて追加**] をクリックして、すべてのプロジェクト項目を追加する。

インポート先 < プロジェクト名 >] リストからプロジェクト項目を削除するには、個々のプロジェクト項目をダブルクリックするか、複数のプロジェクト項目を **Ctrl キー**を押しながらクリックした後に左矢印ボタンをクリックするか、または **すべてクリア**] をクリックします。

5. **OK**] をクリックして、現在のプロジェクトにプロジェクト項目をコピーします。

プロジェクト内に同じ名前の項目が既に存在する場合は、コピーした項目名の末尾に増分数字が付けられます。

プロジェクト項目のインポート

個別のファイルとして Analytics プロジェクトの外部に存在する Analytics プロジェクト項目をインポートすることができます。たとえば、Analytics スクリプトは .aclscript ファイルとして、テーブルレイアウトは .layout ファイルとして保存されています。一度にインポートできるプロジェクト項目は 1 つだけです。

別の Analytics プロジェクトからプロジェクト項目をインポートしたい場合は、"別の Analytics プロジェクトからプロジェクト項目をコピーする" 前のページを参照してください。

方法の説明

1. ナビゲーターの **総覧**] タブで、Analytics プロジェクトのエントリを右クリックし、**プロジェクト項目のインポート > < プロジェクト項目の種類 >]** を選択します。 < プロジェクト項目の種類 > は次のオプションのいずれかです。
 - テーブル
 - ビュー
 - スクリプト
 - ワークスペース
2. **プロジェクト**] ダイアログ ボックスで、適切なファイルの種類を探して選択し、**開く**] をクリックします。

ファイルタイプおよびプロジェクト項目は次のように対応します。

ファイル拡張子	プロジェクト項目
.layout	テーブルレイアウト
.rpt	view
.aclscript	スクリプト
.wsp	ワークスペース

3. 確認ダイアログボックスで、**[OK]**をクリックします。

現在のプロジェクトにプロジェクト項目がインポートされます。プロジェクト内に同じ種類で、同じ名前の項目が既に存在する場合は、インポートした項目名の末尾に増分数字が付けられます。

プロジェクト項目のエクスポート

Analytics プロジェクト項目は、Analytics プロジェクトの外部に保存される個別のファイルとしてエクスポートすることができます。たとえば、Analytics スクリプトは .aclscript ファイルとして保存でき、テーブルレイアウトは .layout ファイルとして保存することができます。別ファイルとしてエクスポートされたプロジェクト項目は、後で任意の Analytics プロジェクトにインポートできます。一度にエクスポートできるプロジェクト項目は1つだけです。

プロジェクト項目とファイルの種類は次のように対応しています。

プロジェクト項目	ファイル拡張子
テーブルレイアウト	.layout
ビュー	.rpt
スクリプト	.aclscript
ワークスペース	.wsp

方法の説明

メモ

項目名を64文字の英数字(ファイル拡張子を含まない)に制限し、項目が Analytics にインポートされる時に名前が切り捨てられないようにします。

名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。先頭に数字を使用することもできません。特殊文字、スペース、先頭の数字はすべて、項目のインポート時にアンダースコア文字に置換されます。

テーブルレイアウト、スクリプト、またはワークスペースのエクスポート

1. ナビゲーターの **総覧** タブで項目を右クリックし、**プロジェクト 項目のエクスポート** を選択します。
2. **ファイルの保存** ダイアログ ボックスで、項目を保存する場所を選択し、必要に応じて名前を変更して、**保存** をクリックします。
3. 確認ダイアログ ボックスで、**OK** をクリックします。
指定した場所にプロジェクト 項目がエクスポートされます。

ビューをエクスポートする

1. ビューに関連付けられているテーブルを開きます。
2. ナビゲーターの **総覧** タブでテーブルを右クリックし、**プロパティ > ビュー** を選択します。
3. ビューを選択して **エクスポート** をクリックします。
4. **ファイルの保存** ダイアログ ボックスで、ビューを保存する場所を選択し、必要に応じてビュー名を変更して、**保存** をクリックします。
5. 確認ダイアログ ボックスで、**OK** をクリックします。
指定した場所にビューがエクスポートされます。

Analytics コマンド ログの使用 方法

Analytics には、2 つの主な目的を果たす強力なログ機能が備わっています。

- **分析の追跡**

ログは各 Analytics セッション中に実行された正確な順番のコマンドを記録し、Analytics プロジェクトの一部として保存します。コマンドの順番を記録しておくことで、いつでも必要なときに監査手順を検証したり、手順を繰り返したりすることができるようになります。

- **Analytics コマンド構文の記録**

ログは、各コマンドの実行に使用された ACLScript 構文を記録します。正確な構文にアクセスすると、次のことができます。

- ユーザー インターフェイスから手動で再作成する代わりに、ログから簡単にコマンドを再実行する
- 選択したログのエントリに基づいてスクリプトを作成する
- ユーザー インターフェイスからアクセスされるコマンドに関連付けられた ACL スクリプト構文を表示して学習する

ログへのセッションの追加

Analytics プロジェクトを開くたびに、新しいセッションが自動的にログに作成されます。各セッションには、セッションの開始時を示す日付とタイムスタンプが含まれています。

ログ エントリのグループを作成したいときはいつでも、手動でセッションをログに追加することもできます。たとえば、実行する監査分析の段階ごとに個別のセッションを追加するとよいでしょう。

セッションを手動で追加する場合は、セッション名を含めることができます。

1. **ログ** タブで右クリックして **新しいセッションの追加** を選択します。
2. (オプション) **セッション名** テキスト ボックスに、そのセッションを識別するための名前を入力します。
セッション名の長さは最大 30 文字です。
3. **OK** をクリックします。

ログへのコメントの追加

Analytics コマンド ログに手動でコマンドを追加できます。分析の手順を文書化したり、監視プロジェクトに関する説明や描写を追加したりするためにコメントを利用します。コメントを追加した場合、そのコメントはログの最後の項目として追加されます。

1. **ツリー > コメント** の順に選択します。
2. **コメント** テキスト ボックスにコメントのテキストを入力し、**OK** をクリックします。

ログ ツリービューには単一行のコメントが表示されます。複数行のコメントを表示するには、ツリービューでコメント エントリをダブルクリックします。

ログを検索する

ログ エントリおよびセッション エントリ内の特定のコマンド名や文字列を見つけるために、ログを検索することができます。

1. ナビゲーターの **ログ** タブをクリックします。
2. 検索の開始位置とするログ エントリを右クリックし、**データ検索** をクリックします。
3. **検索** ダイアログボックスの **検索** フィールドで、検索する文字列を入力します。
4. 必要に応じて、任意の**検索オプション**を追加します。

オプション	説明
単語のみと一致	正確な一致のみが検出されます。 たとえば、"SET L" の検索では、このオプションがオンになっている場合は "SET LEARN" と一致しませんが、オフの場合は一致します。
大文字と小文字を区別する	大文字と小文字が正確に一致する語のみが検出されます。 たとえば、"Comment" の検索では、このオプションがオンになっている場合は "COMMENT" と一致しませんが、オフの場合は一致します。
上、下	検索方向を指定します。

5. **次を検索** をクリックします。

一致が見つかった場合は、検索文字列が含まれる最初のログ エントリが強調表示されます。次の一致へ移動するには、**次を検索** をクリックします。

ログ エントリのコピー

クリップボードへログ エントリをコピーし、Analytics 内のスクリプト エディターやワークスペース エディターや、別のアプリケーションにエントリを貼り付けることができます。ログ エントリのコピーは、新しいファイルやスクリプトにエントリをエクスポートするもう 1 つの方法です。

1. ナビゲーターの **ログ** タブをクリックします。
2. コピーしたい各ログ エントリの横にあるチェック ボックスをオンにします。

次の項目を選択できます。

- 個別のエントリ
- ログセッション
- 日付範囲
- ログ全体

ログセッションまたは日付範囲を選択する場合は、すべてのサブエントリも自動的に選択されません。

3. **ログ** タブで右クリックして **コピー** を選択します。
ログ エントリが Windows クリップボードにコピーされます。
4. ログ エントリを対象に貼り付けます。

ログ エントリのエクスポート

Analytics ログ エントリまたはログの内容全体を外部ファイルまたは現在のプロジェクトの Analytics スクリプトにエクスポートできます。

1. ナビゲーターの **ログ** タブをクリックします。
2. エクスポートしたい各ログ項目の横にあるチェックボックスをオンにします。

次の項目を選択できます。

- 個別のエントリ
- ログセッション
- 日付範囲
- ログ全体

ログセッションまたは日付範囲を選択する場合は、すべてのサブエントリも自動的に選択されません。

3. **ログ** タブを右クリックし、**選択項目の保存 < エクスポートの種類 >** を選択します。 < エクスポートの種類 > は次のオプションのいずれかになります。

HTML	HTML ファイル (.htm)
ログファイル	新しい Analytics ログファイル (.log)
スクリプト	現在のプロジェクトの新しい Analytics スクリプト
ワードパッド	WordPad の一時新規ファイル
テキスト	テキスト ファイル (.txt)

4. ファイル名またはスクリプト名を指定し、**保存** または **OK** をクリックします。
一時 WordPad ファイルを保存するには、WordPad で **名前を付けて保存** を使用します。

ログ エントリの削除

ログから次のエントリを削除できます。

- 個別のエントリ
- ログセッション
- 日付範囲
- ログ全体

ログセッションまたは日付範囲を選択する場合は、すべてのサブエントリも自動的に選択されます。

ログの一部またはログ全体を削除するときには、削除の位置に次のコメントが自動的に挿入されます。**ログの範囲が削除されました。**

1. ナビゲーターの **ログ** タブをクリックします。
2. ログのすべてのエントリを削除するには
 - a. **ログ** タブで右クリックして **ログ全体の削除** を選択します。

- b. 確認ダイアログ ボックスで **[OK]** をクリックします。
3. 個別のエントリを削除するには
 - a. 削除する各エントリ、セッション、または日付範囲の横のチェックボックスを選択します。
 - b. 右クリックし、**[選択した項目の削除]** を選択します。
 - c. 確認ダイアログ ボックスで **[OK]** をクリックします。

ログからのコマンドの再実行

ログに含まれる単一行コマンドを選択し、それをコマンド ラインから再実行することができます。

また、複数行にわたる DEFINE FIELD...COMPUTED コマンドを選択して実行することもできます。GROUP など、ほかの複数行コマンドについてはコマンド ラインから実行できません。これらはスクリプトからのみ実行できます。

コマンドを再実行する場合、そのコマンドをそのまま実行することも、また実行前に変更することもできます。

1. ナビゲーターの **[ログ]** タブをクリックします。
2. 再実行させたいコマンドを含むログ エントリを選択します。

当該のコマンドが、ツールバーの真下、Analytics のインターフェイスの上部近くにある**コマンドライン**にあらかじめ入力されます。

メモ

コマンドラインが表示されない場合は、**[ウィンドウ > コマンドライン]**を Analytics メインメニューから選択します。

3. 必要であれば、**[コマンド ライン]**テキスト ボックスでコマンドを編集します。
4. **実行 ** をクリックします。

Analytics プロジェクトでのノートの使用 方法

特定のプロジェクト項目に関する情報を記録するために、さまざまな種類のノート(注釈)が Analytics で作成できます。ノートが特に役立つのは、定期的に繰り返す処理について詳細を記録しておく場合です。複数の人が、ある 1 つの Analytics プロジェクトで作業する場合、ノートを付けることで、1 人が設計した手続きをほかの人が理解しやすくなります。

ノートは以下の項目に追加することができます。

- Analytics プロジェクト
- テーブルレイアウト
- ビュー
- レコード
- フィールド
- スクリプト
- ワークスペース

Analytics プロジェクトのノートの追加と編集

ある Analytics プロジェクトに関して、今後の参考のために保持しておきたい、あるいは他のユーザーのために文書化しておきたい情報がある場合は、ノート(注釈)をそのプロジェクトに追加して記録することができます。プロジェクト ノートの内容はいつでも編集できます。

方法の説明

1. ナビゲーターの **総覧**] タブで、Analytics プロジェクトを右クリックします。
Analytics プロジェクトが、ツリービューの最上位のエントリとして表示されます。
2. **プロパティ**] を選択します。
3. **プロジェクト プロパティ**] ダイアログ ボックスで **ノート**] タブをクリックします。
4. 新しいノートを入力するか、または既存のノートを編集します。
ノートを削除する場合は、テキストをすべて削除します。
5. **OK**] をクリックしてダイアログ ボックスを閉じ、変更を保存します。

テーブルレイアウトのノートの追加と編集

いつどのようにデータソースがアクセスされたか、定義されている演算フィールド、またテーブルに対して実行する必要がある分析手順などの情報を記録するために、テーブルレイアウトへノートを追加することができます。テーブルレイアウト ダイアログ ボックス内、またはナビゲーターの **総覧**] タブから、テーブルレイアウトのノートを追加することができます。テーブルレイアウトのノートの内容はいつでも編集できます。

ナビゲーターでテーブルレイアウトのノート进行管理している場合は、ノートを追加、削除、編集または読み込む際にテーブルを開く必要はありません。

オプション]ダイアログボックスで **レポートにレポート履歴を含める]** オプションを選択すると(デフォルトでは選択済み)、Analytics 印刷レポートにテーブルレイアウトのノートが追加されます。詳細については、"印刷オプション" ページ 146を参照してください。

方法の説明

概要] タブからノートを追加または編集する

1. ナビゲーターの **総覧]** タブでテーブルを右クリックし、**プロパティ]** を選択します。
2. **テーブルプロパティ]** ダイアログボックスで **ノート]** タブをクリックします。
3. 新しいノートを入力するか、または既存のノートを編集します。
ノートを削除する場合は、テキストをすべて削除します。
4. **OK]** をクリックしてダイアログボックスを閉じ、変更を保存します。

テーブルレイアウト] ダイアログボックスからノートを追加または編集する

1. **編集 > テーブルレイアウト]** の順にクリックします。
2. **テーブルレイアウト オプション]** タブをクリックします。
3. **テーブルレイアウト ノートの編集]**  をクリックします。
4. 新しいノートを入力するか、または既存のノートを編集します。
ノートを削除する場合は、テキストをすべて削除します。
5. **閉じる]**  をクリックします。
ノートは自動的に保存されます。
6. **閉じる]**  をクリックしてテーブルレイアウト ダイアログボックスを終了します。

ビューノートの追加と編集

ビューにノートを追加して、そのビューに関する追加情報を提供することができます。ノートはその特定のビューに固有のものとなり、そのビューから作成される新しいすべてのビューにコピーされます。ビューノートの内容はいつでも編集できます。

オプション]ダイアログボックスで **レポート履歴にビューノートを含める]** オプションを選択すると、Analytics 印刷レポートにビューノートが追加されます。詳細については、"印刷オプション" ページ 146を参照してください。

方法の説明

1. ビュータブの下部で、ノートを追加したいビューのボタンを右クリックし、**プロパティ]** を選択します。
2. **ビュープロパティ]** ダイアログボックスで **ノート]** タブをクリックします。

3. 新しいノートを入力するか、または既存のノートを編集します。
ノートを削除する場合は、テキストをすべて削除します。
4. **[OK]** をクリックしてダイアログ ボックスを閉じ、変更を保存します。

レコード ノートの追加と編集

レコードに関する補足情報を提供したり、関連ファイルへのリンクを作成したりするために、レコードにノートを追加することができます。レコードに追加されたノートは、そのレコードが含まれるテーブルのすべてのビューで利用できます。レコード ノートの内容はいつでも編集できます。

ノート アイコン

ノートが添付されているレコードは、ビューのレコード番号の横にあるノート アイコン  **12** で識別されます。ノートが追加されているレコードが存在するテーブルは、**ナビゲーターの 総覧]** タブで見分けることができます。そのテーブルの標準アイコンの左下にはノート アイコン  が付けられます。

RecordNote フィールド

テーブルで最初にレコード ノートを追加したときに、Analytics が自動的にテーブルレイアウトに **RecordNote** というフィールドを追加します。このフィールドでレコード ノートを保持します。**RecordNote** フィールドをビューに追加すれば、レコード ノートをビューに表示したり、Analytics 印刷レポートに含めたりすることができます。**RecordNote** フィールドが追加されたら、その列をダブルクリックしてレコード ノートの内容を迅速かつ簡単に編集することができます。

手順

方法の説明

ヒント

複数のレコード ノートを同時に追加または編集するには、NOTES コマンドを使用します。

1. ビュー タブのレコード番号列(左端の灰色の先頭列)で、対象とするレコード番号を右クリックし、**[ノートの編集]** を選択します。
2. 新しいノートを入力するか、または既存のノートを編集します。
ノートを削除する場合は、テキストをすべて削除します。
3. 関連ファイルへのリンクを作成する場合は、以下の手順を実行します。
 - a. ノート内で、リンクを挿入したい場所にカーソルを置きます。
 - b. **[ファイルの参照]** をクリックします。 

- c. **ファイルを開く**] ダイアログ ボックスで適切なファイルを選択し、**開く**] をクリックします。

次の構文を使用して、ファイルへのリンクがノートに追加されます。

```
file:///<ファイルへのパス>
```

4. **OK**] をクリックしてダイアログ ボックスを閉じ、変更を保存します。
5. ビューにレコード ノートを表示する、あるいは Analytics 印刷レポートにレコード ノートを含める場合は、以下の手順を実行します。
 - a. 表示領域内で右クリックしてから **列の追加**] を選択します。
 - b. **使用可能なフィールド**] リストで **RecordNote** をダブルクリックし、**OK**] をクリックします。

レコード ノートの保存

必要であれば、レコード ノートはテキスト ファイルや別の Analytics テーブルへ保存することができます。

Analytics テーブルのデータを更新 (**ソース ファイルから更新**) するとき、テーブル内のレコード ノートはすべて自動的に削除されます。レコード ノートはテーブルを更新する前に保存することができます。また、テーブルの更新時にレコード ノートを保存できる Analytics オプションを使用することもできます。保存されるノートの書式は、保存方法に応じて若干異なります。

方法の説明

テーブルを更新する前にノートをテキスト ファイルへ保存する

1. コマンドラインに次の構文を入力します。

```
LIST Recno() RecordNote to <ファイル名.txt>
```

例: `LIST Recno() RecordNote to Ap_trans_record_notes.txt`

テキスト ファイル名にはスペースを使用できません。

2. **実行** ] をクリックします。

テーブルを更新する前にノートを別の Analytics テーブルへ保存する

1. 標準の抽出操作で、**フィールド**] オプションを使用して **RecordNote** フィールドを選択します。
2. 抽出するフィールドを最低でももう1つ選択するか、"Recno()" 式を作成し、抽出する追加フィールドとして式を追加します。

Analyticsでは **RecordNote** フィールド自体を単独で抽出することはできません。

テーブルを更新している間にノートテキスト ファイルへ保存する

テーブルの更新時、ノートをファイルへ保存するかどうかを尋ねられたときに **[はい]** をクリックします。

レコード ノートは <テーブル名.txt> というテキスト ファイルに保存されます。このファイルは、作業中の Analytics プロジェクトと同じフォルダーに保存されます。

レコード ノートの削除

レコード ノートは、個別に削除または条件選択して削除したり、また、テーブル内の全レコード ノートを一括で削除したりすることもできます。

方法の説明

レコード ノートを個別に削除する

1. ビュータブのレコード番号列(左端の灰色の先頭列)で、対象とするレコード番号を右クリックし、**ノートの編集** を選択します。
2. **ノートの編集** ダイアログボックスの内容を、スペースや改行も含めすべて削除したら、**[OK]** をクリックします。

メモ

テーブル内のレコード ノートをすべて個別に削除しても、テーブルレイアウトで自動的に作成された **RecordNote** フィールドは削除されません。これは、**ナビゲーターの総覧** タブでそのテーブルのアイコン上にノート アイコンが引き続き表示されることを意味します。

テーブル内のすべてのレコード ノートを削除したい場合は、全レコードを一括で削除する方法を使用してください。この方法であれば **RecordNote** フィールドも削除されます。

選択したレコード ノートを削除する

1. 削除するノートを含むレコードを特定する IF 構文を使用して、コマンドラインに次の式を入力します。

```
NOTES IF <該当するフィルター条件> CLEAR
```

例: `NOTES IF Location = "03" CLEAR` Location #3 レコードのすべてのノートを削除します

2. **実行**  をクリックします。

テーブルのすべてのレコード ノートを一度に削除する

1. ビューに **RecordNote** 列が表示されている場合は、その列の列見出しを右クリックし、**選択列の削除** を選択してビューから削除します。

2. **編集 > ノート > テーブルから全ノートを削除]**の順にクリックします。
3. 確認ダイアログボックスで、**OK]**をクリックします。

レコード ノートがすべて削除され、**RecordNote** フィールドもテーブルレイアウトから削除されます。画面をリフレッシュすると、**ナビゲーター**の **総覧]**タブで対象テーブルのアイコンからノート アイコンが消えます。

フィールド ノートの追加と編集

フィールドにノートを追加して、そのフィールドに関する追加情報を提供することができます。このノートは、**テーブルレイアウト** ダイアログの **フィールド/式の編集]**タブにある **ノート]**列に表示されます。フィールド ノートの内容はいつでも編集できます。

オプション]ダイアログボックスで **テーブル履歴にフィールド定義を含める]**オプションを選択すると、Analytics 印刷レポートにフィールド ノートが追加されます。詳細については、「印刷オプション」ページ 146 を参照してください。

方法の説明

1. **編集 > テーブルレイアウト]**の順にクリックします。
2. **フィールド/式の編集]**タブでノートを追加するフィールドをダブルクリックします。
3. **フィールド ノートの編集]**  をクリックします。
4. 新しいノートを入力するか、または既存のノートを編集します。

ノートを削除する場合は、テキストをすべて削除します。

5. **閉じる**  をクリックします。
ノートは自動的に保存されます。
6. **入力を受け入れる**  をクリックします。

スクリプト ノートの追加と編集

あるスクリプトに関して、今後の参考のために保持しておきたい、あるいは他のユーザーのために文書化しておきたい情報がある場合は、ノート(注釈)をそのスクリプト プロジェクト項目に追加して記録することができます。

スクリプト プロジェクト項目は、スクリプトの外部にあるもので、スクリプトの本文に追加されるインラインコメントとは異なります。スクリプト プロジェクト項目 ノートの内容はいつでも編集できます。

ノートを追加、編集、削除または読むために、そのスクリプトを開く必要はありません。

方法の説明

1. **ナビゲーター**の **総覧]**タブで、作業対象のスクリプトを右クリックします。
2. **プロパティ]**を選択します。
3. **スクリプト プロパティ]**ダイアログボックスで **ノート]**タブをクリックします。
4. 新しいノートを入力するか、または既存のノートを編集します。

ノートを削除する場合は、テキストをすべて削除します。

5. [OK] をクリックしてダイアログ ボックスを閉じ、変更を保存します。

ワークスペース ノートを追加または編集する

詳細については、"ワークスペース ノートを追加または編集する" ページ 834を参照してください。

Analytics プロジェクトの情報の印刷

現在の Analytics プロジェクト中の次の項目について、Analytics から完全に詳細な情報のレポートを印刷することができます。

- フィールド定義とフィルター構文を含むテーブルレイアウト
- ビュー
- スクリプト
- インデックス
- ワークスペース
- 初期設定
- プロジェクト ノート
- コマンド ログ

項目の一部またはすべてについての情報を印刷するかどうかを選択できます。レポートは、分析が完了したときに使われていたプロジェクト設定を永久的に記録しておくため、あるいは Analytics プロジェクトに関する問題のトラブルシューティングに役立てるために使用できます。

メモ

コマンド ログを印刷するように選択した場合は、ログ全体が印刷されます。これは、ログのサイズによっては多数のページになる可能性があります。

Analytics プロジェクトの情報を印刷するには

1. **ファイルプロジェクト内容の印刷**の順にクリックします。
2. **ドキュメントの印刷**ダイアログボックスで、生成されたレポートに使用するフォントのポイント サイズを **フォント サイズ**ドロップダウンリストから選択します。
3. 次の項目の種類から1つ以上を選択して、プロジェクトに含まれる当該項目のすべての出現に関連する情報を印刷します。
 - **すべてのテーブルレイアウト**
 - **すべてのビュー定義**
 - **すべてのスクリプト定義**
 - **すべてのインデックス定義**
 - **すべてのワークスペース定義**ある項目の種類すべての出現について情報を印刷したいわけではない場合は、関連付けられているチェックボックスをオフにし、**Ctrl** または **Shift** キーを押しながら項目をクリックして、個々の項目を選択します。
4. 追加のプロジェクト情報を含めたり、出力の書式設定方法を指定したりするためには、次のチェックボックスをオンにします。

- **初期設定 - オプション]**ダイアログ ボックスで現在選択されている初期設定の一覧を印刷します。
 - **プロジェクト ノート** - プロジェクトに記録されているすべてのノート (注釈) を印刷します。
 - **ログ** - コマンド ログ全体を印刷します。
 - **カテゴリごとに改ページする** - 各プロジェクト項目のカテゴリ、初期設定、プロジェクト ノート、およびログ エントリの後に改ページを挿入します。このチェック ボックスがオフになっている場合、各カテゴリは前のカテゴリのすぐ後に記載されます。
 - **項目ごとに改ページする** - カテゴリ内の各項目の後に改ページを挿入します。たとえば、3つのスクリプトを選択した場合は、各スクリプト定義の後に改ページが挿入されます。このチェック ボックスがオフになっている場合、各項目はカテゴリ内の前の項目のすぐ後に記載されます。
5. **印刷]**をクリックします。
 6. **印刷]**ダイアログ ボックスで、必要なすべてのプリンター設定を構成し、**印刷]**をクリックします。**印刷]**ダイアログ ボックスでは、印刷ジョブの送信先とするプリンターなどの設定や、用紙サイズ、印刷の向きなどのプリンター固有のオプションの設定を変更することができます。

予期せずに関じられた Analytics プロジェクトの回復

保存されていないプロジェクトが開いている間に Analytics が予期せず終了した場合、次回そのプロジェクトを開くときに、自動保存されたプロジェクトの作業コピー(.ac ファイル)を開くか、保存されている最終保存バージョンのプロジェクト(.acl ファイル)を開くかの選択肢が与えられます。

- **作業コピーを開く**-作業コピーを開いて変更を保存した場合、**最終保存バージョン**のプロジェクト ファイルは上書きされ、元の作業コピーは削除されます。
- **最終保存バージョンを開く**-最終保存バージョンを開いて変更を保存した場合、**最終保存バージョン**のプロジェクト ファイルが更新され、作業コピーは削除されます。
- **選択するオプションがわからない場合は**、**[キャンセル]**をクリックして、作業コピーと最終保存バージョンの両方をバックアップすることができます。そうすると、どちらかのプロジェクト ファイルを使用できるようにしておけます。

ガイドライン

予期せず閉じたプロジェクトを開こうとすると、プロジェクト ファイルを回復するための3つのオプションを提示した ACL Analytics ダイアログ ボックスが表示されます。次のオプションから適切なものを選択します。

- 前回プロジェクトを保存した以降に、プロジェクト項目に変更を加えたり、分析手順を実行したりし、これらの操作のログ エントリを失いたくない場合は、**作業中]**をクリックします。

メモ

作業コピーにはほぼ完全な情報が含まれていますが、コマンドの処理中に Analytics が閉じた場合、作業コピーは壊れている可能性があります。

- プロジェクトの未保存の変更が重要ではない場合は、**前回の保存]**をクリックします。
- どちらかのバージョンのプロジェクト ファイルを使用できる選択肢を確保しておきたい場合は、**[キャンセル]**をクリックします。ダイアログ ボックスを閉じた後に、プロジェクト ファイルが保存されている Windows フォルダーに移動し、別のファイル名を使用して、作業コピーと最終保存バージョンの両方のバックアップを作成できます。

このページは意図的に空白のままにしています

一般的なデータ準備と分析タスク

Analytics を使用してデータを準備または分析しているとき、いくつかの一般的なタスクが頻繁に繰り返し発生します。

- 出力結果の保存
- 出力結果の保存場所の指定
- データの抽出
- データの追加
- データのエクスポート

このセクションでは、これらのタスクについてより詳細に説明します。また、キーフィールド、フィールドの連結、および乱数の生成についても説明します。

結果の保存と出力フォルダーの指定

Analytics テーブルの操作を実行し、その結果を新しい Analytics テーブルまたはテキスト ファイルに保存するとき、結果を保存する方法、および出力フォルダーの場所に関するいくつかのオプションがあります。

メモ

Analytics テーブルには、テーブルレイアウト (ナビゲーターで見ることができます) と、拡張子が .fil の関連するソース データ ファイル (ナビゲーターでは表示されず、Windows フォルダーに格納されています) が含まれています。

テーブルレイアウトとソース データ ファイルの違いを理解することが、結果を保存するとき、および出力フォルダーを指定するときに重要になります。

詳細については、"Analytics テーブルの構造" ページ 118を参照してください。

結果の保存

Analytics テーブルまたはテキスト ファイルに結果を保存する場合には、次のオプションがあります。

- **保存** -新しい Analytics テーブルまたはテキスト ファイルに結果を保存する
- **追加** -既存の Analytics テーブルまたはテキスト ファイルに結果を追加する
- **上書き** -既存の Analytics テーブルまたはテキスト ファイルを上書きする

追加すると、ソース データ ファイルは更新されますが、テーブルレイアウト は変更されません。上書きすると、ソース データ ファイルとテーブルレイアウト の両方が置き換えられます。

メモ

一部の Analytics 操作は、Analytics テーブルまたはテキスト ファイルのいずれかに結果を保存することをサポートしますが、両方への保存はサポートしません。

別のプロジェクトでソース データの後ろに追加または上書きする

一般的に、開いている Analytics プロジェクトで、テーブルの後ろに追加するか、テーブルを上書きします。結果を保存するときに適切な Windows フォルダーへ移動すれば、別のプロジェクトに含まれる Analytics テーブルのソース データ ファイルに追加または上書きすることもできます。

これを行うと、更新または上書きされたテーブルは別のプロジェクトに存続する上、開いているプロジェクトにも追加され、2つのプロジェクトの両方のテーブルレイアウトは同じソース データ ファイルを共有します。

注意

この方法で結果を保存する前に、別のプロジェクトでソース データを上書きすることが意図した結果であることを確信した上で行ってください。

出力フォルダーの指定

結果を保存する際に、考慮すべき「出力フォルダー」の種類が2つあります。

- **Analytics プロジェクト フォルダー** -: 結果の Analytics テーブルレイアウトが入っています(テキストファイルに結果を保存する場合は適用されません)
- **Windows フォルダー** -: Analytics テーブルレイアウトと関連付けられている結果のソースデータファイル(.fil)、または結果のテキストファイルが入っています

Analytics プロジェクト フォルダーは Windows フォルダーではありません

Analytics プロジェクト フォルダーは Windows フォルダーではなく、Analytics プロジェクト フォルダーを作成しても、対応する Windows フォルダーは作成されません。Analytics プロジェクト フォルダーは、Analytics プロジェクト ファイル(.acl) 内部の仮想フォルダーです。Analytics プロジェクト フォルダー間でテーブルレイアウトを移動しても、Windows フォルダーにおけるテーブルのソースデータファイルの場所に何ら影響はありません。

テーブルレイアウトとソースデータファイルの配置オプション

Analytics テーブルに結果を保存する際には、テーブルレイアウトとソースデータファイルの配置に関するオプションがいくつかあります。テーブルレイアウトとソースデータファイルの配置は、それぞれ単独で制御することができます。テーブルレイアウトの配置には SET FOLDER コマンドを使用し、ソースデータファイルの配置には標準の Windows ナビゲーションを使用します。

項目	配置オプション(出力フォルダー)
テーブルレイアウト	<ul style="list-style-type: none">◦ アクティブなテーブルが入っている Analytics プロジェクト フォルダー(デフォルト)◦ SET FOLDER コマンドを使って指定された、アクティブなテーブルのフォルダー以外の Analytics プロジェクト フォルダー
ソースデータファイル(.fil)	<ul style="list-style-type: none">◦ Analytics プロジェクトが入っている Windows フォルダー(デフォルト)◦ Analytics プロジェクトが入っているフォルダー以外の Windows フォルダー◦ AX サーバー上の Prefix フォルダー(サーバーテーブルのみ。デフォルト)◦ AX サーバー上の Prefix フォルダー以外のフォルダー(サーバーテーブルのみ)

SET FOLDER を使用して、Analytics プロジェクト 出力フォルダーを指定する

結果を出力する前に、SET FOLDER コマンドを使用して、テーブルレイアウトに使用する Analytics プロジェクト フォルダーを設定します。SET FOLDER コマンドを Analytics のコマンドラインに入力するか、スクリプトに含めます。複数の例を以下に示します。

出力フォルダーは、リセットするか、プロジェクトを閉じない限り、設定したままの状態であり続けます。プロジェクトを開くときに、出力フォルダーはデフォルトのアクティブなテーブルフォルダーに戻ります。

メモ

SET FOLDER コマンドで指定するファイルパスでは、スラッシュを使う必要があります。

コマンド構文	説明
<code>SET FOLDER /Results</code>	テーブルレイアウトは、アクティブなテーブルフォルダーではなく AnalyticsL プロジェクト フォルダ、 Results に置かれます。
<code>SET FOLDER /Results/Duplicates</code>	テーブルレイアウトは、アクティブなテーブルフォルダーではなく Analytics プロジェクト サブフォルダ、 Duplicates に置かれます。
<code>SET FOLDER /</code>	テーブルレイアウトは、アクティブなテーブルフォルダーではなく Analytics プロジェクト のルートディレクトリに置かれます。
<code>SET FOLDER</code>	出力フォルダーを、デフォルトのアクティブなテーブルフォルダーにリセットします。
<code>DISPLAY OUTPUTFOLDER</code>	現在の Analytics プロジェクト 出力フォルダーを表示します。

テーブルレイアウトと Analytics プロジェクト フォルダとの相互作用

操作の結果生じるテーブルレイアウトと Analytics プロジェクト フォルダとの相互作用について、次のテーブルにまとめてあります。

いかなる場合でも、ソースデータファイルの保存先として選択した Windows フォルダは、Analytics 内のテーブルレイアウトやプロジェクト フォルダの場所に影響を与えません。

アクション	Analytics の出力フォルダーはアクティブなテーブルフォルダー(デフォルト)	Analytics の出力フォルダーは SET FOLDER コマンドによる指定場所
新しい Analytics テーブルに結果を保存する	テーブルレイアウトは、アクティブなテーブルと同じ Analytics プロジェクト フォルダに追加される	テーブルレイアウトは、SET FOLDER コマンドで指定された Analytics プロジェクト フォルダに追加される
プロジェクト内の既存の Analytics テーブルに結果を追加する	既存のテーブルレイアウトは移動されない	既存のテーブルレイアウトは移動されない
結果を保存し、プロジェクト内の既存の Analytics テーブルを上書きする	テーブルレイアウトは、アクティブなテーブルと同じ Analytics プロジェクト フォルダに既に存在しない限り、そのフォルダへ移動される	テーブルレイアウトは、SET FOLDER コマンドで指定された Analytics プロジェクト フォルダに既に存在しない限り、そのフォルダへ移動される
別のプロジェクト内の既存の Analytics テーブルに結果を追加する	テーブルレイアウトは、アクティブなテーブルと同じ Analytics プロジェクト フォルダに追加される	テーブルレイアウトは、SET FOLDER コマンドで指定された Analytics プロジェクト フォルダに追加される

アクション	Analytics の出力フォルダーはアクティブなテーブルフォルダー(デフォルト)	Analytics の出力フォルダーは SET FOLDER コマンドによる指定場所
結果を保存し、別のプロジェクト内の既存の Analytics テーブルを上書きする	別のプロジェクト内のテーブルレイアウトは変更されない 両方のテーブルレイアウトは同じソースデータファイルを共有する	別のプロジェクト内のテーブルレイアウトは変更されない 両方のテーブルレイアウトは同じソースデータファイルを共有する

Windows 出力フォルダーの指定

デフォルトでは、テーブルレイアウトに関連付けられたデータファイル(.fil)は、Analytics プロジェクトが入っている Windows フォルダーに出力されます。データファイルを別の Windows フォルダーに出力するには:

- ユーザー インターフェイスで - コマンド ダイアログボックスを使用するときにフォルダーに移動します
- スクリプトで - コマンドでテーブルを出力するファイルパスを指定します。例:

```
CLASSIFY ON Vendor_Number SUBTOTAL Invoice_Amount TO "C:\Data
analysis\January\Classified_transactions_Jan.FIL"
```

Analytics プロジェクト フォルダーと Windows フォルダーを一致させる

Analytics プロジェクト フォルダーと Windows フォルダーの2つのフォルダー セット間を直接対応させることが重要である場合、あるいは監査のワークフローに有用である場合は、これらのフォルダーの構造を一致させることができます。Analytics プロジェクト フォルダーは Windows フォルダーではなく、Analytics プロジェクト フォルダーを作成しても、対応する Windows フォルダーは作成されません。ただし、必要に応じて、並列フォルダー構造を手動で作成し、手動で管理することができます。そもそも、2つのフォルダーのセット間にプログラムのつながりは存在しません。

Analytics プロジェクト フォルダーと Windows フォルダーを一致させるには

1. Analytics プロジェクトで、必要なフォルダーを作成します。たとえば、"元のデータ"、作業ファイル"、"結果"など。
2. そのプロジェクトを含んでいる Windows フォルダーか、または別の Windows フォルダーに、Analytics プロジェクト フォルダーの構造を正確に複製するサブフォルダーを作成します。
3. 初度監査の内容を適切に編成します。たとえば、Excel ファイルや Access ファイルなどの元のデータファイルを適切な Windows フォルダーに置きます。データ定義ウィザードを使用して、これらのファイルのいずれかを Analytics にインポートするときは、元のデータファイルと同じ Windows フォルダーか、または別の適切な Windows フォルダーに新しいソース データファイル(.fil)を保存します。最後に、新しいソース データファイルを含んでいる Windows フォルダーと一致する Analytics プロジェクト フォルダーに Analytics テーブルレイアウトを保存します。

ヒント:

テーブルレイアウトが適切な Analytics プロジェクト フォルダーに保存されていることを確認するには、フォルダーを右クリックしてインポート処理を開始します。

4. Analytics テーブルに結果を保存する操作を実行する前に、必要であれば、SET FOLDER コマンドを使って、結果のテーブルレイアウト用として適切な Analytics プロジェクト フォルダーを指定します。

詳細については、"結果の保存と出力フォルダーの指定" ページ 196を参照してください。

5. 操作に関連付けられたダイアログ ボックスで、絶対または相対のファイルパスを使用するか、対象フォルダーへ移動することによって、ソース データファイル用の適切な Windows フォルダーを指定します。

例: `C:\Results\Classify.fil` または `Results\Classify.fil`

データの抽出

抽出を使用すると、Analytics テーブルからレコードまたはフィールドの一部または全部を、新しい Analytics テーブルへコピーすることができます。

新しいテーブルは次のいずれかになります。

- ソーステーブル内のすべてのレコードおよびフィールドを含む、ソーステーブルと同一のコピー
- ソーステーブル内のレコードのサブセット
- ソーステーブル内のフィールドのサブセット
- ソーステーブル内のレコードおよびフィールドの両方のサブセット

ソーステーブルに存在する並べ替え順序は、新しいテーブルに保持されます。

メモ

データを抽出し、それを既存の Analytics テーブルの末尾に追加するのは、データを結合する手法の1つです。これは、データの結合に関するセクションで説明します。詳細については、「データの抽出と追加」ページ 940を参照してください。

データを抽出することのメリット

以下に、新しいテーブルへデータを抽出する理由をいくつか挙げます。

- 特定の分析に関連するデータのみサブセットが生成され、ファイルサイズと処理時間が削減される
- フィルターを使用すると、特定の項目をさらに分析するために別のテーブルに抽出できる
- ファイルの作業コピーに元のデータファイルの内容を抽出することにより、元ファイルの整合性が保持される
- 演算フィールドが、実際に計算された値が格納された物理フィールドに変換される
- サーバーテーブルから新しいローカルテーブルにデータが抽出される
- 2つ以上の関連テーブルから新しい Analytics テーブルにデータが抽出される

データの抽出とテーブルのコピーの違い

すべてのデータを抽出することと、ナビゲーターでテーブルをコピーすること(編集コピー)の違いは、抽出では、新しいテーブルレイアウトだけでなく新しいソースデータファイル(.fil)も作成されるのに対し、コピーでは、新しいテーブルレイアウトだけが作成され、依然として元のソースデータファイルと関連付けられ続けるということです。

レコード単位、ビュー単位、フィールド単位の抽出

データを抽出する場合には、次のオプションがあります。

- **レコード** -: レコード全体を抽出します。
- **ビュー** -: ビューにあるすべてのフィールドを抽出します。
- **フィールド** -: フィールドを個別に選択して抽出します。

レコード全体を抽出するときは、テーブルレイアウトには定義されていないすき間に格納されているデータも含め、レコードが完全にコピーされます。

ビュー内の全フィールドまたは個別フィールドを抽出するときは、ソーステーブル内のすべてのフィールドを抽出しても、レコードの定義されていない部分はすべて無視されます。

演算フィールドの抽出

演算フィールドは、レコード単位で抽出を行う場合は演算フィールドのままです。これらは、ビュー内のフィールドまたはビュー単位またはフィールド単位で抽出を行う場合には、適切なデータ型の物理フィールドに変換され、実際に計算された値が格納されます。

演算フィールド内の時刻データの抽出

演算フィールドに、UTC オフセットを伴うローカル時間 (23:59:59-05:00 など) が含まれる場合、そのローカル時間と UTC オフセットはレコードの抽出時に保存されます。

ビューまたはフィールド単位で抽出する場合は、ローカル時間と UTC オフセットはオフセットなしの UTC に変換されます。たとえば、23:59:59-05:00 は 04:59:59 となります。

ビュー単位による抽出の補足説明

抽出 ダイアログボックスで **ビュー** オプションを選択すると、アクティブなビューで現在表示されているデータをすべて抽出することができます。

ビュー単位で抽出する場合は、以下の点に留意してください。

抽出されるフィールド	<p>ビュー内で現在表示されているフィールドのみが抽出されます。テーブルレイアウトには存在していてもビューに表示されていないフィールドについては抽出されません。</p> <p>ビュー内のフィールドはすべて抽出されます。フィールドのサブセットを抽出したい場合は、ビューから不要なフィールドを削除する、ビューを新たに作成し必要なフィールドのみを含める、あるいはビュー単位の抽出ではなくフィールド単位の抽出を使用するようにしてください。</p>
フィールドの順番	<p>フィールドは、ビューに現れる順序と同じ並びで抽出されます。別の並び順でフィールドを抽出する場合は、抽出を実行する前に、ビューでフィールドの配置を変えるか、ビューを新たに作成してフィールドを希望の順序で配置してください。</p>

フィルターリング	ビューにフィルターが適用されている場合は、フィルター条件に一致するデータのみが抽出されます。
レコード ノート	レコード ノートは、 [RecordNote] 列が事前にビューへ追加されていた場合にのみ抽出されます。
代替列見出し	代替列見出しがビューレベルで指定されている場合、ビュー単位の抽出はビューレベルの見出しを保存します。コマンド ログの構文を使用して抽出コマンドを再実行する場合、テーブルレイアウトで指定されている代替列見出しが使用され、ビューレベルの見出しは無視されます。
スクリプト コマンド ライン	ビュー単位の抽出の指定は、スクリプト内またはコマンド ラインからはサポートされません。ビュー単位の抽出は、ACLScript で表現すると、アクティブビューの全フィールドを使用したフィールド単位の抽出 (EXTRACT FIELDS) と実質的に同じになります。フィールドはビューに表示されている順序で抽出されます。

論理フィールドの抽出

論理フィールドを抽出するには、 **[フィールド一覧にフィルターを含める]** (**[ツールオプションインターフェイス]**) がオンになっている必要があります。

コントロール合計の設定

テーブル内のすべてのレコード、ビュー内の全データ、または個別フィールドを抽出する場合は、すべてのデータが実際に抽出されたことを確認するために、数値フィールドにコントロール合計を設定することができます。

[テーブルレイアウト] ダイアログボックスで、フィールドにコントロール合計を設定します。データを抽出したら、新しいテーブルで **[ツールテーブル履歴]** を選択して、入力と出力のコントロール合計を比較します。詳細については、「物理フィールドを定義する」 ページ 790を参照してください。

サーバー テーブルおよびローカル テーブルからのデータの抽出

サーバー テーブルとローカル テーブルの両方からデータを抽出することができます。サーバー テーブルから抽出されたデータは、サーバー上またはローカルコンピューター上のテーブルに保存できます。ローカル テーブルから抽出されたデータは、ローカルコンピューター上のテーブルにのみ保存できます。

手順

Analytics テーブルからレコードまたはフィールドの一部または全部を、新しい Analytics テーブルへ抽出することができます。

メモ

データを抽出し、それを既存の Analytics テーブルの末尾に追加するのは、データを結合する手法の1つです。これは、データの結合に関するセクションで説明します。詳細については、「データを抽出して追加する」ページ 945を参照してください。

方法の説明

1. レコードまたはフィールドを抽出するテーブルを開きます。
2. **データ > 抽出**]の順に選択します。
3. **メイン**]タブで、次のいずれかを選択します。
 - **レコード** - レコード全体を抽出します
レコード内のフィールドは、テーブルレイアウトに現れる順序と同じ並びで抽出されます。
 - **ビュー** - すべてのフィールドを現在のビューに抽出します
フィールドは、ビューに表示される順序で抽出されます。
 - **フィールド** - 選択した個別のフィールドを抽出します
フィールドは、選択した順序で抽出されます。

テーブルの関係における子テーブルからデータを抽出する場合、その子テーブルのフィールドがビューに追加済みであれば、**フィールド**]あるいは**ビュー**]を選択します。**レコード**]オプションを使って子テーブルのデータを抽出することはできません。

メモ

1つまたは複数の演算フィールドを抽出する場合、**レコード**]を選択すると、抽出されたフィールドは計算式として保持されます。

ビュー]または**フィールド**]を選択すると、抽出されたフィールドは適切なデータ型の物理フィールドに変換され、実際に計算された値が格納されます。

4. **フィールド**]を選択した場合は、次のいずれかを行います。
 - **抽出フィールド**]リストから適切なフィールドを選択します。
 - **抽出フィールド**]をクリックして、適切なフィールドを選択するか、または式を作成して、**OK**]をクリックします。

テーブルの関係における子テーブルからフィールドを選択したい場合は、**抽出フィールド**]をクリックします。**選択済みのフィールド**]ダイアログボックスの**対象テーブル**]ドロップダウンリストを使用して、適切な子テーブルを選択できます。
5. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または**[f]**ボタンをクリックし、**式ビルダー**]を利用してIFステートメントを作成します。

メモ

If条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

6. 次のいずれかを実行します。
 - **保存先**]テキストボックスに、新しい Analytics テーブルの名前を指定します。

- **保存先]** ボタンをクリックして、**保存]** または **ファイルを保存する名前]** ダイアログ ボックスで新しい Analytics テーブルの名前を指定するか、または既存のテーブルに上書きする場合はそのテーブルを選択します。

Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所に新しいテーブルを保存したり、その場所にある既存のテーブルを上書きしたりすることもできます。たとえば、**C:\Results\GL_2011.fil** または **Results\GL_2011.fil** と指定します。保存または上書きするテーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

7. **プロジェクトと同一場所接続されたい場合は、保存先のいずれかを指定するか、または別のローカルフォルダーへ移動する場合は、**[ローカル]** を選択します。**
- サーバーの Prefix フォルダーに出力テーブルを保存する場合は、**[ローカル]** を選択解除されたままにしておきます。

メモ

出力結果が Analytics Exchange のサーバー テーブルの分析や処理から生成される場合は、**[ローカル]** を選択します。**[ローカル]** の設定は、結果テーブルを Analytics Exchange へインポートする場合に選択解除することはできません。

処理の完了時に、出力テーブルを自動的に開くようにするには、**出力テーブルの使用]** をオンにします。

- 8.
9. **詳細]** タブをクリックします。
10. **適用範囲]** パネルで、適切な出力オプションを選択します。

- **すべて**
- **先頭**
- **次へ**
- **While**

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。

<p>While</p>	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>	

- 必要であれば **[EOF(ファイルの終わりの処理)]** をオンにして、テーブルの終わりに達したときに、もう一度抽出操作が実行されるようにします。

EOF パラメーターは通常、大規模な分析プロセスの一環としてレコードを抽出するとき、Extract コマンドがスクリプトのグループ内で発生する場合に使用されます。シーケンシャルレコード間の比較に基づいてレコードを抽出する場合は、テーブルの最後のレコードが確実に抽出されるように、EOF を使用する必要があるかもしれません。

- [OK]** をクリックします。
- 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

既存テーブルへの出力結果の追加

新しい Analytics テーブルまたはテキスト ファイルに結果を出力できる Analytics 操作では、既存の Analytics テーブルまたはテキスト ファイルに結果を追加することもできます。追加は、既存のテーブルまたはファイルの終わり、つまり、既存のテーブルまたはファイルの最後のレコードの後に、グループとして出力結果内のレコードを加えます。

並べ替えの仕組み

出力結果と、出力先のテーブルまたはファイルに存在するすべての並べ替え順は、結果として生じる結合したデータ内のそれぞれのレコード セットで別々に保持されます。必要に応じて、結合されたテーブルを順番に並べ替え、すべてのレコードで単一の並べ順を作成することができます。

同じデータ構造の要件

Analytics テーブルへの追加が正常に行われるためには、出力結果と出力先テーブルのレコードがまったく同じ構造である必要があります。次の構造特性がすべて同じでなければなりません。

- データ要素の選択
- フィールドの数と順序
- 対応するフィールドのデータ型と長さ
- 対応する日付および日付時刻フィールドの形式
- レコードの長さ

レコード構造の特性が1つでもが同一でない場合は、雑然としたデータが生じる可能性があります。テキスト ファイルに追加するときは、同一の構造は必須ではありません。

レコード構造の詳細については、「データ構造およびデータ書式の要件」ページ 918を参照してください。

2つのテーブルのデータ構造の比較に、DISPLAY コマンドを使用する方法については、「データ構造の比較」ページ 209を参照してください。

既存のファイルに追加する]オプション

Analytics 操作を実行する前に **既存のファイルに追加する]オプション**を選択しておくと、出力結果は強制的にターゲット テーブルへ追加されます。この追加は、データ構造が同一であるかどうかに関係なく行われます。データ構造が同一である場合にのみ、このオプションを選択してください。

最後に追加]ボタン

Analytics テーブルへの追加を行う際に **既存のファイルに追加する]** オプションの選択をオフにしておく、より安全です。このオプションがオフになっていると、アクティブなテーブルの処理時に、Analytics が出力結果とターゲット テーブルのレコード長を比較します。レコード長が同一である場合は、上書きを確認するメッセージにオプションとして **最後に追加]** が表示されます。

最後に追加]ボタンの制限事項

最後に追加] ボタンが表示されたとしても、それぞれのデータ構造が異なっている可能性もあります。たとえば、出力結果とターゲット テーブルがいずれも 100 文字のレコード長であっても、フィールドの順序が異なっている、フィールド数が異なっている、あるいは各フィールドの開始位置とフィールド長が同じでもデータ型が異なることもあります。

最後に追加]ボタンが表示されないとき

最後に追加] ボタンが表示されない場合はレコード長が同一ではありません。これはデータ構造の 1 つ以上の要素が異なっており、処理の前に手動で一致させる必要があることを示しています。この自動チェックは、ターゲット テーブルが、開いている Analytics プロジェクト内にある場合にのみ発生します。

ファイルを上書きする前にメッセージを表示する]オプション

オプション] ダイアログ ボックスで **ファイルを上書きする前にメッセージを表示する]** オプションが選択されている必要があります(デフォルトで選択済み)。**ファイルを上書きする前にメッセージを表示する]** オプションが選択されていないと、上書きの確認メッセージは表示されず、レコード長の自動比較も行われません。また、**既存のファイルに追加する]** オプションを選択しない限り、出力結果はターゲット テーブルに追加されず、常に上書きされます。

データ構造の比較

DISPLAY コマンドを使用すると、Analytics テーブルのテーブルレイアウトを表示することができます。このテーブルレイアウトによってテーブルのデータ構造が明確になります。

ソーステーブルまたは出力結果をターゲット テーブルに追加するか、2つのテーブルをマージする前に、双方のテーブルのテーブルレイアウトを表示してそれぞれのデータ構造を視覚的に比較すれば、それらの構造が同一かどうかを確認することができます。追加またはマージ操作が正しく動作するには、2つのテーブルのデータ構造が同じでなければなりません。

DISPLAY コマンドでは以下のデータ構造要素が表示されます。

- レコード長
- フィールド名
- フィールドの数
- フィールドの順番
- フィールドの開始位置
- フィールド長
- フィールドのデータ型
- 数値フィールドの小数点以下の桁数
- フィールド書式の詳細
- 演算フィールドの式

データ構造の比較

1. ソーステーブルを開きます。
2. コマンドラインに `DISPLAY` を入力して **実行** をクリックします。 
そのソーステーブルのデータ構造情報が結果タブに表示されます。
3. ピンアイコン  をクリックして結果タブをピン留めし、そのデータ構造情報を保持します。
4. ターゲット テーブルを開きます。
5. コマンドラインに `DISPLAY` を入力して **実行** をクリックします。 
そのターゲット テーブルのデータ構造情報が2番目の結果タブに表示されます。
6. ピンアイコン  をクリックして結果タブをピン留めし、そのデータ構造情報を保持します。
7. 2つの結果タブを見比べながらデータ構造を視覚的に比較します。

データ構造に違いがあった場合は、追加またはマージ操作が正しく動作しないでしょう。データ構造を手動で一致させる必要があるかもしれません。詳細については、「フィールドの一致」ページ 921を参照してください。

2つのテーブルを結合している場合は、Analytics 外でデータを結合するという方法もあります。詳細については、「データの結合の代替方法」ページ 917を参照してください。

データのエクスポート

その他のアプリケーションでデータを使用できるように Analytics データをその他のファイル形式にエクスポートできます。

- Microsoft Excel(*.xlsx、*.xls)
- テキスト(*.txt)
- 区切り文字付きテキスト(*.del)
- カンマ区切り値(*.csv)
- Microsoft Access(*.mdb)
- XML(*.xml)
- JSON(*.json)
- dBASE III PLUS(*.dbf)
- ほかのドキュメントまたはアプリケーションに貼り付けられる Windows のクリップボード

メモ

古い Excel ファイル(*.xls) および Microsoft Access ファイル(*.mdb) にエクスポートするには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、"Microsoft Access データベースエンジンを除外するオプション" ページ 2664を参照してください。

Analytics データを HighBond アプリにエクスポートすることもできます。

- 例外をリザルト アプリへエクスポート
詳細については、"HighBond のリザルト アプリへの例外のエクスポート" ページ 216を参照してください。
- 圧縮された CSV ファイル(*.csv.gz) を HighBond ロボットまたはロボット アプリのワークフローロボットにエクスポート
詳細については、"HighBond のロボット アプリへのデータのエクスポート" ページ 224を参照してください。

エクスポートされたファイルを開く

Analytics の **結果** タブで、エクスポート コマンド ログ エントリの画面表示には次へのリンクが含まれます。

- エクスポートされたファイル
- ファイルを含むフォルダー

リンクを使用すると、直接 Analytics からファイルとファイルを含むフォルダーを開くことができ便利です。

エクスポートされたファイルはファイル拡張子に関連付けられたアプリケーションで開きます(アプリケーションがコンピューターにインストールされている場合)。

Excel へのエクスポート

Analytics テーブルを個々の Excel ワークシートとして、新しく作成された、または既存の Excel ファイルへエクスポートすることができます。既存の Excel ファイルへのエクスポートは ***.xlsx** でのみサポートされています。

文字およびサイズの制限

データを Excel ファイルにエクスポートするときには、以下の制限事項があります。

制限	詳細
レコード数	<ul style="list-style-type: none"> Excel 2007 以降 (*.xlsx) - 16,384 フィールドで最大 1,048,576 レコード (Excel がサポートする最大ワークシートサイズ) Excel 97 および 2003 - 最大レコード件数は 65,536 件 <p>この最大件数を超える Analytics テーブルでもエクスポートは行えますが、上限を超えた分のレコードは無視されエクスポートされません。</p>
フィールドの長さ	<ul style="list-style-type: none"> 特定のフィールド長上限なし フィールド長の合計が、レコード全体の長さ上限である 32 KB (非 Unicode 版 Analytics では 32,765 文字、Unicode 版 Analytics では 16,382 文字) を超えることはできません。 Excel 2.1 の場合は、最大 247 文字
フィールド名の長さ	<ul style="list-style-type: none"> 最大 64 文字 Excel 2.1 の場合は、最大 248 文字

古い Excel ファイル (*.xls) にエクスポートされた日付時刻および時刻データ

古い Excel ファイル (*.xls) にエクスポートされた日付時刻および時刻データは、最初に Excel ファイルを開いたときに正しく表示されない場合があります。日付時刻の場合は日付部分しか表示されず、時刻の場合は "00/01/1900" と表示されるかもしれません。Excel ファイルには日付時刻や時刻データが正しくエクスポートされています。ただし、Excel 側でそのデータが適切に表示されるよう、セルの書式を変更する必要があります。

サーバー テーブルのデータのエクスポート

サーバー テーブルからエクスポートされたデータをサーバーに保存することはできません。サーバー テーブルおよびローカル テーブルのデータは、ローカル コンピューターにエクスポートすることができます。

手順

Analytics テーブルのレコードまたはフィールドの一部または全部を、別のアプリケーションで使用するためにエクスポートすることができます。

方法の説明

エクスポートするフィールドを指定する

1. **データ** > **エクスポート** を選択します。
2. **メイン** タブで、次のいずれかを選択します。
 - **フィールド** - エクスポートするフィールドを指定します
このオプションを選択すると、テーブルレイアウトの物理フィールド名を使用して、フィールドがエクスポートされます。
フィールドの名前変更の詳細については、「テーブルレイアウトのフィールドの名前を変更する」ページ 822を参照してください。
 - **ビュー** - 現在のビューのすべてフィールドをエクスポートします
オプションを選択すると、列表示名を使用して、フィールドがエクスポートされます。フィールドは、ビューに現れる順序と同じ並びでエクスポートされます。
列の名前変更の詳細については、「ビュー内の列名の変更」ページ 851を参照してください。
3. **フィールド** を選択した場合は、次のいずれかを行います。
 - **エクスポート フィールド** リストからエクスポートするフィールドを選択します。
 - **エクスポート フィールド** をクリックして、フィールドを選択するか、または式を作成します。

エクスポートする形式を選択する

エクスポート形式 ドロップダウンリストからエクスポート形式を選択し、以下のガイドラインに従います。

エクスポート形式	ガイドライン
区切り文字付きまたはテキスト (またはカンマ区切り値)	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> • 区切り文字付き - 任意で フィールド名付きでエクスポートする を選び、エクスポートにフィールド名または列名を見出しとして含めます。 区切り文字付きファイルで使用する列 区切り文字 および テキスト修飾子 文字を選択します。 • テキスト - 任意で フィールド名付きでエクスポートする を選び、エクスポートにフィールド名または列名を見出しとして含めます。 エクスポートされたテキスト ファイルのフィールド値は、空白スペースで区切られ、値は修飾されません。

エクスポート形式	ガイドライン
	<p>ヒント</p> <p>カンマ区切り値ファイル(*.csv)にエクスポートするには、区切り文字付き]を選択し、列区切り文字]ドロップダウンリストで、]を選択します。エクスポート先]フィールドでエクスポート ファイル名を指定するときには、.csv ファイル拡張子を含めます。例：<code>vendors.csv</code></p>
Excel (.xlsx)	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <p>新しい Excel ファイルを作成するか、既存の Excel ファイルにエクスポートする</p> <p>ワークシートの追加]テキストボックスのデフォルト名を使用するか、必要に応じて変更します。</p> <p>新しく作成した、または既存の .xlsx Excel ファイルにエクスポートすると、Excel ファイル内にワークシートが自動的に作成されます。ワークシートの名前は、変更しない限り、エクスポート元の Analytics テーブルと同じ名前になります。</p> <p>メモ</p> <p>ワークシート名を指定する場合、名前に含めることができるのは、英数文字とアンダースコア文字(_)だけです。名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。</p> <p>既存の Excel ファイルのワークシートを上書きできますが、そのワークシートが本来、Analytics から Excel にエクスポートして作成された場合にのみ行えます。</p> <p>Excel で直接作成されたワークシート、または名前が変更されているワークシートへの上書きはできません。</p> <p>既存の Excel ファイルを上書きする</p> <p>ワークシートを追加]テキスト ボックス内の名前を削除して、そのテキスト ボックスを空のままにします。</p> <p>既存の Excel ファイルを上書きすると、エクスポート元となる Analytics テーブルと同じ名前のワークシートが、結果として生じる Excel ファイル内に自動的に作成されます。</p>
XML	<ul style="list-style-type: none"> <p>任意で XML スキーマ付きでエクスポートする]をオンにすると、エクスポートされる XML ファイルに XML スキーマを含めることができます。</p> <p>XML スキーマには、フィールドのデータ型など、XML ファイルの構造を記述するメタデータが含まれています。ファイルがエクスポートされたら、スキーマに対してファイルを検証することができます。</p>
Unicode 版の Analytics からエクスポート	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <p>エクスポートしているデータが拡張 ASCII(ANSI)によってサポートされていない場合は、Unicodeを選択 -します</p> <p>エクスポートされたデータは、Unicode UTF-16 LE として暗号化されます。</p> <p>エクスポートしているデータのすべての文字が拡張 ASCII(ANSI)によってサポートされている場合は、Unicode を選択しないでください -</p> <p>エクスポートされたデータは拡張 ASCII(ANSI)としてエンコードされます。サポートされていない文字はエクスポートされたファイルから省略されます。</p>

エクスポート形式	ガイドライン
	<p>メモ</p> <p>Unicode オプションは、クリップボード、区切り、テキスト、またはXML にエクスポートするときのみ使用できます。</p> <p>詳細については、"Diligent Unicode 製品" ページ 2672を参照してください。</p>
HighBond (HighBond ユーザーのみ)	<ul style="list-style-type: none"> "HighBond のリザルト アプリへの例外のエクスポート" ページ 216を参照してください。
ロボット (HighBond ユーザーのみ)	<ul style="list-style-type: none"> "HighBond のロボット アプリへのデータのエクスポート" ページ 224を参照してください。

エクスポートの最終処理

- 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

- 次のいずれかを実行します。
 - 保存先]** テキスト ボックスで、エクスポートされたデータを格納するファイルの名前を指定します。
 - 保存先]** ボタンをクリックして、**保存]** または **ファイルを保存する名前]** ダイアログ ボックスでファイル名を指定するか、または既存のファイルを選択します。

Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

メモ

クリップボードにデータをエクスポートする場合は、ファイルにデータを保存しないため、**保存先]** テキスト ボックスは無効になります。

- 詳細]** タブをクリックします。
- 適用範囲]** パネルで、適切な出力オプションを選択します。
 - すべて
 - 先頭
 - 次へ
 - While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

5. 区切り文字付きテキストファイルまたはテキスト ファイルにエクスポートするとき、エクスポートされたデータを既存ファイルの末尾に追加したい場合は、**既存のファイルに追加する**]をオンにします。
6. **OK**]をクリックします。
7. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

HighBond のリザルト アプリへの例外のエクスポート

HighBond を使用している場合は、Analytics テーブルの例外データをリザルトのテーブルにエクスポートすることができます。例外をエクスポートするには、Analytics からデータをエクスポートする標準的な手順を使用できますが、異なる点が若干あります。

リザルトは、改善およびワークフローを自動化するアプリであり、例外データの管理、およびアンケートによる人間の背景の追加を行い、監視をトリガーとメトリクスで継続します。詳細については、[リザルトでのデータの操作](#)を参照してください。

セキュリティ要件

リザルトの統制テストへ例外データをエクスポートする機能は、特定の HighBond の役割の割り当て、または管理者特権を必要とします。

- リザルト コレクションのプロフェッショナル ユーザーまたはプロフェッショナル部長の役割を担うユーザーは、そのコレクション内のあらゆる統制テストに結果をエクスポートすることができます。

メモ

監査部門長の役割のユーザーのみが統制テストの既存のデータをエクスポートおよび上書きできます。

- HighBond アカウント管理者およびリザルト管理者は、管理している HighBond インスタンス内のすべてのコレクションにおいて、自動的にプロフェッショナル部門長の役割を取得します。

詳細については、[リザルト アプリアクセス許可](#)を参照してください。

エクスポートの制限事項

例外データをリザルト内の統制テストにエクスポートする際の制限事項を下に示します。

これらの制限内では、同じ統制テストに複数回エクスポートできます。統制テストにすでにデータが存在する場合は、それを上書きする、または新しいデータを追加するオプションがあります。

メモ

統制テストに最大 100,000 レコードエクスポートできますが、より小さく、より明確な例外セットを作成する方がよいでしょう。

項目	最大
エクスポートあたりのレコード数	100,000
統制テストあたりのレコード数	100,000
レコードあたりのフィールド数	500
フィールドあたりの文字数	256

Analytics とリザルトとの間でフィールドを一致させる

リザルトと Analytics 間でデータを往復する場合は、リザルト テーブルのすべてのフィールド名がより厳しい Analytics フィールド名要件を満たすことを確認する必要があります。そうでない場合、Analytics とリザルト データが一致しないおそれがあります。

たとえば、リザルト フィールド名の特殊文字は、Analytics にインポートされる時に自動的にアンダースコアに変換されます。これは、フィールド名がリザルトの元の名前と一致しないことを意味します。後から Analytics データをリザルトの元のテーブルにエクスポートする場合は、フィールドが正しく一致しません。

往復するデータでこの問題を回避するには、CSV または Excel ファイルからリザルトにデータをアップロードする前に、以下の Analytics フィールド名要件を満たしていることを確認します。

- 特殊文字またはスペースがない
- 数字で始まらない
- 英数文字またはアンダースコア文字(_)のみを含む

メモ

アンケート フィールドの末尾にデータを追加するときには、Analytics で表示名を変更した場合でも、リザルトの列の表示名は、アンケート構成で指定された名前のままです。

上書きオプションとリザルト 主キー

Analytics から既存のリザルト テーブルに例外データをエクスポートするときには、エクスポートされたデータをテーブルの最後に追加するか、テーブルを完全に上書きするオプションがあります。

リザルト テーブルのフィールドが主キーに指定されていて、エクスポートしているデータに対応するフィールドが含まれている場合は、エクスポート処理の動作が少し異なります。(リザルトでの主キーの指定の詳細については、[主キーの指定](#)を参照してください。)

次の表は、さまざまな可能性をまとめたものです。

	リザルトに主キーがない	リザルトにおける主キー
上書き オプションが選択されていない	エクスポートされたデータは既存のリザルトテーブルの最後に追加されます	<ul style="list-style-type: none"> 一致する値 - リザルトの主キーフィールドと対応するフィールドが Analytics からエクスポートされたフィールドに、一致する値が存在する場合、リザルトのレコードはエクスポートされたレコードに存在する値で更新されます。 一致する値がない - リザルトの主キーフィールドと対応するフィールドが Analytics からエクスポートされたフィールドに、一致する値が存在しない場合、リザルトのレコードは更新されず、エクスポートされたレコードはテーブルの最後に追加されます。
上書き オプションが選択されている	エクスポートされたデータは既存のリザルトテーブルを置換 (上書き) します	エクスポートされたデータは既存のリザルトテーブルを置換 (上書き) します

リザルトへの例外のエクスポート

メモ

HighBond に接続するときには、パスワードを指定する必要がある場合があります。詳細については、「パスワード要件」ページ 222 を参照してください。

エクスポートするフィールドを指定する

1. エクスポートする例外データが格納されたテーブルを開きます。
2. **データ > エクスポート**] を選択します。
3. **メイン**] タブで、次のいずれかを選択します。

- **フィールド** - エクスポートするフィールドを指定します

このオプションを選択すると、テーブルレイアウトの物理フィールド名を使用して、フィールドがエクスポートされます。

フィールドの名前変更の詳細については、「テーブルレイアウトのフィールドの名前を変更する」ページ 822 を参照してください。

- **ビュー** - 現在のビューのすべてフィールドをエクスポートします

オプションを選択すると、列表示名を使用して、フィールドがエクスポートされます。フィールドは、ビューに現れる順序と同じ並びでエクスポートされます。

列の名前変更の詳細については、「ビュー内の列名の変更」ページ 851 を参照してください。

4. **フィールド]**を選択した場合は、次のいずれかを行います。
 - **エクスポート フィールド]**リストからエクスポートするフィールドを選択します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

- **エクスポート フィールド]**をクリックして、フィールドを選択するか、または式を作成します。

エクスポート オプションを選択する

1. **エクスポート 形式]**ドロップダウン リストから **HighBond** をクリックします。
2. 次のいずれかを実行します。

- **リザルト テーブルの最後に追加する**

エクスポートされたデータをリザルトの既存のテーブルの最後に追加する場合は、**[上書き]**をオフにします。

メモ

既存のリザルト フィールドに Analytics フィールドを追加できるのは、両者間で一致する物理フィールド名がある場合に限りです。各アプリケーション内でのフィールドの表示名は問いません。Analytics では、物理フィールド名はテーブルレイアウトの名前です。

2つのアプリケーションにおけるフィールドの順序はフィールド名一致に影響しません。

リザルト テーブルのフィールドの物理名と一致しない物理名があるエクスポートされたフィールドは、テーブルに新しい列を作成します。

- **リザルト テーブルを置換(上書き)する**

リザルトで既存のテーブルを置換する場合は、**[上書き]**を選択します。

詳細については、「上書きオプションとリザルト 主キー」ページ 217を参照してください。

3. (任意) 列の表示名をリザルトにエクスポートするには、**フィールド表示名を含める]**を選択します。

このオプションを選択すると、リザルトで、列の表示名と物理名が、Analytics の名前と同じになります。

フィールド表示名を含める]を選択しない場合、結果は、フィールドまたはビューのどちらでエクスポートしているのかによって異なります。

	フィールドのエクスポート	ビューのエクスポート
フィールド表示名を含めるが選択	リザルトのフィールド名は Analytics のフィールド名です。Analytics の表示名が、リザルトの表示名になります。	
フィールド表示名を含めるが選択されていない	Analytics の表示名が、リザルトのフィールド名になります。	Analytics の表示名が、リザルトのフィールド名と表示名になります。

メモ

Analytics のバージョン 14.1 より前からビューをエクスポートする方法で最初に作成されたリザルト テーブルの最後にビューを追加する場合は、**フィールド表示名を含める**を選択しないでください。このようにすると、リザルトの名前と異なるフィールド名の列がエクスポートされる可能性があります。この場合、リザルトで新しい列が作成され、アプリケーション間でデータの整合性がとれなくなります。

エクスポートの最終処理

- 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

- 次のいずれかを実行します。

エクスポート先テーブルの ID 番号がわかっている場合：

To テキスト ボックスに ID 番号を入力します。

- 番号を入力します。引用符は付けしないでください。たとえば、**99** のように入力します。
- 数字のみを入力します。ファイル名を入力してはいけません。
- 北米(米国)以外のデータセンターにエクスポートする場合は、データセンターのコードも指定する必要があります。統制テストの ID 番号とデータセンターのコードは、アット マーク(@)で区切る必要があります。たとえば、**99@eu**と指定します。データセンターコードは、どの地域の HighBond サーバーにデータをエクスポートするのかを指定します。
 - af** - アフリカ(南アフリカ)
 - ap** - アジア太平洋(シンガポール)
 - au** - アジア太平洋(オーストラリア)

- **ca** - 北米 (カナダ)
- **eu** - ヨーロッパ (ドイツ)
- **sa** - 南米 (ブラジル)
- **us** - 北米 (米国)

組織の HighBond インスタンスに対して承認されているデータセンターコードのみを使用できません。北米のデータセンターがデフォルトであるため、**@us** を指定するのはオプションです。

エクスポート先テーブルの ID 番号がわかっていない場合または新しいテーブルを作成する場合：

- a. **To** を選択し、**宛先テストを選択**] ダイアログボックスで該当する分析フォルダーに移動します。
- b. 次のいずれかを実行します。

- 既存のテーブルを選択し、**OK**] をクリックします。
- **新しいデータアナリティクス**] フィールドに名前を入力し、**作成**] をクリックします。

エクスポート] ダイアログボックスに戻ります。統制テスト ID 番号とデータセンターコードは **To**] テキストボックスに事前に入力されます。

3. **詳細**] タブをクリックします。
4. **適用範囲**] パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While] テキスト ボックスに条件を入力するか、または While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>

メモ

"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。

ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。

5. [OK]をクリックします。

例外データがリザルトへエクスポートされる間は進捗インジケータが表示されます。エクスポートの操作が完了すると、そのエントリがログに記録されます。

パスワード要件

パスワードが必要ではない

Analyticsのコピーをアクティベーションするためにオンラインアクティベーションを使用した場合は、リザルトにエクスポートするパスワードを指定する必要がありません。パスワードは自動的に作成され、コンピューターに保存されたアクティベーション情報に基づいてリザルトに送信されます。

パスワードが必要

次のどちらの状況でも、リザルトにエクスポートするためにパスワードを指定する必要があります。

- オフラインアクティベーションを使用して Analytics のコピーをアクティブ化した場合
- スクリプトを使用してリザルトにエクスポートし、ロボットでスクリプトを実行した場合

必須のパスワード値は、HighBond アクセストークンです。

HighBond アクセストークンを取得

注意

アカウントパスワードのようにアクセストークンを保護してください。HighBond アカウント固有の情報が含まれます。アクセストークンを共有しないでください。

1. 次のいずれかを実行します。

- Analytics メインメニューで [ツール>HighBond アクセストークン] を選択します。
- スクリプト エディター内を右クリックし、[挿入 > HighBond トークン] を選択します。

[API アクセストークンを管理します] ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する** - **トークン** 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - **トークンを作成 > Analytics** をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー** をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログ ボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているデータアクセスとパスワード定義方法に応じて、次のいずれかを実行します。

Analytics ユーザーインターフェイス

コピーしたトークンを、HighBond に手動でアクセスする際に表示されるパスワードプロンプトに貼り付けます。

Analytics スクリプト

- **PASSWORD コマンド** - コピーしたトークンをスクリプト実行中に表示されるパスワードプロンプトに貼り付けます。
- **SET PASSWORD コマンド** - コピーしたトークンを、スクリプト内の SET PASSWORD コマンド構文の適切な位置に貼り付けます。

5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[アクセストークンの作成と管理](#)を参照してください。

HighBond のロボット アプリへのデータの のエクスポート

HighBond でロボット アプリを使用している場合、ローカルな Analytics テーブルのデータを、ロボットの圧縮された CSV (`*.csv.gz`) ファイルにエクスポート (アップロード) できます。HighBond ロボット またはワークフローロボットの **作業データ** タブにエクスポートし、Python/HCL スクリプトを使用してデータを分析または処理できます。ACL ロボットにはエクスポートできません。

ACL スクリプトとオンプレミスのロボット エージェントを使用している組織では、この機能により、ローカルネットワークからクラウドベースのロボット インスタンスにデータを移動する、統合された ACLScript-Python/HCL データ自動化ワークフローを構築できます。

ロボットにエクスポートした後の、圧縮された CSV ファイルへのアクセス方法については、[load_working_file\(\) メソッド](#)を参照してください。ロボット アプリの概要については、「[ロボットを使用した作業の自動化](#)」を参照してください。

メモ

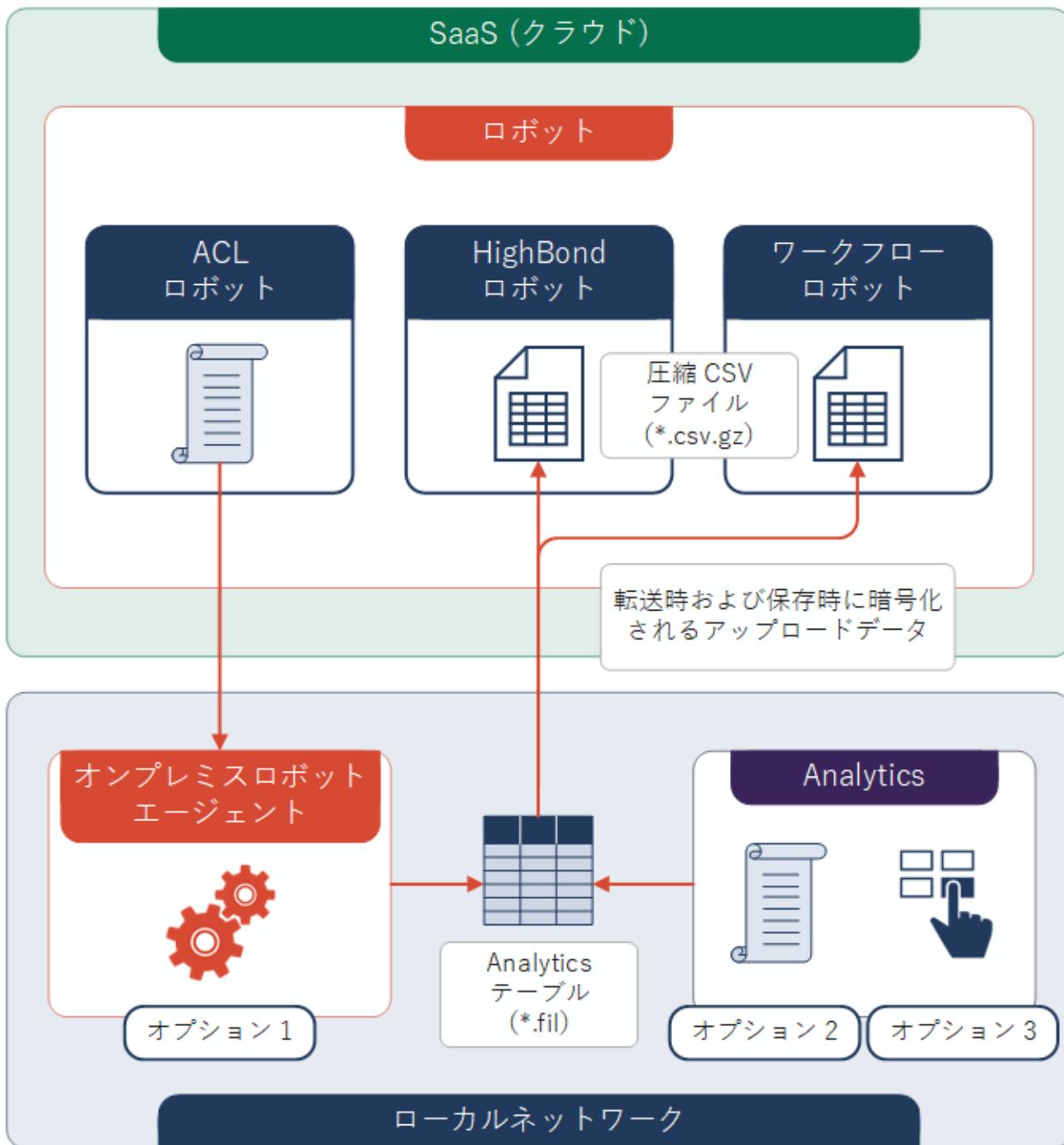
ロボットにアップロードされたデータは、安全な AWS データセンターに保存され、転送時にも保管時にも暗号化されます。

スクリプトによる処理の自動化または手動エクスポート

ロボットにローカルの Analytics データをエクスポートする処理をスクリプトで実行できます。また、Analytics ユーザーインターフェイスを使用して、データを手動でエクスポートすることもできます。

- **オプション 1** - ロボットの ACL ロボットにスクリプトをアップロード (コミット) し、オンプレミスのロボット エージェントを使用してスクリプトを実行します。
- **オプション 2** - Analytics でローカルにスクリプトを実行します。
- **オプション 3** - Analytics ユーザーインターフェイスを使用して、データを手動でエクスポートします。Analytics からデータを手動でエクスポートするための標準的な手順を使用しますが、いくつかのわずかな違いがあります。

スクリプトを使用したプロセスの自動化の詳細については、「EXPORT コマンド」ページ 1784を参照してください。



セキュリティ要件

圧縮 CSV ファイルをロボット アプリにアップロードする機能には、特定のロボットロールの割り当て、または管理者権限が必要です。

- ロボットの編集者ロールまたは所有者ロールを持つユーザーは、圧縮された CSV ファイルをロボットにアップロードできます。レビュー担当者ロールを持つユーザーは、アップロードできません。
- ロボット管理者は自動的にすべてのロボットのコラボレーターになり、所有者のロールを持ちます。
- Professional サブスクリプションを持つ HighBond システム管理者は、自動的にロボット管理者になります。

詳細については、[ロボットアプリのアクセス許可](#)を参照してください。

ロボットにデータを手動でエクスポート

メモ

HighBond に接続するときには、パスワードを指定する必要がある場合があります。詳細については、「パスワード要件」ページ 228を参照してください。

エクスポートするフィールドを指定する

1. エクスポートするデータが格納されたテーブルを開きます。
2. **データ > エクスポート**] を選択します。
3. **メイン**] タブで、次のいずれかを選択します。
 - **フィールド** - エクスポートするフィールドを指定します
このオプションを選択すると、テーブルレイアウトの物理フィールド名を使用して、フィールドがエクスポートされます。
フィールドの名前変更の詳細については、「テーブルレイアウトのフィールドの名前を変更する」ページ 822を参照してください。
 - **ビュー** - 現在のビューのすべてフィールドをエクスポートします
オプションを選択すると、列表示名を使用して、フィールドがエクスポートされます。フィールドは、ビューに現れる順序と同じ並びでエクスポートされます。
列の名前変更の詳細については、「ビュー内の列名の変更」ページ 851を参照してください。
4. **フィールド**] を選択した場合は、次のいずれかを行います。
 - **エクスポート フィールド**] リストからエクスポートするフィールドを選択します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

- **エクスポートフィールド**] をクリックして、フィールドを選択するか、または式を作成します。

メモ

フィールドは選択した順序でエクスポートされます。

エクスポート オプションを選択する

1. **[エクスポート形式]**ドロップダウンリストから **[ロボット]**を選択します。
2. 省略可能。既存のファイルを宛先ロボットの同じ名前に置き換える場合、**[上書き]**を選択します。

メモ

既存の *.csv.gz ファイルへのデータの追加はサポートされていません。**[上書き]**が選択されておらず、宛先ロボットに同じ名前のファイルが既に存在している場合、エクスポートはエラーになります。

エクスポートの最終処理

1. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

2. **[保存先]** をクリックし、**[送信先の選択]** ダイアログボックスで該当するロボットに移動します。
ロボットはアルファベット順に表示され、HighBond ロボットが最初に表示され、次にワークフローロボットが表示されます。
3. ロボットを選択します。
ロボットを選択すると、ロボットのリストの下に、ロボットの種類を含む通知が表示されます。
4. **[環境]** で、**[開発]** または **[本番]** を選択してファイルのエクスポート先のロボットモードを指定します。
5. **[OK]** をクリックします。
[エクスポート] ダイアログボックスに戻ると、**[保存先]** テキストボックスに宛先ロボットの HighBond API URL が設定されています。
6. **[詳細]** タブをクリックします。
7. **[適用範囲]** パネルで、適切な出力オプションを選択します。
 - **すべて**
 - **先頭**

- 次へ
 - While
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While テキスト ボックスに条件を入力するか、または While ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>	

8. [OK] をクリックします。

データがロボットへエクスポートされる間は進捗インジケータが表示されます。エクスポートの操作が完了すると、そのエントリがログに記録されます。

データが正常にエクスポートされると、**<Analytics テーブル名>.csv.gz** という名前のファイルが宛先ロボットの **作業データ** タブに表示されます。

パスワード要件

パスワードが必要

次のどちらの状況でも、ロボットにエクスポートするためにパスワードを指定する必要があります。

- スクリプトを使用してロボットにエクスポートし、ロボットでスクリプトを実行した場合
- オフラインアクティベーションを使用して Analytics のコピーをアクティブ化した場合

必須のパスワード値は、HighBond アクセストークンです。

パスワードが必要ではない

Analytics のコピーをアクティベーションするためにオンラインクティベーションを使用した場合は、ロボットにエクスポートするパスワードを指定する必要がありません。パスワードは自動的に作成され、コンピューターに保存されたアクティベーション情報に基づいてロボットに送信されます。

HighBond アクセストークンを取得

注意

アカウント パスワードのようにアクセストークンを保護してください。HighBond アカウント固有の情報が含まれます。アクセストークンを共有しないでください。

1. 次のいずれかを実行します。
 - Analytics メインメニューで **[ツール>HighBond アクセストークン]** を選択します。
 - スクリプト エディター内を右クリックし、 **[挿入 > HighBond トークン]** を選択します。

[API アクセストークンを管理します] ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **[API トークンの管理]** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロファイルからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。
 - **[既存のトークンを使用する - トークン]** 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **[新しいトークンを作成する - トークンを作成 > Analytics]** をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **[コピー]** をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているデータアクセスとパスワード定義方法に応じて、次のいずれかを実行します。

Analytics ユーザーインターフェイス

コピーしたトークンを、HighBond に手動でアクセスする際に表示されるパスワードプロンプトに貼り付けます。

Analytics スクリプト

- **PASSWORD コマンド** - コピーしたトークンをスクリプト実行中に表示されるパスワードプロンプトに貼り付けます。
- **SET PASSWORD コマンド** - コピーしたトークンを、スクリプト内の SET PASSWORD コマンド構文の適切な位置に貼り付けます。

5. Launchpad で、トークンが含まれているダイアログボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[アクセストークンの作成と管理](#)を参照してください。

キーフィールドについて

キーフィールドを使用する Analytics の操作を以下に示します。

- 結合
- 関連付け
- マージ
- 並べ替え
- インデックス

「キーフィールド」という用語にはさまざまな意味があり、実行される操作に応じて、キーフィールドは異なる機能を持つことができます。Analytics におけるキーフィールドは、リレーショナルデータベース用語におけるキーフィールドの典型的な定義とも若干異なります。

結合、関連付け、またはマージにおけるキーフィールド

Analytics における結合、関連付け、およびマージの操作は、すべて 2 つ以上のテーブルのデータを結合するものです。この場合の「キーフィールド」は、2 つのテーブルを結び付ける共通のフィールドを指します。そのフィールドの値が比較され照合されます。マージの場合は、比較して 1 つにまとめられます。

Analytics でデータを結合するための別の方法である [最後に追加] では、キーフィールドを使用しません。

主テーブル、副テーブル、キーフィールド

結合またはマージを行うときに開いている最初のテーブルが主テーブルとなり、選択したキーフィールドが主キーフィールドとなります。2 番目に開いたテーブルが副テーブルとなり、選択したキーフィールドが副キーフィールドとなります。

テーブルを関連付ける場合、主は「親」と呼ばれ、副は「子」と呼ばれます。

データを結合するとき、主と副のテーブルおよびキーフィールドは自由に選択できます。Analytics は、キーフィールドのペアが同一のデータ構造であることを必要とする場合でも、フィールドの特定の選択を強制しません。

ソースデータから保持されない一意キーと外部キーの指定

Analytics テーブルにインポートされたデータ、または Analytics サーバーに存在するデータは、非リレーショナルのフラット ファイル(.fil ファイル)に格納されています。.fil ファイルでは、以前に、リレーショナルデータベースの主キー、一意キー、外部キー、または副キーとして機能していたかもしれないフィールドが、キーでないフィールドと何ら異なる扱いを受けません。

Analytics コマンドで、リレーショナルデータベースの主キー(従業員 ID など)のみが、Analytics テーブルでキーを指定するときに、主キーまたは親キーとなります。ユーザーは、Analytics で結合や関係を構築する

方法について決定を下すとき、どのフィールドがソースデータベースの主キーや一意キーであったかを知る必要があるかもしれません。しかし、Analyticsにはこの情報が格納されていません。

同じ状況が、Analytics データベースプロファイルを使用して、直接データベーステーブルにアクセスするときにも当てはまります。Analyticsは、データベースのどのフィールドがキーフィールドであるかについて情報を持っていないので、データベースクエリを構築する際は、自身でこの情報を知る必要があります。

Analytics キーフィールドは一意性が強制されない

Analyticsは、Analytics コマンドで、指定するキーフィールドに一意性を強制しません。主キーフィールドと副キーフィールドの両方に同じ値が存在できます。

並べ替えまたはインデックス作成におけるキーフィールド

Analyticsにおける並べ替えおよびインデックス作成は、テーブルにシーケンシャルな順序を強要する、単一テーブルの操作です。この場合の「キーフィールド」は、並べ替えまたはインデックス付けの基礎となるフィールド、並べ替えまたはインデックスの基準とするフィールドを指します。

「並べ替えキー」または「インデックスキー」と同等です

Analyticsにおけるソート キーフィールドやインデックス キーフィールドは、一般的なコンピューター用語またはデータベース用語における「ソート キー」や「インデックス キー」に相当します。一意性は強制されません。

データを並べ替えたりインデックス付けしたりするとき、キーフィールドは自由に選択できます。元のソースデータのソート キーまたはインデックス キーであった可能性のあるフィールドについて、それらのフィールドの値が現状でもシーケンシャル順であるとしても、Analyticsにはフィールドに関する情報が何も格納されていません。

キーおよび入れ子になった並べ替えまたはインデックス

並べ替えやインデックスが入れ子になっている場合、「主」のソートまたはインデックス キーは、「副」のソートまたはインデックス キーよりも優先されます。主キーと副キーは、単にそれらを選んだ順番によって確定されます。

フィールドの連結

分析において、テーブル内の2つ以上のフィールドを1つのデータ要素として検査または処理する必要がある場合、それらのフィールドを連結(加算)した演算フィールドを作成することができます。この演算フィールド内の結合済みデータに対して検査または処理が行えます。

たとえば、ファーストネーム、ミドルネームおよびラストネームに分かれている3つのフィールドを連結してフルネームを含む1つのフィールドを作成したり、あるいはベンダーIDフィールドと地区コードフィールドを連結して、テーブルの各小売チェーンの直販店別に一意の識別子を作成したりすることができます。

メモ

連結できるのは文字のフィールドのみです。必要であれば、連結前にAnalytics関数を使用して文字以外のデータを文字データへ変換することができます。

1. テーブルを開いて、**編集 > テーブルレイアウト**]の順に選択します。
2. **新しい式の追加**  をクリックします。
3. 連結するフィールド用の名前を **名前**]に入力します。
4. **f(x)**  をクリックして**式ビルダー**を開きます。
5. 2つ以上のフィールドと加算演算子(+)を使用して式を作成します。

必要であれば、式内で空白などの区切り文字を含めたり、TRIM()関数を使用してフィールドの先頭にある空白を取り除くことができます。例:

```
TRIM(名 + " " + TRIM(ミドルネーム) + " " + 姓)
```

6. **OK**]をクリックします。
"式の型の不一致"エラーになった場合、フィールドの1つ以上が文字のフィールドでない可能性があります。
7. **入力を受け入れる** ] をクリックし、**閉じる** ] ボタンをクリックして **テーブルレイアウト**]ダイアログボックスを終了します。
演算フィールドをビューに追加する方法については、"ビューに列を追加する" ページ 849を参照してください。

乱数の生成

Analytics を使って乱数一式を生成できます。一式のサイズや(その一部の)範囲などの、いくつかのパラメーターを指定できます。

通常、Analytics 以外のアプリケーションには生成された値一式が使用されます。たとえば、ランダム選択したハードコピーファイルが抽出されたりします。

メモ

ランダム選択の結果を、統計的に有効なものにするか、または母集団全体を代表するものにする必要がある場合には、上記とは異なる正式なサンプリング処理を行う必要があります。詳細については、「データのサンプリング」ページ 1021を参照してください。

乱数を生成するには

1. ツール> 乱数の生成]の順に選択します。
2. [メイン]タブで、次の情報を入力します。
 - **個数** - 生成される乱数一式のサイズ。
最大 32767 の数字を生成できます。
 - **シード値** - 省略可能。乱数ジェネレーターを初期化するために使用される値。
シード値を指定したり、'0' のシード値を入力したりできます。あるいは、シード値を Analytics にランダムに選択させたい場合は、シード値をブランクのままにすることができます。
シード値を指定する場合、任意の数字を指定することができます。シード値がそれぞれ一意であると、異なる乱数一式になります。同じシード値を指定すると、同じ乱数一式が生成されます。特定の乱数一式を複製したい場合は、シード値を明示的に指定し、これを保存します。
 - **最小値** - 乱数一式に可能な最小の数値。有効な任意の数値または数式が使用できます。
 - **最大値** - 乱数一式に可能な最大の数値。有効な任意の数値または数式が使用できます。
 - **列数** - 乱数一式の表示に使用する列数。デフォルトの列数は6です。
 - **一意** - 一意の数値のみが乱数一式に含まれるよう指定します。
デフォルトの動作は、乱数一式に重複が可能になっています。

メモ

乱数一式の指定サイズが**最小値**と**最大値**の範囲の75%を超える場合は、**一意**を選択してはいけません。選択すると、乱数の選択肢が破棄される数が多くなりすぎます。

- **並べ替え** - 乱数一式が昇順で表示されるように指定します。
デフォルトでは、ランダムに選択されている順序で数字が表示されます。

- **既存のファイルに追加する** - 既存のファイルを上書きしないで、既存のファイルの最後に出力結果を追加する必要があることを指定します。
3. **出力]**タブをクリックします。
 4. **出力先]**パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics 表示領域の **結果]**タブに乱数一式を表示します。
 - **ファイル** - 乱数一式をテキスト ファイルに保存します。
 5. 出力の種類として **ファイル]**を選択した場合は、**出力指定]**パネルの **名前]**テキスト ボックスにファイル名を指定します。または、**名前]**をクリックして、既存のファイルを参照して選択します。

既存のファイルに追加する]チェック ボックスがオンになっている場合は、同じ名前のファイルがあれば、そのファイルに追加されます。オフになっている場合は、ファイルを上書きするか、または出力に追加するかどうかの確認を求められます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所に保存したり、その場所にあるファイルに出力を追加したりすることもできます。たとえば、**C:\Output\random.txt** または **Output\random.txt** のように指定します。
 6. **OK]**をクリックします。

ランダムに選択したレコードの生成

Analytics を使って、ランダムに選択したレコードを生成できます。

次に示す方法ではレコードをランダムに選択していますが、この選択結果の出力はレコードの母集団全体を代表しているとは見なすことができません。

選択したレコードを分析し、その分析結果を母集団全体に適用するには、選択が統計的に有効で、母集団全体を代表している必要があります。詳細については、「データのサンプリング」ページ 1021を参照してください。

手順

1. ランダムに選択したいレコードが含まれるテーブルを開きます。
2. メインメニューから [サンプリング > レコード/金額単位サンプリング > サンプル] を選択します。
3. [サンプルの種類] から "レコード" を選択します。
4. [サンプルパラメーター] から "ランダム" を選択します。
5. 次の値を指定します。
 - **サイズ** - ランダムに選択したいレコードの数
 - **シード値** - (省略可能) Analytics 乱数ジェネレーターを初期化するための値
シード値には任意の数値を指定できます。同じシード値を再指定することで、同じランダム選択レコードを再作成できます。
シード値を Analytics にランダムに選択させたい場合は、シード値には '0' または空白を指定します。
 - **母集団** - テーブル内のレコードの総数
 - **保存先** - 出力テーブルの名前
6. [OK] をクリックします。

このページは意図的に空白のままにしています

データの定義とインポート

Analytics でデータを分析する前に、Analytics テーブルを作成してデータを含める必要があります。データを定義およびインポートして、Analytics テーブルを作成します。ソースデータのタイプに関係なく、定義とインポートは、同じ基本プロセスに従います。

1	移動または接続する	ソース データ ファイルに移動したり、ソース データを含むファイルまたはデータベースに接続します。
2	定義する	<p>ソース データを定義します。これは、Analytics がこのデータを読み取れるように、ソース データの構造および特性に関する情報を明らかにすることを意味します。</p> <p>メモ Analytics が一部のソース データ形式を自動的に定義するため、データをユーザーが定義する必要はありません。</p>
3	直接的にインポートまたは読み取る	ソース データをネイティブの Analytics データ ファイルにインポートするか、Analytics データ ファイルを作成せずにソースから直接データを読み取ります。
4	Analytics テーブルに名前を付けて保存する	自動的に作成された Analytics テーブルに名前を付け、保存します。

メモ

データソースに接続するとき、またはデータソースからインポートするときには、Analytics は厳密に読み取り専用です。詳細については、「Analytics によるデータアクセスは読み取り専用です」ページ 243を参照してください。

データの定義とインポートをするためのコンポーネント

Analytics は、データの定義、ソースからの直接的なデータのインポートまたは読み取り、および Analytics テーブルの作成を行うためのコンポーネントを 2 つ提供します。

- データ定義ウィザード
- データアクセス ウィンドウ

データ定義ウィザード

データ定義ウィザードは、さまざまなデータソースにアクセスする標準的な方法(ほとんどの場合はデータベース)を提供する、ページベースのウィザードです。

ウィザードを使用するデータの定義およびインポートの基本的なプロセスは一貫していますが、表示されるページの選択および順序は使用しているデータソースの種類によって異なります。

新しいプロジェクトとテーブル

データ定義ウィザードは、新しい Analytics プロジェクトを作成するとき、およびプロジェクトに新しい Analytics テーブルを追加するとき([インポート > ファイル]、または [ファイル > 新規 > テーブル]) デフォルトで自動的に表示されます。

データ アクセス ウィンドウ

データ アクセス ウィンドウは、データベースまたはファイルのいずれかでソース データにアクセスするために使用できる複数のデータ コネクタを含む視覚的なインターフェイスです。データ コネクタは、ネイティブの Analytics ODBC ドライバー、またはインストールしたどのような Windows ODBC ドライバーでも使用します。

データ アクセス ウィンドウを使用してデータ ソースに接続したら、データの定義およびインポートを行うための標準のオプション一式が用意されます。

- テーブルの検索および選択
- フィールドの選択
- テーブルの結合
- データのフィルターリング
- データ インポートのプレビュー
- データ インポートのサイズの見積もり
- フィールド長の仕様
- SQL インポート ステートメントを直接編集するための SQL モード

データの定義とインポートするためのコンポーネント

多くの場合、アクセスするデータソースの種類が、使用する必要のあるコンポーネントを指示します。Analytics でアクセスできるすべてのデータソースのリスト、およびデータソースのアクセスに使用する必要のあるコンポーネントについては、"Analytics でアクセスできるデータソース" ページ 242を参照してください。

一部のデータソースについては、Microsoft Excel ファイルや Access ファイルをインポートする時など、データ定義ウィザードまたはデータアクセス ウィンドウのいずれかを使用できます。

一般に、データアクセス ウィンドウは、モダンで視覚的なインターフェイスであり、データ定義ウィザードよりも大幅に使いやすくなっています。

以下のテーブルは、異なる 2 つのコンポーネントで利用可能なさまざまなオプションを比較したものです。

データの定義とインポート

オプション	データ定義ウィザード	データアクセス ウィンドウ
テーブルの選択	はい	はい
テーブルの検索	いいえ	はい
フィールドの選択	データソースによって異なる	はい
複数のテーブルのインポート	いいえ (Excel でははい)	はい (最大 5 つ)
テーブルの結合	いいえ	はい
データのフィルター	いいえ	はい
データ インポートのプレビュー	はい (基本)	はい (モダンなインターフェイス、簡単に更新可能)
データ インポートのサイズの見積もり	いいえ	はい
フィールド長の指定	はい	はい
フィールド名の変更	データソースによって異なる	はい (SQL モード)
フィールドのデータ型の変更	データソースによって異なる	いいえ (データ型はインポート 後に変更可能)

Analytics でアクセスできるデータソース

Analytics では、さまざまなファイルタイプ、データベース、クラウド データソースにアクセスできます。

データソースに関する情報：

- **アクセス方法** - データ定義ウィザードまたはデータアクセス ウィンドウを使用してデータにアクセスするかどうか、またはいずれかを使用できるかどうか
- **データ読み取り方法** - 結果の Analytics テーブルが Analytics データファイル(.fil) から読み取られるか、データソースから直接読み取られるかどうか

.fil ファイルのデータは固定であり、手動で更新する必要がありますが、直接アクセスされるデータソースは Analytics テーブルが開くたびに最新の情報で更新されます。

メモ

Analytics データファイル(.fil) でサポートされる最大レコード長は 32 KB です。32 KB を超えるレコードがあると、インポート処理は失敗します。

データアクセス]ウィンドウを使用して、ODBC データソースにアクセスする

データアクセス]ウィンドウを使用して、ODBC 対応データソースへの接続を作成できます。

次の ODBC オプションは データアクセス]ウィンドウで使用できます。

- **ACL コネクタ** - Analytics には、Oracle、Microsoft SQL Server および Salesforce などのデータソースへのさまざまな標準データ接続があります。標準データコネクタのリスト全体については、"データベースとクラウド データ サービス" ページ 244 を参照してください。
- **ACL DSN コネクタ(バンドル)** - には、データパートナーの CData Software が提供する追加のデータコネクタが含まれます。
- **Windows DSN コネクタ** - コンピューターに既にインストールおよび構成された Windows ODBC ドライバーまたは DSN を使用します。
- **その他のコネクタ** - 必要な ODBC ドライバーをインストールし、データアクセス]ウィンドウで使用します。

メモ

データアクセス]ウィンドウを使用してデータベースまたはクラウド データ サービスに接続するときには、特定の要件または前提条件が存在します。詳細については、"データベースまたはクラウド データ サービスに接続する前に" ページ 376 を参照してください。

Analytics によるデータ アクセスは読み取り専用です

データソースに接続するとき、またはデータソースからインポートするときには、Analytics は厳密に読み取り専用です。Analytics はデータソースのデータを追加、更新、削除できません。また、いかなる方法でも、データソースを修正することはできません。この制限は、ファイルベースのデータソース、データベース、クラウド データ サービスといった Analytics がアクセスできるすべてのデータソースに適用されます。

インポートされたデータから作成される Analytics データファイル(.fil) も Analytics によって読み取り専用として処理されます。データソースからファイルを更新する以外に、Analytics は .fil ファイルを修正できません。

.fil ファイルは作成するために使用されるデータソースとは完全に別です。.fil ファイルを削除すると、データソースに影響しません。

ファイルベースのデータソース

データソース	データ定義ウィザードの使用	データアクセスウィンドウの使用	Analytics テーブルの読み取り先
Adobe Acrobat (.pdf)	はい	いいえ	Analytics データファイル(.fil)
ACCPAC マスターファイル	はい	いいえ	データソース
dBASE 互換ファイル (.dbf)	はい	いいえ	データソース
区切り文字付きテキスト (.csv または .txt)	はい	はい	Analytics データファイル(.fil)
Microsoft Access (.mdb または .accdb)	はい	はい	Analytics データファイル(.fil)
Microsoft Excel (.xls または .xlsx)	はい	はい	Analytics データファイル(.fil)
印刷イメージ(レポート) (.txt)	はい	いいえ	Analytics データファイル(.fil)
SAP プライベート ファイル形式/DART (.dat)	はい	いいえ	Analytics データファイル(.fil)

データソース	データ定義ウィザードの使用	データアクセスウィンドウの使用	Analytics テーブルの読み取り先
XBRL (.xml または .xbrl)	はい	いいえ	Analytics データ ファイル(.fil)
XML (.xml)	はい	いいえ	Analytics データ ファイル(.fil)
SAP (Direct Link 経由、オプションのユーティリティ)	いいえ	いいえ	Analytics データ ファイル(.fil)
HighBond			
プロジェクト アプリ	はい	いいえ	Analytics データ ファイル(.fil)
リザルト アプリ	はい	いいえ	Analytics データ ファイル(.fil)
外部定義			
AS/400 FDF (.fdf)	はい	いいえ	データソース
COBOL (.cob)	はい	いいえ	データソース
PL/1 (.txt)	はい	いいえ	データソース

データベースとクラウド データ サービス

以下のデータベースとクラウド データ サービスにアクセスするには、データアクセスウィンドウと Analytics データコネクタを使用します。インポートされたデータは Analytics データファイル(.fil)に保存されます。詳細については、「データアクセスウィンドウを使用したデータのインポート」ページ 372を参照してください。

メモ

データアクセスウィンドウを使用すると、次のネイティブ データコネクタだけでなく、すべての ODBC 準拠データソースにアクセスできます。詳細については、「[データアクセス](#)」ウィンドウを使用して、ODBC データソースにアクセスする」ページ 242を参照してください。

ビッグデータと NoSQL

- [Amazon Athena](#)
- [Amazon DynamoDB](#)

	<ul style="list-style-type: none"> ◦ Amazon Redshift ◦ Apache Cassandra ◦ Apache Drill ◦ Apache HBase ◦ Apache Hive ◦ Apache Phoenix ◦ Apache Spark ◦ Azure Data Catalog ◦ Azure Data Lake Storage ◦ Azure Table ◦ CockroachDB ◦ Couchbase ◦ Elasticsearch ◦ Google BigQuery ◦ IBM Cloudant ◦ MarkLogic ◦ MongoDB ◦ Parquet ◦ Presto ◦ Snowflake
<p>ERP および CRM システム</p>	<ul style="list-style-type: none"> ◦ Concur ◦ Dynamics 365 Business Central ◦ Dynamics 365 Finance and Operations ◦ Dynamics 365 Sales ◦ Dynamics CRM ◦ Epicor ERP ◦ Exact Online ◦ NetSuite ◦ Odo ◦ Oracle HCM Cloud ◦ Salesforce ◦ SAP ◦ SAP ByDesign ◦ ServiceNow ◦ SugarCRM ◦ SuiteCRM ◦ Workday
<p>財務会計ツール</p>	<ul style="list-style-type: none"> ◦ ADP ◦ Dynamics GP ◦ Dynamics NAV ◦ QuickBooks ◦ QuickBooks Online ◦ QuickBooks POS ◦ Sage 50 UK ◦ Sage Cloud Accounting ◦ Sage Intacct ◦ SAP Concur
<p>マーケティングおよびアナリティクス</p>	<ul style="list-style-type: none"> ◦ Google Analytics ◦ LinkedIn ◦ Marketo ◦ Oracle Eloqua ◦ Oracle Sales Cloud ◦ Splunk ◦ SurveyMonkey ◦ Twitter

コラボレーション ソリューション	<ul style="list-style-type: none"> ◦ Active Directory ◦ Airtable ◦ AWS Data Management ◦ Azure Management ◦ Basecamp ◦ DocuSign ◦ 電子メール ◦ Excel ◦ Excel Online ◦ Exchange ◦ Google Contacts ◦ Google Sheets ◦ Jira ◦ Kintone ◦ Microsoft Teams ◦ SAP SuccessFactors ◦ SharePoint ◦ Slack ◦ Zendesk
ファイルおよび API 統合	<ul style="list-style-type: none"> ◦ Amazon S3 ◦ Box ◦ CSV ◦ DigitalOcean ◦ Dropbox ◦ Google Cloud Storage ◦ Google Drive ◦ JSON ◦ LDAP ◦ Microsoft OneDrive ◦ OData ◦ REST サービス ◦ RSS/ATOM ◦ SFTP
eコマース ソリューション	<ul style="list-style-type: none"> ◦ Edgar Online ◦ Open Exchange Rates ◦ ShipStation ◦ Square ◦ Stripe ◦ UPS ◦ USPS
関係 データベース	<ul style="list-style-type: none"> ◦ Azure Analysis Services ◦ Cloudera Impala ◦ Microsoft Access ◦ Microsoft SQL Server ◦ MySQL ◦ Oracle ◦ SAP Hybris Cloud for Customer ◦ Sybase ◦ Sybase IQ ◦ Teradata ◦ xBase
セキュリティ ツール	<ul style="list-style-type: none"> ◦ Qualys ◦ Tenable.sc

その他のコネクタ	◦ Rsam
----------	------------------------

データ定義ウィザードを使用したデータの定義とインポート

データ定義ウィザードは Analytics のコンポーネントであり、単一のウィザードベースのプロセスで次のタスクを実行するために使用できます。

- データの定義
- データのインポート
- 新しい Analytics テーブルを作成する

さまざまなデータソースにアクセスするための標準的な方法を提供します。データソースから Analytics テーブルを作成する基本的なプロセスは一貫していますが、データソースによっては、ウィザードに異なるページやオプションが表示される場合があります。たとえば、Excel ファイル、PDF ファイル、または XML ファイルをインポートしている場合は、ウィザードはまったく異なる方法で動作します。

データ定義ウィザードは、新しい Analytics プロジェクトを作成するとき、およびプロジェクトに新しい Analytics テーブルを追加するときに([インポート > ファイル]、または [ファイル > 新規 > テーブル]) デフォルトで自動的に表示されます。

メモ

データアクセス ウィンドウを使用してデータをインポートすることもできます。詳細については、「データアクセス ウィンドウを使用したデータのインポート」ページ 372を参照してください。

データソースに接続するとき、またはデータソースからインポートするときには、Analytics は厳密に読み取り専用です。詳細については、「Analytics によるデータアクセスは読み取り専用です」ページ 243を参照してください。

データの定義

インポートするときには、データを定義しなければならない場合があります。つまり、次のようなメタデータを指定します。

- フィールド名
- フィールド長
- フィールドのデータ型
- 数値および日付時刻値の形式

次の図は、データ定義ウィザードを使用してインポートされる Excel ワークシートの DATE フィールドの定義を示します。

データ定義ウィザード - Excel インポート

データのプレビュー
 ウィザードはフィールド プロパティを識別しました。この画面では、ウィザードの推奨設定の変更が行えます。フィールドを選択する場合はその列見出しをクリックしてください。

このフィールドを無視する(I)

フィールド プロパティ

名前(M) 型(Y)

列見出し(T) 値(V)

長さ 入力書式(N)

	CARDNUM	CODES	DATE	CUSTNO	DESCRIPTION	AMOUNT
1	8590125272447003	4131	2006-05-27	925007	Tour Buses	\$108.01
2	8590128346463420	4214	2006-05-28	051593	Delivery Services - Local	\$71.57
3	8590128263176714	4784	2006-05-29	503458	Toll and Bridge Fees	\$5.83
4	8590128006917664	5992	2006-05-30	925007	Florists	\$152.97
5	8590129400665510	4131	2006-05-31	051593	Tour Buses	\$390.33
6	8590121044024386	3640	2006-05-01	202028	Hyatt	\$295.75
7	8590126590309991	5734	2006-05-02	962353	Computer Software Stores	\$270.25
8	8590128379233112	3637	2006-05-03	503458	Ramada Inn	\$223.18
9	8590127539604447	4411	2006-05-05	444413	Cruise Lines	\$1,583.02
10	8590124910032035	5811	2006-05-05	250402	Caterers	\$307.27
11	8590120923083354	3638	2006-05-06	051593	Howard Johnson	\$176.29
12	8590124581105605	5814	2006-05-07	812465	Fast food restaurants	\$12.15
13	8590125757369024	3640	2006-05-08	444413	Hyatt	\$259.05
14	8590120772096763	3640	2006-05-09	250402	Hyatt	\$253.97
15	8590126686866268	4121	2006-05-10	444413	Taxis/Limousines	\$35.31
16	8590125255144947	5462	2006-05-11	359310	Bakeries	\$107.95
17	8590124434068810	3649	2006-05-12	284354	Radisson	\$281.85
18	8590126821848844	5943	2006-05-13	812465	Stationery Stores, Office and School Supply Stores	\$119.32

< 戻る(B) 次へ(N) > キャンセル ヘルプ

自動化の度合いの多様化

可能な限り、データ定義ウィザードはソースデータを自動的に定義する次の方法のいずれかを使用します。

- ソースファイルに含まれるレイアウト情報を読み取る
- ソースファイルを解析し、データのパターンを識別する
- レイアウト情報をデータベースに問い合わせる

データ定義のプロセスは、レイアウト情報を何も含まないファイルではもっと込み入ったものになります。このような場合、データ定義ウィザードは必要な情報を提供するように入力を要求します。

Microsoft Excel データのインポート

Microsoft Excel データを分析のために Analytics にインポートするには、多種多様なツールを使用することができます。

メモ

古い Excel ファイル (*.xls) からインポートするには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、"Microsoft Access データベースエンジンを除外するオプション" ページ 2664 を参照してください。

機能の仕組み

データ定義ウィザードを使用することで、1 つ以上の Excel ファイルの選択、インポートする 1 つまたは複数のワークシートの指定、Excel データの Analytics へのインポートを行うことができます。インポートされたデータを基に、1 つまたは複数の新しい Analytics テーブルおよび関連するデータファイル (.fil) が作成されます。インポートされた各ワークシートが個別の Analytics テーブルとなります。

Analytics データファイルには、元の Excel ファイルから完全に独立した、Excel データのコピーが格納されます。

Microsoft Excel がインストールされていないコンピューター上でも、Excel ファイルからデータをインポートすることができます。

1 つのワークシートのインポートと複数のワークシートのインポート

1 回の操作で複数の Excel ワークシートをインポートするオプションと 1 つの Excel ワークシートをインポートするオプションがあります。どちらのオプションを使用するかによって、インポート処理が多少異なってきます。

- **1 つのワークシート** - インポート処理時にソースの Excel データを手動で定義することもできます。
- **複数のワークシート** - ソースの Excel データが Analytics によって自動的に定義されるので、インポート処理時に手動での定義はできません。

たとえば、複数のワークシートのインポート時には、以下を行うことはできません。

- データ型やフィールド長の指定
- インポート対象からフィールドを選択的に除外する

データを Analytics にインポート後、データ定義を [テーブルレイアウト] ダイアログボックスで必要に応じて調整することができます。

名前付き範囲のインポート

ワークシート全体をインポートする代わりに、ワークシートの定義された部分である、名前付き範囲をインポートできます。名前付き範囲の作成については、Excel のヘルプを参照してください。

複数のワークシートの結合

複数のワークシートを個々の Analytics テーブルにインポート後、それらを 1 つの Analytics テーブルに結合できます。たとえば、12 個の月次テーブルのデータをすべてのデータを含む単一の年次テーブルに結合できます。個別の Analytics テーブルにインポートした後にのみ、ワークシートを組み合わせることができます。

複数の Analytics テーブルを結合する方法については、「テーブルの追加」ページ 928 を参照してください。

ヒント

作業を減らすには、新しい結合されたテーブルでデータ定義に必要な調整を行う前に、複数のテーブルを結合してみてください。

ガイドライン

Excel でデータをインポートする手順については、下のガイドラインを参照してください。

データ型と欠落データ

Excel データをインポートする際に最適の結果を得るには、インポート対象のすべてのワークシートにおいて次のことを確認します。

- すべての列が同じデータ型である
- 空白行や空白列がない

列と文字の最大数

Excel 2007 以降

`.xlsx` または `.xlsm` ファイルからインポートできる Excel の列の最大数、およびフィールド内の最大文字数は特定の数に制限されません。

これらの Excel ファイルの種類からのインポートは、Analytics データファイル (.fil) における 32 KB のレコード長の制限によって制御されます。ソース Excel データ内のいずれかのレコードによって 32 KB より長い Analytics レコードが作成される場合、インポートは失敗します。

メモ

新しいテーブルを Analytics で開くとき、デフォルト ビューには最大 256 個の列が表示されます。テーブルにそれ以上の列がある場合は、必要に応じて、それらの列を手動でビューに追加することができます。

Excel 97 - 2003

.xls ファイル(Excel 97 - 2003) のインポート古い種類の処理を用いており、次を上限としています。

- 255 列
- フィールドにつき 255 文字
- レコードにつき 32 KB
- 65,000 行

サポートされている Excel のバージョン

Excel 3.0 以降の任意バージョンのデータを Excel 2016 にインポートできます。

Excel 3.0 より前のバージョンのデータをインポートしたい場合は、その Excel ファイルを **.csv** など Analytics がインポートできる別のファイル形式として保存します。

Excel 保護ビュー

ワークブックの保護ビューが有効な場合、analytics は Excel ワークブックからインポートできません。まず、ワークブックで編集を有効にし、ワークブックを保存して閉じてから、インポートを実行する必要があります。

Web ベースの Excel ファイルはサポート外

Analytics では、Google スプレッドシートなどの Web アプリケーションから作成された Excel ファイルは直接インポートできません。まず Excel でファイルを開き、別のファイル名で保存してから、Analytics を使用して新しいファイルをインポートする必要があります。

上書きの動作

Excel データをインポートし、Analytics プロジェクト内の既存のテーブルと同じ名前を持つ新しい Analytics テーブルを作成する場合は、その既存テーブルを上書きすることもできます。

詳しく表示

上書きされるテーブルの 2 つの部分

Analytics テーブルの上書きは、テーブルの以下の 2 つの部分が上書きされるため、複雑です。

- テーブルレイアウト。これはナビゲーターに表示される部分です。
- 関連するソース データ ファイル。これは Windows フォルダーに保存されている部分です

(テーブルレイアウトとソース データ ファイルについては、"Analytics テーブルの構造" ページ 118を参照してください)。

テーブルのこれら 2 つの部分は、互いとは関係なく上書きされます。これら 2 つの部分が新しいテーブルの対応する部分と同じ名前を持っている場合には、古い方の 2 つの部分が両方とも上書きされます。このシナリオは最も一般的です。

これに対し、テーブルレイアウトまたはソース データ ファイルが新しいテーブルの対応する部分と異なる名前を持つ場合は、新しいテーブルの対応する部分と同じ名前を持つ部分のみが上書きされます。

この上書き動作は、インポート対象となる Excel ワークシートが 1 つか複数かに関係なく該当します。

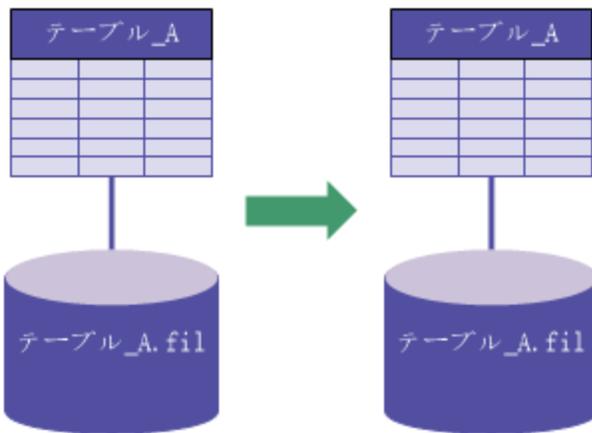
複数のワークシートをインポートする場合の上書き

複数の Excel ワークシートをインポートする場合、上書きの動作は **既存のテーブルを上書きする** 設定と **出力パス** 設定によっても変わります。

以下のセクションでは、複数のワークシートをインポートする場合にあり得る様々な上書き結果をまとめます (最も一般的なシナリオから説明します)。

名前が同じ: 新しいテーブル、既存のテーブルレイアウト、既存のソース データ ファイル

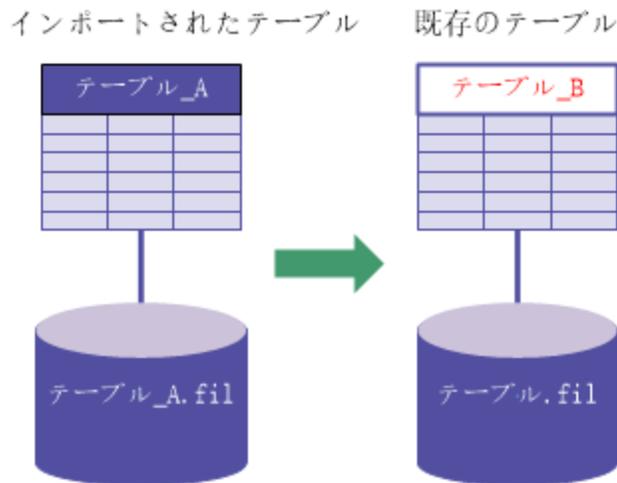
インポートされたテーブル 既存のテーブル



	「出力パス」が既存のソース データ ファイルと同じ	「出力パス」が既存のソース データ ファイルと異なる
既存のテーブルを上書きする] がオン	<ul style="list-style-type: none"> ○ 既存のテーブルレイアウトは上書きされる ○ 既存のソース データ ファイルは上書きされる 	<ul style="list-style-type: none"> ○ 既存のテーブルレイアウトは上書きされ、新しいソース データ ファイルにリンクされる ○ 新しいソース データ ファイルが作成される

	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
		<ul style="list-style-type: none"> 既存のソースデータファイルが保持されるが、リンク解除される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが、各名前の末尾に数字を付けて作成される 例： <ul style="list-style-type: none"> テーブルレイアウト -Table_A2 ソースデータファイル-Table_A2.fil 既存のテーブルレベルとソースデータファイルが保持される 	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが、各名前の末尾に数字を付けて作成される 例： <ul style="list-style-type: none"> テーブルレイアウト -Table_A2 ソースデータファイル-Table_A2.fil 既存のテーブルレベルとソースデータファイルが保持される

名前が異なる: 既存のテーブルレイアウト

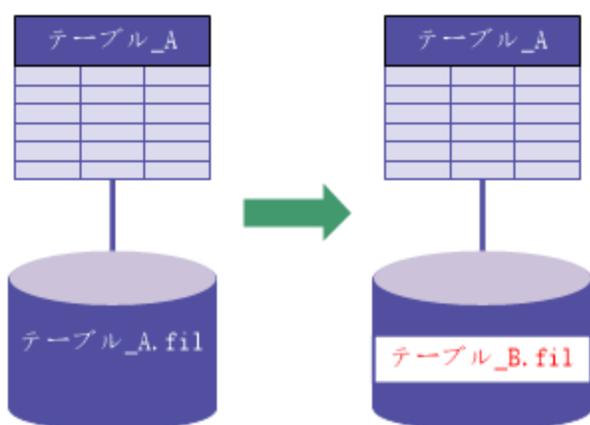


	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
既存のテーブルを上書きする]がオン	<ul style="list-style-type: none"> 新しいテーブルレイアウトが作成される 既存のソースデータファイルは上書きされる 新しいテーブルレイアウトも既存のテーブルレイアウトもソースデータファイルにリンクされる 	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが作成される 既存のテーブルレベルとソースデータファイルが保持される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが、各名前の末尾に数字を付けて作成される 例： 	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが作成される 既存のテーブルレベルとソースデータファイルが保持される

	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
	<ul style="list-style-type: none"> • テーブルレイアウト -Table_A2 • ソースデータファイル-Table_A2.fil <ul style="list-style-type: none"> ◦ 既存のテーブルレベルとソースデータファイルが保持される 	

名前が異なる: 既存のソースデータファイル

インポートされたテーブル 既存のテーブル



	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
既存のテーブルを上書きする]がオン	<ul style="list-style-type: none"> ◦ 既存のテーブルレイアウトは上書きされ、新しいソースデータファイルにリンクされる ◦ 新しいソースデータファイルが作成される ◦ 既存のソースデータファイルが保持されるが、リンク解除される 	<ul style="list-style-type: none"> ◦ 既存のテーブルレイアウトは上書きされ、新しいソースデータファイルにリンクされる ◦ 新しいソースデータファイルが作成される ◦ 既存のソースデータファイルが保持されるが、リンク解除される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> ◦ 新しいテーブルレイアウトとソースデータファイルが、各名前の末尾に数字を付けて作成される <p>例:</p> <ul style="list-style-type: none"> • テーブルレイアウト -Table_A2 • ソースデータファイル-Table_A2.fil <ul style="list-style-type: none"> ◦ 既存のテーブルレベルとソースデータファイルが保持される 	<ul style="list-style-type: none"> ◦ 新しいテーブルレイアウトとソースデータファイルが、各名前の末尾に数字を付けて作成される <p>例:</p> <ul style="list-style-type: none"> • テーブルレイアウト -Table_A2 • ソースデータファイル-Table_A2.fil <ul style="list-style-type: none"> ◦ 既存のテーブルレベルとソースデータファイルが保持される

1 つの Excel ワークシートのインポート

1 つの Excel ワークシートまたは名前付き範囲をインポートして、新しい Analytics テーブルを作成することができます。インポート処理時にソースの Excel データを手動で定義することもできます。

方法の説明

メモ

インポート処理を開始する前に、Excel ファイルが閉じていることを確認します。

Excel ファイルの特定と選択

1. [インポート > ファイル] を選択します。
2. **定義するファイルの選択** ダイアログボックスで、Excel ファイルを特定して選択し、**開く** をクリックします。

Microsoft Excel ファイルは、**.xls** または **.xlsx** ファイル拡張子を持ちます。

3. **ファイル形式** ページで、**Excel ファイル** オプションが選択されていることを確認し、**次へ** をクリックします。

インポートするワークシートの指定

1. **データソース** ページで、インポートするワークシートまたは名前付き範囲を選択します。

メモ

名前付き範囲を表示するには、**システムテーブルのみ** を選択解除します。

ワークシートは、ワークシート名に付加されているドル記号 (\$) で識別されます。ドル記号は一時的に追加されたものであり、Analytics テーブル名には表示されません。

2. ページのデフォルト設定を確認し、必要な更新を行って、**次へ** をクリックします。

設定	説明
先頭の行をフィールド名として使用する	<p>ワークシートまたは名前付き範囲の第 1 行の値は、Analytics テーブルレイアウトのフィールド名として使用されます。</p> <p>メモ この設定を使用する場合、フィールド名として使用される行は、行番号が 開始する行 で指定される行です。</p>
開始する行	<p>ワークシートの読み取りを開始する行番号。</p> <p>この設定により、ワークシートの前半にある、インポートしたくない情報が含まれる行をスキップできます。たとえば、ワークシートの先頭 3 行にヘッダー情報が含まれている場合、4 行目からデータの読み取りを開始するには、「4」と入力します。</p>

設定	説明
	<p>メモ</p> <p>名前付き範囲の開始行は、開始する行の設定値に関係なく、常に名前付き範囲の1行目になります。</p>
<p>すべてのフィールドを文字タイプとしてインポートします</p>	<p>インポートされたすべてのフィールドに文字データ型を割り当てます。</p> <p>ヒント</p> <p>インポートされたすべてのフィールドに文字データ型を割り当てると、Excel ファイルのインポート処理が容易になります。</p> <p>Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。 [すべてのフィールドを文字タイプとしてインポートします] は、実際に文字データ型を使用する必要がある場合に、Analytics によって数値のデータ型が自動的に割り当てられた、識別子のフィールドを持つテーブルをインポートする際に有用となります。</p>
<p>先頭の 100 レコード</p>	<p>Analytics はワークシートまたは名前付き範囲の先頭の 100 レコードだけを使用して、Analytics テーブルのフィールドのデータ型と長さを決定します。</p> <p>大型の Excel ファイルの場合、先頭 100 レコード] を使用すると、インポート処理の速度が大幅に上がります。</p> <p>注意</p> <p>最初の 100 レコードの値がすべての後続の値のデータ型と長さを正確に反映していることが確実である場合にのみ、このオプションを選択してください。</p> <p>先頭 100 レコードの後の値が別のデータ型か、長い場合は、結果として生じる Analytics テーブルには不正確なデータや切り詰められたデータが含まれます。</p> <p>重要なフィールドに不正確なデータや切り詰められたデータがあると、以降のデータ分析の結果が無効になる可能性が非常に高くなります。</p>
<p>Excel ワークシートまたは名前付き範囲全体</p>	<p>Analytics はワークシートまたは名前付き範囲内のすべてのレコードを使用して、Analytics テーブルのフィールドのデータ型と長さを決定します。</p> <p>大きい Excel ファイルの場合、すべてのレコードを使用してデータ型とフィールド長を決定すると、インポート処理の速度は大幅に低下します。</p> <p>メモ</p> <p>Excel データの値のデータ型または長さが一貫しているかどうかを確認できない場合にのみ、このオプションを選択します。</p>

Analytics フィールド プロパティの編集

Analytics は、Excel データ内の各フィールドのプロパティについて、最良の提案を行います。デフォルト設定を受け入れることも、次の手順に従って手動でフィールドを定義することもできます。

1. フィールドのプロパティを確認するには、**Excel Import(Excel からのインポート)]** ページのプレビュー テーブルで列見出しを選択します。
2. フィールドごとに、以下に一覧されたプロパティに Analytics によって割り当てられた設定を確認し、必要に応じて更新を行います。
3. プロパティの確認と編集が完了したら、**次へ]** をクリックします。

プロパティ	説明
このフィールドを無視する	フィールドのデータはインポートされません。
名前	テーブルレイアウトのフィールド名。 Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
列見出し	デフォルトの Analytics ビューにおけるフィールドの列見出し。 列見出しを指定しない場合は、 名前] の値が使用されます。
長さ	<p>テーブルレイアウトにおけるフィールドの長さ。文字で長さを指定します。</p> <p>日付時刻フィールドに時刻データがなく、その日付の後に 00:00:00 と表示されている場合は、フィールドの長さを短くして、空の時刻データを切り捨てることができます。</p> <p>メモ 最大フィールド長は 32,767 文字 (非 Unicode エディションの場合) または 16,383 文字 (Unicode エディションの場合) です。上限までのフィールド長全体が Analytics にインポートされますが、テーブルビューに表示されるのは最初の 256 文字だけです。残りのデータも存在していて分析できますが、ビューには表示されません。データ全体を表示するには、テーブルレイアウト] ダイアログ ボックスを開きます。</p> <p>ヒント 前の画面で 先頭 100 レコード] を選択したが、このフィールドの後続の値の長さがわからない場合は、フィールド長を大きくしてください。</p>
	<p>メモ データソース] ページで すべてのフィールドを文字タイプとしてインポートします] を選択した場合、以下のオプションは無効になるので適用されません。</p>
型	Analytics のフィールドに割り当てられるデータ型。 Analytics によって割り当てられたデータ型を保持するか、ドロップダウン リストから適切なデータ型を選択することができます。
値	フィールドの最初の値を表示する読み取り専用プロパティ。 この値は、行う編集に基づき動的に更新します。
10 進型 (Decimal)	<p>数値フィールドのみ。ソース データの小数点以下桁数。</p> <p>メモ 小数位] テキスト ボックスは、数値のデータ型を選択すると自動的に表示されず。</p>

プロパティ	説明
入力形式	<p>日付時刻フィールドのみ。ソースデータの日付時刻値の書式。</p> <p>実際のデータに対応する書式を選択します。また、必要であれば、データに対応する書式を作成します。指定する書式はソースデータの書式と正確に一致する必要があります。</p> <p>日付と時刻の書式の詳細については、"ソースデータに含まれる日付および時刻データの書式" ページ 366を参照してください。</p> <p>メモ</p> <p>入力書式] テキスト ボックスは、日付時刻データ型を選択したときに自動的に表示されます。</p>

Analytics データ ファイルの保存

[データ ファイルを別名で保存] ダイアログ ボックスで、Analytics データ ファイルの名前を入力して **保存**] をクリックします。

Analytics によってデータ ファイル名 があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

Analytics によって開かれたデフォルト のロケーションを使用しない場合は、異なるフォルダーに移動して、データ ファイルを保存することもできます。

インポートの最終処理

1. **最終**] ページで、新しい Analytics テーブルの設定を確認したら、**完了**] をクリックします。
変更する場合は、**戻る**] をクリックし、ウィザードの適切なページへと進みます。
2. プロジェクトに追加中のテーブル名を入力するか、デフォルトの名前を保持し、そこで、**OK**] をクリックします。
新しい Analytics テーブルは、インポートされたファイルからのデータと一緒に作成されます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

複数の Excel ワークシートのインポート

1 つ以上の Excel ファイルから複数の Excel ワークシートまたは名前付き範囲を 1 回の操作でインポートします。各 ワークシートまたは名前付き範囲は、Analytics プロジェクトにインポートされると、独立した Analytics テーブルになります。

ソースの Excel データは、Analytics により自動的に定義されるため、インポート時に手動で定義することはできません。

データを Analytics にインポート後、データ定義を **テーブルレイアウト** ダイアログボックスで必要に応じて調整することができます。

方法の説明

メモ

インポート処理を開始する前に、すべての Excel ファイルが閉じていることを確認します。

インポートするワークシートと名前付き範囲のすべての最初の行は、一貫性がある方法を使用してください。最初の行は、すべてのデータセットで、フィールド名、データのいずれかです。1つのインポート処理で2つの内容を混在させないでください。

データセットの最初の行へのアプローチに一貫性がない場合、2つの別のインポート処理を使用してください。

Excel ファイルの特定と選択

1. **[インポート > ファイル]** を選択します。
2. **定義するファイルの選択** ダイアログボックスで、Excel ファイルを特定して選択し、**開く** をクリックします。

Microsoft Excel ファイルは、**.xls** または **.xlsx** ファイル拡張子を持ちます。

隣り合う複数のファイルを選択する場合は、**Shift キー**を押しながらクリックします。また、隣り合っていない複数のファイルを選択する場合は、**Ctrl キー**を押しながらクリックします。

3. **ファイル形式** ページで、**Excel ファイル** オプションが選択されていることを確認し、**次へ** をクリックします。

インポートするワークシートの指定

1. **データソース** ページで、インポートするワークシートまたは名前付き範囲を選択します。

メモ

名前付き範囲を表示するには、**システムテーブルのみ** を選択解除します。

個々のワークシートまたは名前付き範囲を選択します。または、Excel ファイル内のすべてのワークシートまたは名前付き範囲を選択する場合には、最初のチェックボックスをオンにします。

ワークシートは、ワークシート名に付加されているドル記号 (\$) で識別されます。ドル記号は一時的に追加されたものであり、結果として生じる Analytics テーブル名には表示されません。

2. Analytics によって割り当てられた設定を確認し、必要な更新を行って、**次へ** をクリックします。

設定	説明
テーブル名	Analytics プロジェクト内のテーブルの名前。 Analytics によって割り当てられた名前を保持します。または、テーブル名をダブルクリックし、

設定	説明
	<p>別の名前を入力して Enter キーを押します。</p> <p>メモ このテーブル名は、データをインポートする際に作成された新しいテーブルレイアウトと新しいソース データ ファイルの両方に適用されます。</p>
先頭の行をフィールド名として使用する	<p>各ワークシートまたは名前付き範囲の第 1 行の値は、生成されるテーブルレイアウトのフィールド名として使用されます。</p> <p>メモ この設定を使用する場合、フィールド名として使用される行は、行番号が 開始する行 で指定される行です。 この設定は、インポートするすべてのワークシートまたは名前付き範囲にグローバルに適用されます。</p>
既存のテーブルを上書きする	<p>Analytics プロジェクト内の同名の既存テーブルが上書きされます。 詳細については、「上書きの動作」 ページ 252を参照してください。</p>
開始する行	<p>ワークシートの読み取りを開始する行番号。 この設定により、ワークシートの前半にある、インポートしたくない情報が含まれる行をスキップできます。たとえば、各ワークシートの先頭 3 行にヘッダー情報が含まれている場合、4 行目からデータの読み取りを開始するには、「4」と入力します。</p> <p>メモ 名前付き範囲の開始行は、 開始する行 の設定値に関係なく、常に名前付き範囲の 1 行目になります。</p>
ファイル名を含める	<p>Excel ファイルを Analytics テーブルの名前の前に追加します。</p> <p>ヒント 別のファイルのワークシートの名前が同じ場合は、Excel ファイル名を前に付けると、テーブル名の競合を回避できます。</p>
すべてのフィールドを文字タイプとしてインポートします	<p>インポートされたすべてのフィールドに文字データ型を割り当てます。</p> <p>ヒント インポートされたすべてのフィールドに文字データ型を割り当てると、Excel ファイルのインポート処理が容易になります。 Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。 [すべてのフィールドを文字タイプとしてインポートします] は、実際に文字データ型を使用する必要がある場合に、Analytics によって数値のデータ型が自動的に割り当てられた、識別子のフィールドを持つテーブルをインポートする際に有用となります。</p>
先頭の 100 レコード	<p>Analytics はワークシートまたは名前付き範囲の先頭の 100 レコードだけを使用して、生成する Analytics テーブルのフィールドのデータ型と長さを決定します。</p>

設定	説明
	<p>大型の Excel ファイルの場合、先頭 100 レコード]を使用すると、インポート処理の速度が大幅に上がります。</p> <p>注意</p> <p>初の 100 レコードの値がすべての後続の値のデータ型と長さを正確に反映していることが確実である場合にのみ、このオプションを使用してください。</p> <p>先頭 100 レコードの後の値が別のデータ型か、長い場合は、結果として生じる Analytics テーブルには不正確なデータや切り詰められたデータが含まれます。</p> <p>重要なフィールドに不正確なデータや切り詰められたデータがあると、以降のデータ分析の結果が無効になる可能性が非常に高くなります。</p>
Excel ワークシート または名前付き範囲全体	<p>Analytics はワークシートまたは名前付き範囲内のすべてのレコードを使用して、生成する Analytics テーブルのフィールドのデータ型と長さを決定します。</p> <p>大きい Excel ファイルの場合、すべてのレコードを使用してデータ型とフィールド長を決定すると、インポート処理の速度は大幅に低下します。</p> <p>メモ</p> <p>ワークシートの列内の値のデータ型または長さが一貫しているかどうかを確認できない場合に、このオプションを使用してください。</p>
出カパス	<p>新しい Analytics データ ファイル(.fil) を保存するフォルダーを指定します。</p> <p>出カパス]を空白のままにした場合には、Analytics データ ファイルは Analytics プロジェクトが格納されているフォルダーに保存されます。</p>

インポートの最終処理

最終] ページで、新しい Analytics テーブルの設定を確認したら、**完了**] をクリックします。

変更する場合は、**戻る**] をクリックし、ウィザードの適切なページへと進みます。

新しい Analytics テーブルは、インポートされたワークシートまたは名前付き範囲内のデータを使って作成されます。

メモ

最終] ページの**テーブル名**の末尾に数値が付いている場合は、同名のテーブルが既に Analytics プロジェクトに存在しているケースで、**既存のテーブルを上書きする**] をオフにしていたことを意味します。

詳細については、「上書きの動作」 ページ 252を参照してください。

Microsoft Access データベース ファイルのインポート

Microsoft Access データベース ファイルを定義およびインポートして、Analytics テーブルを作成できます。Access ファイルは Access 2000 から Access 2010 までの任意のバージョンにできます。前のバージョンの Access からファイルをインポートするには、Analytics が定義およびインポートできる別のファイル形式でファイルを保存できます。

Microsoft Access がインストールされていないコンピューター上でも、Access ファイルをインポートすることができます。

メモ

Microsoft Access データベースファイルからインポートするには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、「Microsoft Access データベースエンジンを除外するオプション」ページ 2664を参照してください。

1. **ファイル > 新規作成 > テーブル**の順にクリックします。
2. **データソースのプラットフォームの選択**ページが表示されている場合は、**ローカル**を選択し、**次へ**をクリックします。
3. **ファイル**を選択して **次へ**をクリックします。
4. **定義するファイルの選択**で、Analytics テーブルの作成元とするファイルを探して選択し、**開く**をクリックします。Microsoft Access データベース ファイルは、.mdb または .accdb ファイル拡張子を持ちます。
5. **ファイル形式**ページで、**Access データベース**オプションが選択されていることを確認し、**次へ**をクリックします。
6. **データソース**ページで、次の手順を実行します。
 - a. インポートするテーブルまたはビューを選択します。利用可能なオプションは、**Access テーブル/ビューの選択**リストに列挙されています。
 - b. テキスト フィールドからインポートされる文字の最大数を増減したい場合は、**文字フィールドの最大長**テキスト ボックスに新しい値を入力します。
1 ~ 255 文字を入力できます。
 - c. メモフィールドや長いテキスト フィールドからインポートされる文字の最大数を増減したい場合は、**メモフィールドの最大長**テキスト ボックスに新しい値を入力します。
1 ~ 32767 文字(非 Unicode 版の Analytics)、または 16383 文字(Unicode 版の Analytics)を指定できます。
 - d. **次へ**をクリックします。

7. **データ ファイルを別名で保存**]ダイアログ ボックスで、必要に応じて、Analytics データ ファイルのファイル名と場所を変更し、**保存**]をクリックします。
8. **最終**]ページで、新しい Analytics テーブルの設定を確認したら、**完了**]をクリックします。
9. プロジェクトに追加する Analytics テーブルの名前を入力し、**OK**]をクリックします。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

区切り文字付きテキスト ファイルのインポート

区切り文字付きテキスト ファイルを分析のために Analytics にインポートするには、多種多様なツールを使用することができます。

機能の仕組み

データ定義ウィザードを使用することで、1 つまたは複数の区切り文字付きテキスト ファイルを選択してそのデータを Analytics にインポートできます。インポートされたデータを基に、1 つまたは複数の新しい Analytics テーブルおよび関連するデータ ファイル (.fil) が作成されます。インポートした個々の区切り文字付きテキスト ファイルは、個別の Analytics テーブルとなります。

Analytics データ ファイルには、元の区切り文字付きテキスト ファイルから完全に独立した、区切り文字付きデータのコピーが格納されます。

お使いのローカルコンピューターまたはネットワークドライブにある区切り文字付きテキスト ファイルをインポートできます。Analytics Exchange のユーザーは、Analytics サーバーにある区切り文字付きテキストにもアクセスできます。

1 つのファイルのインポートと複数のファイルのインポート

1 つの区切り文字付きテキスト ファイルをインポートすることも、複数の区切り文字付きファイルを 1 回の操作でインポートすることもできます。どちらのオプションを使用するかによって、インポート処理が多少異なってきます。

- **1 つのファイル**- インポート処理時に、ファイルレベルのプロパティとフィールド レベルのプロパティをどちらも手動で定義することができます。
- **複数のファイル**- インポート処理時にファイルレベルのプロパティのみを手動で定義することができます。フィールド レベルのプロパティは、Analytics により自動的に定義されるため、インポート時に手動で定義することはできません。

たとえば、複数のファイルのインポート時に以下を行うことはできません。

- フィールドのデータ型の指定
- インポート対象からフィールドを選択的に除外する

データを Analytics にインポート後、データ定義を [テーブルレイアウト] ダイアログボックスで必要に応じて調整することができます。

複数のファイルの結合

複数の区切り文字付きテキスト ファイルを個々の Analytics テーブルにインポート後に、それらを1つの Analytics テーブルとして結合できます。たとえば、12 箇の月次テーブルのデータをすべてのデータを含む単一の年次テーブルに結合できます。ファイルを結合できるのは、それらのファイルを個別の Analytics テーブルにインポートした後に限ります。

複数の Analytics テーブルを結合する方法については、「テーブルの追加」 ページ 928 を参照してください。

ヒント

作業を減らすには、新しい結合されたテーブルでデータ定義に必要な調整を行う前に、複数のテーブルを結合してみてください。

区切り文字付きのテキスト ファイルの構造化

区切り文字付きテキスト ファイルのファイル拡張子は通常、.txt または .csv ですが、他のファイル拡張子も使用できます。区切り文字付きテキスト ファイルは、主にスプレッドシート やデータベース アプリケーションからデータを Analytics にインポートするために使用します。データベースまたはスプレッドシートの各行はそれぞれ区切り文字付きテキスト ファイルの1行となり、各行または各レコードが行区切り文字で区切られます。有効な行区切り文字：

- CR - キャリッジ リターン
- LF - 改行
- CRLF - キャリッジ リターン改行 (標準 DOS/Windows 文字シーケンス)

フィールド区切り文字

区切り文字付きテキスト ファイルの各レコードのフィールドはフィールド区切り文字で区切られます。主に3種類の区切り文字付きファイルがあります。これらはそれぞれ使用する次のようなフィールド区切り文字に基づいています。

- **カンマ区切り (.csv)** - 各レコード内のフィールドを区切るためにカンマが使用されます。
- **タブ区切り** - 各レコード内のフィールドを区切るためにタブが使用されます。
- **テキスト ファイル (.txt)** - カンマ、タブ、または別のフィールド区切り文字が、各レコードのフィールドを区切るために使用されます。その他、一般的なフィールド区切り文字にはスペース、パイプ (|) およびセミコロン (;) があります。

テキスト修飾子文字

フィールド区切り文字が使用されている場合、テキスト修飾子文字を使用して文字フィールド値を囲み、その値をフィールド区切り文字と区別します。一般的なテキスト修飾子は二重引用符 (") または一重引用符 (') です。

たとえば、カンマがフィールド区切り文字の場合に、値 \$1,000 をテキスト修飾子で囲んで "\$1,000" とすると、元の値は、2つの値(\$1と000)ではなく、単一の値として解釈されます。

区切られたテキスト ファイル

次の例では、区切り文字付きテキスト ファイルの最初の4件の行を表示します。

- 最初の行にはフィールド名が格納されています。
- フィールド区切り文字はカンマです。各行には7つのフィールドがありカンマで区切られています。
- テキスト修飾子は二重引用符です。各行の最後のフィールドにはテキスト修飾子があり、ドルの値に含まれるカンマはフィールド区切り文字としては解釈されません。

```
First_Name, Last_Name, CardNum, EmpNo, HireDate, Salary, Bonus_2011
Lila, Remlawi, 8590122497663807, 000008, 12/28/2007, 52750, "$1,405.40"
Vladimir, Alexov, 8590122281964011, 000060, 10/5/2007, 41250, "$4,557.43"
Alex, Williams, 8590124253621744, 000104, 8/12/2010, 40175, "$7,460.02"
```

上書きの動作

区切り文字付きデータをインポートし、Analytics プロジェクト内の既存のテーブルと同じ名前を持つ新しい Analytics テーブルを作成する場合は、その既存のテーブルが上書きされます。

詳しく表示

上書きされるテーブルの2つの部分

Analytics テーブルの上書きは、テーブルの以下の2つの部分が上書きされるため、複雑です。

- テーブルレイアウト。これはナビゲーターに表示される部分です。
- 関連するソース データ ファイル。これは Windows フォルダーに保存されている部分です

(テーブルレイアウトとソース データ ファイルについては、"Analytics テーブルの構造" ページ 118を参照してください)。

テーブルのこれら2つの部分は、互いとは関係なく上書きされます。これら2つの部分が新しいテーブルの対応する部分と同じ名前を持っている場合には、古い方の2つの部分が両方とも上書きされます。このシナリオは最も一般的です。

これに対し、テーブルレイアウトまたはソース データ ファイルが新しいテーブルの対応する部分と異なる名前を持つ場合は、新しいテーブルの対応する部分と同じ名前を持つ部分のみが上書きされます。

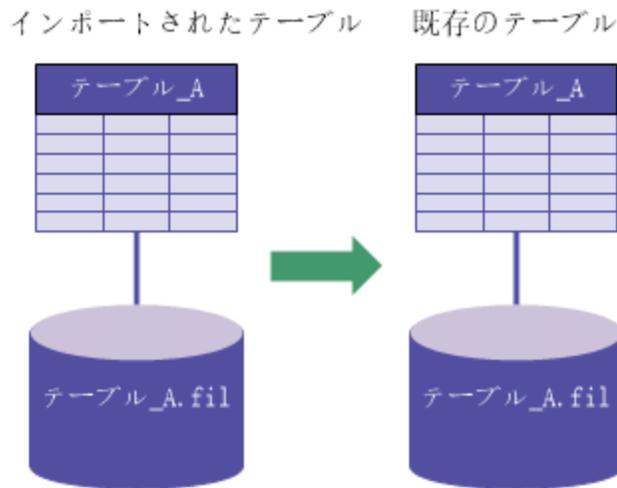
この上書き動作は、インポートする区切り文字付きテキスト ファイルが1つか複数かに関係なく該当しません。

複数の区切り文字付きテキスト ファイルをインポートする場合の上書き

複数の区切り文字付きテキスト ファイルをインポートする場合、上書きの動作は **既存のテーブルを上書きする]設定と 出力パス]設定によって変わります。**

以下のセクションでは、複数の区切り文字付きテキスト ファイルをインポートする場合にあり得るさまざまな上書き結果をまとめます (最も一般的なシナリオから説明します)。

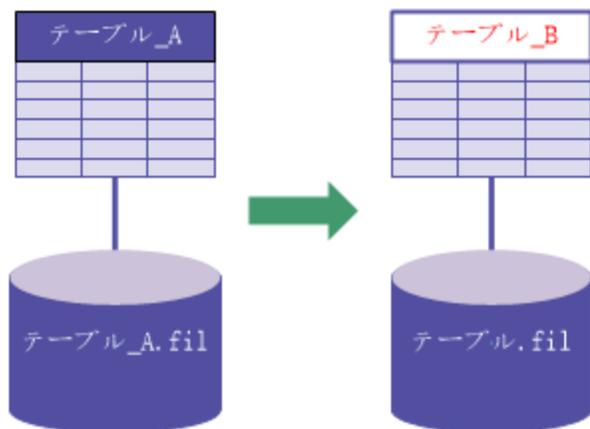
名前が同じ: 新しいテーブル、既存のテーブルレイアウト、既存のソース データ ファイル



	「出力パス」が既存のソース データ ファイルと同じ	「出力パス」が既存のソース データ ファイルと異なる
既存のテーブルを上書きする]がオン	<ul style="list-style-type: none"> 既存のテーブルレイアウトは上書きされる 既存のソース データ ファイルは上書きされる 	<ul style="list-style-type: none"> 既存のテーブルレイアウトは上書きされ、新しいソース データ ファイルにリンクされる 新しいソース データ ファイルが作成される 既存のソース データ ファイルが保持されるが、リンク解除される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示される 	<ul style="list-style-type: none"> "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示される

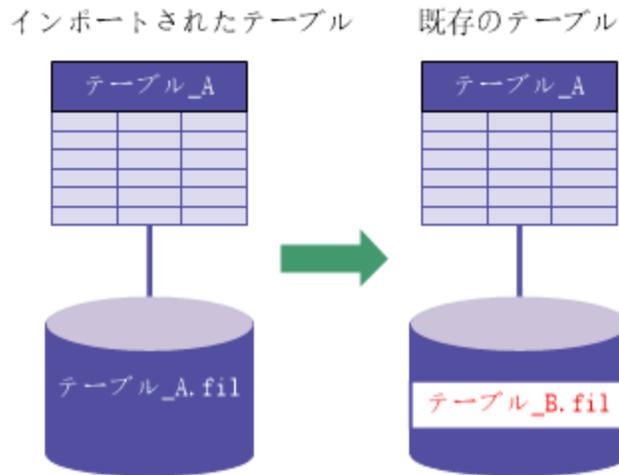
名前が異なる: 既存のテーブルレイアウト

インポートされたテーブル 既存のテーブル



	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
既存のテーブルを上書きする]がオン	<ul style="list-style-type: none"> 新しいテーブルレイアウトが作成される 既存のソースデータファイルは上書きされる 新しいテーブルレイアウトも既存のテーブルレイアウトもソースデータファイルにリンクされる 	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが作成される 既存のテーブルレベルとソースデータファイルが保持される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示される 	<ul style="list-style-type: none"> 新しいテーブルレイアウトとソースデータファイルが作成される 既存のテーブルレベルとソースデータファイルが保持される

名前が異なる: 既存のソースデータファイル



	「出力パス」が既存のソースデータファイルと同じ	「出力パス」が既存のソースデータファイルと異なる
既存のテーブルを上書きする]がオン	<ul style="list-style-type: none"> 既存のテーブルレイアウトは上書きされ、新しいソースデータファイルにリンクされる 新しいソースデータファイルが作成される 既存のソースデータファイルが保持されるが、リンク解除される 	<ul style="list-style-type: none"> 既存のテーブルレイアウトは上書きされ、新しいソースデータファイルにリンクされる 新しいソースデータファイルが作成される 既存のソースデータファイルが保持されるが、リンク解除される
既存のテーブルを上書きする]がオフ	<ul style="list-style-type: none"> "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示される 	<ul style="list-style-type: none"> "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示される

1つの区切り文字付きテキストファイルのインポート

1つの区切り文字付きテキストファイルをインポートして新しいAnalyticsテーブルを作成します。インポート処理時に、ファイルレベルのプロパティとフィールドレベルのプロパティをどちらも手動で定義することができます。

方法の説明

1 つの区切り文字付きファイルの検索と選択

1. [インポート > ファイル] を選択します。
2. **定義するファイルの選択** ダイアログ ボックスで、区切り文字付きテキスト ファイルを探して選択し、**開く** をクリックします。

区切り文字付きテキスト ファイルには、**.txt** や **.csv** など、複数の異なるファイル拡張子を使用できます。

区切り文字付きファイルのプロパティの指定

1. **文字セット** ページで、正しい文字セット オプションが選択されていることを確認し、**次へ** をクリックします。
2. **ファイル形式** ページで、**区切り文字付きテキスト ファイル** オプションが選択されていることを確認し、**次へ** をクリックします。
3. **区切り文字付きファイルのプロパティ** ページで、Analytics によって、以下に一覧されたプロパティへと割り当てられた設定をレビューし、必要な更新を行い、**次へ** をクリックします。

プロパティ	説明
開始する行	<p>ファイルの読み取りを開始する行番号。</p> <p>この設定により、インポートしない情報を含むファイルが始まる行をスキップできます。たとえば、ファイルの先頭 3 行にヘッダー情報が含まれている場合、4 行目からデータの読み取りを開始するには、「4」と入力します。</p>
フィールド幅	<p>プレビュー テーブルの選択された列見出しに、結果として得られるテーブルレイアウトのフィールド長を指定します。文字数を指定します。</p> <p>Analytics は、フィールドの最長値に一致する長さを自動的に割り当てます。割り当てられた長さを維持するか、別の長さを指定することができます。指定されたフィールド長を超えるフィールド値は、結果 Analytics テーブルで切り捨てられます。</p> <p>ヒント</p> <p>更新されたソース データから結果として得られる Analytics テーブルを定期的に更新する、またはインポート コマンドを再利用する予定である場合は、Analytics によって割り当てられたものより長いフィールド長を入力します。</p> <p>ソース データの更新された値が現在のいずれかの値より長い場合は、それより長いフィールド長によって余分のスペースが与えられます。</p> <p>メモ</p> <p>最大フィールド長は 32,767 文字 (非 Unicode エディションの場合) または 16,383 文字 (Unicode エディションの場合) です。上限までのフィールド長全体が Analytics にインポートされますが、テーブルビューに表示されるのは最初の 256 文字だけです。残りのデータも存在していて分析できますが、ビューには表示されません。データ全体を表示するには、テーブルレイアウト ダイアログ ボックスを開きます。</p>
先頭の行をフィールド名として使用する	<p>ファイルの 1 行目の値は、生成されるテーブルレイアウトのフィールド名として使用されます。</p>

プロパティ	説明
	<p>メモ</p> <p>この設定を使用する場合、フィールド名として使用される行は、行番号が 開始する行]で指定される行です。フィールド名が正しくない場合は、データ定義ウィザード]の後続ページで更新できます。</p>
連続した修飾子は1文字として扱う	<p>重複する修飾文字が無視されます。</p> <p>たとえば、"Diligent Corporation""(2つの二重引用符で終わっています)は、このオプションがオンの場合には、"Diligent Corporation"と同等になります。</p>
フィールドの区切り文字	<p>ファイルのフィールドを区切る文字であり、以下のものがあります。</p> <ul style="list-style-type: none"> カンマ TAB セミコロン その他 - フィールドの区切り文字として使用する文字を指定することができます。
テキスト修飾子	<p>フィールドに含まれる値を特定するテキスト記号であり、以下のものがあります。</p> <ul style="list-style-type: none"> 二重引用符 引用符 なし - テキスト修飾子が使用されないことを示します その他 - テキスト修飾子として使用する文字を指定できます。
キャリッジリターン(CR)およびラインフィード(LF)の削除	<p>キャリッジリターン(CR)やラインフィード(LF)文字の配置が誤ってインポートされたデータをクレンジングします。</p> <p>CR/LFの配置が正しくないと、レコード内の改行が正しくなくなる可能性があります。有効化されると、このオプションはCR/LF文字をスペースで置き換えます。テキスト修飾子のペアの中に発生するCR/LF文字のみが置き換えられます。</p> <p>Windowsファイルの場合、キャリッジリターンの削除]とラインフィードの削除]の両方を選択します。</p> <p>テキスト修飾子]が"なし"である場合は、これら2つのオプションは無効になっています。</p>
すべての文字タイプ(「すべて文字型」の意)	<p>インポートされたすべてのフィールドに文字データ型を割り当てます。</p> <p>ヒント</p> <p>インポートされたすべてのフィールドに文字データ型を割り当てると、区切り文字付きテキストファイルのインポート処理が容易になります。</p> <p>Analyticsにインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。</p> <p>すべての文字]オプションは、実際に文字データ型を使用する必要がある場合に、Analyticsによって数値のデータ型が自動的に割り当てられた、識別子のフィールドを持つテーブルをインポートする際に有用となります。</p>
NULLの置換	<p>誤って配置されてNUL文字がインポートされたデータをクレンジングします。</p> <p>NUL文字の配置が正しくないと、レコード内でギャップが生じ、フィールドの分割が正しくなくなる可能性があります。有効化されると、このオプションはNUL文字をスペースで置き換えます。</p>

Analytics データ ファイルの保存

データ ファイルを別名で保存] ダイアログ ボックスで、Analytics データ ファイルの名前を入力して **保存**] をクリックします。

Analytics によってデータ ファイル名 があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

Analytics によって開かれたデフォルト のロケーションを使用しない場合は、異なるフォルダーに移動して、データ ファイルを保存することもできます。

Analytics フィールド プロパティの編集

フィールド プロパティの編集] ページで、Analytics によって、以下に一覧されたプロパティへと割り当てられた設定をレビューし、必要な更新を行い、**次へ**] をクリックします。

メモ

列に関連付けられたプロパティを確認するには、プレビュー テーブルで列見出しを選択します。

プロパティ	説明
このフィールドを無視する	結果として得られるテーブルレイアウトからフィールドを除外します。 このフィールドのデータは、インポートは行われますが、定義されないため、新しい Analytics テーブルには表示されません。必要に応じ、あとで定義して、テーブルに追加することができます。
名前	テーブルレイアウトのフィールド名。 Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
列見出し	デフォルトの Analytics ビューにおけるフィールドの列見出し。 列見出しを指定しない場合は、 名前]の値が使用されます。
<h3>メモ</h3> <p>区切り文字付きファイルのプロパティ] ページで すべての文字] を選択した場合、以下のオプションは適用されず、無効になります。</p>	
型	作成される Analytics テーブルのフィールドに割り当てられるデータ型。 Analytics によって割り当てられたデータ型を保持するか、ドロップダウン リストから適切なデータ型を選択することができます。 Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。
値	フィールドの最初の値を表示する読み取り専用プロパティ。 この値は、行う編集に基づき動的に更新します。
10 進型 (Decimal)	数値フィールドのみ。ソースデータの小数点以下桁数。

プロパティ	説明
	<p>メモ</p> <p>小数位 テキスト ボックスは、数値のデータ型を選択すると自動的に表示されます。</p>
入力形式	<p>日付時刻フィールドのみ。ソースデータの日付時刻値の書式。</p> <p>指定する書式はソース データの書式と正確に一致する必要があります。</p> <p>日付と時刻の書式の詳細については、"ソース データに含まれる日付および時刻データの書式" ページ 366を参照してください。</p>

インポートの最終処理

1. **最終** ページで、新しい Analytics テーブルの設定を確認したら、**完了** をクリックします。
変更する場合は、**戻る** をクリックし、ウィザードの適切なページへと進みます。
2. プロジェクトに追加中のテーブル名を入力するか、デフォルトの名前を保持し、そこで、**OK** をクリックします。

新しい Analytics テーブルは、インポートされたファイルからのデータと一緒に作成されます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

複数の区切り文字付きテキスト ファイルのインポート

1 回の操作で複数の区切り文字付きテキスト ファイルをインポートします。各区切り文字付きファイルは、Analytics プロジェクトにインポートされると、独立した Analytics テーブルになります。

インポート処理時にファイルレベルのプロパティのみを手動で定義することができます。フィールドレベルのプロパティは、Analytics により自動的に定義されるため、インポート時に手動で定義することはできません。

データを Analytics にインポート後、データ定義を **テーブルレイアウト** ダイアログ ボックスで必要に応じて調整することができます。

方法の説明

メモ

インポートするファイルのすべての最初の行の内容は、一貫している必要があります。最初の行は、すべてのファイルにおいて、フィールド名またはデータのいずれかにする必要があります。1つのインポート処理で2つの内容を混在させないでください。

ファイルの最初の行の内容が一貫していない場合は、2つの異なるインポート処理を行ってください。

複数の区切り文字付きファイルの検索と選択

1. [インポート > ファイル] を選択します。
2. **定義するファイルの選択** ダイアログボックスで、区切り文字付きテキスト ファイルを探して選択し、**開く** をクリックします。

以下のファイル拡張子を持つ区切り文字付きテキスト ファイルがサポートされています。**.txt**、**.csv**、**.del**、**.dat**

隣り合う複数のファイルを選択する場合は **Shift** キーを押しながらクリックし、隣り合っていない複数のファイルを選択する場合は **Ctrl** キーを押しながらクリックします。

最初のインポート準備を行う

1. **区切り文字付きファイルのプロパティ** ページで、インポートするファイルを選択します。
デフォルトで選択されているファイルをそのまま使用することも、インポートしたくない任意のファイルを選択解除することもできます。すべてのファイルを選択解除するか選択する場合は、最初のチェックボックスをオンにします。
2. Analytics によって割り当てられた設定を確認し、必要な更新を行って、**次へ** をクリックします。

設定	説明
テーブル名	<p>Analytics プロジェクト内のテーブルの名前。</p> <p>Analytics によって割り当てられた名前を保持します。または、テーブル名をダブルクリックし、別の名前を入力して Enter キーを押します。</p> <p>メモ このテーブル名は、データをインポートする際に作成された新しいテーブルレイアウトと新しいソース データ ファイルの両方に適用されます。</p>
既存のテーブルを上書きする	<p>Analytics プロジェクト内の同名の既存テーブルが上書きされます。</p> <p>詳細については、「上書きの動作」 ページ 267を参照してください。</p>
出カパス	<p>新しい Analytics データ ファイル(.fil) を保存するフォルダーを指定します。</p> <p>出カパス を空白のままにした場合には、Analytics データ ファイルは Analytics プロジェクトが格納されているフォルダーに保存されます。</p>

3. "既存のファイルまたはテーブル名が検出されました" というエラーメッセージが表示されたら、**OK** をクリックし、以下のいずれかまたは両方を行います。

- 既存のテーブルレイアウトまたは同名の関連するデータファイルを上書きしても構わない場合は、**既存のテーブルを上書きする**]をオンにします。
 - 既存の任意のテーブルレイアウトまたは関連するデータファイルが上書きされないように、インポートするテーブルの名前を **テーブル名**]の設定で必要に応じて変更します。
4. **確認**]ダイアログボックスにおいて、続行する場合は **[はい]**、戻ってファイルの選択を変更する場合は **[いいえ]**をクリックします。

区切り文字付きファイルのプロパティの指定

メモ

指定したプロパティは、インポートするすべてのファイルに適用されます。ファイルの構造が一貫していない場合は、一部のファイルのプロパティが不正になるため、インポートで問題が発生します。

1. **区切り文字付きファイルのプロパティ**] ページで、Analytics によって、以下に一覧されたプロパティへと割り当てられた設定をレビューし、必要な更新を行い、**次へ**]をクリックします。

プロパティ	説明
開始する行	<p>ファイルの読み取りを開始する行番号。</p> <p>この設定により、ファイルの前半にある、インポートしたくない情報が含まれる行をスキップできます。たとえば、各ファイルの先頭 3 行にヘッダー情報が含まれている場合、4 行目からデータの読み取りを開始するには、「4」と入力します。</p>
フィールド幅	<p>プレビュー テーブルの選択された列見出しに、結果として得られるテーブルレイアウトのフィールド長を指定します。文字数を指定します。</p> <p>Analyticsは、フィールドの最長値に一致する長さを自動的に割り当てます。割り当てられた長さを維持するか、別の長さを指定することができます。指定されたフィールド長を超えるフィールド値は、結果 Analytics テーブルで切り捨てられます。</p> <p>ヒント</p> <p>更新されたソース データから結果として得られる Analytics テーブルを定期的に更新する、またはインポート コマンドを再利用する予定である場合は、Analytics によって割り当てられたものより長いフィールド長を入力します。</p> <p>ソース データの更新された値が現在のいずれかの値より長い場合は、それより長いフィールド長によって余分のスペースが与えられます。</p> <p>メモ</p> <p>最大フィールド長は 32,767 文字 (非 Unicode エディションの場合) または 16,383 文字 (Unicode エディションの場合) です。上限までのフィールド長全体が Analytics にインポートされますが、テーブルビューに表示されるのは最初の 256 文字だけです。残りのデータも存在していて分析できますが、ビューには表示されません。データ全体を表示するには、テーブルレイアウト]ダイアログボックスを開きます。</p>
先頭の行をフィールド	各ファイルの 1 行目の値は、生成されるテーブルレイアウトのフィールド名として使用されます

プロパティ	説明
ド名として使用する	<p>す。</p> <p>メモ この設定を使用する場合、フィールド名として使用される行は、行番号が 開始する行]で指定される行です。 この設定は、インポートするすべてのファイルにグローバルに適用されます。</p>
連続した修飾子は 1 文字として扱う	<p>重複する修飾文字が無視されます。</p> <p>たとえば、"Diligent Corporation""(2つの二重引用符で終わっています)は、このオプションがオンの場合には、"Diligent Corporation"と同等になります。</p>
フィールドの区切り文字	<p>ファイルのフィールドを区切る文字であり、以下のものがあります。</p> <ul style="list-style-type: none"> ○ カンマ ○ TAB ○ セミコロン ○ その他 - フィールドの区切り文字として使用する文字を指定することができます。
テキスト修飾子	<p>フィールドに含まれる値を特定するテキスト記号であり、以下のものがあります。</p> <ul style="list-style-type: none"> ○ 二重引用符 ○ 引用符 ○ なし - テキスト修飾子が使用されないことを示します ○ その他 - テキスト修飾子として使用する文字を指定できます。
キャリッジ リターン(CR) の削除およびライン フィード(LF) の削除	<p>キャリッジ リターン(CR) やライン フィード(LF) 文字の配置が誤ってインポートされたデータをクレンジングします。</p> <p>CR/LF の配置が正しくないと、レコード内の改行が正しくなくなる可能性があります。有効化されると、このオプションは CR/LF 文字をスペースで置き換えます。テキスト修飾子のペアの中に発生する CR/LF 文字のみが置き換えられます。</p> <p>Windows ファイルの場合、 キャリッジ リターンの削除]と ライン フィードの削除]の両方を選択します。</p> <p>テキスト修飾子]が"なし"である場合は、これら2つのオプションは無効になっています。</p>
すべての文字タイプ (「すべて文字型」の意)	<p>インポートされたすべてのフィールドに文字データ型を割り当てます。</p> <p>ヒント インポートされたすべてのフィールドに文字データ型を割り当てると、区切り文字付きテキスト ファイルのインポート処理が容易になります。 Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。 すべての文字]オプションは、実際に文字データ型を使用する必要がある場合に、Analytics によって数値のデータ型が自動的に割り当てられた、識別子のフィールドを持つテーブルをインポートする際に有用となります。</p>
NULL の置換	<p>誤って配置されて NUL 文字がインポートされたデータをクレンジングします。</p> <p>NUL 文字の配置が正しくないと、レコード内でギャップが生じ、フィールドの分割が正しくなく</p>

プロパティ	説明
	なる可能性があります。有効化されると、このオプションは NUL 文字をスペースで置き換えます。

インポートの最終処理

最終] ページで、新しい Analytics テーブルの設定を確認したら、**完了]** をクリックします。

変更する場合は、**戻る]** をクリックし、ウィザードの適切なページへと進みます。

新しい Analytics テーブルは、インポートされたファイルからのデータを使って作成されます。

印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート

印刷イメージファイル(レポートファイルとも呼ばれる)は、印刷されたレポートの電子的なコピーです。Adobe PDFファイルは、標準のPDF形式で保存されたアプリケーションファイルまたはスキャンされたファイルです。印刷イメージファイルまたはPDFファイルを定義およびインポートする方法は、ほとんど同じです。このため、このセクションのほとんどのトピックが両方のファイル形式に該当します。

PDFは印刷イメージファイルよりも定義とインポートの手順が難しくなる場合があります。これは、ソースPDFで正しく整列しているように見えるデータ列でも、AnalyticsがPDFを解析(ファイル定義処理の一部)した後にはずれが生じる可能性があるためです。AnalyticsにはXpdfとVeryPDFの2つのPDFパーサーがあります。両方のパーサーを試し、どちらの方がより良い結果をもたらすかを確認できます。

注意

コントロール合計を使用し、インポートされた印刷イメージファイルまたはPDFファイルから作成されたAnalyticsテーブルに、ソースファイルのすべてのデータが含まれることを検証します。印刷イメージファイルまたはPDFファイルを定義するときには、誤ってレコードを除外する可能性があります。分析を開始する前に、Analyticsに完全なデータセットがあることを必ず確認してください。

印刷イメージファイルまたはPDFファイルを正しく定義するための重要な点

印刷イメージファイルまたはPDFファイルの定義には注意が必要です。科学的というより技術的な部分があり、効果的な手法を計画するためには、ソースファイル内のデータの配置を注意深く分析することが求められます。一般的に、印刷イメージファイルまたはPDFファイルのデータの配置は、他のファイル形式の配置よりも標準化されていません。このため、定義処理が複雑になります。定義を成功させるには、繰り返し処理が必要になることがあります。

以下に、スムーズに処理を進めるために役立つ重要な点や手法を概説します。ファイルを定義する前、定義するとき、または問題が発生した場合は、以下の点をよく確認することをお勧めします。

一般的な点

- "ファイル定義の処理は反復的" 見開きページ
- "繰り返し実践することで上達する" 見開きページ

整列していないデータ

- "整列していないデータへの対処" ページ 281

フィールドとレコード

- "フィールドは水色、レコードは灰色、未定義データは白色" ページ 282
- "詳細、ヘッダー、フッターの3種類のデータを定義できる" ページ 283
- "ソースファイルのフィールド名を選択しない" ページ 286

レコードの取り込み

- "レコードのセットを取り込むための一意の値を指定する" ページ 286
- "一意の値を選択するためのヒント" ページ 287
- "レコードのセットを正確に取り込む" ページ 288
- "複数の条件を使用してレコードのセットを取り込む" ページ 291
- "ファイル全体のレコード定義とフィールド定義を確認する" ページ 292
- "複数行のレコードおよびフィールドを定義できる" ページ 292

その他の考慮事項

- "必要なデータのみを定義してインポートする" ページ 292
- "結果の Analytics テーブルのフィールドの順序を制御する" ページ 292
- "Analytics が行う可能性があるファイルの自動定義" ページ 293
- "コントロール合計を使用して、結果の Analytics テーブルを検証する" ページ 293

一般的な点

ファイル定義の処理は反復的

通常、印刷イメージファイルまたは PDF ファイルの適切な定義には繰り返し作業が伴い、ある程度の試行錯誤が必要な場合もあります。次の個別のタスクの一部またはすべてを実行する必要があります。

- 1 つ以上のフィールドを定義する
- 一意の値に基づいて、詳細レコードのセットを定義する
- 1 つ以上のヘッダーレコードまたはフッターレコードを定義する
- 取り込まれたレコード セットを調整するための条件を修正または作成する
- 各フィールド定義およびレコード定義の正確性を確認する
- 不正確なフィールド定義またはレコード定義を編集する
- 整列していないデータを処理する方法の 1 つとして、ファイルを複数回処理する

繰り返し実践することで上達する

印刷イメージファイルまたは PDF ファイルの定義(特に、整列していないデータがあるファイルの定義)は、最初はかなり難しく感じるかもしれません。実践を積むことで、ソースファイルのデータの構造を見極め、定義するための適切な方法を見つけることがうまくなります。

Analytics に含まれている以下の 2 つのサンプルファイルを使用して実践できます。

- **REPORT3.TXT** は簡単に定義できます。Analytics はファイルの詳細レコードを自動定義しますが、エラーが含まれるため、自動定義を編集する必要があります。
- **Inventory.pdf** の定義はより難しくなります(ページごとにファイルを解析する場合を除く)。これは整列していないデータが含まれているためです。Analytics はデータが整列していないファイルのどの部分も自動定義できません。このため、一から手作業で定義を作成する必要があります。

Analytics は別のサンプルファイル **Report.txt** の詳細フィールドおよびレコードは完璧に自動定義します。**データ定義ウィザード** で **Report.txt** の自動定義を学習しておく役立つこともあります。

整列していないデータ

整列していないデータへの対処

データ定義ウィザードでは、解析された PDF または印刷イメージ ファイル("解析された PDF ファイル内の整列しているデータと整列していないデータ" 下 を参照) でデータ列が整列していないために、使用可能な Analytics テーブルの作成が難しくなったり、作業に時間がかかったりすることがあります。整列していないデータが重要な問題である場合は、次のいずれかの方法を検討してください。

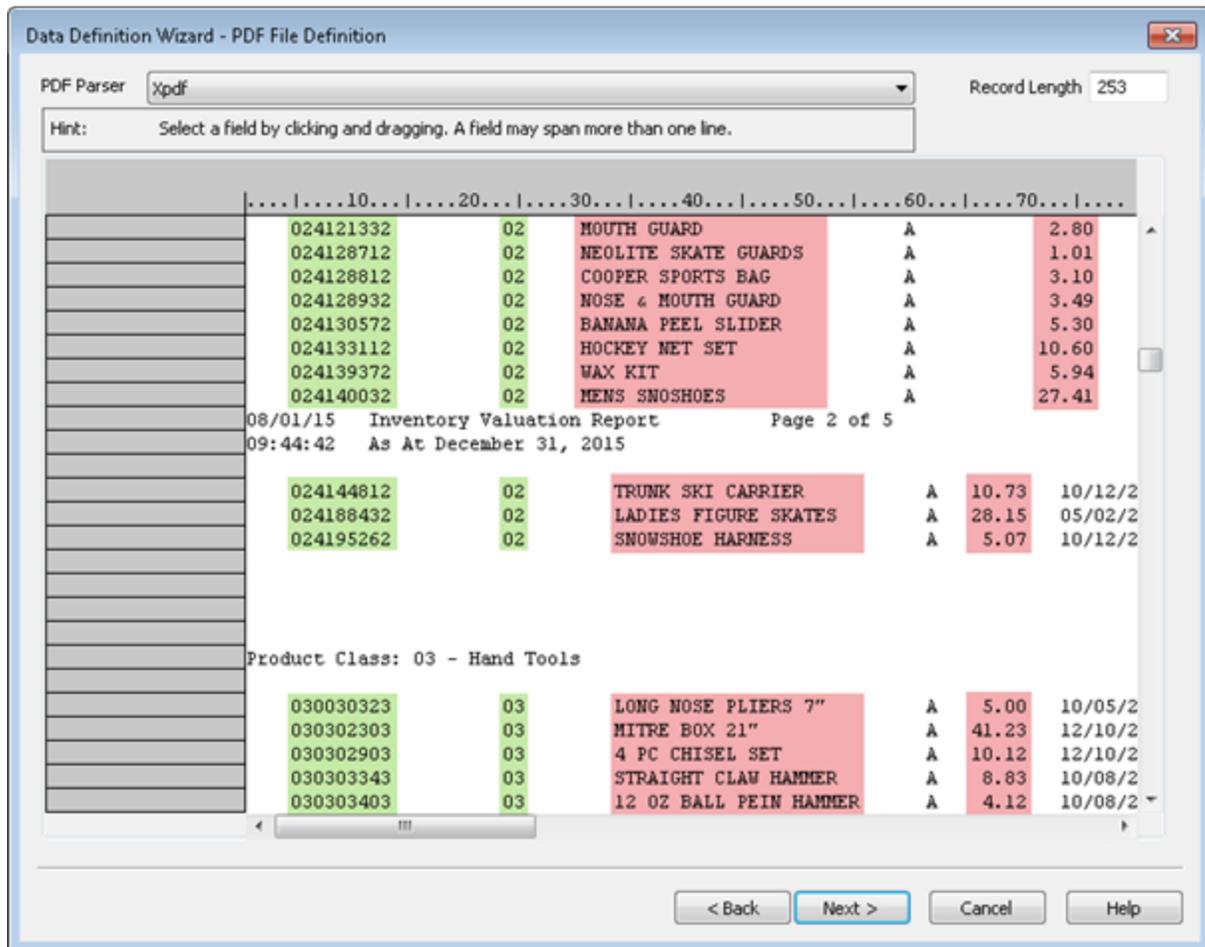
メモ

状況に最も適した方法は、定義しようとしているデータの性質と、Analytics における使用経験によって異なります。Analytics を使い始めたばかりならば、別の形式のデータを要求することを検討してください。

- ファイルのソースの画面に戻って、別の形式のデータを要求します。
 - 変換ソフトウェアを使用してファイルを変換してみます。たとえば、PDF ファイルを Excel ファイルまたはテキスト ファイルに変換するソフトウェアなどです。変換されたファイルを Analytics にインポートします。
 - PDF のデータをコピーして、テキスト エディターに貼り付けます。その後、そのテキスト ファイルを Analytics にインポートします。
 - 次の方法の 1 つ以上を使用して、整列していないフィールドを定義します。
 - 整列していないフィールドの左端と右端の文字を取り込むのに十分な長さがあるフィールド定義を作成します。
 - 重複フィールドの定義を作成します。
 - 複数の整列していないフィールドを含めることができる、単一の長いフィールド定義を作成します。
- 詳細については、"印刷イメージまたは PDF ファイルで整列していないフィールドの定義" ページ 326を参照してください。
- ソースファイルを 2 回以上インポートします。各インポートで、レコードの異なるサブセットを定義します。結果として生じる Analytics テーブルを追加して、完全なデータセットを作成します。
- 詳細については、"印刷イメージまたは PDF データのサブセットの定義とインポート" ページ 330を参照してください。

解析された PDF ファイル内の整列しているデータと整列していないデータ

以下のように、解析された PDF ファイルの左端の 2 つのデータ列は整列しています。残りのデータ列は整列していません。



フィールドとレコード

フィールドは水色、レコードは灰色、未定義データは白色

データ定義ウィザードを使用して、印刷イメージファイルまたは解析されたPDFファイルのフィールドやレコードを定義するにつれて、3つの色によってデータの状態が示されます。

- 水色の強調表示は、データが定義済みフィールドの一部であることを示します。また、すべての定義済みフィールドは定義済みレコードの一部です。
- 灰色の強調表示は、データは定義済みレコードの一部であるが、定義済みフィールドの一部ではないことを示します。
- 白色の背景は、データが完全に未定義であることを示します。

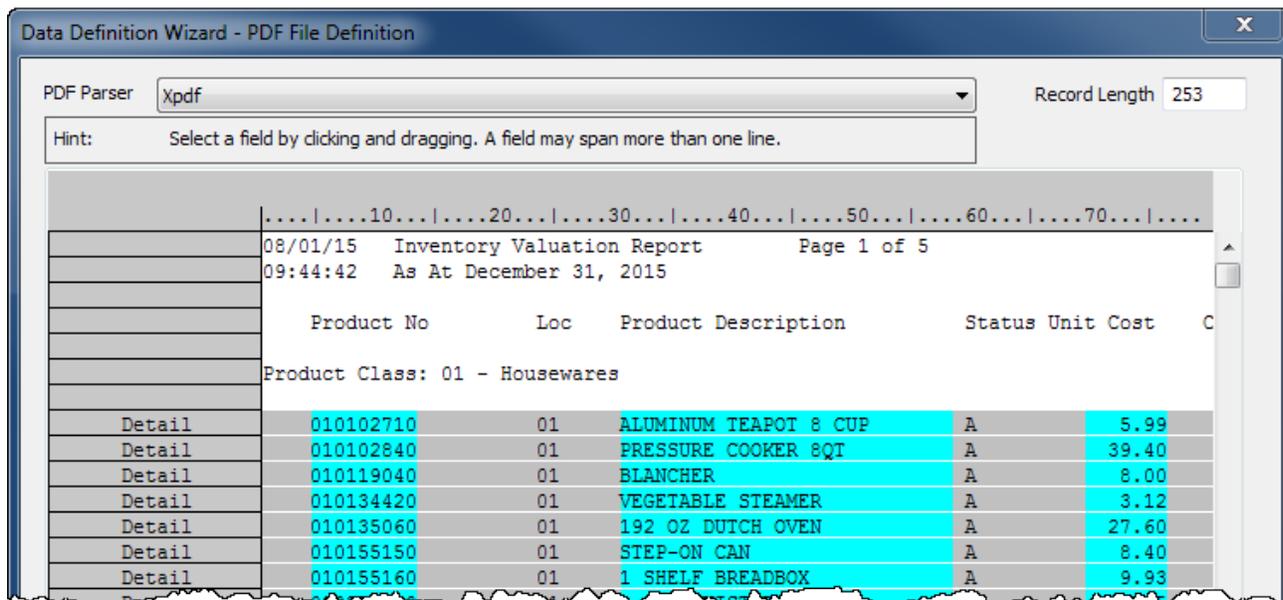
メモ

水色で強調表示されたフィールドのみが、結果として生じる Analytics テーブルの一部になります。

定義済みレコード内の灰色で強調表示されたデータは、フィールドとしても定義されない限り、無視されます。定義済みフィールドの間にあるレコードの灰色の部分は、結果の Analytics テーブルで省略されます。

完全に未定義のデータは無視されます。このようなデータを結果の Analytics テーブルに含めたい場合は、フィールドやレコードを追加定義する必要があります。

定義済みフィールド、定義済みレコード、および未定義データ



詳細、ヘッダー、フッターの3種類のデータを定義できる

データ定義ウィザードでは、印刷イメージファイルまたはPDFファイル内の3種類のデータを定義できます。

データの種類	説明	例	での該当箇所 "PDFファイル内の異なる種類のデータ" 見開きページ
詳細データ	レコードに配置される、ファイルの基本的な内容。 詳細データの定義は必須です。詳細データを定義せずに、印刷イメージファイルまたはPDFファイルを定義することはできません。	<ul style="list-style-type: none"> クレジットカード取引 インベントリレコード 	#2、青の枠線

データの種類	説明	例	での該当箇所 "PDF ファイル内の異なる種類のデータ" 見開きページ
ヘッダーデータ	詳細レコードのブロックまたはサブセットの上に表示される識別情報。 ヘッダーデータの定義は任意です。ヘッダー情報情報が不要な場合は、定義する必要はありません。	<ul style="list-style-type: none"> クレジット カード取引が発生した店番号と場所 “製品クラス” 情報 	#1、赤の枠線
フッターデータ	詳細レコードのブロックまたはサブセットの下に表示される情報。 フッターデータの定義は任意です。フッター情報が不要な場合は、定義する必要はありません。	<ul style="list-style-type: none"> 店別の小計されたクレジット カード取引 “クラス合計” 	#3、アクアブルーの枠線

追加のガイドライン

- 任意の順序で、詳細、ヘッダー、またはフッターデータを定義できます。順序は強制されません。
- フィールド名を指定することもできます("PDF ファイル内の異なる種類のデータ" 下で緑色の輪郭で示されています)。フィールド名を指定する方法は、詳細、ヘッダー、フッターデータを定義する処理によって異なります。

メモ

ヘッダーデータを使用して、印刷イメージまたは PDF ファイルに表示される可能性があるフィールド名を定義しようとししないでください。

PDF ファイル内の異なる種類のデータ

以下の例では、棚卸資産評価レポートの PDF に含まれる異なる種類のデータを強調表示しています。

08/01/15 Inventory Valuation Report Page 1 of 5
09:44:42 As At December 31, 2015

Product No	Loc	Product Description	Status	Unit Cost	Cost Date	Sale Price	QoH	Inventory Val at Cost	Market Value
Product Class: 01 - Housewares									
010102710	01	ALUMINUM TEAPOT 8 CUP	A	5.99	10/02/2015	7.99	144	862.56	1,150.56
010102840	01	PRESSURE COOKER 8QT	A	39.40	19/11/2015	64.98	400	15,760.00	25,992.00
010119040	01	BLANCHER	A	8.00	15/08/2015	13.99	190	1,520.00	2,658.10
010134420	01	VEGETABLE STEAMER	A	3.12	15/08/2015	3.99	50	156.00	199.50
010135060	01	192 OZ DUTCH OVEN	A	27.60	19/11/2015	39.98	230	6,348.00	9,195.40
010155150	01	STEP-ON CAN	A	8.40	15/09/2015	12.99	132	1,108.80	1,714.68
010155160	01	1 SHELF BREADBOX	A	9.93	12/06/2015	13.99	56	556.08	783.44
010155170	01	4 PC CANISTER SET	A	7.05	12/06/2015	10.99	96	676.80	1,055.04
010207220	01	NAPKIN & RELISH HOLDER	A	3.22	12/06/2015	5.79	212	682.64	1,227.48
010226620	01	CAKE DECORATING SET	A	10.80	10/02/2015	15.99	48	518.40	767.52
010310890	01	MINCER	A	14.14	18/04/2015	19.99	86	1,216.04	1,719.14
010311800	01	PASTA NOODLE MAKER	A	24.88	20/12/2015	54.99	64	1,592.32	3,519.36
010311990	01	DIET SCALE	A	2.98	12/06/2015	3.99	290	864.20	1,157.10
010551340	01	DISH DRAINER	D	6.56	10/04/2015	5.99	412	2,702.72	2,467.88
010631140	01	CAKE PAN	A	3.09	15/08/2015	3.59	140	432.60	502.60
010631190	01	LOAF PAN	A	3.10	15/08/2015	3.79	36	111.60	136.44
010803760	01	7 PC KITCHEN TOOL SET	A	-3.21	20/12/2015	6.99	48	-154.08	335.52
Class Totals:							2,634	34,954.68	54,581.76
Product Class: 02 - Sporting Goods									
023903712	02	FISH N FILLET BOARD	A	10.73	01/11/2015	14.95	120	1,287.60	1,794.00
023946372	01	TEHO ROD AND REEL	A	6.43	01/11/2015	7.95	110	707.30	874.50
023973042	02	MORA ICE DRILL EXTN	A	3.32	01/11/2015	4.80	75	249.00	360.00
024100312	02	COOPER HOCKEY HELMET	A	8.42	02/10/2015	10.95	95	799.90	1,040.25
024104312	02	SNOW-JET MOBILE	A	12.10	01/08/2015	13.50	12	145.20	162.00
024106512	02	HOCKEY PANTS	A	14.80	05/02/2015	18.95	125	1,850.00	2,368.75
024108612	02	ESKIMO TOBOGGAN 6FT	A	15.87	01/11/2015	17.95	45	714.15	807.75
024112162	02	CURLING SLIDERS	A	5.18	01/02/2015	6.49	310	1,605.80	2,011.90
024121332	02	MOUTH GUARD	A	2.80	02/10/2015	3.70	345	966.00	1,276.50
024128712	02	NEOLITE SKATE GUARDS	A	1.01	02/10/2015	1.19	450	454.50	535.50
024128812	02	COOPER SPORTS BAG	A	3.10	05/02/2015	2.95	170	527.00	501.50
024128932	02	NOSE & MOUTH GUARD	A	3.49	02/10/2015	3.85	300	1,047.00	1,155.00
024130572	02	BANANA PEEL SLIDER	A	5.30	01/08/2015	7.50	90	477.00	675.00
024133112	02	HOCKEY NET SET	A	10.60	02/10/2015	13.95	200	2,120.00	2,790.00
024139372	02	WAX KIT	A	5.94	10/12/2015	7.95	235	1,395.90	1,868.25
024140032	02	MENS SNOSHoes	A	27.41	10/12/2015	32.95	75	2,055.75	2,471.25

解析された PDF ファイルの詳細データとヘッダー データ

以下の例は、**データ定義ウィザード**で解析されたときの棚卸資産評価レポートです。5つのフィールドを持つ1つの詳細レコードと、2つのフィールドを持つ1つのヘッダーレコードが定義されています。

Detail	Product ID	Product Name	Category	Price
Detail	010631140	01 CAKE PAN	A	3.09
Detail	010631190	01 LOAF PAN	A	3.10
Detail	010803760	01 7 PC KITCHEN TOOL SET	A	-3.21
Header1 Product Class: 02 - Sporting Goods				
Detail	023903712	02 FISH N FILLET BOARD	A	10.73
Detail	023946372	01 TEHO ROD AND REEL	A	6.43
Detail	023973042	02 MORA ICE DRILL EXTN	A	3.32
Detail	024100312	02 COOPER HOCKEY HELMET	A	8.42
Detail	024104312	02 SNOW-JET MOBILE	A	2.10

ヘッダーおよびフッターデータの処理方法

データ定義ウィザードは、ヘッダーデータやフッターデータを、フィールドを持つレコードのように扱いますが、結果として生じる Analytics テーブルの実際のレコードセットになるのは詳細データのみです。定義するヘッダーまたはフッターデータは、詳細レコードに追加される 1 つ以上のフィールドになります。

追加されたヘッダー - およびフッターフィールドは、レコードの個々のブロックまたはサブセット内のすべてのレコードに対して同じ値を繰り返します。たとえば、レコードのあるブロックでは "Store 3"、その次のブロックでは "Store 4" などとなります。

ソースファイルのフィールド名を選択しない

印刷イメージまたは PDF ファイルでフィールド名を選択することにより、フィールド名を定義しないでください。直観に反しているように感じるかもしれませんが、ソースファイルのフィールド名はすべて選択しないでください。代わりに、**[フィールド定義]** ダイアログボックスに名前を入力して、フィールド名を作成します。ソースファイルのフィールド名を選択すると、Analytics はそのフィールド名を、フィールドに含まれるデータとして扱います。

レコードのセットを取り込むための一意の値を指定する

レコードのセットを正確に取り込むには、レコードのセットに対して一意の値を選択または指定することが重要です。つまり、その値は、セット内の全レコードの特定のバイト位置(文字位置)に現れ、ソースファイル内のどこか別の位置には現れません。一意の値は、1 文字でも複数の文字であってもかまいません。

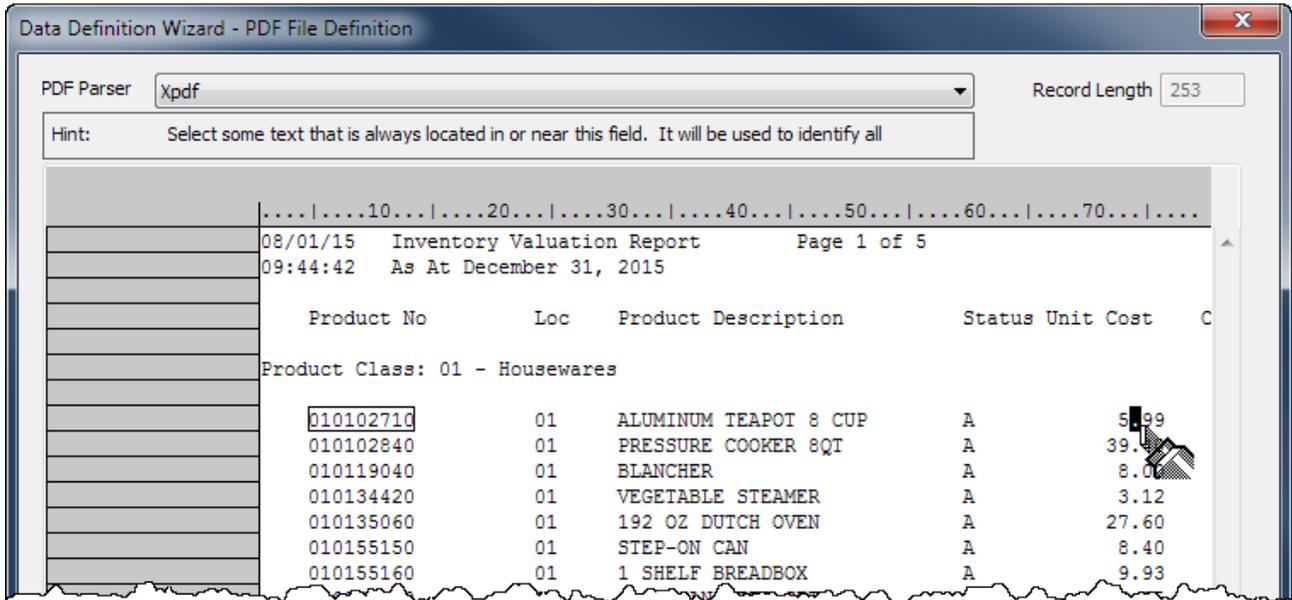
たとえば、「レコードのセットに対して一意の値を選択する」次のページでは、一意の値として "Unit Cost" フィールドの小数点を選択されています。これは、フィールド内のすべての金額で同じ位置に現れ、上下のフィールドのその位置には現れません。

一意の値は、次の2つの場所のいずれかで選択または指定できます。

- 最初のデータフィールドの定義を開始するために選択する、最初のデータ値
- 最初のデータ値と同じ行

レコードのセットに対して一意の値を選択する

以下の例では、一意の値は最初のデータ値と同じ行にあります。最初のデータ値(選択後、ボックスで囲まれています)は、"Product No" フィールドの最初の製品番号です。



一意の値を選択するためのヒント

一意の値を選択するには、レコードセットの上下のデータと比較したときに、一貫して配置されている1つ以上の文字が一意であるか、または一意に配置されているレコードデータを探します。

以下に挙げる値は、一意の値になり得る良い候補です。これらは通常、すべてのレコードで同じ位置に現れ、レコードセット外と同じ位置には現れません。

- 数値の小数点
- 日付の1つまたは両方のスラッシュ
- ID番号の1つ以上のハイフン
- 標準の接頭辞を形成する文字列
- ヘッダーまたはフッターデータで、"Customer ID:" や "Subtotal:" のように一貫して現れるラベル

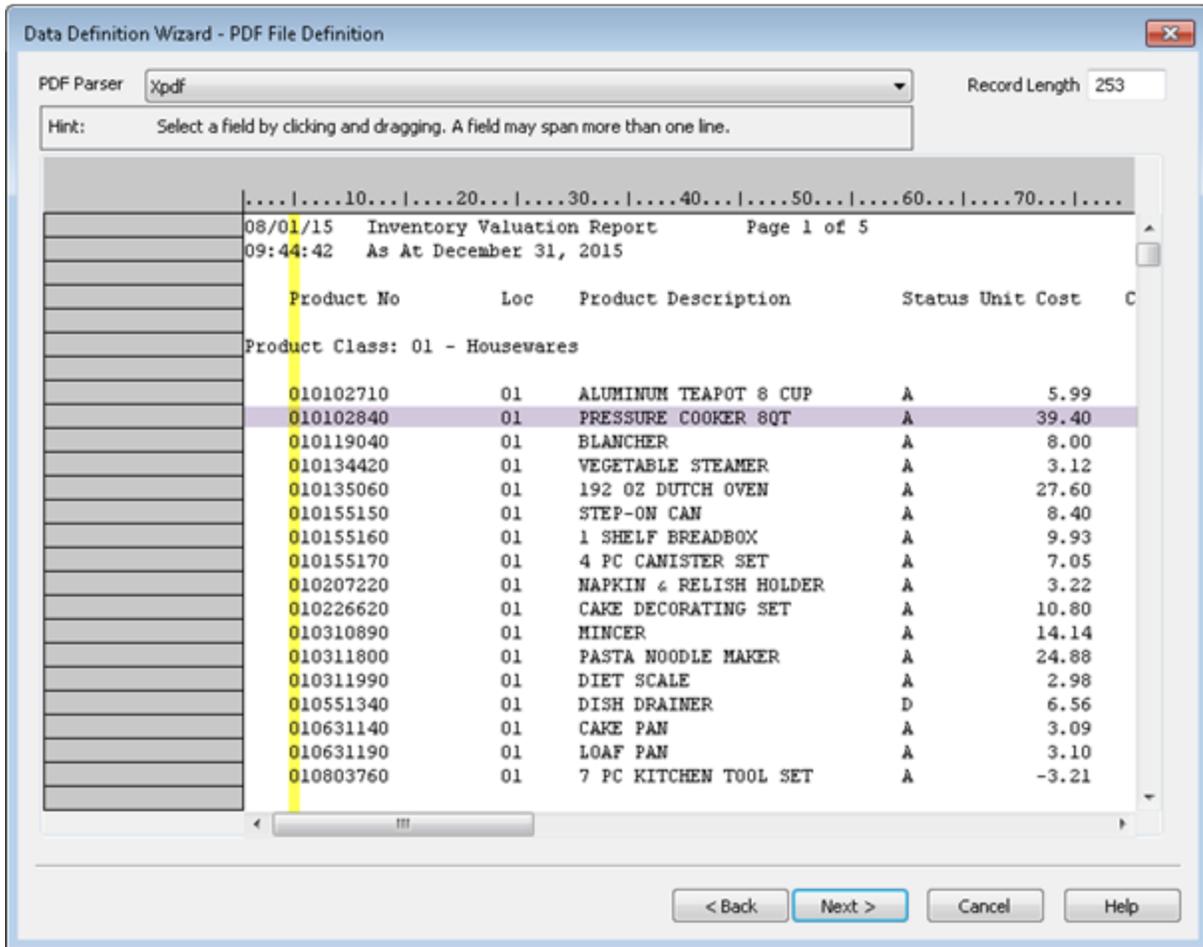
一意の値の最初の選択により、[レコード定義]ダイアログボックスで"正確に一致する"条件を作成します。以下の例では、条件は、すべてのレコードがレコードセットに含まれるようにするには、小数点がバイト位置74に現れる必要があるということを示しています。

必要に応じて、「正確に一致する」を「数字」や「空白以外」などの一般的な一致に変更することができます。これにより、一意の値を指定するときの柔軟性が増します。詳細については、「レコード定義での作業」ページ 322を参照してください。

レコードのセットを正確に取り込む

レコードのセットを正確に取り込むことは難しい場合があります。取り込むレコードのセットに対して一意だと思われる値を選択しても、必要なレコードの一部が取り込まれていなかったり、レコード以外の追加データが取り込まれていたりすることを見つける場合があります。

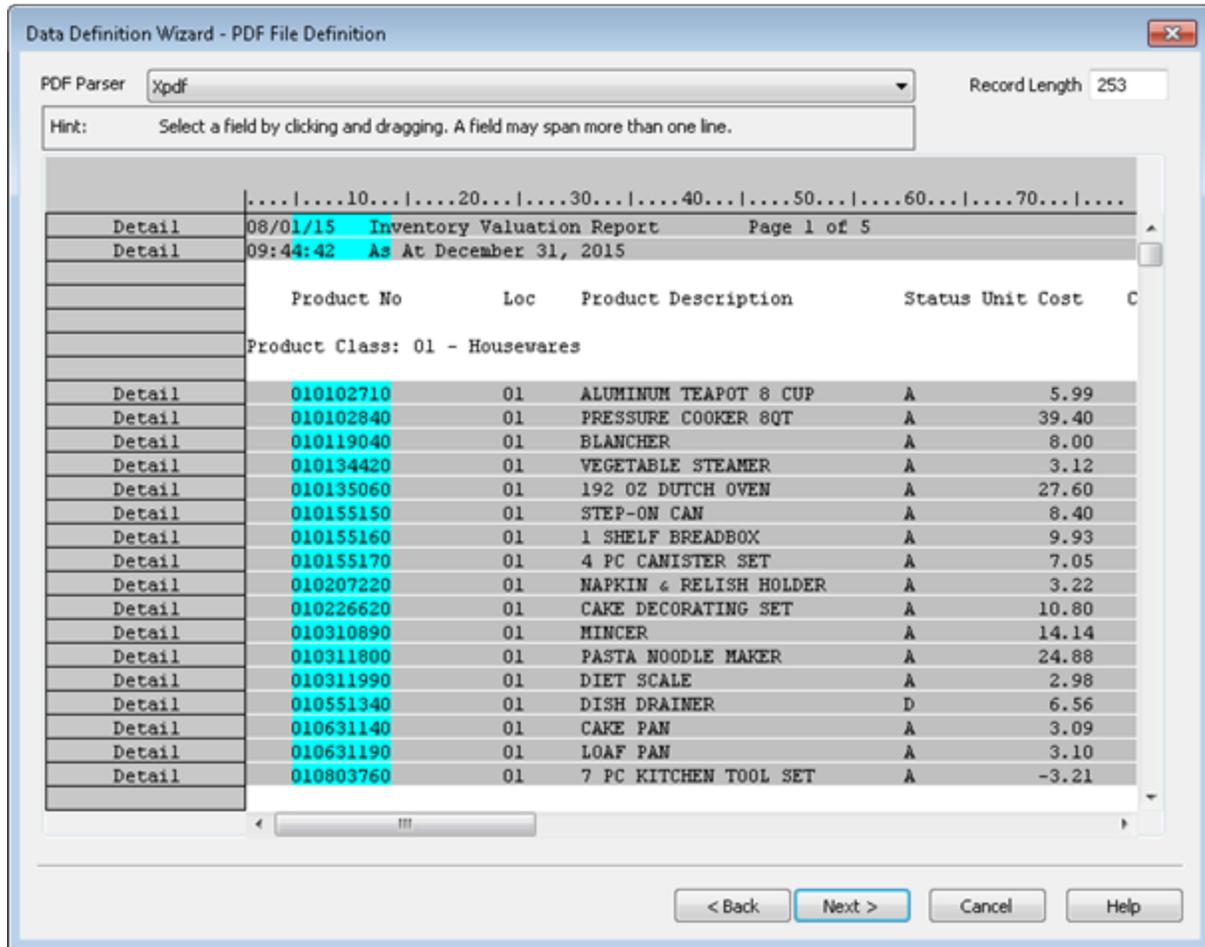
この状況をより適切に理解するには、印刷イメージやPDF ファイルを、行と列で構成されるグリッドとして考えてみると役に立つかもしれません。各列は必ず1つの文字または1つのスペースであり、ファイルの一番上からファイルの一番下まで広がっていると想像してください。



特定の位置にある値を選択または指定して、レコードのセットを取り込む場合、Analytics は、値を検索するときに、ファイルの一番上から一番下まで、その位置にある任意の文字を検討対象とします。文字は、レコード データと見なす行の外側にある場合でも検討されます。指定した値が十分に正確でないと、レコード以外の追加のデータが取り込まれ、レコードのセットに含まれることがあります。

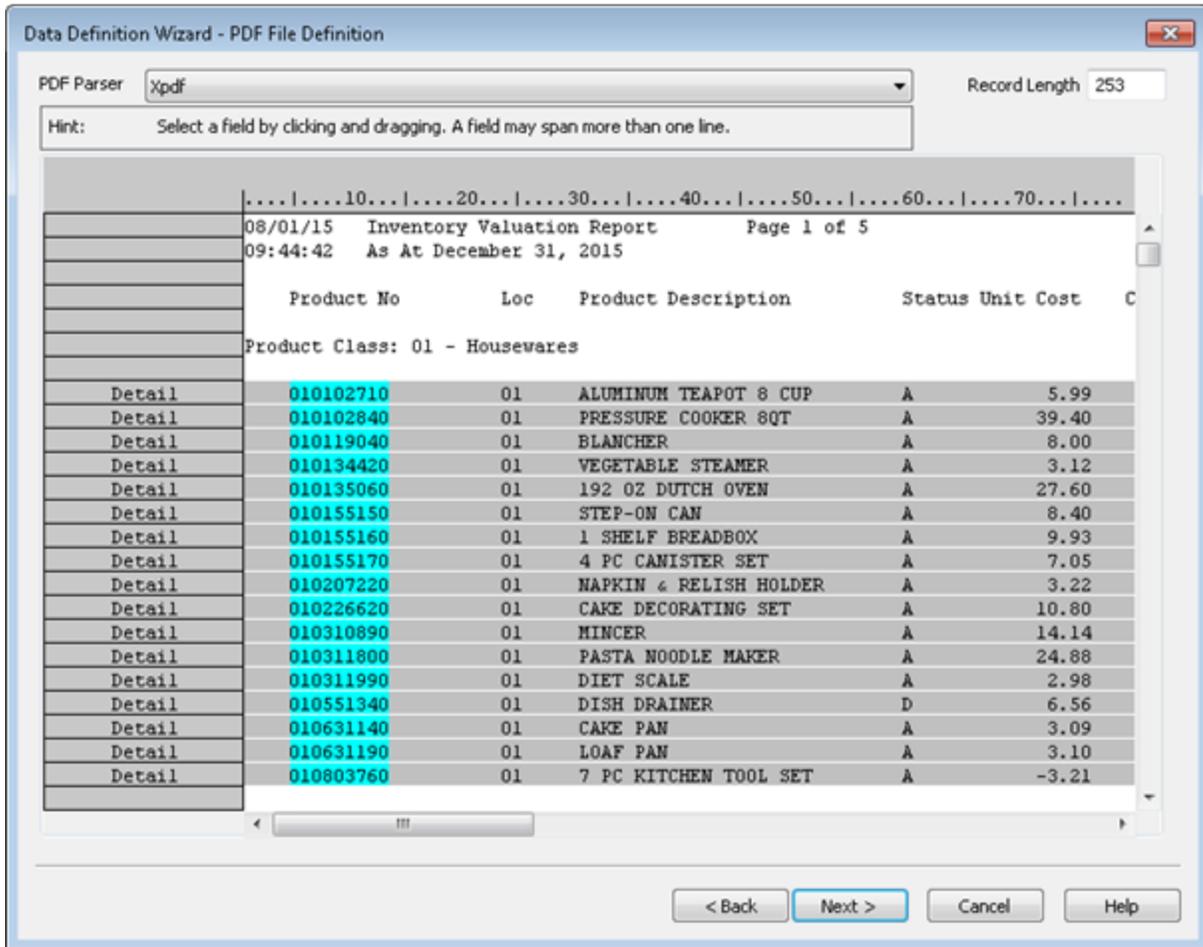
不正確に定義されたデータ フィールド

上記の例では、レコード セットを取り込むための一意の値として、"Product No" フィールドの最初の位置で数字の値を指定した場合、ファイル内のその位置にある数字と、製品番号の実際の最初の数字が取り込まれます。以下の例を参照してください。



正確に定義されたデータフィールド

しかし、フィールドの9桁すべてを含む数値を指定した場合は、意図したレコードのセットのみを取り込むために十分に正確な条件を作成することになります。



複数の条件を使用してレコードのセットを取り込む

日付フィールドのスラッシュなどの1つの条件では、レコードのセットを正確に取り込むのに十分ではない場合があります。もしかしたら、日付の値の一部が欠落しているかもしれません。あるいは、偶然、取り込むレコードセットの外側にある、ヘッダー情報またはフッター情報内の同じ位置にスラッシュが現れる場合もあります。このような場合は、別の条件を追加して、取り込まれたレコードのセットを微調整できます。

メモ

条件を使用して、ソースファイルの行を**包含**または**除外**することができます。

複数の条件の例：

- 日付フィールドの最初のスラッシュと日付フィールドの2番目のスラッシュを持つ行を含める
- 数値フィールドに小数点がある行を含め、"Subtotal" という単語を含む行を除外する
- 最初に指定されたバイト位置のセットに英文字がある行を含めるか、2番目に指定されたバイト位置のセットに英文字がある行を含める

詳細については、「レコード定義での作業」ページ 322を参照してください。

ファイル全体のレコード定義とフィールド定義を確認する

レコードとフィールドを定義するときには、必ず、ファイルをスクロールして、定義の精度を確認してください。空白の値、予期しない文字、および整列していないデータがあると、次のような問題が発生する可能性があります。

- ファイルの一部のレコードが除外される
- レコード以外のデータがレコードとして取り込まれる
- フィールド データがフィールド定義内に不完全な状態で含まれており、値が切り捨てられる
- 2つの異なるフィールドのデータが1つのフィールド定義に表示される

レコード定義が正しくない場合は、レコードのセットを取り込むために使用する条件を変更または作成する必要があります。詳細については、「レコード定義での作業」ページ 322を参照してください。

フィールド定義が正しくない場合は、定義を編集する必要があります。詳細については、「フィールド定義での作業」ページ 317を参照してください。

複数行のレコードおよびフィールドを定義できる

ソースファイル内の複数行にわたるレコードやフィールドのデータを定義することができます。たとえば、各レコードの住所データを複数の行に配置できます。詳細については、「複数行のレコードおよびフィールドの操作」ページ 332を参照してください。

必要なデータのみを定義してインポートする

分析に必要なデータフィールドを定義することで、時間を無駄にしたり、定義およびインポート処理を複雑にしないでください。ヘッダーレコードやフッターレコードは、それらが有益な情報を加える場合にのみ定義してください。含めるデータ要素が追加されるごとに、複雑さが増し、定義処理をより難しくすることがあります。

結果の Analytics テーブルのフィールドの順序を制御する

詳細レコードでフィールドを定義する順序が、結果の Analytics テーブルに表示される順序になります。定義処理中に詳細フィールドを削除して、もう一度追加すると、元の位置が失われ、詳細フィールドの最後に配置されます。内部的に入れ替えようと、詳細フィールドのまともりはそのままです。

ヒント

最初の詳細フィールドを使用して詳細レコードを取り込むが、そのフィールドを結果の Analytics テーブルの先頭に表示したくない場合は、フィールドを使用してレコードを取り込んだ後、フィールドを削除して再度追加します。

ヘッダーフィールドおよびフッターフィールドは、これらを定義した順序で結果の Analytics テーブルに表示されます。ヘッダーフィールドとフッターフィールドは、最初の詳細フィールドを定義していない場合には詳細フィールドの前に表示され、定義している場合には詳細フィールドの後に表示されます。

また、印刷イメージまたはPDFファイルをAnalyticsにインポートした後に、フィールドを並べ替えるオプションもあります。ビュー内で列をドラッグして、順序を変えることができます。さらに、テーブルレイアウトのフィールドが物理的に並べ替えられている状態で新しいテーブルを作成したい場合は、ビュー単位で抽出することもできます。詳細については、「データの抽出」ページ 201を参照してください。Analytics内でフィールドを並べ替える方が、**データ定義ウィザード**で正確なフィールドの順序を維持しようとするよりも簡単な場合があります。

Analyticsが行う可能性があるファイルの自動定義

Analyticsは、データの繰り返しパターンを特定できる場合、印刷イメージまたはPDFファイルを自動定義することがあります。**データ定義ウィザード**のソースファイルの初期表示に水色のフィールド定義と灰色のレコード定義が含まれる場合、Analyticsはファイルの一部またはすべてを自動定義しています。

ファイル全体のフィールド定義とレコード定義を確認し、自動定義が完全に正確であると判断した場合には、ファイル定義の作業はほぼ完了しています。**データ定義ウィザード**の次のページに進むことができます。

多くの場合、Analyticsの自動定義は完全に正確ではありません。そこで、自動定義を編集するか、自動定義全体を削除して最初から手動定義を開始するか、簡単な作業を決定する必要があります。いつでも削除またはやり直しができるため、最初に一部を編集してみて、自分の要件と自動定義に大幅な隔たりがある場合には、その時点で削除することができます。

メモ

詳細レコードのみが自動定義されます。ヘッダーまたはフッターデータは、必要であれば、手動で定義してください。

コントロール合計を使用して、結果のAnalyticsテーブルを検証する

データ分析を開始する前に、コントロール合計を使用し、印刷イメージファイルまたはPDFファイルから作成されたAnalyticsテーブルに、ソースファイルにあるすべてのデータが含まれることを検証します。**不完全なAnalyticsテーブルがあると、分析が無効になります。**

コントロール合計を使用してAnalyticsテーブルを検証するには

1. 次のいずれかを実行します。
 - ソースファイルでレコードがグループ化されている場合は、Analyticsテーブルを分類化または要約し、同じ方法でレコードをグループ化します。
分類化または要約を行うときには、ソースファイルの1つ以上の小計フィールドと一致する **[小計フィールド]** を選択します。
詳細については、「データの分類化」ページ 1319と「データの要約」ページ 1326を参照してください。
 - ソースファイルでレコードがグループ化されていない場合は、ソースファイルでも合計が算出されているAnalyticsテーブルのフィールドを合計します。
詳細については、「フィールドの合計」ページ 908を参照してください。

2. 結果を画面または新しい Analytics テーブルに出力し、Analytics の小計または合計をソースファイルの数値と比較します。

すべての数値が同じ場合は、完全なデータセットがあります。

1 つ以上の数値が同じではない場合、Analytics テーブルのデータはソースファイルのデータと異なっています。データのサブセットをインポートして、Analytics で完全なデータセットを再作成した場合は、Analytics テーブルに重複するレコードが存在する可能性があります。重複レコードの削除については、「重複の削除」ページ 1272を参照してください。

重複するレコードが問題ではない場合は、ソースファイルの定義とインポートをやり直さなければなりません。定義をやり直す場合は、フィールド定義およびレコード定義をよく確認し、データを正確に取り込むようにしてください。

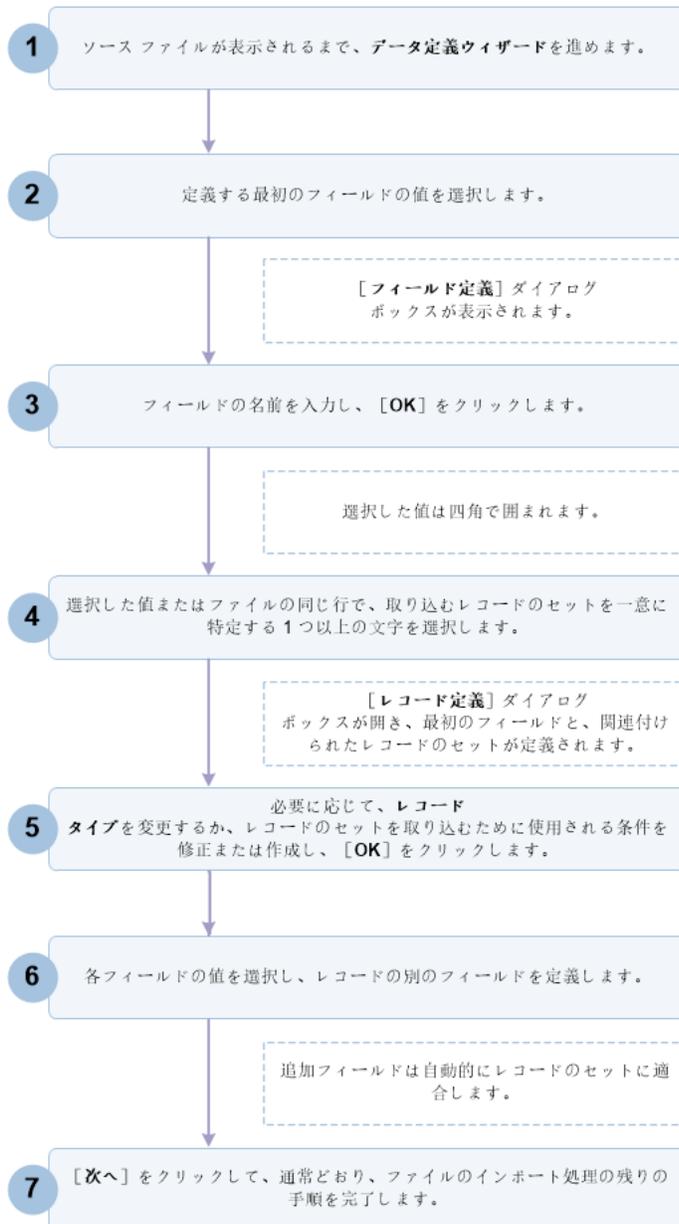
クイックスタート：印刷イメージまたはPDFファイルを定義する方法

印刷イメージまたはPDFファイルを手動で定義し始めるには、初期データ値を選択してから、関連付けられたレコードのセットを取り込みます。

- "印刷イメージまたはPDFファイルを定義するワークフロー" 見開きページ - Analytics にインポートする前に、印刷イメージまたはPDFファイルを定義するための基本ワークフローを示します。
- "クイックスタートの手順" 見開きページ - 画面キャプチャの段階的な手順

これらのタイプのファイルを定義する方法を理解するには、Analytics に含まれるサンプルファイルのいずれかを定義してみることをお勧めします。"REPORT3.TXT" は簡単に定義できます。"Inventory.pdf" の定義は難しくなります。

印刷イメージまたは PDF ファイルを定義する ワークフロー



クイックスタートの手順

以下の手順は、印刷イメージまたは PDF ファイルを定義する基本手順について説明します。詳細な手順については、「印刷イメージファイルの定義とインポート」ページ 300または「PDF ファイルの定義とイン

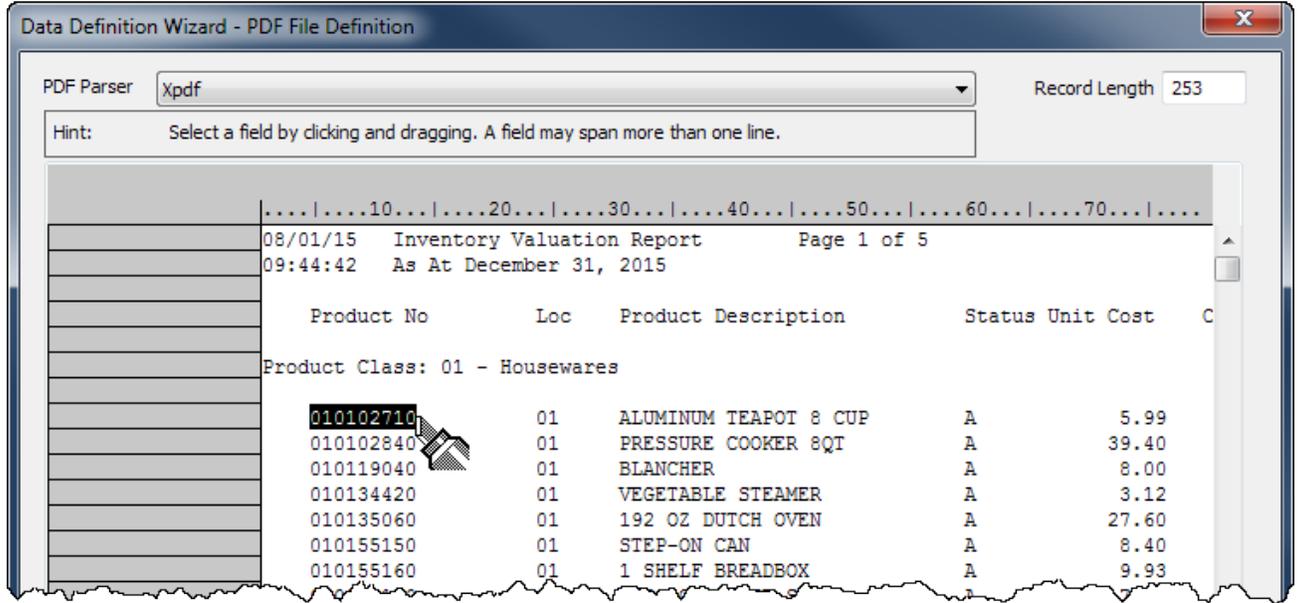
ポート" ページ 308を参照してください。

1. ソースファイルが表示されるまで、**データ定義ウィザード**を進めます。

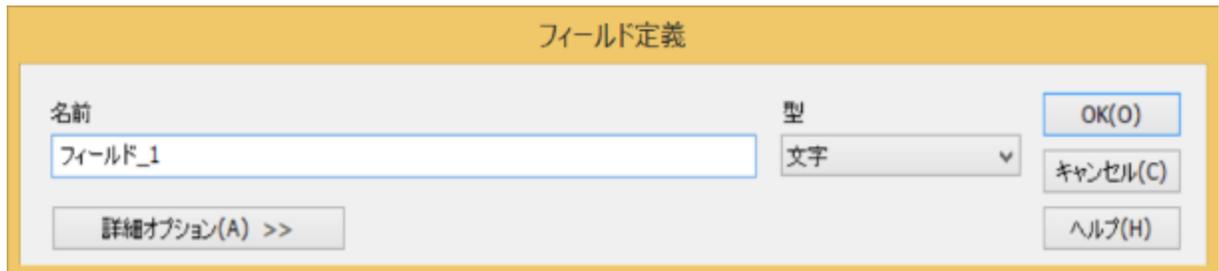
詳細な手順については、"印刷イメージファイルの定義とインポート" ページ 300または"PDF ファイルの定義とインポート" ページ 308を参照してください。

2. 定義する最初のフィールドの値を選択します。

以下の例では、"Product No" フィールドの最初の値が選択されています。



フィールド定義 ダイアログボックスが表示されます。



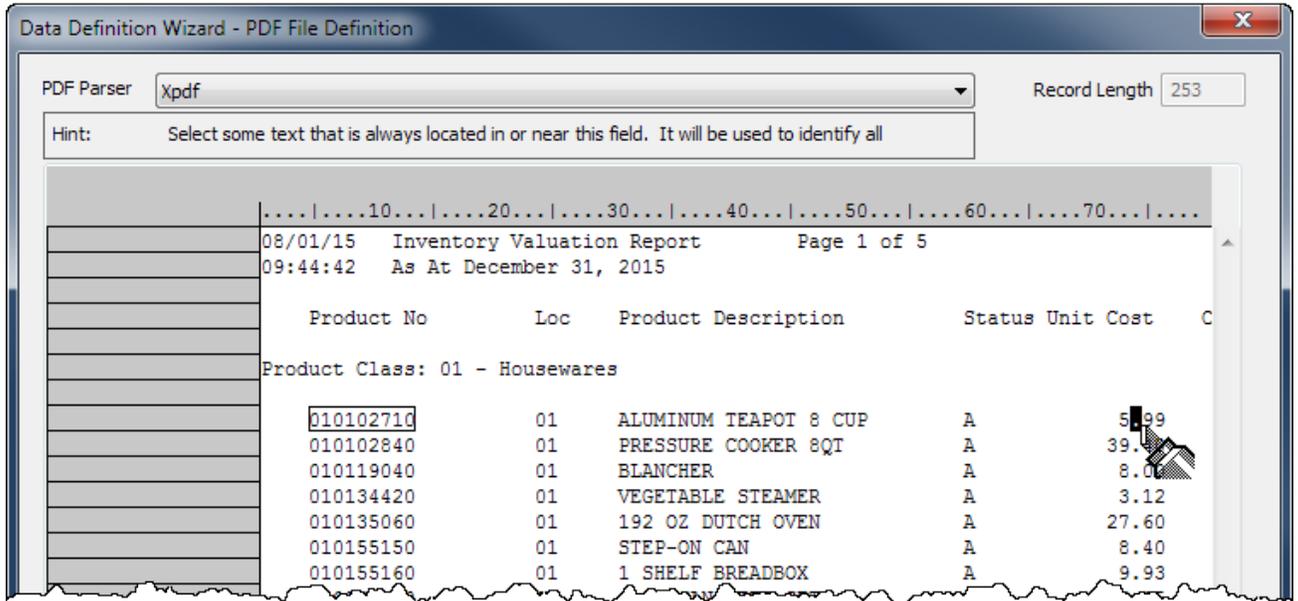
3. フィールドの名前を入力し、必要に応じて、データ型を更新して、**OK**をクリックします。

選択した値は四角で囲まれます。

4. 選択した値またはファイルの同じ行で、取り込むレコードのセットを一意に特定する1つ以上の文字を選択します。

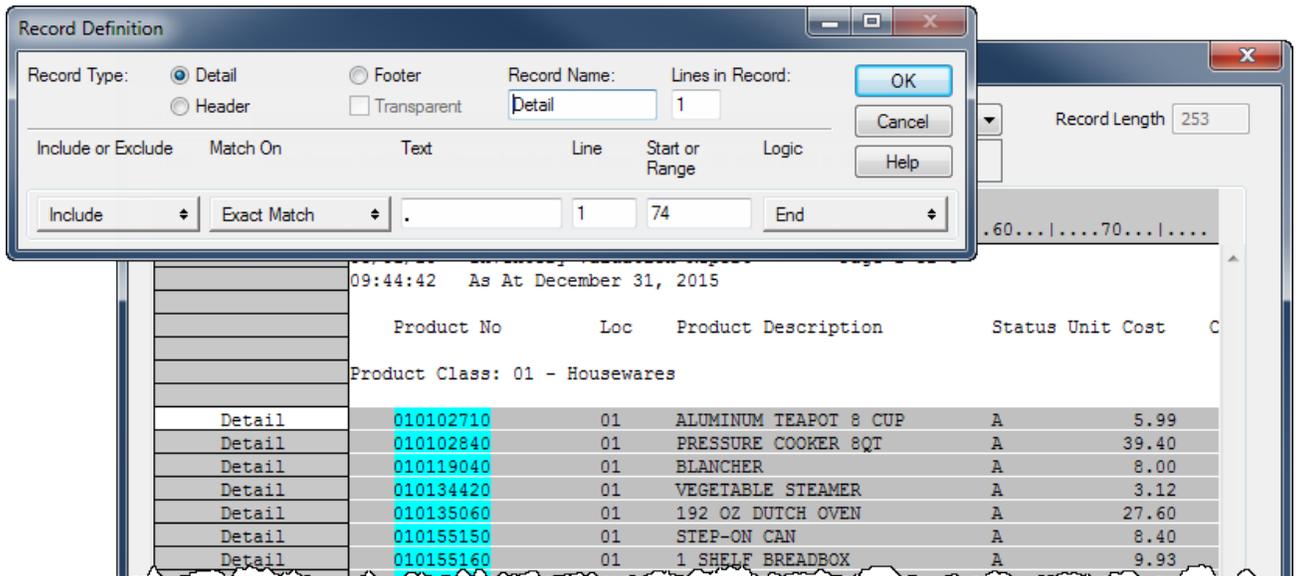
詳細については、"印刷イメージ(レポート)ファイルとPDF ファイルの定義とインポート" ページ 279を参照してください。

以下の例では、"Unit Cost" フィールドの小数点を選択されています。



レコード定義] ダイアログボックスが開き、最初のフィールドと、関連付けられたレコードのセットが定義されます。

フィールドは水色、レコードは灰色で示されます。未定義のデータは白色の背景です。



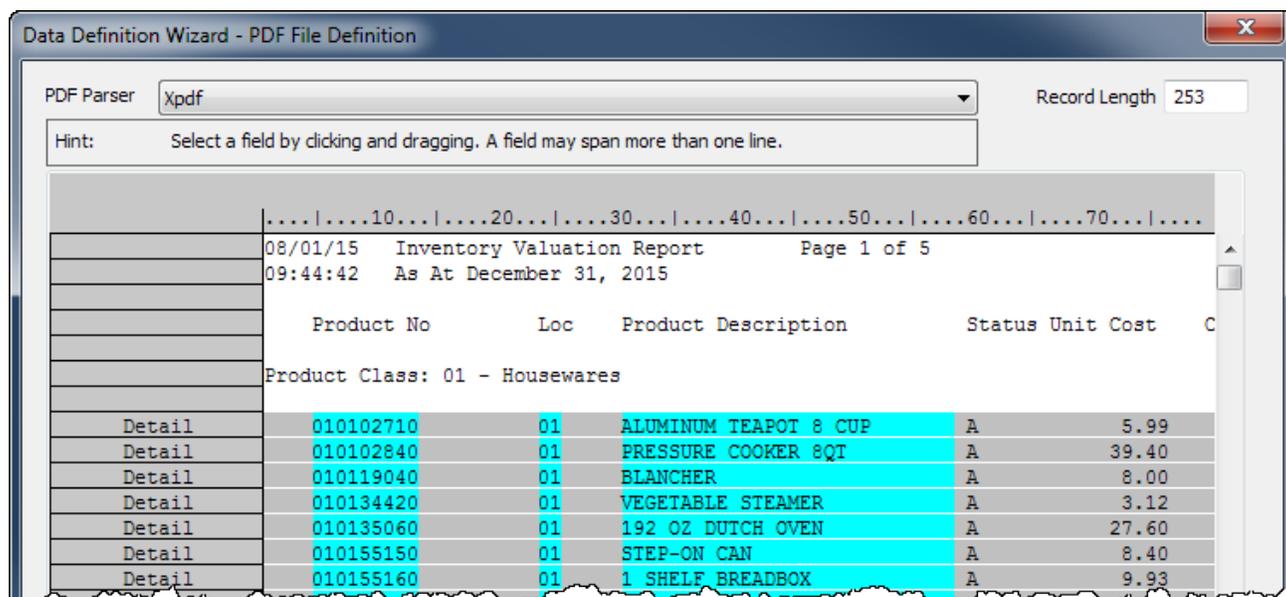
- 必要に応じて、**レコードタイプ**を変更するか、レコードのセットを取り込むために使用される条件を修正または作成し、**[OK]**をクリックします。

詳細については、「レコード定義での作業」ページ 322を参照してください。

- 各フィールドの値を選択し、レコードの別のフィールドを定義します。
追加フィールドは自動的にレコードのセットに適合します。

フィールド値の長さが異なる場合は、最も長い値を選択するか、あるいは余分な空白スペースを選択してフィールド内のどこかにある長い値に対応できるようにします。

以下の例では、"Product No"、"Loc"、"Product Description" の3つのフィールドが定義されます。



- 必要なフィールドの定義が完了したら、**次へ**]をクリックします。

定義およびインポート処理の残りの手順は、Excel や区切り文字付きテキスト ファイルなどの他のデータ形式を定義およびインポートする手順と同様です。

詳細な手順については、"印刷イメージファイルの定義とインポート" 見開きページまたは"PDF ファイルの定義とインポート" ページ 308を参照してください。

印刷イメージファイルの定義とインポート

印刷イメージファイルを定義およびインポートして、Analytics テーブルを作成できます。

データ定義ウィザードを使用して印刷イメージファイル进行处理するときには、Analytics がファイルのすべてまたは一部を自動的に定義する場合があります。あるいは、ファイルを手動で定義しなければならない場合があります。

メモ

印刷イメージファイルの定義は簡単ではありません。問題が発生した場合は、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を確認してください。

印刷イメージファイルの検索と選択

1. [ファイル > 新規作成 > テーブル] の順にクリックします。
2. [データソースのプラットフォームの選択] ページが表示されている場合は、[ローカル] を選択し、[次へ] をクリックします。
3. [ローカルデータソースの選択] ページで [ファイル] を選択し、[次へ] をクリックします。
4. [定義するファイルの選択] ダイアログボックスで、Analytics テーブルの作成元とする印刷イメージファイルを探して選択し、[開く] をクリックします。

通常、印刷イメージファイルのファイル拡張子は .txt です。

5. [文字セット] ページで、正しい文字セット オプションが選択されていることを確認し、[次へ] をクリックします。
6. [ファイル形式] ページで、[印刷イメージ(レポート)ファイル] オプションを必ず選択し、[次へ] をクリックします。

印刷イメージファイルが解析され、[印刷イメージファイルの定義] ページでは解析されたファイルが表示されます。

印刷イメージファイルの定義

1. [印刷イメージファイルの定義] ページで、上下左右にスクロールし、解析されたファイルを調べます。

強調表示は Analytics ではファイルに自動定義されたデータがあるかどうかを示します。

強調表示	意味
水色の強調表示	フィールドとして自動定義されたデータ。
灰色の強調表示	レコードとして自動定義されたデータ。 レコード定義は、レコードで定義されている1つ以上のフィールドによって異なります。
白色の背景	未定義のデータ。 Analyticsではデータのパターンを検出できず、自動定義できませんでした。

2. 次のいずれかを実行します。

自動定義の結果	対応
Analytics がファイルを自動定義し、更新する場合	Analytics がファイル全体を完全に自動定義し、次のことをしない場合 <ul style="list-style-type: none"> 汎用フィールド名の更新 ヘッダーまたはフッターデータを詳細データに追加に移動します"印刷イメージ ファイルの定義を確定する" ページ 305
Analytics がファイルを自動定義し、更新する場合	Analytics がファイル全体を完全に自動定義し、次のことをする場合 <ul style="list-style-type: none"> 汎用フィールド名の更新 ("Field_1"、"Field_2" など)、移動します"自動定義の編集" 見開きページ 詳細データにヘッダーデータまたはフッターデータを追加する場合は、進みます。"印刷イメージ ファイルの手動による定義" ページ 303 <p>ヒント データ定義ウィザードの次のページで汎用フィールド名を更新することもできます。この方が便利な場合があります。</p>
自動定義にエラーが含まれる場合	自動定義 <ul style="list-style-type: none"> にエラーが含まれる場合 必要なデータを除外します 必要ではないデータを含めます <p>次のいずれかを実行する必要があります。</p> <ul style="list-style-type: none"> "自動定義の編集" 見開きページ 自動定義全体の削除 "印刷イメージ ファイルの手動による定義" ページ 303 <p>ヒント 自動定義に重大なエラーが含まれる場合は、自動定義全体を削除し、ファイルを手動で定義する方が簡単な場合があります。</p>
解析されたファイルが完全に未定義の場合	解析されたファイルが完全に未定義の場合(白色の背景)、"印刷イメージ ファイルの手動による定義" ページ 303

自動定義の編集

自動定義(または手動定義)を編集したい場合は、**印刷イメージファイルの定義** ページで次のいずれかを実行します。

タスクの編集	説明
フィールド定義の編集	<p>水色で強調表示されたフィールドを右クリックして、フィールドの編集を選択するか、フィールドをダブルクリックします。</p> <p>次を含む多数の変更を行うことができます。</p> <ul style="list-style-type: none"> ○ フィールド名の更新 ○ データ型の更新 ○ 詳細オプションの下： <ul style="list-style-type: none"> ● フィールド長の変更(フィールド幅) ● フィールドの開始位置の変更 <p>詳細については、"フィールド定義での作業" ページ 317を参照してください。</p>
レコード定義の編集	<p>灰色のレコードを右クリックして、レコードの編集を選択するか、レコードをダブルクリックします。</p> <p>2つの主な変更を行うことができます。</p> <ul style="list-style-type: none"> ○ レコードの分類の更新。詳細、ヘッダー、フッターは任意です。 ○ レコードのセットを取り込むために Analytics が使用する条件を修正します <p>詳細については、"レコード定義での作業" ページ 322を参照してください。</p>
フィールド定義またはレコード定義の削除	<p>フィールドまたはレコードを右クリックし、フィールドの削除またはレコードの削除を選択します。</p> <p>Analytics テーブルに必要なフィールド、または自動定義でエラーが発生したために手動で定義したいフィールドの定義を削除できます。</p> <p>レコード定義を削除する場合、レコードに含まれるすべてのフィールド定義も削除され、ファイルのレコード定義のすべてのインスタンスが削除されます。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>削除しているのは、フィールド定義またはレコード定義のみであり、実際のデータを削除しているわけではありません。必要に応じて、同じフィールドまたはレコードデータを再定義できます。</p> </div> <div style="border-left: 2px solid #008000; padding-left: 10px; margin-top: 10px;"> <p>ヒント</p> <p>レコードを選択して削除する場合は、レコードの編集を選択し、レコードのセットを取り込むために Analytics で使用される条件を微調整します。</p> <p>詳細については、"レコード定義での作業" ページ 322を参照してください。</p> </div>

印刷イメージファイルの手動による定義

ヒント

始める前に、"クイックスタートの手順" ページ 296というスクリーンキャプチャー付きの以下の基本的な手順をまず確認すると役立つでしょう。

メモ

保存されているフィールドとレコード定義がある場合はこれらを使用すると、印刷イメージファイルを定義することもできます。

詳細については、"保存されているフィールドとレコード定義のセットを使用した印刷イメージファイルの定義" ページ 305を参照してください。

1. **印刷イメージファイル定義** ページで、データ値を選択し、テーブル内のフィールドの1つを定義することから始めます。

たとえば、SSN フィールドの社会保障番号を選択できます。データ値を選択するときには、**フィールド定義** ダイアログボックスが表示されます。

ガイドライン:

- データの任意の場所で値を選択できます。テーブルの最初のフィールドを使用したり、フィールドの最初の値を選択したりする必要はありません。
- 値には詳細データ、ヘッダーデータ、またはフッターデータを選択できます。
- フィールド名を選択しないでください。ソースファイルのすべてのフィールド名は選択しないようにしてください。ソースファイルのフィールド名を選択すると、Analytics はそのフィールド名を、フィールドに含まれるデータとして扱います。
- フィールド値の長さが異なる場合は、最も長い値を選択するか、あるいは余分な空白スペースを選択して、フィールドの低い位置にある可能性のある、現在表示されていない長い値に対応できるようにします。

初期データ値を使用して、レコードのセットを一意に識別する場合は、"フィールド定義での作業" ページ 317を参照してください。

2. フィールドの名前を入力し、必要に応じて、データ型を更新して、**OK** をクリックします。
3. 選択したデータ値またはファイルの同じ行で、ソースファイルのレコードのセットを一意に識別する文字または文字列を選択します。

たとえば次を選択します。

- 日付の値のスラッシュ
- 数値の小数点
- 選択したデータ値を含む行のいずれかの場所にある一意に識別可能な値

一意の文字を選択する場合は、**レコード定義** ダイアログボックスが開き、文字を含むすべてのレコードが灰色に強調表示されます。

詳細については、"印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート" ページ 279 を参照してください。

ソースファイルで 1 行を超えているレコードを定義する必要がある場合は、"複数行のレコードおよびフィールドの操作" ページ 332を参照してください。

4. 必要に応じて、**[レコードタイプ]**を更新し、定義しているデータのタイプ(詳細、ヘッダー、またはフッター)に合わせます。
5. 必要に応じて、レコードのセットを取り込むために使用する条件を変更します。

たとえば、別の条件を追加し、最初に取り込まれたレコードの一部を省略することができます。

詳細については、"レコード定義での作業" ページ 322を参照してください。

6. **[OK]**をクリックします。

定義したフィールドは水色で強調表示され、取り込まれたレコードの関連付けられたセットは灰色で強調表示されます。

7. 上下にスクロールして、定義されたフィールドと、取り込まれたレコードの関連付けられたセットを検証します。
8. フィールドが正しく定義されていない場合、または取り込まれたレコードのセットを調整する必要がある場合は、フィールドまたはレコードをダブルクリックし、**[フィールド定義]**ダイアログボックスまたは**[レコード定義]**ダイアログボックスで必要な編集を行います。

詳細については、"フィールド定義での作業" ページ 317と"レコード定義での作業" ページ 322を参照してください。

9. 各フィールドの代表データ値を選択し、レコードの残りのフィールドを定義します。

追加フィールドは自動的にレコードのセットに適合します。

ガイドライン:

- 結果の Analytics テーブルで必要なフィールドのみを定義します。
- 各フィールド定義で上下にスクロールし、定義済みのフィールドを検証します。必要に応じて定義を編集します。

たとえば、データ値が完全にフィールドに含まれていない場合は、長さまたはフィールドの開始位置を調整する必要があります。

詳細については、"自動定義の編集" ページ 302を参照してください。

- ソースファイルで 1 行を超えているフィールド値を定義する必要がある場合は、"複数行のレコードおよびフィールドの操作" ページ 332を参照してください。

ヒント

詳細フィールドを定義する順序が、結果の Analytics テーブルに表示される順序になります。

定義処理中に詳細フィールドを削除して、もう一度追加すると、元の位置が失われ、詳細フィールドの最後に配置されます。

- 別のレコードを定義する場合は、手順の1～9を繰り返してください。

ガイドライン:

- データ値を選択し、レコードの新しいフィールドおよび関連付けられたセットの定義を開始するときには、表示されているダイアログボックスで **新規レコード**] が選択されていることを確認し、**OK**] をクリックします。
- ヘッダーレコードまたはフッターレコードは複数定義できますが、詳細レコードは1つしか定義できません。異なるレコードタイプを定義する順序は強制されません。

保存されているフィールドとレコード定義のセットを使用した印刷イメージファイルの定義

印刷イメージクエリファイルに保存されている前のファイル定義セッションからのフィールドとレコードの定義を使用して、印刷イメージファイルを定義することができます。印刷イメージクエリファイルはすでに存在している必要があり、保存されている定義は現行データと一致しなければなりません。

メモ

印刷イメージクエリファイルを読み込むと、任意の現行のフィールドとレコードの定義が削除されます。

- 印刷イメージファイル定義] ページで **読み込み**] をクリックします。
- 以前に保存されている印刷イメージクエリファイルに移動してこれを選択し、**開く**] をクリックします。

定義が現在のデータに適用されます。

印刷イメージクエリファイルのファイル拡張子は .txt です。

メモ

現行データと一致する、またはほとんど一致することが分かっている定義のあるファイルのみを読み込んでください。

- ファイルの読み込み後、次のいずれかを実行します。
 - ファイル全体が完全に定義されている場合 -"印刷イメージファイルの定義を確定する" 下に移動します
 - ファイル定義の調整が必要な場合 -"自動定義の編集" ページ 302 に移動します

印刷イメージファイルの定義を確定する

- 省略可能。フィールドとレコード定義の現在のセットを保存する場合は、次のことを実行します。
 - 保存**] をクリックします。
 - 印刷イメージクエリファイルの名前を入力し、**保存**] をクリックします。

メモ

フィールドとレコードの定義には多くの作業を要する場合があります。そのためこれらを保存することをお勧めします。

インポートされたデータに調整が必要で、再定義、再インポートが必要であることが後で分かった場合は、保存されている定義はゼロから再び作成する必要はありません。

- すべてのフィールドおよびレコード定義が完了したら、**次へ**]をクリックします。

メモ

必要であれば、ウィザードのこの時点に戻り、フィールドとレコードの定義を更新できます。

Analytics データ ファイルの保存

データ ファイルを別名で保存] ダイアログ ボックスで、Analytics データ ファイルの名前を入力して **保存**] をクリックします。

Analytics によってデータ ファイル名 があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

Analytics によって開かれたデフォルトのロケーションを使用しない場合は、異なるフォルダーに移動して、データ ファイルを保存することもできます。

Analytics フィールド プロパティの編集

フィールド プロパティの編集] ページで、Analytics によって、以下に一覧されたプロパティへと割り当てられた設定をレビューし、必要な更新を行い、**次へ**]をクリックします。

メモ

列に関連付けられたプロパティを確認するには、プレビューテーブルで列見出しを選択します。

プロパティ	説明
このフィールドを無視する	結果として得られるテーブルレイアウトからフィールドを除外します。 このフィールドのデータは、インポートは行われますが、定義されないため、新しい Analytics テーブルには表示されません。必要に応じ、あとで定義して、テーブルに追加することができます。
名前	テーブルレイアウトのフィールド名。 Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
列見出し	デフォルトの Analytics ビューにおけるフィールドの列見出し。 列見出しを指定しない場合は、 名前]の値が使用されます。

<p>型</p>	<p>Analytics のフィールドに割り当てられるデータ型。</p> <p>Analytics によって割り当てられたデータ型を保持するか、ドロップダウンリストから適切なデータ型を選択することができます。</p> <p>Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。</p>
<p>値</p>	<p>フィールドの最初の値を表示する読み取り専用プロパティ。</p> <p>この値は、行う編集に基づき動的に更新します。</p>
<p>10 進型(Decimal)</p>	<p>数値フィールドのみ。ソースデータの小数点以下桁数。</p> <p>メモ [小数位]テキスト ボックスは、数値のデータ型を選択すると自動的に表示されません。</p>
<p>入力形式</p>	<p>日付時刻フィールドのみ。ソースデータの日付時刻値の書式。</p> <p>指定する書式はソースデータの書式と正確に一致する必要があります。</p> <p>日付と時刻の書式の詳細については、"ソースデータに含まれる日付および時刻データの書式" ページ 366を参照してください。</p>

インポートの最終処理

1. **最終]** ページで、新しい Analytics テーブルの設定を確認したら、**完了]** をクリックします。
変更する場合は、**戻る]** をクリックし、ウィザードの適切なページへと進みます。
2. プロジェクトに追加中のテーブル名を入力するか、デフォルトの名前を保持し、そこで、**OK]** をクリックします。

新しい Analytics テーブルは、インポートされたファイルからのデータと一緒に作成されます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

PDF ファイルの定義とインポート

Adobe PDF ファイルを定義およびインポートして、Analytics テーブルを作成できます。

データ定義ウィザードを使用して PDF ファイルを処理するときには、Analytics がファイルのすべてまたは一部を自動的に定義する場合があります。あるいは、ファイルを手動で定義しなければならない場合があります。

メモ

PDF ファイルの定義は簡単ではありません。問題が発生した場合は、「印刷イメージ(レポート) ファイルと PDF ファイルの定義とインポート」 ページ 279を確認してください。

PDF ファイルの検索と選択

1. **ファイル > 新規作成 > テーブル** の順にクリックします。
2. **データソースのプラットフォームの選択** ページが表示されている場合は、**ローカル** を選択し、**次へ** をクリックします。
3. **ローカル データソースの選択** ページで **ファイル** を選択し、**次へ** をクリックします。
4. **定義するファイルの選択** ダイアログボックスで、Analytics テーブルの作成元とする PDF ファイルを探して選択し、**開く** をクリックします。

Adobe PDF ファイルのファイル拡張子は .pdf です。

5. **ファイル形式** ページで、**PDF Adobe Acrobat ファイル** オプションを必ず選択し、**次へ** をクリックします。

PDF ファイルの定義

1. **PDF ファイルの定義** ページで、必要に応じて、PDF ファイルのパスワードを入力し、**次へ** をクリックします。
2. **すべてのページ**ではなく、解析対象の特定のページまたはページ範囲を指定する場合は、**ページ指定** を選択し、1 つ以上のページ番号を指定します。

カンマで区切られた個々のページ (1,3,5)、ページの範囲 (2-7)、またはその組み合わせ (1, 3, 5-7, 11) を指定することができます。

ヒント

ある状況下においては、ページごとにPDF ファイルを解析すると、データの配置にずれが生じる場合があります。

この方法を使用する場合は、ファイルを2回以上インポートし、複数のAnalytics テーブルを作成して、Analytics に作成したテーブルを追加する必要があります。

詳細については、「印刷イメージまたはPDF データのサブセットの定義とインポート」ページ 330を参照してください。

3. **[PDF パーサー]**はデフォルト設定の**"Xpdf"**にするか、**"VeryPDF"**を選択します。
初めてファイルをインポートし、特に理由がない場合は、設定を**Xpdf**のままにしてください。
ファイルで**Xpdf**を使用して、既にデータの整列に問題が発生している場合は、**VeryPDF**を選択して解析結果が改善されるか確認してください。
4. **[次へ]**をクリックします。
PDF ファイルが解析されると **[PDF ファイル定義]** ページが更新され、解析されたファイルが表示されます。
5. 上下左右にスクロールし、解析されたファイルを検証します。
強調表示はAnalytics ではファイルに自動定義されたデータがあるかどうかを示します。

強調表示	意味
水色の強調表示	フィールドとして自動定義されたデータ。
灰色の強調表示	レコードして自動定義されたデータ。 レコード定義は、レコードで定義されている1つ以上のフィールドによって異なります。
白色の背景	未定義のデータ。 Analyticsではデータのパターンを検出できず、自動定義できませんでした。

6. 省略可能。解析されたファイルのデータの配置がずれていた場合、**[戻る]**をクリックし、**[PDF パーサー]**でパーサー選択を切り替え、**[次へ]**をクリックします。
PDF ファイルは、選択したパーサーを使用して再解析されます。これにより、データの配置が向上する場合があります。
ファイルを再解析するときには、すべての既存のフィールドおよびレコード定義が削除されます。
7. 次のいずれかを実行します。

自動定義の結果	対応
Analytics がファイルを自動定義し、更新する場合	Analytics がファイル全体を完全に自動定義し、次のことをしない場合 <ul style="list-style-type: none"> ○ 汎用フィールド名の更新 ○ ヘッダーまたはフッター データを詳細データに追加 に移動します"PDF ファイルの定義を確定する" ページ 314

自動定義の結果	対応
Analytics がファイルを自動定義し、更新する場合	<p>Analytics がファイル全体を完全に自動定義し、次のことをする場合</p> <ul style="list-style-type: none"> 汎用フィールド名の更新 ("Field_1"、"Field_2" など)、移動します "自動定義の編集" 下 詳細データにヘッダー データまたはフッター データを追加する場合は、進みます。"PDF ファイルの手動による定義" 次のページ <p>ヒント データ定義ウィザードの次のページで汎用フィールド名を更新することもできます。この方が便利な場合があります。</p>
自動定義にエラーが含まれる場合	<p>自動定義</p> <ul style="list-style-type: none"> にエラーが含まれる場合 必要なデータを除外します 必要ではないデータを含めます <p>次のいずれかを実行する必要があります。</p> <ul style="list-style-type: none"> "自動定義の編集" 下 自動定義全体の削除 "PDF ファイルの手動による定義" 次のページ <p>ヒント 自動定義に重大なエラーが含まれる場合は、自動定義全体を削除し、ファイルを手動で定義の方が簡単な場合もあります。</p>
解析されたファイルが完全に未定義の場合	<p>解析されたファイルが完全に未定義の場合 (白色の背景)、"PDF ファイルの手動による定義" 次のページ</p>

自動定義の編集

自動定義 (または手動定義) を編集したい場合は、[PDF ファイルの定義](#) ページで次のいずれかを実行します。

タスクの編集	説明
フィールド定義の編集	<p>水色で強調表示されたフィールドを右クリックして、フィールドの編集 を選択するか、フィールドをダブルクリックします。</p> <p>次を含む多数の変更を行うことができます。</p> <ul style="list-style-type: none"> フィールド名の更新 データ型の更新 詳細オプションの下: <ul style="list-style-type: none"> フィールド長の変更 (フィールド幅) フィールドの開始位置の変更 <p>詳細については、"フィールド定義での作業" ページ 317を参照してください。</p>
レコード定義の編集	<p>灰色のレコードを右クリックして、レコードの編集 を選択するか、レコードをダブルクリックしま</p>

タスクの編集	説明
	<p>す。</p> <p>2つの主な変更を行うことができます。</p> <ul style="list-style-type: none"> レコードの分類の更新。詳細、ヘッダー、フッターは任意です。 レコードのセットを取り込むためにAnalyticsが使用する条件を修正します <p>詳細については、「レコード定義での作業」ページ 322を参照してください。</p>
<p>フィールド定義またはレコード定義の削除</p>	<p>フィールドまたはレコードを右クリックし、フィールドの削除またはレコードの削除を選択します。</p> <p>Analyticsテーブルに必要なフィールド、または自動定義でエラーが発生したために手動で定義したいフィールドの定義を削除できます。</p> <p>レコード定義を削除する場合、レコードに含まれるすべてのフィールド定義も削除され、ファイルのレコード定義のすべてのインスタンスが削除されます。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>削除しているのは、フィールド定義またはレコード定義のみであり、実際のデータを削除しているわけではありません。必要に応じて、同じフィールドまたはレコードデータを再定義できます。</p> </div> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>ヒント</p> <p>レコードを選択して削除する場合は、レコードの編集を選択し、レコードのセットを取り込むためにAnalyticsで使用される条件を微調整します。</p> <p>詳細については、「レコード定義での作業」ページ 322を参照してください。</p> </div>

PDF ファイルの手動による定義

ヒント

始める前に、「クイックスタートの手順」ページ 296というスクリーンキャプチャー付きの以下の基本的な手順をまず確認すると役立つでしょう。

メモ

保存されているフィールドとレコード定義がある場合はこれらを使用すると、PDF ファイルを定義することもできます。

詳細については、「保存されているフィールドとレコード定義のセットを使用した PDF ファイルの定義」ページ 313を参照してください。

1. **PDF ファイル定義** ページで、データ値を選択し、テーブル内のフィールドの1つを定義することから始めます。

たとえば、SSN フィールドの社会保障番号を選択できます。データ値を選択するときには、**フィールド定義** ダイアログボックスが表示されます。

ガイドライン:

- データの任意の場所で値を選択できます。テーブルの最初のフィールドを使用したり、フィールドの最初の値を選択したりする必要はありません。
- 値には詳細データ、ヘッダーデータ、またはフッターデータを選択できます。
- フィールド名を選択しないでください。ソースファイルのすべてのフィールド名は選択しないようにしてください。ソースファイルのフィールド名を選択すると、Analytics はそのフィールド名を、フィールドに含まれるデータとして扱います。
- フィールド値の長さが異なる場合は、最も長い値を選択するか、あるいは余分な空白スペースを選択して、フィールドの低い位置にある可能性のある、現在表示されていない長い値に対応できるようにします。

初期データ値を使用して、レコードのセットを一意に識別する場合は、"フィールド定義での作業" ページ 317を参照してください。

2. フィールドの名前を入力し、必要に応じて、データ型を更新して、**[OK]**をクリックします。
3. 選択したデータ値またはファイルの同じ行で、ソースファイルのレコードのセットを一意に識別する文字または文字列を選択します。

たとえば次を選択します。

- 日付の値のスラッシュ
- 数値の小数点
- 選択したデータ値を含む行のいずれかの場所にある一意に識別可能な値

一意の文字を選択する場合は、**[レコード定義]**ダイアログボックスが開き、文字を含むすべてのレコードが灰色に強調表示されます。

詳細については、"印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート" ページ 279を参照してください。

ソースファイルで1行を超えているレコードを定義する必要がある場合は、"複数行のレコードおよびフィールドの操作" ページ 332を参照してください。

4. 必要に応じて、**[レコードタイプ]**を更新し、定義しているデータのタイプ(詳細、ヘッダー、またはフッター)に合わせます。
5. 必要に応じて、レコードのセットを取り込むために使用する条件を変更します。

たとえば、別の条件を追加し、最初に取り込まれたレコードの一部を省略することができます。

詳細については、"レコード定義での作業" ページ 322を参照してください。

6. **[OK]**をクリックします。
定義したフィールドは水色で強調表示され、取り込まれたレコードの関連付けられたセットは灰色で強調表示されます。
7. 上下にスクロールして、定義されたフィールドと、取り込まれたレコードの関連付けられたセットを検証します。
8. フィールドが正しく定義されていない場合、または取り込まれたレコードのセットを調整する必要がある場合は、フィールドまたはレコードをダブルクリックし、**[フィールド定義]**ダイアログボックスまたは**[レコード定義]**ダイアログボックスで必要な編集を行います。

詳細については、"フィールド定義での作業" ページ 317と"レコード定義での作業" ページ 322を参照してください。

9. 各フィールドの代表データ値を選択し、レコードの残りのフィールドを定義します。

追加フィールドは自動的にレコードのセットに適合します。

ガイドライン:

- 結果の Analytics テーブルで必要なフィールドのみを定義します。
- 各フィールド定義で上下にスクロールし、定義済みのフィールドを検証します。必要に応じて定義を編集します。
たとえば、データ値が完全にフィールドに含まれていない場合は、長さまたはフィールドの開始位置を調整する必要があります。
詳細については、"自動定義の編集" ページ 310を参照してください。

- ソースファイルで 1 行を超えているフィールド値を定義する必要がある場合は、"複数行のレコードおよびフィールドの操作" ページ 332を参照してください。

ヒント

詳細フィールドを定義する順序が、結果の Analytics テーブルに表示される順序になります。

定義処理中に詳細フィールドを削除して、もう一度追加すると、元の位置が失われ、詳細フィールドの最後に配置されます。

10. 別のレコードを定義する場合は、手順の 1 ~ 9 を繰り返してください。

ガイドライン:

- データ値を選択し、レコードの新しいフィールドおよび関連付けられたセットの定義を開始するときには、表示されているダイアログボックスで **新規レコード** が選択されていることを確認し、**OK** をクリックします。
- ヘッダーレコードまたはフッターレコードは複数定義できますが、詳細レコードは 1 つしか定義できません。異なるレコードタイプを定義する順序は強制されません。

保存されているフィールドとレコード定義のセットを使用した PDF ファイルの定義

印刷イメージクエリファイルに保存されている前のファイル定義セッションからのフィールドとレコードの定義を使用して、PDF ファイルを定義することができます。印刷イメージクエリファイルはすでに存在している必要があり、保存されている定義は現行データと一致しなければなりません。

メモ

印刷イメージ クエリファイルを読み込むと、任意の現行のフィールドとレコードの定義が削除されます。

1. **PDF ファイル定義** ページで **読み込み** をクリックします。
2. 以前に保存されている印刷イメージ クエリファイルに移動してこれを選択し、**開く** をクリックします。

定義が現在のデータに適用されます。

印刷イメージ クエリファイルのファイル拡張子は .txt です。

メモ

現行データと一致する、またはほとんど一致することが分かっている定義のあるファイルのみを読み込んでください。

3. ファイルの読み込み後、次のいずれかを実行します。
 - **ファイル全体が完全に定義されている場合** - "PDF ファイルの定義を確定する" 下
に移動します
 - **ファイル定義の調整が必要な場合** - "自動定義の編集" ページ 310
に移動します

PDF ファイルの定義を確定する

1. 省略可能。フィールドとレコード定義の現在のセットを保存する場合は、次のことを実行します。
 - a. **保存** をクリックします。
 - b. 印刷イメージ クエリファイルの名前を入力し、**保存** をクリックします。

メモ

フィールドとレコードの定義には多くの作業を要する場合があります。そのためこれらを保存することをお勧めします。

インポートされたデータに調整が必要で、再定義、再インポートが必要であることが後で分かった場合は、保存されている定義はゼロから再び作成する必要はありません。

2. すべてのフィールドおよびレコード定義が完了したら、**次へ** をクリックします。

メモ

必要であれば、ウィザードのこの時点に戻り、フィールドとレコードの定義を更新できます。

Analytics データ ファイルの保存

データ ファイルを別名で保存] ダイアログ ボックスで、Analytics データ ファイルの名前を入力して **保存**] をクリックします。

Analytics によってデータ ファイル名 があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

Analytics によって開かれたデフォルトのロケーションを使用しない場合は、異なるフォルダーに移動して、データ ファイルを保存することもできます。

Analytics フィールド プロパティの編集

フィールド プロパティの編集] ページで、Analytics によって、以下に一覧されたプロパティへと割り当てられた設定をレビューし、必要な更新を行い、**次へ**] をクリックします。

メモ

列に関連付けられたプロパティを確認するには、プレビュー テーブルで列見出しを選択します。

プロパティ	説明
このフィールドを無視する	結果として得られるテーブルレイアウトからフィールドを除外します。 このフィールドのデータは、インポートは行われますが、定義されないため、新しい Analytics テーブルには表示されません。必要に応じ、あとで定義して、テーブルに追加することができます。
名前	テーブルレイアウトのフィールド名。 Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
列見出し	デフォルトの Analytics ビューにおけるフィールドの列見出し。 列見出しを指定しない場合は、 名前] の値が使用されます。
型	Analytics のフィールドに割り当てられるデータ型。 Analytics によって割り当てられたデータ型を保持するか、ドロップダウン リストから適切なデータ型を選択することができます。 Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。
値	フィールドの最初の値を表示する読み取り専用プロパティ。 この値は、行う編集に基づき動的に更新します。
10 進型(Decimal)	数値フィールドのみ。ソース データの小数点以下桁数。

	<p>メモ</p> <p>[小数位]テキスト ボックスは、数値のデータ型を選択すると自動的に表示されます。</p>
入力形式	<p>日付時刻フィールドのみ。ソースデータの日付時刻値の書式。</p> <p>指定する書式はソースデータの書式と正確に一致する必要があります。</p> <p>日付と時刻の書式の詳細については、「ソースデータに含まれる日付および時刻データの書式」ページ 366を参照してください。</p>

インポートの最終処理

1. **最終]** ページで、新しい Analytics テーブルの設定を確認したら、**完了]** をクリックします。
変更する場合は、**戻る]** をクリックし、ウィザードの適切なページへと進みます。
2. プロジェクトに追加中のテーブル名を入力するか、デフォルトの名前を保持し、そこで、**OK]** をクリックします。
新しい Analytics テーブルは、インポートされたファイルからのデータと一緒に作成されます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

フィールド定義での作業

フィールド定義は、印刷イメージまたはPDFファイルの単一のフィールドを表す情報です。印刷イメージまたはPDFファイルがイメージで、フィールドとレコードを識別するメタデータがない場合、1つ以上のフィールド定義を指定し、ファイルのフィールドを特定し、その周りのデータまたは空白とは区別する必要があります。

ファイル定義処理中に1つ以上のフィールド定義がAnalyticsによって自動的に作成されるか、あるいは、手動でフィールド定義を作成する必要があります。

初期データ値を使用して、レコードのセットを一意に識別する

印刷イメージまたはPDFファイルを手動で定義するには、まず、最初のデータ値を選択してから、関連付けられるレコードのセットを取り込みます。初期データ値のすべてまたは一部を使用して、レコードのセットを一意に識別する場合は、初期データ値を含むフィールドを選択するときに以下のガイドラインに従ってください。

- レコードのどの場所に位置しているフィールドでも可能です。レコードの最初のフィールドでなくてもかまいません。
- データに一貫した構造を持つフィールドを探します。例：
 - MM/DD/YYYY などの一貫した形式を持つ日付フィールド
 - SSN(社会保障番号)フィールド
 - クレジットカード番号フィールド
 - 一貫した構造のIDまたは数値フィールド

異なる内容のフィールドを使用するよりも、一貫した構造のフィールドを使用する方が成功しやすくなります。

- フィールドで1つ以上の一貫して配置された文字は、そのフィールドの上下のデータと比較するときに、一意であるか、一意に配置されている必要があります。
- 値が欠落しているフィールドは使用しないでください。値が欠落しているフィールドを使用することはできますが、ファイル定義の処理が複雑になります。

メモ

レコードのセットを一意に識別するために使用する値は、初期データ値または初期データフィールドに含める必要がありません。初期データ値を含む行のどの場所にあってもかまいません。詳細については、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を参照してください。

フィールド定義]ダイアログボックス

フィールド定義]ダイアログボックスでは、印刷イメージまたはPDFファイルのフィールドを表す情報を指定します。

以下の図は、フィールド定義]ダイアログボックスの **詳細オプション]** が展開されている状態を示します。

The screenshot shows the 'Field Definition' dialog box with the following details:

- 名前:** フィールド_9
- 型:** 数値
- 詳細オプション(A) <<** (Expanded)
- 開始する行:** 1
- 開始する列:** 57
- フィールドの幅:** 5
- フィールドの高さ:** 1
- 小数位:** 2
- 書式:** -9,999,999.99
- レポート用の列のデフォルト設定:**
 - 幅:** 5
 - 代替列見出し:** (Empty)
- 単一フィールドに変換する:**
- 空白の場合は埋める:**
- 空白行で終わる:**
- 総計を出力しない:**
- コントロール合計:**

次の表では、フィールド定義]ダイアログボックスの各項目の目的が説明されています。

項目名	目的
名前	Analytics によって割り当てられた汎用フィールド名以外のフィールド名を指定します。 指定した名前は結果の Analytics テーブルの物理フィールド名になります。つまり、テーブルレイアウトのフィールド名です。
型	フィールドのデータ型を指定します。 "文字"、"数値"、および"日付時刻"から選択します。数値または日付時刻フィールドの値に一貫性がない場合は、文字データとしてフィールドを定義し、インポートして試すことができます。
開始する行	フィールドの開始を含むレコードの行を指定します。 例:

項目名	目的
	<ul style="list-style-type: none"> フィールドを含む各レコードが単一の行に表示される場合、値は"1"でなければなりません。 フィールドを含む各レコードが2行にまたがり、フィールドが2行目から開始する場合、値は"2"でなければなりません。
開始する列	<p>フィールドが始まるバイト位置を指定します。</p> <p>たとえば、行の先頭の3つの空白スペースがフィールドの最初の文字の前にある場合は、開始する列の値は'4'(非 Unicode Analytics)または'7'(Unicode Analytics)である必要があります。</p> <p>メモ</p> <p>フィールドの開始位置は、定義およびインポート処理の成功にとって非常に重要です。フィールドが定義されたら、ソースファイルをスクロールして、開始位置にフィールドのすべての値が入ることを確認してください。必要に応じて開始位置を調整します。</p> <p>Unicode データでは、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>
フィールドの幅	<p>フィールドの長さをバイト単位で指定します。</p> <p>指定した長さは結果の Analytics テーブルの物理フィールド長になります。つまり、テーブルレイアウトのフィールド長です。</p> <p>メモ</p> <p>フィールドの長さは、定義およびインポート処理の成功にとって非常に重要です。フィールドが定義されたら、ソースファイルをスクロールして、フィールドの長さが十分で、すべての値がフィールドに入ることを確認してください。必要に応じて長さを調整します。</p> <p>Unicode データでは、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>
フィールドの高さ	<p>フィールドの単一の値を構成する行数を指定します。</p> <p>例:</p> <ul style="list-style-type: none"> 各値が単一の行に表示される場合は、フィールドの高さは"1"でなければなりません。 各値が2行にまたがる場合は、フィールドの高さは"2"でなければなりません。 各値が [ルート] フィールドの内容のように複数の行数にわたる場合は、フィールドの高さを設定し、最大行数になる値が入るようにします(以下の「空白行で終わる」を参照)。
小数点以下桁数 (数値フィールドのみ)	<p>数値の小数点以下の桁数を指定します。</p>
書式 (数値と日付時刻 フィールドのみ)	<p>数値または日付時刻データの書式を指定します。</p> <p>この書式は、ソースファイルの数値または日付時刻値の書式と一致する必要があります。</p> <p>例:</p> <ul style="list-style-type: none"> -1,234.00 などの数値がフィールドに表示される場合は、書式 -9,999,999.99 を選択または指定する必要があります。 31/12/2015 などの日付がフィールドに表示される場合は、書式 DD/MM/YYYY を選択または指定する必要があります。書式で MMM を使用すると、省略名を使用する月、または完全な名前の月と一致する必要があります。

項目名	目的
	<p>ヒント</p> <p>ソースファイルの数値または日付時刻データの書式が一貫していない場合は、文字データとしてインポートし、結果の Analytics テーブルで Analytics 関数を使用して書式のばらつきを統一することができます。</p>
<p>単一フィールドに変換する</p> <p>(文字フィールドのみ)</p> <p>(複数行のフィールドのみ)</p>	<p>ソースファイルで定義された複数行のフィールドがすべての行のデータを含む単一のフィールドとして Analytics にインポートされることを指定します。</p> <p>たとえば、複数の行にまたがる住所データを定義する場合は、単一のフィールドに変換する]を選択すると、1行にすべての住所データを含む単一フィールドを作成します。</p> <p>単一のフィールドに変換する]をオフ(デフォルト設定)にすると、複数行のフィールドが、各行に単一行のデータを含む複数のフィールドとして Analytics にインポートされます。</p>
<p>空白の場合は埋める</p>	<p>新しいフィールド値が現れるまで、現フィールド値が後続の空白値にコピーされることを指定します。</p> <p>たとえば、Product Class(製品クラス)フィールドの値 "01" が製品クラス 01 レコードの最初のブロックにのみ表示される場合に、空白の場合は埋める]を選択すると、すべてのレコードに値 "01" が表示されます。</p>
<p>空白行で終わる</p> <p>(複数行のフィールドのみ)</p>	<p>空白の行があるときに複数行のフィールドの値が終了することを指定します。</p> <p>このオプションは、複数行のフィールドの値が異なる行数にまたがるときに発生する状況に対応します。フィールドの高さ]を設定し、最大行数にまたがる値を収容できるようにする必要があります。ただし、これにより、少ない行数の値とフィールドまたはレコードの境界の間で不一致が発生することがあります。空白行で終わる]を選択すると、フィールドの高さ、およびフィールドまたはレコードの境界のサイズが動的に変更され、各値が占める行数に合うようになります。</p> <p>メモ</p> <p>1つ以上の空白行が複数行フィールドの各値を区切る場合にのみ、この機能は動作します。</p>
<p>レポート用の列のデフォルト設定</p> <ul style="list-style-type: none"> ○ 幅 ○ 代替列見出し ○ 合計を出力しない <p>(数値フィールドのみ)</p> <ul style="list-style-type: none"> ○ コントロール合計 <p>(数値フィールドのみ)</p>	<p>メモ</p> <p>レポート用の列のデフォルト設定]セクションの設定は任意です。この設定はデータ定義ウィザードのフィールドの処理に影響しません。同じプロパティは後から Analytics で設定できます。</p> <p>フィールドが結果の Analytics テーブルと Analytics レポートのデフォルトビューに表示される際のフィールドのプロパティを指定します。</p> <ul style="list-style-type: none"> ○ 幅 - フィールド用の表示幅をバイトで指定します。 ○ 代替列見出し - Analytics ビューおよびレポートにフィールドの内容を表示する場合、この値は列サイズとして使用されます。 ○ 代替列タイトル - Analytics ビューおよびレポートにフィールドを表示する場合、フィールド名の代わりに使用する列見出しを指定します。 ○ 総計を出力しない - Analytics レポートでフィールドの値が自動的に合計されないことを指定します。 <p>デフォルトでは、Analytics は自動的にレポート内の数値フィールドの合計を計算します。フィールドに、単価などの合計を出しても意味のないデータが含まれている場合は、この動作を止めさせることができます。</p> <ul style="list-style-type: none"> ○ コントロール合計 - フィールドをコントロール合計フィールドとして指定します。

項目名	目的
	<p>コントロール合計は数値フィールドの値の合計です。これを使用して、データの整合性を確認することができます。新規のテーブルにデータを抽出した場合、または新規のテーブルでデータを並べ替えた場合、コントロール合計フィールドの入力値合計と出力値合計がテーブル履歴に格納されます。入力は元テーブルを指しています。出力は新規テーブルを指しています。2つの合計が一致する場合は、抽出または並べ替え処理で失われたデータはありません。</p> <p>複数のフィールドに対してコントロール合計を指定すると、テーブル履歴はその中で開始位置が最も左にある数値フィールドについてのみ集計を報告します。</p> <p>メモ</p> <p>フィールド定義ダイアログボックスの コントロール合計 設定を選択しても、印刷イメージまたはPDFファイルをAnalyticsにインポートするときにコントロール合計は作成されません。この目的でコントロール合計を作成するには、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を参照してください。</p>

レコード定義での作業

レコード定義は、印刷イメージまたはPDFファイルのレコードのセットを取り込む情報、またはレコードのセットを表す情報です。印刷イメージまたはPDFファイルがイメージで、フィールドとレコードを識別するメタデータがない場合、1つ以上のレコード定義を指定し、ファイルのレコードを特定し、その周りのデータまたは空白とは区別する必要があります。

印刷イメージまたはPDFファイルを定義するときには、詳細レコードのセットを定義する必要があります。ヘッダーまたはフッターレコードも定義できますが、必須ではありません。

ファイル定義処理中に詳細レコード定義がAnalyticsによって自動的に作成されるか、あるいは、手動で1つ以上のレコード定義を作成する必要があります。

初期データ値の選択から開始する

レコード定義では、最初に初期データフィールドの初期データ値を選択します。初期データ値または初期データ値を含む行の文字はレコード定義の基礎となるもので、初期データフィールドに関連付けられたレコードのセットを特定します。詳細については、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を参照してください。

初期データフィールドと、関連付けられたレコードのセットが定義されたら、レコードの残りの部分を個別のデータ要素に分割するために必要な数の追加フィールド定義を指定します。

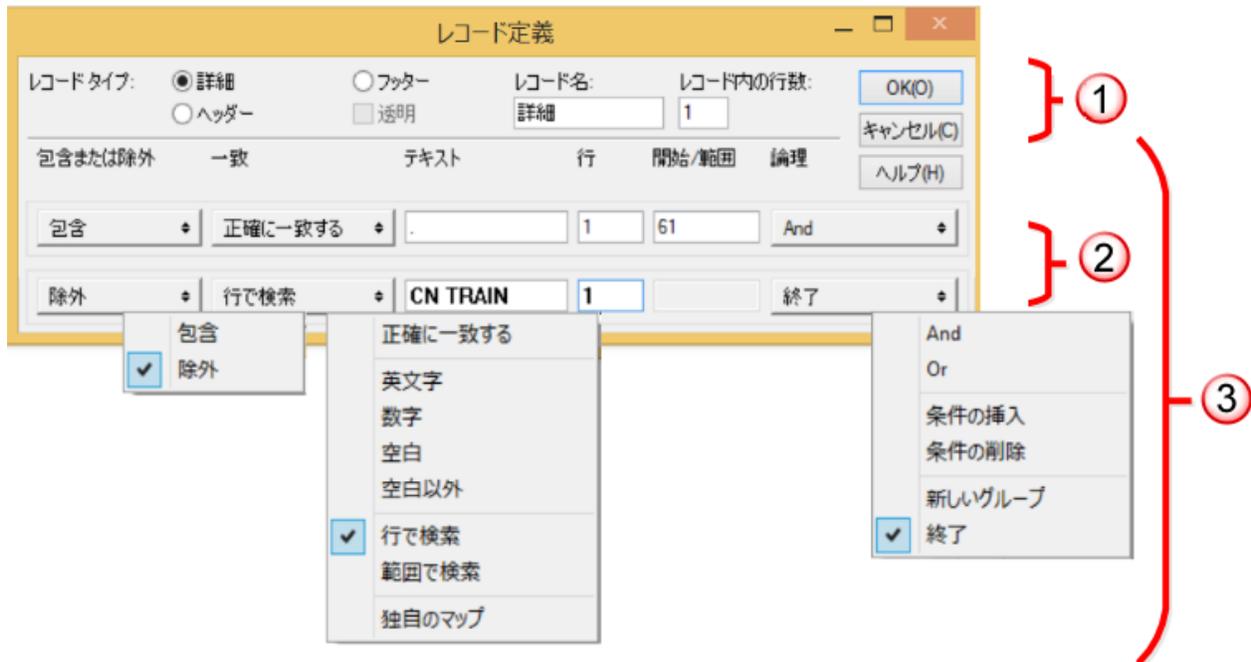
[レコード定義]ダイアログボックス

[レコード定義]ダイアログボックスでは、ソースファイルのレコードのセットを表す情報を指定します。次の2つの主要なタスクを実行できます。

- 詳細、ヘッダー、またはフッターというレコードが表すデータのタイプを指定する
- レコードのセットを取り込むために使用する条件を変更または作成する

何もない状態から開始するわけではありません。Analyticsでは既に一部の情報が指定されており、通常はこれを微調整するか追加する必要があります。

以下の図は、[レコード定義]ダイアログボックスで、2つの条件が設定され、3つのメニューが開いているところを示します。それらのメニューと、隣接するテキストフィールドを使用して、レコードのセットを取り込む条件を修正または作成します。



[レコード定義]ダイアログボックスには次の要素があります。

1. レコードタイプの指定
2. 個別の条件
3. 条件ビルダー

次の表では、[レコード定義]ダイアログボックスの各項目の目的が説明されています。

項目名	目的
レコードタイプ: <ul style="list-style-type: none"> ○ 詳細 ○ ヘッダー ○ フッター 	<p>詳細、ヘッダー、またはフッターというレコードが表すデータのタイプを指定します。</p> <ul style="list-style-type: none"> ○ レコード明細 - ファイルの主要な情報 たとえば、期限の過ぎた請求書を一覧表示するファイルでは、請求書のエントリが詳細レコードになります。詳細レコードのセットはファイルで1つだけ定義できます。 ○ ヘッダーレコード - 明細レコードのブロックまたはサブセットの上に表示される識別情報 たとえば、各顧客の会計情報(ヘッダーレコード)に続いて、各顧客の未払いの請求書(詳細レコード)の一覧を記載するファイルがあるかもしれません。必要であれば、2つ以上のヘッダーレコードセットを定義することができます。 ○ フッターレコード - 明細レコードのブロックまたはサブセットの下に表示される情報 たとえば、各顧客の未払い請求書の小計の一覧ファイルがあります(フッターレコード)。必要であれば、2つ以上のフッターレコードセットを定義することができます。 <p>メモ ヘッダーまたはフッターデータは、最初は、データ定義ウィザードの個別のレコードのように処理されますが、結果のAnalyticsテーブルでは、このデータは繰り返される値を含む1つ以上の追加フィールドになり、詳細レコードに追加されます。</p>
透明	ヘッダーレコードが複数行の詳細レコードに分割されないことを指定します。

項目名	目的
(ヘッダーレコードにのみ適用)	ヘッダーレコードがソースファイルの複数行の詳細レコードを分割する場合(改ページで発生する場合があります)、 透明]を選択すると、結果のAnalyticsテーブルの詳細レコードが統合されます。
レコード名	<p>データ定義ウィザードの左端の列に表示されるデフォルトレコード名をカスタマイズできます。</p> <p>複数のヘッダーまたはフッターレコードを作成する場合、デフォルト名のカスタマイズが便利なときがあります。値はデータ定義ウィザードにのみ表示され、結果のAnalyticsテーブルには表示されません。</p>
レコード内の行数	<p>ソースファイルの単一のレコードを構成する行数を指定します。</p> <p>たとえば、ソースファイルの各詳細レコードが単一の行に表示される場合は、値は"1"でなければなりません。各詳細レコードが3行にまたがる場合は、値は"3"でなければなりません。</p>
包含はまたは除外 (条件ビルダーの一部)	<p>条件に一致するレコードをレコードのセットに含めるかセットから除外するかどうかを指定します。</p> <p>このメニューには次のオプションがあります。</p> <ul style="list-style-type: none"> ○ 包含 - 条件に一致するレコードを含めます ○ 除外 - 条件に一致するレコードを除外します
一致 (条件ビルダーの一部)	<p>ファイルのレコードのセットを一意に識別するために使用する方法または使用する文字のタイプを指定します。</p> <p>このメニューには次のオプションがあります。</p> <ul style="list-style-type: none"> ○ 正確に一致する - 一致するレコードには、テキスト]フィールドにある文字または文字列が、レコードの指定された行の指定された開始位置に含まれている必要があります。 ○ 英文字 - 一致するレコードには、レコードの指定された行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の英文字を含んでいる必要があります。 ○ 数字 - 一致するレコードには、レコードの指定された行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の数字を含んでいる必要があります。 ○ 空白 - 一致するレコードには、レコードの指定された行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の空白スペースを含んでいる必要があります。 ○ 空白以外 - 一致するレコードには、レコードの指定された行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の空白以外の文字(特殊文字を含む)を含んでいる必要があります。 ○ 行で検索 - 一致するレコードには、テキスト]フィールドにある文字または文字列が、レコードの指定された行のどこかに含まれている必要があります。 ○ 範囲で検索 - 一致するレコードには、テキスト]フィールドにある文字または文字列が、レコードの指定された行の指定された範囲のどこかに含まれている必要があります。 ○ 独自のマップ - 一致するレコードには、テキスト]フィールドにある文字のパターンと一致する文字が、レコードの指定された行の指定された開始位置に含まれている必要があります。 <p>カスタムマップ]オプションは、MAP()関数と同じ構文を使用します。</p>
テキスト (条件ビルダーの一部)	<p>正確に一致する]、行で検索]、または範囲で検索]の場合、ファイルのレコードのセットを一意に識別する文字または文字列を指定します。</p> <p>独自のマップ]の場合、ファイルのレコードのセットを一意に識別する文字パターンを指定します。</p> <p>このフィールドは、ほかの一致]オプションでは無効です。</p>
直線 (条件ビルダーの一部)	<p>条件が適用されるレコードの行を指定します。</p> <p>たとえば、独自のマップを使用して郵便番号と一致させ、郵便番号が3行レコードの3行目に表示される場合は、行]で"3"を指定する必要があります。</p>

項目名	目的
	単一行のレコードの場合、値は常に"1"です。
開始/範囲 (条件ビルダーの一部)	次のいずれかを指定します。 <ul style="list-style-type: none"> ◦ 条件と比較するレコードの開始バイト位置 ◦ 条件と比較するレコードのバイトの範囲 ソースファイル内で位置または範囲を強調表示することで、 開始/範囲 フィールドの値を自動的に設定することができます。位置または範囲の番号を手動で入力することもできます。範囲の場合、「 開始位置のバイト:終了位置のバイト 」という構文を使用します。
論理 (条件ビルダーの一部)	条件を追加または削除し、条件間に論理関係を指定できます。最大 8 つの条件を追加できません。 <p>このメニューには次のオプションがあります。</p> <ul style="list-style-type: none"> ◦ And - 論理 AND を使用して別の条件を追加します ◦ Or - 論理 OR を使用して別の条件を追加します ◦ 条件の挿入 - 適用される条件の下に空の条件を挿入します 条件は最初に論理 AND で挿入されます。論理 OR に変更することができますが、挿入された条件の値を指定した後に限られます。 <ul style="list-style-type: none"> ◦ 条件の削除 - 適用される条件を削除します ◦ 新しいグループ - 個別の条件グループを作成します 新しいグループ オプションでは、複数の条件グループを作成し、別の論理ブロックとして動作させることができます。グループは、論理 OR または論理 AND を使用して、相互に関連付けられます。 <ul style="list-style-type: none"> ◦ 終了 - 最終条件として条件を指定します 条件の 終了]を選択すると、他のグループを含む後続の条件がすべて削除されます。 <div style="border-left: 2px solid green; padding-left: 10px;"> <p>ヒント</p> <p>条件に欠落した値がある場合は、論理]ボタンが応答しなくなることがあります。論理]ボタンを再有効化するには、欠落した値を入力します。</p> </div>

印刷イメージまたは PDF ファイルで整列していないフィールドの定義

以下の手順は、印刷イメージまたは PDF ファイルで整列していないフィールドを定義する手順について説明します。この方法では、**データ定義ウィザード**で整列していないフィールドをおおまかに定義するだけで構いません。Analytics にデータが取り込まれたら、Analytics 関数を使用する 1 つ以上の演算フィールドを作成し、すべての値の調整など、フィールドのデータを正確に形成する必要があります。

演算フィールドの作成については、「条件付き演算フィールドを定義する」ページ 806を参照してください。

印刷イメージまたは PDF ファイルで整列していないフィールドを定義するには

1. 通常の方法で、整列しているフィールドを定義します。

最低限、レコードのセットの定義として、1 つ以上のフィールドを定義する必要があります。データの多くが整列していない場合でも、ファイルのすべてのレコード データを取り込むレコードのセットを作成してみてください。

レコードのセットの定義の詳細については、「クイックスタート:印刷イメージまたは PDF ファイルを定義する方法」ページ 295を参照してください。

2. 次の方法の 1 つ以上を使用して、整列していないフィールドを定義します。

- **整列していないフィールドの左端と右端の文字を取り込むのに十分な長さがあるフィールド定義を作成する**

Analytics では、ALLTRIM() 関数を使用してフィールドを調整します。

- **必要に応じて、重複フィールドの定義を作成する**

場合によっては、データが整列していないことが原因で、2 つの異なるフィールドの値が重なり合ってしまうことがあります。各フィールドを個別に定義し、各フィールドに属するすべての値が該当するフィールド定義によって取り込まれるようにします。

同じバイト位置は、1 つのフィールド定義の最後と他のフィールド定義の最初に含まれます。共有されたバイト位置では、一貫した構造のデータを取り込んでみてください。たとえば、一部のレコードに単一の文字列があり、別のレコードには別々の 2 つの文字列があるか文字がないものではなく、1 つの完全な文字列を使用します。

Analytics では、ALLTRIM()、REGEXREPLACE()、および SPLIT() 関数を使用して、フィールドを調整し、不要な文字を取り除きます。

- **複数の整列していないフィールドを含めることができる、単一の長いフィールド定義を作成します。**

レコードのセットの大部分が整列していない場合は、長いフィールド定義を使用して、レコードデータの問題がある部分を取り込むことができます。整列していないデータのブロックの左端と右端の文字を取り込むのに十分な長さがあるフィールドである必要があります。

整列していないデータがレコードのセットの別の部分で発生する場合は、さらに別の長いフィールド定義を作成し、整列していない各部分を取り込みます。

メモ

1つのフィールドで、その値に含まれる単語数が異なる場合は、これらの値に対して個別のフィールド定義を作成するか、これらの値が複数の整列していないフィールドを含む長いフィールド定義の末尾で最後のフィールドを表すようにしてください。サンプル "Inventory.pdf" の "Product Description" フィールドは、値に異なる単語数が含まれるフィールドの例です。

Analytics では、ALLTRIM()、REGEXREPLACE()、および SPLIT() 関数を使用して、単一のフィールドを個別の整列したデータ要素に分割します。

3. ファイル全体をチェックし、フィールド定義の水色の強調表示以外で整列していないフィールドの値がないことを確認します。必要に応じて、フィールド定義の長さを調整します。
4. 文字データ型が、**フィールド定義** ダイアログボックスの各フィールド定義で指定されていることを確認します。
5. 通常どおり、インポート処理を完了します。

フィールドプロパティの編集 ページで、ASCII または UNICODE データ型がすべてのフィールドで指定されていることを確認します。

詳細については、"印刷イメージファイルの定義とインポート" ページ 300と"PDFファイルの定義とインポート" ページ 308を参照してください。

6. 整列していないフィールドで、重複するフィールドのデータがない場合は、次の式を使用する演算フィールドを Analytics で作成します。

```
ALLTRIM(整列していないフィールド名)
```

最初と最後のスペースがフィールドから削除され、フィールドのすべての値が調整されます。

7. 整列していないフィールドで、重複するフィールドのデータを含む場合は、Analytics で次の手順を実行します。
 - a. 次の式を使用する最初の演算フィールドを作成し、フィールド値と不要な文字の間の1つ以上のスペースを単一のスペースで置換します。

```
ALLTRIM(REGEXREPLACE(整列していないフィールド名, "\s+", " "))
```

この式は、整列していないフィールドの先頭と末尾のスペースも除去します。

- b. 次の式のいずれかを使用する2番目の演算フィールドを作成して、フィールド値を抽出し、不要な文字を破棄します。
 - 不要な文字がフィールドの末尾にある場合は、次の式を使用します。

```
SPLIT(初期演算フィールド名, " ", 1)
```

- 不要な文字がフィールドの先頭にある場合は、次の式を使用します。

```
SPLIT(初期演算フィールド名," ", 2)
```

ヒント

不要な文字がフィールドの末尾にあたり、先頭にあたりする場合、あるいは一部のレコードにのみ現れる場合は、整列していないフィールドの別の部分に別のSPLIT()式を適用する、条件付き演算フィールドを作成する必要があります。たとえば、条件 RECNO() > 100 では、先頭から 100 件のレコードを超えるレコードにのみ、ある特定の式を適用できます。

詳細については、「条件付き演算フィールドを定義する」ページ 806を参照してください。

8. 複数の整列していないフィールドを含めることができる、単一の長いフィールド定義については、Analytics で次の手順を実行します。
 - a. 次の式を使用する最初の演算フィールドを作成し、データ要素間の 1 つ以上のスペースを単一のスペースで置換します。

```
ALLTRIM(REGEXREPLACE(長いフィールド名, "\s+", " "))
```

この式は、長いフィールドの先頭と末尾のスペースも除去します。

ヒント

式で OMIT() 関数を使用すると、一貫性がなく出現して後続の処理を複雑にするデータの部分を削除するのに便利なことがわかるでしょう。たとえば、OMIT (ALLTRIM(REGEXREPLACE(長いフィールド名, "\s+", " ")), "-") は上記の式と同じ処理を実行することに加え、ハイフンも削除します。

- b. 次の式を使用する 2 番目の演算フィールドを作成し、最初のデータ要素を抽出します。

```
SPLIT(初期演算フィールド名," ", 1)
```

- c. 同じ式のバリエーションを使用して、必要な数の演算フィールドを作成し、すべてのデータ要素を抽出します。

例：

```
SPLIT(initial_computed_field_name," ", 2)
SPLIT(初期演算フィールド名," ", 3)
```

連続するデータ要素を指定するには、SPLIT() 関数のセグメントパラメーターの数を増やします。

メモ

複数の単語を含むフィールド値(サンプル "Inventory.pdf" の "Product Description" フィールドなど) の場合、この方法によって、各単語を別々のフィールドに分離できます。必要に応じて、この別々のフィールドを連結し、値を返すこともできます。詳細については、"フィールドの連結" ページ 233を参照してください。

9. すべてのデータ要素を個別のフィールドに抽出した後、次の手順を実行して、数値および日付時刻データを該当するデータ型に変換します。
 - a. 数値フィールドについては、次の式を使用する演算フィールドを作成します。

```
VALUE(フィールド名, 小数点以下桁数)
```

詳細については、"VALUE() 関数" ページ 2559を参照してください。

- b. 日付フィールドについては、次の式を使用する演算フィールドを作成します。

```
CTOD(フィールド名, "日付書式")
```

詳細については、"CTOD() 関数" ページ 2200を参照してください。

日付時刻または時刻値を変換するには、CTODT() または CTOT() 関数を使用します。

ヒント

関数を適用すると同時にデータ型を変換して、整列していないデータを修正すると、作業を軽減し、作成する演算フィールドを減らすことができます。例：

```
VALUE(ALLTRIM(整列していないフィールド名), 2)
```

10. 必要な演算フィールドをすべて作成したら、テーブルビューに追加します。
最初の演算フィールドをビューに追加する必要はありません。整列していないフィールドまたは長いフィールドをビューから削除できます。
詳細については、"ビューに列を追加する" ページ 849と"ビューからの列の削除" ページ 850を参照してください。

印刷イメージまたは PDF データのサブセットの定義とインポート

データが整列していないことが原因で、印刷イメージまたは PDF ファイルの完全なレコードのセットを定義することが難しいか不可能である場合は、ファイルからデータの複数のサブセットを定義してインポートできます。その後、Analytics で結果の Analytics テーブルを追加して、完全なデータセットを作成できます。

データ定義ウィザードのソースファイルに、レコードのブロックが含まれており、その中ですべてのフィールドが各ブロック内に配置される場合、この方法が最適です。データがより不規則に配置されている場合は、"印刷イメージまたは PDF ファイルで整列していないフィールドの定義" ページ 326を参照してください。

ヒント:

PDF 定義の場合、ページごとに PDF ファイルを解析することができます。場合によっては、改ページをまたいでデータのずれが発生します。ページサイズのデータのサブセットを使用すると、整列の問題を解決できる場合があります。

印刷イメージまたは PDF データのサブセットを定義してインポートするには

1. 通常の方法で、定義とインポート処理を実行します。ただし、次の違いがあります。

- **同じファイルを複数回定義してインポートする**

各繰り返しで、レコードの異なるサブセットを定義します。各サブセットのフィールドは、内部的に整列している必要があります。

レコードのサブセットは連続している必要はありません。たとえば、ファイルの開始および終了ではレコードのフィールドが互いに整列しているが、ファイルの中央にあるフィールドとは配置がずれている場合があります。

- **各サブセットに含まれるレコードを追跡する方法を考える**

意図せずに同じレコードを複数回取り込んだ場合は、Analytics で再作成されたデータセットから重複レコードを削除できます。詳細については、"重複の削除" ページ 1272を参照してください。

- **各繰り返しで、データ構造が一貫していることを確認する**

名前、長さ、データ型、および対応するフィールドの順序が一貫していることを確認します。データ構造のこの一貫性を維持することで、結果 Analytics テーブルの追加が大幅に簡単になります。

ヒント:

最初のサブセットをインポートした後、Analytics で結果テーブルを開き、コマンドラインで「**DISPLAY**」と入力して、テーブルレイアウトのデータ構造を表示します。レコードとフィールドの後続のサブセットを作成するためのガイドとして、その表示されたテーブルレイアウト情報を使用します。

作業を保存するには、レコードのサブセットを定義およびインポートするときに、Analytics における汎用フィールド名 ("Field_1"、"Field_2" など) を使用します。Analytics でデータセットを再作成した時点で、再作成されたテーブルのすべてのフィールド名を変更できます。

2. 各 Analytics データファイルと各 Analytics テーブルレイアウトを保存するときには、それぞれの名前の末尾に増分数字を付け、既に作成されたテーブルが上書きされないようにします。たとえば、"Table_1.fil"、"Table_2.fil" などです。
3. ソースファイルのすべてのレコードを定義およびインポートしたら、複数の Analytics テーブルを追加します。

詳細については、「データの抽出と追加」 ページ 940を参照してください。

複数行のレコードおよびフィールドの操作

ソースファイル内の複数行にわたるレコードやフィールドのデータを定義することができます。例：

- 複数の行に配置された住所データまたはコメント データ
- 単一フィールドに積み重ねられた異なる型のデータ
- 複数行フィールドの値に含まれる行数が異なるデータ

次のセクションでは、このようにデータが配置されたファイルを定義する方法について説明します。

複数行のレコードと複数行のフィールド

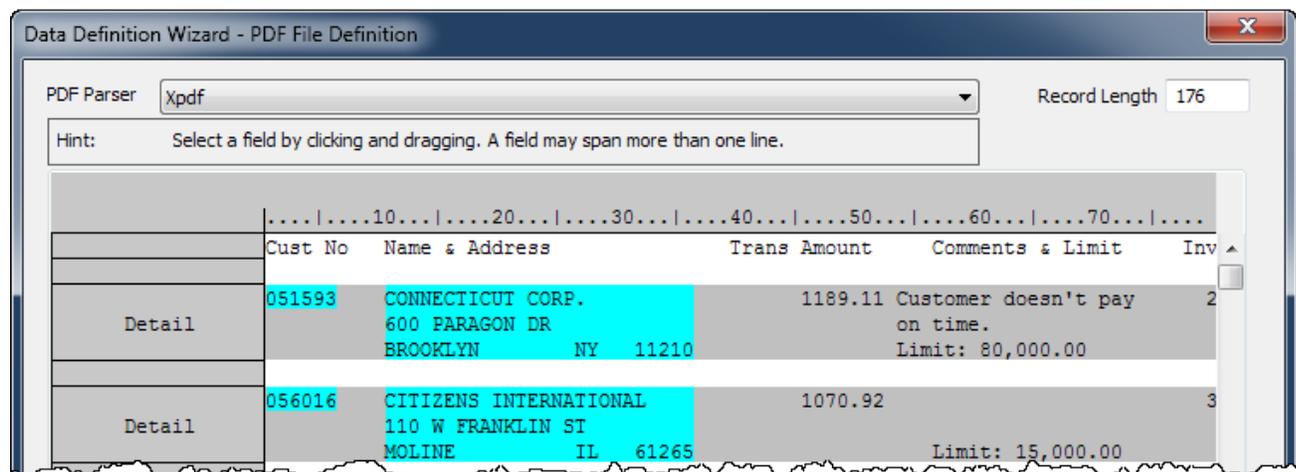
ソースファイルで、複数行のレコードには、すべてが同じレコードに属する2行以上のデータが含まれています("1行のフィールドと複数行のフィールドがある複数行のレコード" 下を参照してください。)

複数行のレコードのフィールドには、複数行のフィールドがある場合とない場合があります。たとえば、2行のレコードは、印刷イメージまたはPDFレポートのレイアウトで2番目の行に折り返された、一連の1行のフィールド値である可能性もあります。

複数行のフィールドには、ソースファイルで2行以上にまたがるフィールド値が含まれます。たとえば、複数行に配置された住所、複数行のテキストを含むコメント フィールドなどです。フィールドが複数行の場合は、フィールドを含むレコードも複数行である必要があります。

1行のフィールドと複数行のフィールドがある複数行のレコード

以下の例では、3行で構成される1件のレコードを示します。このレコードには、1行の"Cust No" フィールドと3行の"Name & Address" フィールドが含まれます。

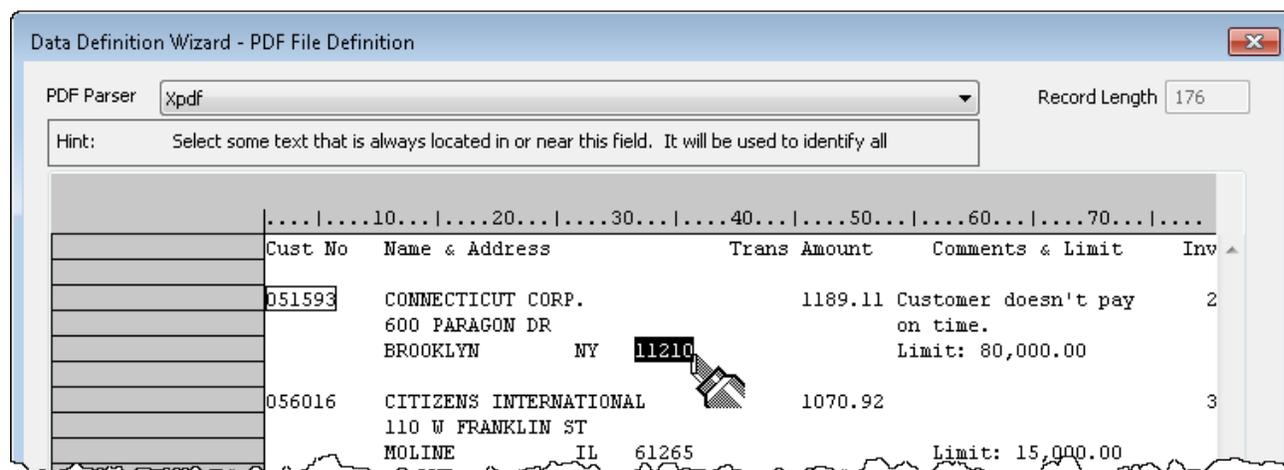


複数行のレコードの定義

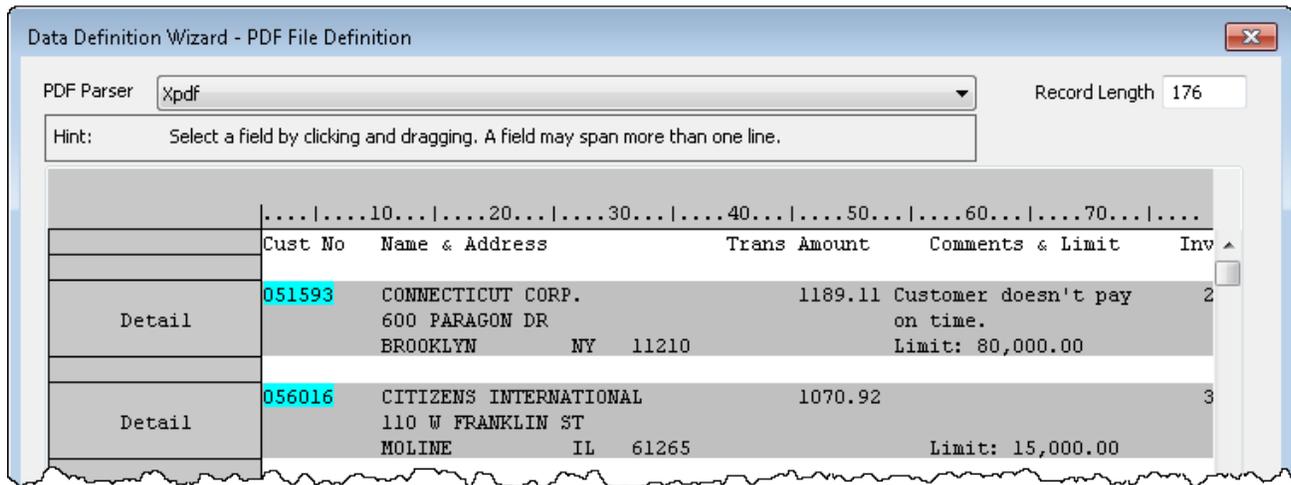
これらの方法のいずれかを使用して、複数行のレコードを定義できます。

- レコードの最初の行の初期データ値と、レコードの最後の行の一意の値を選択します。
"レコードの最初の行の初期データ値と、最後の行の一意の値" 下では、顧客番号が最初の行で選択され、郵便番号が最後の行で選択されています。"定義された複数行のレコード" 見開きページは結果の複数行レコードを示します。
- レコードの最初の行を定義し、その後、[レコード定義]ダイアログボックスの[レコード内の行数]設定で適切な行数を指定し、レコード定義を編集します。レコードの最後の行で一意の値を指定できない場合は、この方法を使用できます。

レコードの最初の行の初期データ値と、最後の行の一意の値



定義された複数行のレコード



複数行のレコードのフィールドの定義

複数行のレコードのフィールドを定義するときには、次のオプションがあります。

- 単一行のフィールドとして単一の行を占める値を定義できます。

"単一行のフィールドと複数行のフィールド" 次のページでは、顧客番号、市、州、郵便番号、取引金額、および上限の値がすべて単一行のフィールドとして定義されています。

単一行のフィールドを定義するには、フィールドの値のいずれかの単一のインスタンスを選択します。

- 2つの方法のいずれかで複数行を示す値を定義できます。
 - 単一の行のデータを含む複数のフィールド

"単一行のフィールドと複数行のフィールド" 次のページでは、顧客名と住所がこの方法で定義されています。

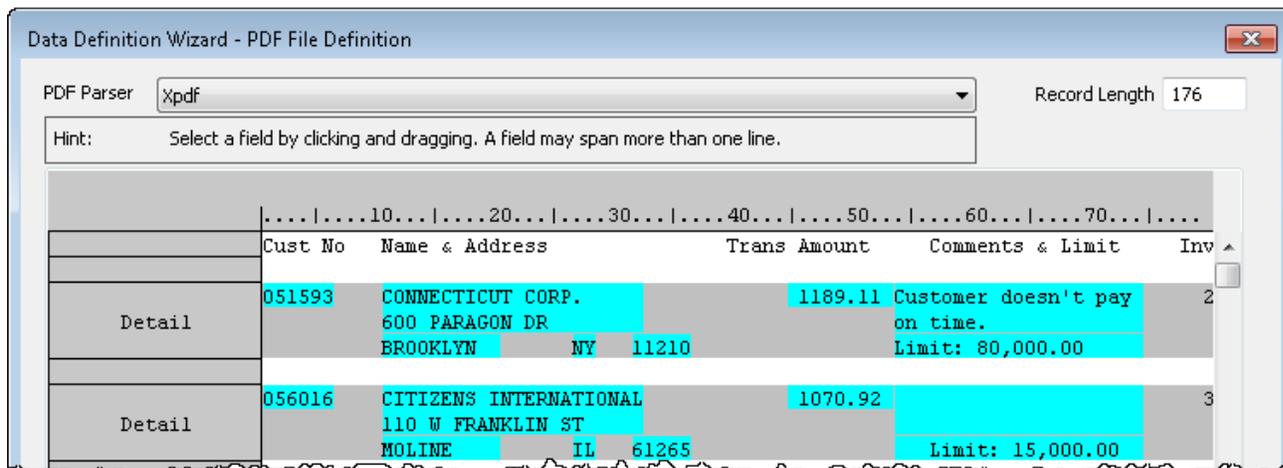
複数行のフィールド、単一行のフィールドを定義するには、複数行データの単一のインスタンスを選択します。デフォルトでは、Analytics は各行に個別のフィールドを作成します。各フィールドには、フィールド名の末尾に増分数字が追加されます。

- すべての行のデータを含む単一のフィールド

"単一行のフィールドと複数行のフィールド" 次のページでは、コメント値は単一のフィールドとして定義されます。

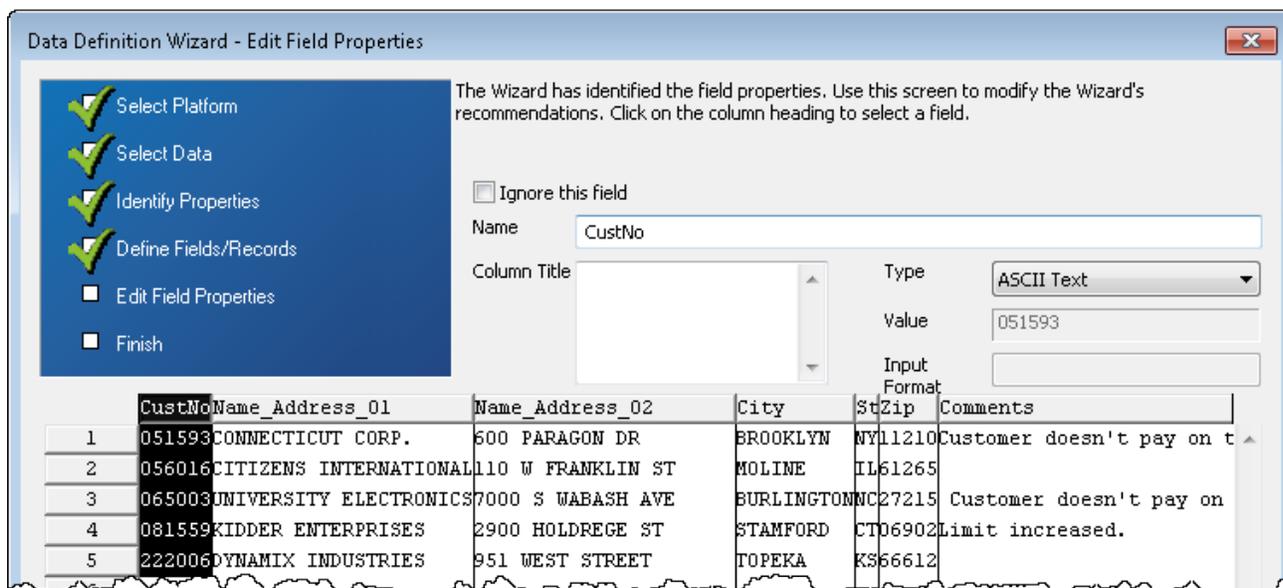
すべての行のデータを含む単一のフィールドを定義するには、複数行データの単一のインスタンスを選択します。[フィールド定義]ダイアログボックスの[詳細オプション]で、[単一フィールドに変換する]を選択します。

単一行のフィールドと複数行のフィールド



結果の Analytics テーブルのフィールドのプレビュー

以下の例は、"単一行のフィールドと複数行のフィールド" 上の定義済みフィールドが結果の Analytics テーブルに表示される方法を示します。



高さが異なるフィールドの定義

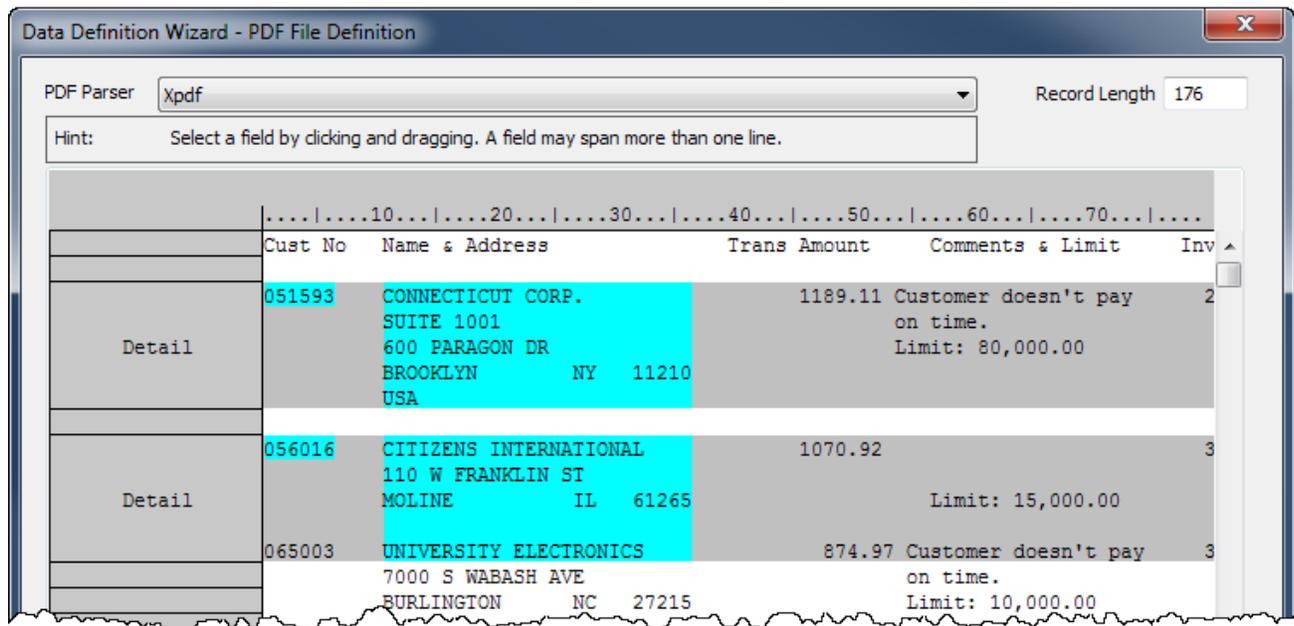
[フィールド定義] ダイアログボックスの [空白行で終わる] 設定を使用して、値が占める行数が異なる複数行のフィールドを定義できます。この設定は、空白の行があるときに値が終了することを指定します。1

つ以上の空白行が複数行フィールドの各値を区切る場合にのみ、この機能は動作します。

フィールドの境界とフィールド値の不一致

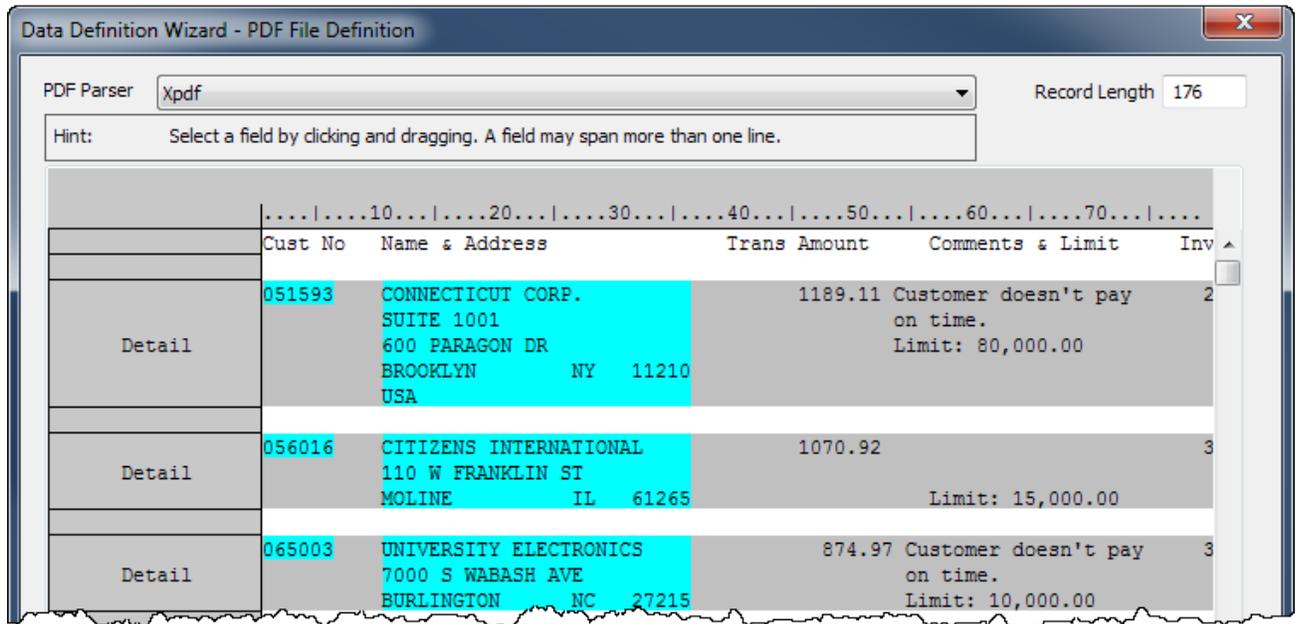
以下の例は、フィールド値の高さが異なるときに発生する可能性がある問題を示します。

フィールドの高さが"5"に設定され、"Name & Address"フィールドの最初の値のすべての5行が取り込まれます。2番目の値には3行しかないため、5行のフィールドの高さではデータを取り込みすぎてしまい、2番目の値すべてと3番目の最初の行取り込むことになります。結果として、フィールドの境界とフィールド値の間に不一致が発生し、レコードの境界も不一致になります。



フィールド値に合わせてサイズ変更されたフィールドの境界

以下の例は **空白行で終わる**を選択することでこの問題を解決する方法を示します。フィールドとレコードの境界のサイズが動的に変更され、各フィールド値が占める行数に合うようになります。



ACCPAC マスター ファイルのインポート

ACCPAC マスター ファイルを定義およびインポートして、Analytics テーブルを作成できます。このファイルは、ローカルコンピューターまたはネットワークドライブ上、あるいは Analytics サーバー(インストールされている場合)上のフォルダーにあります。

1. **ファイル > 新規作成 > テーブル**の順にクリックします。

データ定義ウィザードに表示される最初のページは構成によって異なります。Analytics サーバーとの統合が有効になっている場合は、**データソースのプラットフォームの選択**ページが表示され、そうでない場合は **ローカルデータソースの選択**ページが表示されます。

2. ファイルの場所を選択するために、次の手順のいずれかを実行します。

- **データソースのプラットフォームの選択**ページが表示されていて、ファイルを定義するのに Analytics を使用したい場合は、**ローカル**を選択して **次へ**をクリックします。**ローカルデータソースの選択**ページで **ファイル**を選択し、**次へ**をクリックします。
- **データソースのプラットフォームの選択**ページが表示されていて、ファイルを定義するのに ACL サーバーを使用したい場合は、**Analytics サーバー**を選択し、ドロップダウンリストから Windows サーバー プロファイルを選択して **次へ**をクリックします。**ACL サーバーデータソースの選択**ページで **プラットフォーム**を選択し、**次へ**をクリックします。
- **ローカルデータソースの選択**ページが表示されている場合は、**ファイル**を選択して **次へ**をクリックします。

3. **定義するファイルの選択**ダイアログボックスで、Analytics テーブルの作成元とするファイルを探して選択し、**開く**をクリックします。
4. **文字セット**ページで、正しい文字セット オプションが選択されていることを確認し、**次へ**をクリックします。
5. **ファイル形式**ページで、**ACCPAC マスター ファイル** オプションが選択されていることを確認し、**次へ**をクリックします。
6. **フィールドの識別**ページで、次の操作を行って、レコードで識別されるフィールドを変更します。
 - 既存のフィールド区切りを削除するには、削除したいフィールド区切り線をクリックします。
 - 既存のフィールド区切りを移動するには、フィールド区切り線をクリックして新しい位置までドラッグします。
 - 新しいフィールド区切りを作成するには、フィールド区切りを追加したい位置のグリッドをクリックします。
7. レコード内のすべてのフィールドが識別されたら、**次へ**をクリックします。
8. **フィールド プロパティの編集**ページでは、プレビューテーブルで変更したいフィールドの列見出しを選択し、以下のプロパティを更新することにより、各フィールドの名前やプロパティを変更することができます。
 - **このフィールドを無視する** - このフィールドを Analytics テーブルに含めたくない場合は、このチェックボックスをオンにします。

- **名前]** - テーブルレイアウトのフィールド用に Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
 - **列見出し]** - Analytics のデフォルトのビューに表示する列見出しを入力します。列見出しが指定されていない場合は、**名前]**の値が使用されます。
 - **型** - ドロップダウンリストから適切なデータ型を選択します。Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。
小数点以下の桁数]および **入力書式]**テキスト ボックスは、そのオプションの対象となるデータ型を選択したときに自動的に表示されます。
 - **値** - フィールドの最初の値を表示する読み取り専用プロパティです。この値は、編集作業によって更新されます。
 - **小数点以下の桁数(数値フィールドのみ)** - 小数点以下の適切な桁数を指定します。
 - **入力書式(日付時刻フィールドのみ)** - その日付に対応する書式を指定します。日付と時刻の書式の詳細については、"ソース データに含まれる日付および時刻データの書式" ページ 366を参照してください。
9. 変更したいフィールド プロパティの編集が完了したら、**次へ]**をクリックします。
 10. **最終]**ページで、新しい Analytics テーブルの設定を確認したら、**完了]**をクリックします。
 11. プロジェクトに追加する Analytics テーブルの名前について、デフォルトの名前をそのまま用いるか、または別の名前を入力して、**OK]**をクリックします。

dBASE 互換ファイルのインポート

dBASE ファイル形式の標準に準拠する任意のファイルをインポートし、Analytics テーブルを作成することができます。データ定義ウィザードは、dBASE 互換ファイルのデータを分析し、この情報を使用して自動的に Analytics テーブルレイアウトを定義します。

dBASE ファイル形式

dBASE ファイル形式は、さまざまなデータベースやビジネスアプリケーションでオプションとして提供されているため、多様なデータソースからデータにアクセスするために使用できます。エクスポートされたファイルが dBASE 標準に準拠している場合は、これが、Analytics を使用してデータにアクセスする便利な方法です。ファイルには必要なフィールド定義がすべて含まれています。エクスポートされたデータが dBASE 標準に完全には準拠していない場合、Analytics はファイルを定義できない可能性があります。

Record_Deleted フィールド

dBASE ファイルには必ず、削除されたレコードを追跡するために使用される "Record_Deleted" というフィールドが含まれています。Analytics は対応するフィールドをテーブルレイアウトに定義しますが、これは必須ではないので、デフォルトビューからこのフィールドを削除した後、テーブルレイアウトから削除してもかまいません。

dBASE ファイル条件

dBASE ファイルにエクスポートされたデータは、Analytics にインポートするには、以下の条件を満たしている必要があります。

列名	列名には最大長 10 文字を指定できます。各フィールドの最初の 10 文字は一意である必要があります。そうでないと、重複するフィールドをエクスポートできません。
フィールド名	フィールド名が最初の行に指定されており、データは 2 行目から始まる必要があります。
データ型	各列は、1 つのデータ型の値のみを含んでいる必要があります。たとえば、フィールドの最初の値に文字データが含まれていたら、そのフィールドは文字データとしてエクスポートされます。
フィールドと数値	数字のみを含んでいるフィールドは、数値データとしてエクスポートされます。場合によっては、これは Analytics で間違ったデータ型を持つフィールドになります。たとえば、請求書番号は数値ですが、これは多くの場合文字フィールドに格納されます。この場合は、 テーブルレイアウト ダイアログボックスでデータ型を変更する必要があります。

手順

1. **[ファイル > 新規作成 > テーブル]**の順にクリックします。

データ定義ウィザードに表示される最初のページは構成によって異なります。Analytics サーバーとの統合が有効になっている場合は、**[データソースのプラットフォームの選択]**ページが表示され、そうでない場合は **[ローカルデータソースの選択]**ページが表示されます。

2. 次の手順のいずれかを完了し、ファイルの場所を選択します。
 - **[データソースのプラットフォームの選択]**ページが表示されていて、ファイルを定義するのに Analytics を使用したい場合は、**[ローカル]**を選択して **[次へ]**をクリックします。**[ローカルデータソースの選択]**ページで **[ファイル]**を選択し、**[次へ]**をクリックします。
 - **[データソースのプラットフォームの選択]**ページが表示されていて、ファイルを定義するのに ACL サーバーを使用したい場合は、**[Analytics サーバー]**を選択し、ドロップダウンリストから Windows サーバー プロファイルを選択して **[次へ]**をクリックします。**[ACL サーバーデータソースの選択]**ページで **[プラットフォームファイル]**を選択し、**[次へ]**をクリックします。
 - **[ローカルデータソースの選択]**ページが表示されている場合は、**[ファイル]**を選択して **[次へ]**をクリックします。
3. **[定義するファイルの選択]**ダイアログボックスで、Analytics テーブルの作成元とするファイルを探して選択し、**[開く]**をクリックします。

dBASE 互換ファイルのファイル拡張子は **.dbf** です。
4. **[ファイル形式]**ページで、**[dBASE 互換ファイル]**オプションが選択されていることを確認し、**[次へ]**をクリックします。
5. **[最終]**ページで、新しい Analytics テーブルの設定を確認したら、**[完了]**をクリックします。
6. プロジェクトに追加する Analytics テーブルの名前を入力し、**[OK]**をクリックします。

SAP 監査形式ファイルのインポート

SAP 監査形式ファイル(旧称 SAP プライベート ファイル形式)を定義およびインポートして、Analytics テーブルを作成できます。SAP 監査形式ファイルのファイル拡張子は .dat です。

お使いのローカルコンピューターまたはネットワークドライブにある監査形式ファイルを定義およびインポートできます。

メモ

Diligent は、SAP システムに直接アクセスし、データを ACL Analytics にインポートするための 2 つのユーティリティを備えています。

- **SAP コネクター** - は、追加のサブスクリプションで使用可能な Analytics コネクターの 1 つです。
- **Direct Link** - は、Diligent から購入することができるオプションのアドオンです。

SAP AIS および SAP DART

SAP 監査形式ファイルは SAP データ抽出から生成されます。SAP AIS (Audit Information System) および SAP DART (Data Retention Tool) はいずれも、データ抽出を生成できる SAP ユーティリティです。

これらの SAP ユーティリティを使用して抽出を作成するときには、SAP ユーザーは、ファイル出力形式を指定できます。一部の Analytics 対応出力形式は、スプレッドシート、テキスト、dbf、および SAP 監査形式です。これらの抽出はデータ定義ウィンドウを使用して、Analytics にインポートできます。使用する最適な形式は SAP 監査形式です。これは、Analytics にインポートする作業が最も少なくともすむためです。

手順

1. **ファイル > 新規作成 > テーブル** の順にクリックします。

データ定義ウィザードに表示される最初のページは構成によって異なります。Analytics サーバーとの統合が有効になっている場合は、**データソースのプラットフォームの選択** ページが表示され、そうでない場合は **ローカルデータソースの選択** ページが表示されます。

2. 次の手順のいずれかを完了し、ファイルの場所を選択します。
 - **データソースのプラットフォームの選択** ページが表示されていて、ファイルを定義するのに Analytics を使用したい場合は、**ローカル** を選択して **次へ** をクリックします。**ローカルデータソースの選択** ページで **ファイル** を選択し、**次へ** をクリックします。
 - **データソースのプラットフォームの選択** ページが表示されていて、ファイルを定義するのに ACL サーバーを使用したい場合は、**Analytics サーバー** を選択し、ドロップダウンリストからサーバープロファイルを選択して **次へ** をクリックします。**ACL サーバーデータソースの選択** ページで **プラットフォーム** を選択し、**次へ** をクリックします。

- **ローカルデータソースの選択**]ページが表示されている場合は、**[ファイル]**を選択して **次へ]**をクリックします。
- 3. **定義するファイルの選択**]で、Analytics テーブルの作成元とするファイルを探して選択し、**開く]**をクリックします。
- 4. **文字セット**]ページで、正しい文字セット オプションが選択されていることを確認し、**次へ]**をクリックします。
- 5. **ファイル形式**]ページで、**SAP プライベート ファイル形式 / DART]**オプションが選択されていることを確認し、**次へ]**をクリックします。
- 6. **SAP プライベート ファイル形式**]ページで、フィールドの名前付けについて最適なオプションを選択します。
 - **ACL フィールド名に現地語のフィールドの説明を使う**-- 標準のドイツ語のフィールド名ではなく、SAP システム用に構成済みのローカライズされたフィールドの説明を使用するには、このオプションを選択します。Analytics テーブルを単一言語だけで使用する場合は、このオプションをお勧めします。
 - **ACL フィールド名に標準配布のSAPドイツ語略記を使う**-ドイツ語のフィールド名を使用したい場合、あるいは Analytics テーブルを複数の言語で使用する場合は、このオプションを選択します。
- 7. **次へ]**をクリックします。
- 8. **変換された SAP ファイルを別名で保存**]ダイアログボックスで、ファイル名を入力し、必要に応じて Analytics データファイルのフォルダーの場所を変更し、**保存]**をクリックします。
- 9. **最終]**ページで、新しい Analytics テーブルの設定を確認したら、**完了]**をクリックします。
- 10. プロジェクトに追加する Analytics テーブルの名前を入力し、**OK]**をクリックします。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

XML ファイルのインポート

XML ファイルを定義およびインポートして、Analytics テーブルを作成できます。**データ定義ウィザード**を使用すると、インポートする XML 要素を選択したり、結果の Analytics テーブルの構造を設定したり、インポートする要素の列見出しやデータ型をカスタマイズしたりすることができます。

メモ

場合によっては、結果の Analytics テーブルの 1 つ以上のデータ定義を調整し、ビューに表示されるデータが正確にソース XML ファイルのデータを反映させなければならないことがあります。Analytics テーブルレイアウトでフィールド定義を調整します。

Analytics は XML ファイルに含まれる正確な未加工データをインポートし、テーブルレイアウトではこのソースデータを確認できます。場合によっては、テーブル定義およびインポート処理中に作成されたフィールド定義がソースデータを誤って解釈し、後続の調整で定義が必要になることがあります。たとえば、数値フィールドが誤って日付フィールドとして解釈され、数値ではなく日付が最初にビューに表示される場合があります。

1. **ファイル > 新規作成 > テーブル** の順にクリックします。
2. **データソースのプラットフォームの選択** ページが表示されている場合は、**ローカル** を選択し、**次へ** をクリックします。
3. **ローカルデータソースの選択** ページで **ファイル** を選択し、**次へ** をクリックします。
4. **定義するファイルの選択** ダイアログボックスで、Analytics テーブルの作成元とする XML ファイルを探して選択し、**開く** をクリックします。

XML ファイルのファイル拡張子は .xml です。

5. **ファイル形式** ページで、**XML ファイル** オプションが選択されていることを確認し、**次へ** をクリックします。
6. **XML データ構造の選択** ページを使用して、結果の Analytics テーブルに含める 1 つまたは複数の XML データ構造を選択します。
 - a. データ構造を選択するには、関連付けられた XML 要素の名前をクリックします。
 - b. **追加** をクリックして、データ構造を **プレビュー** ペインに追加します。

メモ

大きい XML ファイルのデータをプレビューすると、速度が低下することがあります。このため、**自動プレビュー** オプションは、2 GB を超えるサイズの XML ファイルの場合には、自動的にオフにされます。

- c. Analytics テーブルに含めるすべてのデータ構造を選択して追加します。
- d. 必要に応じて、**プレビュー** ペインでデータ構造を選択し、**削除** をクリックして削除します。
- e. **次へ** をクリックします。

XML データ構造は XML 要素および属性のコレクションです。詳細については、「XML データ構造の選択」ページ 346 を参照してください。

7. **[XML 要素の選択]** ページで、XML 要素と属性を選択して微調整し、**[次へ]** をクリックします。
詳細については、"XML 要素の選択および構成" ページ 347を参照してください。
8. **[データのプレビュー]** ページで、必要に応じて、フィールドの名前またはプロパティを修正します。
フィールド プロパティを修正するには、ページの下半分にあるプレビュー テーブルで適切な列見出しを選択し、以下のプロパティを更新します。
 - **[名前]** - テーブルレイアウトのフィールド用に Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
 - **[列見出し]** - Analytics のデフォルトのビューに表示する列見出しを入力します。列見出しが指定されていない場合は、**[名前]** の値が使用されます。
 - **[型]** - ドロップダウン リストから適切なデータ型を選択します。Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。
[小数点以下の桁数] および **[入力書式]** テキスト ボックスは、そのオプションの対象となるデータ型を選択したときに自動的に表示されます。
 - **[値]** - フィールドの最初の値を表示する読み取り専用プロパティです。この値は、編集作業によって更新されます。
 - **[小数点以下の桁数(数値フィールドのみ)]** - 小数点以下の適切な桁数を指定します。
 - **[入力書式(日付時刻フィールドのみ)]** - その日付に対応する書式を指定します。日付と時刻の書式の詳細については、"ソース データに含まれる日付および時刻データの書式" ページ 366を参照してください。
9. **[次へ]** をクリックします。
10. **[データ ファイルを別名で保存]** ダイアログ ボックスで、Analytics データ ファイルのファイル名を入力し、必要であれば、そのファイルを保存する場所を変更して、**[保存]** をクリックします。
11. **[最終]** ページで、新しい Analytics テーブルの設定を確認したら、**[完了]** をクリックします。
12. プロジェクトに追加する Analytics テーブルの名前について、デフォルトの名前をそのまま用いるか、または別の名前を入力して、**[OK]** をクリックします。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字()を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

13. 新しい Analytics テーブルのデータを確認し、必要に応じてフィールド定義を更新します。
フィールドで誤ったデータ型が指定されている場合、データがビューに表示されなかったり、データが誤って解釈される可能性があります。たとえば、数値は日付として解釈される場合があります。
14. フィールド定義を更新するには、次の手順を実行します。
 - a. **[編集 > テーブルレイアウト]** の順にクリックします。
 - b. **[フィールド/式の編集]** タブで変更するフィールドをダブルクリックします。
 - c. フィールド定義に対して必要な変更を行い、**[入力を受け入れる]** をクリックします。
たとえば、**[型]** を **[日付時刻]** から **[数値]** に変更しなければならない場合があります。

XML データ構造の選択

ウィザードの **XML データ構造の選択** ページを使用して、Analytics テーブルに含める 1 つまたは複数の XML データ構造を選択します。

XML データ構造は、Analytics が XML ファイルを分析するときに識別する、要素、入れ子になった子要素、および属性から成ります。それらは、XML ファイルの階層的な表現である **XML データ構造** ツリービューに表示されます。各 XML データ構造は、表アイコン  と、XML 要素の名前、およびそれに含まれる入れ子の要素または属性の名前で表されます。

XML データ構造を選択する前に、XML ファイルを再調査して、監査の目的に適した Analytics テーブルの構造を決定しておく必要があります。そのテーブル構造を念頭において、意図するテーブル構造の列と一致する列を含んでいる XML データ構造のみを選択します。ウィザードの後続のページを使用して、含める個々の要素を微調整したり、列プロパティを変更したりすることができます。

複雑な構造を持つ XML ファイルを分析する場合は、その XML ファイルを 2 回以上インポートする必要があるかもしれません。作成する Analytics テーブルごとに異なるデータ構造を選択し、その後、Analytics でテーブルを結合したり関連付けることができます。

データ構造を選択するには

1. **XML データ構造** ツリービューで、XML 要素の名前をクリックし、結果の Analytics テーブルに含めるために、関連付けられたデータ構造を選択します。

サンプル構造 ペインに、データ構造の列名がサンプルデータと共に表示されます。@ 記号で始まる名前は、XML 要素ではなく XML 属性から派生したものです。

メモ

データ構造の中に、入れ子要素の複数のインスタンスが存在する場合、その繰り返し要素はツリービューのデータ構造に表示されないことがあります。繰り返し要素の特定のインスタンスは、ウィザードの次のページで選択することができます。

2. **追加** をクリックして、データ構造を **プレビュー** ペインに追加します。

選択したデータ構造のすべての表示される要素と属性、およびその要素および属性名に対応する列名が表示されます。

自動プレビュー がオンになっている場合は、データが表示されます。

メモ

大きい XML ファイルのデータをプレビューすると、速度が低下することがあります。このため、**自動プレビュー** オプションは、2 GB を超えるサイズの XML ファイルの場合には、自動的にオフにされます。

3. 手順 1 と 2 を繰り返して、Analytics テーブルに含めるほかの任意のデータ構造を追加します。
4. **次へ** をクリックします。

XML 要素の選択および構成

ウィザードの **XML 要素の選択** ページを使用して、結果の Analytics テーブルに表示される個別の XML 要素および属性を選択または微調整します。必要に応じて、列プロパティを変更できます。

XML 要素 ツリービューには、XML ファイルに含まれるすべての要素と属性の階層的な表現が提供されています。緑色のチェックマークは、前もって選択した要素または属性を示します。前に複数のデータ構造を選択した場合、それらは 1 つの **プレビュー** テーブルに統合されています。**自動プレビュー** がオンになっている場合は、選択した列のデータが表示されます。

メモ

大きい XML ファイルのデータをプレビューすると、速度が低下することがあります。このため、**自動プレビュー** オプションは、2 GB を超えるサイズの XML ファイルの場合には、自動的にオフにされます。

要素を選択および構成するには

1. 列を削除する場合は、**プレビュー** テーブルで列を選択して **削除** をクリックします。
2. 列を追加する場合は、**XML 要素** ツリービューで要素または属性を選択して **列の作成** をクリックします。

メモ

要素を追加することで、意図する Analytics テーブル構造が乱れたり、テーブルにギャップが生じたりする場合は、おそらく同じ種類のデータを単一の列にマージする必要があります(下記の手順 4)。

要素を追加することで、重複レコードや複数の同一レコード(追加された要素を除いて同一)が作成される場合は、おそらく、その要素の複数のインスタンスが親要素内に存在しており、インスタンスを別々の列に分割する必要があります。

3. 列を移動する場合は、対象の列を選択して左矢印または右矢印ボタンをクリックするか、もしくは、列を新しい位置までドラッグします。
4. 同じ型のデータを 1 つの列にマージするには、**プレビュー** テーブルで列を選択し、**XML 要素** ツリービューで追加する要素または属性を選択したら、**列へ追加** をクリックします。
5. 列のプロパティを変更する場合は、**プレビュー** テーブルで列を選択して **列プロパティ** をクリックします。

XML 列プロパティ ダイアログボックスでは、以下の変更を行い、**OK** をクリックします。

- 列の名前を変更します。
- 列のデータ型を変更します。
- 列にテキスト データ型が割り当てられている場合は、列の各行で列の名前を繰り返します。
- データ構造で 1 つの要素が繰り返されている場合は、繰り返し要素の特定のインスタンスを列に割り当てます。たとえば、データ構造に複数の `<description>` 要素がある場合は、最初のインスタンスのみを Description 列に割り当てます。
- マージされた列から特定の列を削除します。

6. [次へ](#)]をクリックします。

XML 列プロパティの修正

選択した列のプロパティの変更は、**[XML 列プロパティ]**ダイアログボックスを使用して行います。フィールドを Analytics テーブルに定義する方法や、XML ファイルから取り出すデータを指定できます。

[ソース]列には、フィールドのデータソースとして選択された各要素の階層が表示されます。たとえば、`/catalog/cd/title/` という値は、XML ファイルの次のような入れ子の構造を表しています。

```
<catalog>
  <cd>
    <title></title>
  </cd>
</catalog>
```

列のプロパティを変更するには

1. 列の名前を変更する場合は、**[列名]**テキストボックスに新しい名前を入力します。
フィールドのデフォルト値は、最初のソース要素の名前です。**[列名]**の値が、Analytics テーブルレイアウトのフィールドの名前になります。
2. 列のデータ型を変更する場合は、**[列の型]**ドロップダウンリストで"テキスト"、"数値"、または"日付時刻"から選択します。
Analytics が日付時刻書式を判断できない場合は、データと一致する日付書式を入力してください。**[日付時刻]**を選択すると、**[日付書式]**フィールドが現れます。
列のデータ型を変更したら、列にデータが表示されなくなった場合は、データ型とデータが一致していません。
3. 列から XML 要素を削除する場合は、**[ソース]**列で要素を選択して **[削除]**をクリックします。
4. 各行に、要素に格納されているデータではなく XML 要素の名前を表示したい場合は、**[種類]**ドロップダウンリストで"名前"を選択します。このオプションは、列の型がテキストである列にのみ適用されます。
5. 階層の同一レベルに同じ XML 要素の複数のインスタンスがあるときに、1つの要素または1つのサブセットを選択したい場合は、**[インスタンス]**フィールドを使用して、選択するインスタンスを指定することができます。

たとえば、**[インスタンス]**フィールドに値 2 を入力すると、次の例から 2 番目の `<description>` 要素の値のみが列に選択されます。

```
<organization>
  <company/>
  <description/>
  <department/>
  <description/>
</organization>
```

1つの番号、カンマで区切られた複数の番号、数値の範囲、あるいはこれら3つの組み合わせを入力できます。たとえば、1番目のインスタンスと、5番目から10番目のインスタンスを1つの列に含める場合は、「1, 5-10」と入力します。デフォルトでは、初めにすべてのインスタンスが列に表示されます。

ヒント:

繰り返し要素のさまざまなインスタンスを Analytics テーブルの異なる列に追加したい場合は、インスタンスごとに列を作成し、[インスタンス] フィールドを適切な値に設定する必要があります。

6. [OK] をクリックします。

XML ファイルについて

XML (eXtensible Markup Language) は、コンピューターシステム間やアプリケーション間でデータをやり取りするためによく使われるマークアップ言語です。は、正しく構成された XML ドキュメントをインポートすることができます。正しく構成されたドキュメントとは、XML 構文の規則に従っているものを指します。

XML ファイル構造

XML ファイルは標準的な方法で構造化され、階層レベルをいくつでもサポートします。階層レベルは、XML 要素をほかの要素の入れ子にすることで表されます。データ構造とは、Analytics テーブルに割り当てることができる要素のグループです。データ定義ウィザードでは、各データ構造はテーブルアイコン  で識別されます。XML ファイルを定義する場合、Analytics は、Analytics テーブルの作成に利用できるファイル内のあらゆるデータ構造を識別します。

Analytics テーブルへの XML のデータ構造のマッピング

データ構造が1つしかない単純な XML ドキュメントをインポートする場合は、Analytics テーブルに直接割り当てることができます。いくつかのデータ構造がある複雑な XML ファイルの場合は、複数のデータ構造を結合して1つの Analytics テーブルにすることができます。複数のデータ構造を結合することで、要望した結果が得られない場合は、各データ構造を別個のテーブルにインポートすることができます。その後、**テーブルの関連付け** コマンドを使用して、結果として生じた Analytics テーブル間の関係を定義できません。

ファイル拡張子

通常、XML ファイルは標準のファイル拡張子 (.xml) を使用します。別のファイル拡張子を使用し、ドキュメントの先頭行でそのドキュメントが XML ファイルであることを特定する場合があります。標準のファイル拡張子が使用されていない場合は、データ定義ウィザードでファイル形式として **[XML ファイル]** を手動で選択する必要があります。

XML 要素と属性

XML は要素と属性を使用してデータの構造と内容を識別します。Analytics は要素と属性の両方をインポートすることができます。

要素とは、タグで区切られたXMLデータの単位です。各XML要素はほかの要素で囲むことができます。次の例では、**name** 要素によって "John Smith" という値を定義しています。

```
<name>John Smith</name>
```

属性を使用すると、要素についての追加情報を提供できます。次の例では、**type** 属性によって、**account** 要素が checking account(当座預金口座)であることを指定しています。

```
<account type="checking">991110101</account>
```

データ定義ウィザードでは、要素名と区別するために、属性名の先頭には自動的に @ 記号が付けられます。たとえば、"type" という名前の属性は "@type" と表示されます。

XML サンプル

通常、XML ファイルには要素と属性が混在しており、少なくとも1つのデータ構造が含まれています。次の例では、2つのクライアントレコードを含む単純なXMLファイルの内容を示します。

```
<?xml version="1.0"?>
<accounts>
  <client>
    <name>John Smith</name>
    <ID>JS004</ID>
    <account type="checking">991110101</account>
    <account type="savings">991110808</account>
  </client>
  <client>
    <name>Jane Smith</name>
    <ID>JS005</ID>
    <account type="checking">771110103</account>
    <account type="savings">771110303</account>
  </client>
</accounts>
```

XBRL ファイルのインポート

XBRL ファイルを定義およびインポートして、Analytics テーブルを作成できます。**データ定義ウィザード**を使用すると、インポートする要素を選択したり、インポートする要素の列見出しやデータ型をカスタマイズしたりすることができます。

1. **ファイル > 新規作成 > テーブル**]の順にクリックします。
2. **データソースのプラットフォームの選択**]ページが表示されている場合は、**ローカル**]を選択し、**次へ**]をクリックします。
3. **ローカルデータソースの選択**]ページで **ファイル**]を選択し、**次へ**]をクリックします。
4. **定義するファイルの選択**]ダイアログボックスで、Analytics テーブルの作成元とするファイルを探して選択し、**開く**]をクリックします。

XBRL 2.1 ファイルは、.xbrl または .xml ファイル拡張子を持ちます。XBRL ファイルとその他 XML ファイルとの違いは、XBRL ファイルの最上位、つまりルート of 要素タグが `<xbrl>` であるということです。

5. **ファイル形式**]ページで、**XBRL 2.1 ファイル**]オプションが選択されていることを確認し、**次へ**]をクリックします。
6. **インポートする XBRL コンテキストの選択**]ページで、Analytics テーブルに含める XBRL コンテキストを選択し、**次へ**]をクリックします。この処理の詳細については、「XBRL コンテキストの選択」ページ 356を参照してください。
7. **インポートする要素の選択**]ページで、Analytics テーブルに含める要素を選択し、**次へ**]をクリックします。この処理の詳細については、「XBRL 要素の選択」ページ 355を参照してください。
8. **データのプレビュー**]ページでは、ページの下半分にあるプレビューテーブルで適切なフィールドの列見出しを選択し、以下のプロパティを更新することにより、各フィールドの名前やプロパティを変更することができます。

- **このフィールドを無視する** - このフィールドを Analytics テーブルに含めたくない場合は、このチェックボックスをオンにします。
- **名前**] - テーブルレイアウトのフィールド用に Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
- **列見出し**] - Analytics のデフォルトのビューに表示する列見出しを入力します。列見出しが指定されていない場合は、**名前**]の値が使用されます。
- **型** - ドロップダウンリストから適切なデータ型を選択します。Analytics でサポートされているデータ型の詳細については、「Analytics のデータ型」ページ 811を参照してください。

小数点以下の桁数]および **入力書式**]テキストボックスは、そのオプションの対象となるデータ型を選択したときに自動的に表示されます。

- **値** - フィールドの最初の値を表示する読み取り専用プロパティです。この値は、編集作業によって更新されます。
- **小数点以下の桁数** (数値フィールドのみ) - 小数点以下の適切な桁数を指定します。

- **入力書式**(日付時刻フィールドのみ) - その日付に対応する書式を指定します。日付と時刻の書式の詳細については、"ソース データに含まれる日付および時刻データの書式" ページ 366を参照してください。
9. 変更したいフィールド プロパティの編集が完了したら、**次へ]**をクリックします。
 10. **データ ファイルを別名で保存]**ダイアログ ボックスで、Analytics データ ファイルのファイル名を入力し、必要であれば、そのファイルを保存する場所を変更して、**保存]**をクリックします。
 11. **最終]**ページで、新しい Analytics テーブルの設定を確認したら、**完了]**をクリックします。
 12. プロジェクトに追加する Analytics テーブルの名前について、デフォルト の名前をそのまま用いるか、または別の名前を入力して、**OK]**をクリックします。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

XBRL 要素の選択

Analytics テーブルに含める XBRL 要素の選択は、ウィザードの **[インポートする要素の選択]** ページを使用して行います。

要素 テーブルには、前に選択したコンテキストと関連付けられているすべての要素が列挙されます。デフォルトでは、すべての要素を Analytics テーブルに含めるよう、初めにすべてが選択されています。

インポートする要素を選択するには

1. 次のいずれかを実行します。
 - 個々の要素の隣にあるチェックボックスをオン(選択)またはオフ(選択解除)にします。
 - **すべて選択** をクリックすると、すべての要素が選択されます。
 - **すべて選択解除** をクリックすると、すべての要素の選択が解除されます。
 - **選択範囲の反転** をクリックすると、選択していない要素がすべて選択され、選択している要素の選択がすべて解除されます。

[インポート サイズ] パネルには、Analytics テーブルにインポートされる個々のレコードの数と、XBRL ファイル内のレコードの総数が表示されます。

2. XBRL 要素の選択が完了したら、**次へ** をクリックします。

XBRL コンテキストの選択

XBRL のコンテキスト タイプと個々のコンテキストの選択は、ウィザードの **[インポートする XBRL コンテキストの選択]** ページを使用して行います。

Analytics テーブルにインポートされる XBRL データはすべて、同じコンテキスト タイプ(Instant(時点)、Period(期間)、またはForever(永遠))と関連付けられている必要があります。複数のコンテキスト タイプを単一の Analytics テーブルにインポートすることは、サポートされていません。

メモ

複雑な XBRL ファイルで作業している場合は、ファイル内のさまざまなコンテキストに対して複数の Analytics テーブルを定義し、その後 **テーブルの関連付け** コマンドを使用して、テーブル間の関係を定義する必要があるかもしれません。あるいは、関係が定義しやすくなるかもしれません。

XBRL コンテキストを選択するには

1. Analytics テーブルに追加したい **[コンテキスト タイプ]** を選択します。

XBRL ファイルの中で、コンテキスト タイプが一致するすべてのコンテキストが **使用可能なコンテキスト** ペインに表示されます。

2. **使用可能なコンテキスト** ペインで1つ以上のコンテキストを選択し、右矢印ボタンをクリックして、それらを **選択済みのコンテキスト** ペインへ移動させます。

すべて追加 をクリックして、使用可能なすべてのコンテキストを追加することもできます。 **選択済みのコンテキスト** ペインに1つ以上のコンテキストを追加すると、Analytics テーブルにインポートされる個々のレコードの数が **[インポート サイズ]** ペインに表示されます。

どちらかのペインで個々のコンテキストを選択すると、**[コンテキスト プロパティ]** 領域にエンティティと期間が表示されます。コンテキストにシナリオ情報が関連付けられている場合は、**[シナリオの表示]** をクリックすることができます。

3. Analytics テーブルに必要なコンテキストをすべて追加したら、**次へ** をクリックします。

XBRL ファイルについて

XBRL (eXtensible Business Reporting Language) とは、ビジネス データや財務 データの定義と交換を行うために用いる XML ベースの標準言語です。XBRL は、財務データを監督機関に提出したり、また企業間や企業内のシステム間でデータをやり取りするのに使用できます。Analytics は現在の標準 XBRL 2.1 をサポートしています。

XBRL インスタンスのドキュメントは XML ファイルです。これは XBRL 要素を使用しており、XBRL 標準に準拠しています。XBRL 要素は「項目」とも呼ばれます。XBRL インスタンスのドキュメント タイプには貸借対照表、総勘定元帳、財務諸表などがあります。

XBRL ドキュメントは有効な XML ファイルで、通常は標準 XML ファイルの拡張子 (.xml) を使用します。ルート要素 `<xbrl>`、つまりファイルの先頭にある要素によって、そのファイルが XBRL ファイルであることが識別されます。

XBRL コンテキスト

XBRL ドキュメントには、1 つ以上のコンテキストを定義する要素とデータが含まれています。コンテキストは、ファイルに含まれるデータの本体を分類します。ファイル内の各要素または項目は、特定のコンテキストを参照する必要があります。コンテキストは次の事項を指定します。

- **エンティティ** - データに関係する企業、事業単位、など。
- **期間** - データが有効である期間。指定可能な期間が 3 つあります。
 - **Instant(時点)** - データは特定の日付または日付時刻に有効です。たとえば、2012 年 1 月 1 日午前 9 時の時点の銀行残高の値などです。
 - **Period(期間)** - データは特定の日付の範囲、または日付時刻の範囲に有効です。たとえば、2012 年 1 月 1 日から 2012 年 12 月 31 日の間に発生した財務取引などです。
 - **Forever(永遠)** - データは日付や時刻に依存しません。たとえば、勘定科目番号は無期限、つまり未定義の期間有効です。
- **シナリオ(任意)** - 関連する要素に関する追加のコンテキスト情報。たとえば、要素に含まれている値は、実費用、見積もり、または予算であるかどうか、などです。

XBRL ドキュメントには複数のコンテキストを含むことができます。たとえば、2011 年 1 月 1 日から 2011 年 12 月 31 日までを期間とするコンテキストと、2012 年 1 月 1 日から 2012 年 12 月 31 日までを期間とする別のコンテキストをドキュメントに含めることができます。**データ定義ウィザード**では、各コンテキストは別個のデータブロックとして利用でき、それぞれインポートするかしないかを選択できます。

Analytics サーバーのデータベース プロファイルデータの定義

御社に1つ以上のWindows Analytics サーバー製品がインストールされ、構成されている場合は、データソースに接続するためにサーバープロファイルおよびデータベースプロファイルを作成しておく、直接 Oracle、SQL Server、または IBM DB2 のデータベースのデータにアクセスする Analytics テーブルを作成することができます。

Analytics テーブルを開くたびに、Analytics テーブルの作成に使用されたクエリを再実行することで、データベースから最新のデータが取得されます。WHERE 句と ORDER 句は、Analytics テーブルが作成された後に変更することができます。

Analytics サーバー テーブルの制限事項

データベースプロファイルを使用して Analytics テーブルを定義するには、次の制限が適用されます。

- Analytics の結合で Analytics サーバー テーブルを使用する場合は、これを主テーブルにする必要があります。
- Analytics でテーブルの関連付けを行う際に使用できる Analytics サーバー テーブルは1つだけで、これを親テーブルにする必要があります。
- Analytics サーバー テーブルにインデックスを設定することはできません。データを整列させる必要がある場合は、テーブルを定義するときに ORDER 句を指定するか、または Analytics でテーブルプロパティを変更して、ORDER 句を追加します。
- **クイックソート 昇順** および **クイックソート 降順** オプションは、Analytics サーバー テーブルのビューウィンドウでは有効になりません。

手順

1. **ファイル > 新規作成 > テーブル** の順にクリックします。
2. **データソースのプラットフォームの選択** ページで、**ACL サーバー** を選択し、ドロップダウンリストから使用する Windows サーバープロファイルを選択したら、**次へ** をクリックします。
3. **ACL サーバー データソースの選択** ページで、**データベースプロファイル** を選択し、ドロップダウンリストから使用するデータベースプロファイルを選択したら、**次へ** をクリックします。
4. **データベース/スキーマの選択** ページで、**スキーマ** ドロップダウンリストからアクセスするスキーマ (Oracle) またはデータベース (SQL Server および IBM DB2) を選択し、**次へ** をクリックします。
5. **テーブルの選択** ページで、**使用可能なテーブル** リスト内の項目を選択して右矢印ボタンをクリックすることにより、クエリに追加するデータベーステーブル、ビュー、およびシノニム/エイリアスを選択します。テーブルを最大5つまで選択できますが、テーブルを2つ以上選択する場合、追加選択したテーブルはそれぞれ、前に選択したテーブルに関連付ける必要があります。複数のテーブルを選択した場合、特に、多数のレコードを含むテーブルを選択した場合には、通常、データが Analytics に表示されるまでの待ち時間がより長くなります。

複数のテーブルを選択すると、Analytics は [リレーションシップの識別] ダイアログ ボックスを表示します。このダイアログ ボックスを使用して、追加しようとしているテーブルの中で、既に選択されているテーブルと関連しているフィールドを識別する必要があります。

6. 必要なテーブルを選択し終わったら、**次へ**]をクリックします。
7. 省略可能。 **Condition 句**] ページで、 **CONDITION 句を編集する**] チェック ボックスをオンにして、編集用のテキスト ボックスを有効にします。SQL 構文を使用してテキスト ボックスに条件を入力したら、 **次へ**]をクリックします。クエリに2つ以上の関連テーブルが含まれている場合は、値だけが **CONDITION 句**] テキスト ボックスに表示されます。Condition 句(条件句)は、テーブルを関連付ける方法を指定します。
8. **列の選択**] ページで、追加する列を **使用可能な列**] リストから選択し、右矢印ボタンをクリックしてそれを **選択済みの列**] リストに追加します。もしくは、 **すべて追加**] をクリックして、すべての列を **選択済みの列**] リストに追加します。複数のテーブルを選択した場合は、 **ソーステーブルの選択**] ドロップダウン リストでテーブルをそれぞれ選択し、必要な列を選択することにより、テーブルごとに少なくとも1つの列は選択する必要があります。含める列をすべて選択し終わったら、 **次へ**] をクリックします。
9. 省略可能。 **Where および Order 句**] ページで、データベースから返される結果を制限したい場合は WHERE ステートメントを入力します。また、1つ以上の列を基準にして結果を並べ替えたい場合は ORDER 句を入力し、 **次へ**] をクリックします。どちらの種類のスートメントも、コマンド キーワードを指定せずに、SQL 構文を使用してステートメントを入力する必要があります。たとえば、「WHERE Value > 1000.00」は「Value > 1000.00」と入力します。
10. **最終**] ページで、新しい Analytics テーブルの設定を確認したら、 **完了**] をクリックします。
11. プロジェクトに追加する Analytics テーブルの名前を入力し、 **OK**] をクリックします。

外部定義ファイルの定義

外部定義ファイルには、外部ファイル内のデータファイルのレコードとフィールドのレイアウト情報が格納されます。Analytics は、必要なフィールド情報について外部定義ファイルを分析し、Analytics テーブルレイアウトを作成することができます。その後、Analytics テーブルレイアウトは、Analytics データファイル (.fil) を作成せずに直接データファイルを読み取るために使用されます。

AS400 FDF ファイルの定義

AS/400 ファイルの定義ファイルから Analytics テーブルを定義することができます。この種類のファイルは、データファイルとは別に、外部ファイルにフィールドの書式に関する情報を格納します。Analytics は、書式情報を使用して Analytics テーブルを作成し、データファイルから直接データを読み取ります。

1. **[ファイル > 新規作成 > テーブル]** の順にクリックします。
2. **[データソースのプラットフォームの選択]** ページが表示されている場合は、**[ローカル]** を選択し、**[次へ]** をクリックします。
3. **[ローカルデータソースの選択]** ページで **[その他]** を選択し、**[次へ]** をクリックします。
4. **[外部定義ソースの選択]** ページで **[AS400 FDF]** を選択し、**[次へ]** をクリックします。
5. **[変換する AS/400 FDF ファイルの選択]** ダイアログボックスで、使用する外部定義ファイルを探して選択し、**[開く]** をクリックします。
6. **[AS/400 変換]** ページで、**[次へ]** をクリックします。
7. **[最終]** ページで、新しい Analytics テーブルの設定を確認したら、**[完了]** をクリックします。
8. プロジェクトに追加する Analytics テーブルの名前を入力し、**[OK]** をクリックします。
9. 関連するデータファイル (.dat) が見つからない場合は、ファイルの場所の入力を求められます。
[データファイルの場所 '<ファイル名>'] ダイアログボックスで、必要なファイルを探して選択し、**[開く]** をクリックします。

COBOL または PL/1 ファイルの定義

COBOL コピーブックファイルまたは PL/1 ファイルから Analytics テーブルを定義することができます。両方の種類のファイルは、データファイルとは別に、外部ファイルにフィールドの書式に関する情報を格納します。Analytics は、書式情報を使用して Analytics テーブルを作成し、データファイルから直接データを読み取ります。

1. **[ファイル > 新規作成 > テーブル]** の順にクリックします。
データ定義ウィザードに表示される最初のページは構成によって異なります。Analytics サーバーとの統合が有効になっている場合は、**[データソースのプラットフォームの選択]** ページが表示され、そうでない場合は **[ローカルデータソースの選択]** ページが表示されます。
2. **[データソースのプラットフォームの選択]** ページが表示されている場合は、**[ローカル]** を選択し、**[次へ]** をクリックします。
3. **[ローカルデータソースの選択]** ページで **[その他]** を選択し、**[次へ]** をクリックします。
4. **[外部定義ソースの選択]** ページで、必要に応じて **[COBOL]** または **[PL/1]** を選択し、**[次へ]** をクリックします。

5. **外部定義の選択**] ページで、次の手順を行います。
 - a. 次のいずれかを実行します。
 - **外部定義**] ドロップダウン リストから、既存の定義ファイルまたはデータ セットを選択します。
 - **参照**] をクリックし、**変換するファイルの選択**] ダイアログ ボックスで定義ファイルまたはデータ セットを検索して、**開く**] をクリックします。
 - b. 次の手順のいずれかを実行して、単一の定義または複数の定義を選択します。
 - 個々の定義を **選択済みの定義**] に追加するには、**使用可能な定義**] リスト内の定義を選択して右矢印ボタンをクリックするか、または **使用可能な定義**] リスト内の定義をダブルクリックします。
 - 複数の定義を追加するには、**選択した複数のファイル定義を連結させますか**] チェック ボックスをオンにしてから、追加する各定義を選択して右矢印ボタンをクリックするか、または **すべて追加**] をクリックして、**使用可能な定義**] リストに挙げられているすべての定義を追加します。
 - c. **次へ**] をクリックします。
6. **変換プロパティの選択**] ページで、必要に応じて、次のプロパティの一方または両方を選択して、**次へ**] をクリックします。
 - **フィールド名から先頭のファイル インジケータを削除する** - 各フィールド名の前にプレフィックスが指定されている場合は、Analytics テーブルレイアウトに追加されるフィールド名からそのプレフィックスが取り除かれます。たとえば、**Test-Field1** ならば、プレフィックスの **Test-** が削除されるでしょう。

このチェック ボックスがオフになっている場合、識別されたプレフィックス値はすべてフィールド名に含まれますが、その際、ハイフンはアンダースコアに変換されます(つまり、**Test-Field1** は **Test_Field1** として追加されます)。
 - **IBM 可変長** - 処理するデータ ファイルが IBM の可変長ファイルである場合、Analytics テーブルレイアウトにレコード長は指定されません。
7. **最終**] ページで、新しい Analytics テーブルの設定を確認したら、**完了**] をクリックします。
8. プロジェクトに追加する Analytics テーブルの名前を入力し、**OK**] をクリックします。
9. 関連するデータ ファイルが見つからない場合は、ファイルの場所の入力を求められます。ファイルを検索するには、次の手順を行います。
 - a. **ファイルの場所の選択**] ダイアログ ボックスが表示されたら、**クライアント**] を選択して、お使いのコンピューターからアクセス可能なファイルを選択するか、または **サーバー**] を選択して、サーバー ファイルをアクセスするために使用するサーバー プロファイルを選択し、**OK**] をクリックします。
 - b. **ファイルの選択**] ダイアログ ボックスで、データ ファイルを探して選択し、**開く**] をクリックします。

Analytics テーブルの手動による定義

Analytics は、ファイルベースのデータソースのファイル形式を識別できない場合には、**データ定義ウィザード**の **ファイル形式** ページで、**その他のファイル形式** を選択します。このオプションは、ファイルを手動で定義したい場合にも選択することもできます。このオプションが選択されている場合、**データ定義ウィザード** は、ファイルを定義するために必要な手順について順を追って説明しますが、ファイルからテーブルレイアウトを作成するには、必要なすべての情報を Analytics に提供しなければなりません。

Analytics テーブルを手動で作成するには

1. **ファイル > 新規作成 > テーブル** の順にクリックします。

データ定義ウィザード に表示される最初のページは構成によって異なります。Analytics サーバーとの統合が有効になっている場合は、**データソースのプラットフォームの選択** ページが表示され、そうでない場合は **ローカルデータソースの選択** ページが表示されます。

2. ファイルの場所を選択するために、次の手順のいずれかを実行します。

- **データソースのプラットフォームの選択** ページが表示されていて、ファイルを定義するのに Analytics を使用したい場合は、**ローカル** を選択して **次へ** をクリックします。**ローカルデータソースの選択** ページで **ファイル** を選択し、**次へ** をクリックします。
- **データソースのプラットフォームの選択** ページが表示されていて、ファイルを定義するのに ACL サーバーを使用したい場合は、**Analytics サーバー** を選択し、ドロップダウンリストからサーバープロファイルを選択して **次へ** をクリックします。**ACL サーバーデータソースの選択** ページで **プラットフォーム ファイル** を選択し、**次へ** をクリックします。
- **ローカルデータソースの選択** ページが表示されている場合は、**ファイル** を選択して **次へ** をクリックします。

3. **定義するファイルの選択** ダイアログボックスで、Analytics テーブルの作成元とするファイルを探して選択し、**開く** をクリックします。

4. **文字セット** ページで、正しい文字セット オプションが選択されていることを確認し、**次へ** をクリックします。

5. **ファイル形式** ページで、**データ定義ウィザード** がより具体的な種類としてファイルを識別できない場合は、**その他のファイル形式** オプションが選択されます。これは、Analytics テーブルを手動で定義する必要があるということです。また、Analytics テーブルを手動で定義したい場合には、別のファイルの種類に割り当てられているファイルを分類し直すことができます。**次へ** をクリックして、次のページに進みます。

6. **ファイルプロパティ** ダイアログボックスで、次の手順を行います。

- a. **データ定義ウィザード** で識別されるファイルの種類を確認します。種類を変更する場合は、以下のオプションからファイル定義の種類を選択します。

- **固定長** - ファイル内のすべてのレコードが同じ長さで、各フィールドがどのレコードでも同じ場所に配置されている場合は、このオプションを選択します。
- **可変長** - ファイル内のレコードの長さが異なる場合は、このオプションを選択します。ウィザードの次のページでは、ファイルの種類をさらに分類することができ、長さが変化する単一レコードタイプを含んでいるファイルであるか、印刷イメージ(レポート)ファイルか、または複数のレコードタイプを含んでいるファイルかを判断します。

- **最後までスキップ** - Analytics テーブルレイアウトにフィールドを定義せずに、**データ定義ウィザード**の **最終**]ページまで移動するには、このオプションを選択します。**テーブルレイアウト** ダイアログボックスで手動でテーブルレイアウトを定義したい場合、または**データ定義ウィザード**を使用してファイルを定義できない場合は、このオプションを選択してください。
 - b. ファイルの先頭から指定されたバイト数をスキップするには、**スキップするバイト数**]テキストボックスに0より大きい値を入力します。たとえば、最初の300バイトにはヘッダー情報しか含まれていない場合は、「300」を入力して、テーブル定義からファイルのこのセクションを除くことができます。
 - c. **データ定義ウィザード**で識別されたレコード長を増減するには、**レコード長**]テキストボックスの値を変更します。レコード長は、固定長ファイルの各レコードの長さ、または可変長レコードの最長レコードの長さを指します。フィールドの値が右にずれている場合は、おそらくレコード長の値を増やす必要があります。フィールドの値が左にずれている場合は、おそらくレコード長の値を減らす必要があります。
 - d. データを16進数形式で表示するには、**16進**]チェックボックスをオンにします。このオプションは、印刷できない文字や圧縮データ(IBMメインフレームコンピューターから作成されるパック化された数値データなど)を処理する場合に便利です。
 - e. **次へ**]をクリックします。
7. 前の手順で **固定長**]または **可変長**]を選択した場合は、**ファイルの種類**]ページで次のオプションのいずれかを選択して、**次へ**]をクリックします。
- **データファイル(単一レコードタイプ)** - レコードの各フィールドが固定された開始点と終了点を持っており、ファイル内の各レコードが同じ長さである場合は、このオプションを選択します。
 - **印刷イメージファイル(レポートファイル)** - データファイルが、一貫した書式を含んでいる印刷されたレポートの電子版である場合は、このオプションを選択します。この種類のデータファイルは、報告される情報が入っている詳細レコードを持ち、多くの場合、顧客の詳細や合計などの追加情報を含むヘッダーレコードおよびフッターレコードを持っています。

メモ

このオプションを使用して印刷イメージファイルの定義を試みる前に、**データ定義ウィザード**の **ファイル形式**]ページ(上記の手順5)で **印刷イメージ(レポート)ファイル**]オプションを選択してみてください。これは、より簡単なプロセスを使用して印刷イメージファイルを定義します。

- **複数レコードタイプファイル** - データファイルに複数のレコードタイプが含まれているが、レポートとしてフォーマットされていない場合は、このオプションを選択します。
 - **フィールドの識別をスキップする** - Analytics テーブルレイアウトにフィールドを定義せずに、**データ定義ウィザード**の **最終**]ページまで移動するには、このオプションを選択します。
8. 前の手順で **データファイル(単一レコードタイプ)**]を選択した場合は、次の手順を行います。
- a. **フィールドの識別**]ページで、次の操作を行ってレコードで識別されるフィールドを変更し、**次へ**]をクリックします。
 - 既存のフィールド区切りを削除するには、削除したいフィールド区切り線をクリックします。
 - 既存のフィールド区切りを移動するには、フィールド区切り線をクリックして新しい位置までドラッグします。
 - 新しいフィールド区切りを作成するには、フィールド区切りを追加したい位置のグリッドをクリックします。

- b. **フィールド プロパティの編集**] ページで、次の手順を行います。
 - a. ページの下半分にあるプレビュー テーブルで、プロパティを編集するフィールドの列見出しをクリックします。
 - b. そのフィールドを Analytics テーブルレイアウトに含めたくない場合は、**[このフィールドを無視する]** チェック ボックスをオンにします。
 - c. **名前**] では、テーブルレイアウトのフィールド用に Analytics で割り当てた名前をそのまま使用するか、別の名前を入力します。
 - d. **列見出し**] には、Analytics のデフォルトのビューに表示する列見出しを入力します。列見出しが指定されていない場合は、**名前**] の値が使用されます。
 - e. **型**] ドロップダウン リストからフィールドの適切なデータ型を選択します。Analytics でサポートされているデータ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。
 - f. 選択したデータ型が"数値"である場合は、**[小数点以下の桁数]** テキスト ボックスに、小数点以下の桁数を指定できます。**[小数点以下の桁数]** テキスト ボックスは、このオプションの対象となるデータ型を選択したときに自動的に表示されます。
 - g. データ型に"日付時刻"を選択した場合は、その日付に対応する書式を **[入力書式]** テキスト ボックスで指定します。**[入力書式]** テキスト ボックスは、このオプションの対象となるデータ型を選択したときに自動的に表示されます。日付と時刻の書式の詳細については、"ソースデータに含まれる日付および時刻データの書式" ページ 366を参照してください。
 - h. 変更したいフィールドをすべて編集し終わったら、**次へ**] をクリックします。
9. 手順 7 で **印刷イメージ ファイル(レポート ファイル)**] または **複数レコード タイプ ファイル**] を選択した場合は、次の手順を行います。
 - a. **[レコード定義のイントロダクション]** ページでは、**データ定義ウィザード** がデータ ファイルで自動的に識別したすべてのレコードが表示されます。新しいレコード タイプを作成したい、またはウィザードによって識別された既存のレコード タイプを変更したい場合は、**[レコード タイプを追加/編集する]** をオンにし、**次へ**] をクリックして続行します。
 - b. **[レコード/行タイプの識別]** ページで、次の手順によってデータ ファイルのレコード タイプを指定します。
 - 新しいレコード タイプを追加するには、**定義するタイプを選択してください。**] ドロップダウン リストでレコード タイプの名前を入力します。プレビュー テーブルでレコードを定義するテキストを選択し、レコードにテキストを含める(**包含**] ボタン) か、またはレコードからテキストを除外する(**除外**] ボタン) かを指定します。また、そのレコードがヘッダーレコードである場合は、**[ヘッダー行]** チェック ボックスをオンにします。
 - 既存のレコード タイプを編集するには、**定義するタイプを選択してください。**] ドロップダウン リストで編集するレコード タイプを選択します。プレビュー テーブルでテキストを選択し、レコードにテキストを含める(**包含**] ボタン) か、またはレコードからテキストを除外する(**除外**] ボタン) かを指定することで、レコードを定義する一意のテキストを変更します。前に選択したテキストを削除するには、リストでエントリを選択し、**削除**] ボタンをクリックします。また、そのレコードがヘッダーレコードである場合は、**[ヘッダー行]** チェック ボックスをオンにします。
 - c. **フィールドの識別**] ページで、上記の手順 8a で説明されている手順を行います。
 - d. **フィールド プロパティの編集**] ページで、上記の手順 8b で説明されている手順を行います。
10. **最終**] ページで、新しい Analytics テーブルの設定を確認したら、**完了**] をクリックします。

11. プロジェクトに追加する Analytics テーブルの名前について、デフォルトの名前をそのまま用いるか、または別の名前を入力して、**OK**をクリックします。

ソースデータに含まれる日付および時刻データの書式

Analytics テーブルを定義する場合、日付、日付時刻または時刻データのソースにおける書式(入力書式)は、Analytics で自動的に認識される可能性があります。たとえば、Analytics は YYYYMMDD という書式になっている日付は自動的に認識します。ソースの書式が自動認識されない場合は、その書式を手動で指定する必要があります。

ソース書式と表示書式

ソースの日付時刻データの書式を指定することと、Analytics での日付時刻データの表示方法を指定することは同じではありません。ソース書式は、Analytics がソースファイルで日付時刻データを読み取る方法を制御します。指定する書式文字とソースデータの実際の書式が一对一で対応している必要があります。

Analytics でソースの日付時刻データが正しく定義されると、それをさまざまな書式で表示できるようになります。日付時刻データの表示にどのような書式を選択しても、そのソースの書式にはなんの影響もありません。

日付時刻データの表示に関する説明については、「日付と時刻オプション」ページ 135を参照してください。

ソース書式を手動で指定する

データ定義ウィザードでテーブルを定義しているときに、手動で日付時刻ソース形式を指定できます。あるいは、Analytics の **テーブルレイアウト** ダイアログボックスで、後から書式を指定できます。

区切り文字を除き、ソースデータの書式と完全に一致するように書式を指定し、Analytics でそのソースデータが正しく表示されるようにする必要があります。

日付時刻の書式で区切り文字を指定するためのガイドライン

ソースの日付時刻データには多くの場合、次のような区切り文字が含まれています。

- 年、月、日の各要素間のスラッシュ(/)などの文字
- 時、分、秒の各要素間のコロン(:)などの文字
- スペースまたは"T"などの文字 - 日付時刻値の日付と時刻の間に入ります。
- "T"などの文字または小数点 - 単独の時刻値時の前に付けられます。

データの定義とインポート

- プラス(+)またはマイナス(-)記号 - タイムゾーンインジケータ付きのローカル時刻の場合、UTC オフセットの前に付けられます。

ソースデータにあるこれらの区切り文字の中には、Analytics で自動的に認識されるものもあります。

日付時刻の書式で区切り文字を指定するときには、次のガイドラインに従ってください。区切り文字を省いたり、適切に指定しなかった場合は、日付時刻データが表示されないか、または正しく表示されません。

メモ

日付時刻の書式における特定の区切り文字の指定は、その文字の機能に応じて、**必須**、**省略可能**、あるいは**拒否**される可能性があります。

区切り文字の機能	書式での指定	対象のソースデータ	指定する書式
日付の日、月、年の要素を区切る	必須	31/12/2014	DD/MM/YYYY
時刻の時、分、秒の要素を区切る	省略可能	23:59:59	hh:mm:ss hhmmss
日付時刻値の日付と時刻の部分を区切る (スペース)	省略可能	31/12/2014 23:59:59	DD/MM/YYYY hh:mm:ss DD/MM/YYYYhh:mm:ss DD/MM/YYYY hhmmss DD/MM/YYYYhhmmss
日付時刻値の日付と時刻の部分を区切る (Tまたはt)	拒否	31/12/2014T235959	DD/MM/YYYY hhmmss DD/MM/YYYYhhmmss
単独の時刻値の前に付ける (Tまたはt)	拒否	T235959	hhmmss
数値データ型を使用する日付時刻値の日付と時刻の部分を区切る (小数点)	省略可能	31122014.235959	DDMMYYYY.hhmmss DDMMYYYYhhmmss
数値データ型を使用する単独の時刻値の前に付ける (小数点)	省略可能	.235959	.hhmmss hhmmss
UTC オフセットの前に付ける (プラスまたはマイナス記号)	必須	T235959-0500	hhmmss-hhmm hhmmss+hhmm

日と時刻の区切り文字

Analytics がソース データから日付時刻値を読み込むためには、ソース データの日付と時刻の要素がスペースか区切り文字で区切られている必要があります。例：

- 2014/12/31 23:59:59
- 20141231.235959

日付時刻データ型、または文字データ型を使用する日付時刻値の場合、Analytics では以下の区切り文字を認識します。

- <日付> <時刻> (スペース1つ)
- <日付>T<時刻> (大文字のT)
- <日付>t<時刻> (小文字のt)

数値データ型を使用する日付時刻値の場合、Analytics では次の区切り文字のみを認識します。

- <日付>.<時刻> (小数点)

メモ

Analytics は日付時刻データ型または文字データ型を使用する日付時刻値を読み込むことができ、また、区切り文字としてピリオド(小数点)を用いることができます(<日付>.<時刻>)。ただし、ピリオドの区切り文字は、状況によっては結果が信頼性に欠けるため、正式にはサポートされません。

単独の時刻データ

Analytics がソース データから単独の時刻値(23:59:59 など)を読み込むためには、ソース データの時刻値の前にスペースまたは区切り文字が付けられているか、あるいはその時刻の各要素がコロンで区切られている必要があります。例：

- 23:59:59
- .235959

時刻データ型、または文字データ型を使用する日付時刻値の場合、Analytics では以下の区切り文字を認識します。

- _<時刻> (スペース1つ)
- T<時刻> (大文字のT)
- t<時刻> (小文字のt)
- <hh>:<mm>:<ss> (コロン)

数値データ型を使用する時刻値の場合、Analytics では次の区切り文字のみを認識します。

- .<時刻> (小数点)

メモ

Analytics は、日付時刻データ型または文字データ型を使用し、区切り文字としてピリオド(小数点)を用いる(<時刻>)時刻値を読み取ることができます。ただし、ピリオドの区切り文字は、状況によっては結果が信頼性に欠けるため、正式にはサポートされません。

日付書式

現在、さまざまな仕様の日付書式が使用されています。**データ定義ウィザード**や**テーブルレイアウト**ダイアログボックスで、一般的な日付書式の中から書式を選択することができます。必要に応じて日付書式をソースデータの書式と一致するように編集または作成することもできます。

日付書式は、日付データ、または日付時刻データの日付部分に適用されます。以下に、一般的な日付書式をいくつか示します。

一般的な日付書式	型	2014年12月31日の場合
YYYY-MM-DD	ISO	2014-12-31
MM/DD/YYYY	アメリカ	12/31/2014
DD/MM/YYYY DD.MM.YYYY DD-MM-YYYY	ヨーロッパ	31/12/2014 31.12.2014 31-12-2014
YYYYDDD	ユリウス日(通日)	2014365

日、月、および年の文字

日付書式を定義するときに、ソースデータのどの要素が年、月、日のそれぞれを表すのかを指定します。Analytics では、以下に示す書式文字を使用して、日付の年月日の要素を表します。

メモ

これらの文字はデフォルトです。**[オプション]**ダイアログボックスで変更できます。

スラッシュ記号(/)などの区切り文字がソースデータに存在する場合は、同じ記号を日付書式内の同じ相対位置に挿入する必要があります。そうしないと、Analytics は日付を正しく解釈することができません。

書式文字	日付要素
DD	日(1 ~ 31)
DDD	ユリウス日/通日(1 ~ 366)

書式文字	日付要素
MM	月 (1 ~ 12)
MMM	月名 (Jan ~ Dec)
YY	短い年書式 (00 ~ 99)
YYYY	長い年書式 (1900 ~ 9999)

ソースデータに対する日付書式を指定する例

Analytics 日付書式	ソースデータ
YYYY-MM-DD	2014-12-31
YYYYMMDD	20141231
MM/DD/YYYY	12/31/2014
MM/DD/YY	12/31/14
DD/MM/YYYY	31/12/2014
YYDDD	14365
MMM DD, YYYY	Dec 31, 2014
DD MMM YYYY	31 Dec 2014

時刻書式

Analytics では最も一般的な時刻書式の表記 (hh:mm:ss) と、この書式の細かなバリエーションをサポートします。データ定義ウィザードやテーブルレイアウト ダイアログボックスで、一般的な時刻書式の中から書式を選択することができます。必要に応じて時刻書式をソースデータの書式と一致するように編集または作成することもできます。

時刻書式は、時刻データ、または日付時刻データの時刻部分に適用されます。

時間、分、および秒の文字

時刻書式を定義するとき、ソースデータのどの要素が時、分、秒のそれぞれを表すのかを指定します。また、AM/PM インジケータや UTC オフセット インジケータが存在する場合は、それらの要素も指定し

ます。Analytics では、以下に示す書式文字を使用して、時刻データのさまざまな要素を表します。

メモ

以下に示す時間、分、秒の文字はデフォルトです。[オプション]ダイアログボックスで変更できます。

書式文字	時刻要素
hh	時(00 ~ 23)
mm	分(00 ~ 59)
ss	秒(00 ~ 59)
:(コロン)	時刻要素の区切り文字
A または P	AM/PM インジケータ (A および P)
AM または PM	AM/PM インジケータ (AM および PM)
+ または -	UTC オフセット インジケータ (+ および -)

ソースデータに対する時刻書式を指定する例

Analytics 時刻書式	ソースデータ
hh:mm	23:59
hh:mm A	11:59 P
hhmm PM	1159 PM
hh:mm:ss	23:59:59
hh:mm:ss P	11:59:59 P
hhmmss AM	115959 PM
hh:mm:ss+hh:mm	23:59:59-05:00

データ アクセス ウィンドウを使用したデータのインポート

データ アクセス ウィンドウは、データ コネクタを含む Analytics のコンポーネントであり、さまざまなソースのデータを Analytics にインポートするために使用できます。データ アクセス ウィンドウには、インポートするデータのセットを正確に形成するための機能もあります。

データ アクセス ウィンドウを使用して接続できるデータソースの一覧については、「Analytics でアクセスできるデータソース」ページ 242を参照してください。

メモ

データ定義ウィザードを使用して、データをインポートすることもできます。詳細については、「データ定義ウィザードを使用したデータの定義とインポート」ページ 248を参照してください。

データソースに接続するとき、またはデータソースからインポートするときには、Analytics は厳密に読み取り専用です。詳細については、「Analytics によるデータアクセスは読み取り専用です」ページ 243を参照してください。

データ コネクタ

データ コネクタは Analytics と、Microsoft SQL Server、Oracle、Salesforce、Microsoft Excel などの ODBC 準拠データベースとファイル形式の間でインターフェイスを提供する ODBC ドライバーです。

ODBC は Open Database Connectivity を表します。これは Microsoft の標準であり、アプリケーションは SQL (構造化問い合わせ言語) を使用して、外部データベースまたはファイルのデータにアクセスできます。

メモ

データ アクセス ウィンドウは、インポート専用ツールです。外部データベースまたはファイルのデータにアクセスするために使用される SQL インポート文を編集できません。SQL を編集してデータソースに書き込むことはできません。

データ アクセス ウィンドウの概要

データ アクセス ウィンドウに含まれる機能は以下のとおりです。

データの定義とインポート

The screenshot shows the 'Data Access' window in ACL for Windows. It is divided into several sections:

- 1 接続 (Connection):** Shows the connection name 'Microsoft Access Driver (*.mdb)' and the database path 'C:\Users\Admin\Documents\ACL Data\Sample Data Files\Sample'.
- 2 テーブルを検索... (Search for tables...):** A search box for finding tables.
- 3 利用可能なテーブル (Available tables):** A list of tables including 'Customer', 'Orders', and 'Product'.
- 4 ステージング領域 (Staging area):** A diagram showing the mapping of source tables to a target table. 'Customer' and 'Orders' are mapped to 'Product'.
- 5 フィルター適用済み - 編集 (Filter applied - Edit):** A section for configuring filters. A filter is set for 'Customer.Region' to be 'である' (is) 'NY'.
- 6 インポートのプレビュー (Import preview):** A table showing the data being imported. It has 28 records and a total size of 29.7 KB.
- 7 最大文字フィールド長 (Maximum character field length):** A field set to 50.
- 8 SQLモード (SQL mode):** A toggle switch.

番号	機能	説明
1	接続	現在アクティブなデータ接続に関する情報を表示します。 接続の名前、データベースの名前、またはソース データを含むファイルの場所が含まれます。
2	テーブルの検索	ソース データの使用可能なテーブルのリストを進行的にフィルタリングする検索ボックス。 検索ボックスに文字を入力すると、使用可能なテーブルリストがフィルタリングされ、一致する文字列を含むテーブル名のみが抽出されます。
3	利用可能なテーブル	インポートで使用できるソース データのテーブル。 ソース データの最初の 200 テーブルが表示されます。追加のテーブルが存在している

番号	機能	説明
		場合は、リンクをクリックして、一度に最大 500 テーブルのブロックで表示できます。
4	ステージング領域	インポート用に選択したテーブルを含むデータ アクセス ウィンドウの領域。 ステージングエリアはテーブル間で結合を実行する場所であり、インポートされるテーブルのフィールドを選択します。
5	[フィルター] パネル	インポートされるデータセットのレコードを正確に指定する簡易または複合フィルターを作成するためのパネル。
6	インポートのプレビュー	Analytics にインポートされる正確なデータのプレビュー。 テーブルの結合、フィールドの省略、フィルターの作成でデータを操作するときには、プレビューを更新し、変更の効果を確認できます。 [サイズの推定] オプションでは、インポートのレコード数の推定と、作成される Analytics データ ファイル (.fil) のサイズを表示します。
7	フィールド構成	3つのオプションでは次のことができます。 <ul style="list-style-type: none"> インポートされる文字またはメモフィールドの最大フィールド長を調整する すべてのフィールドを文字データ型としてインポートする
8	SQL モード	SQL インポート文を直接編集できるテキスト エディター。 SQL の知識があるユーザーは、ユーザー インターフェイスでは使用できないデータ インポートの要素を制御できます。

データ接続の管理

データ アクセス ウィンドウでは、Analytics データ コネクタを使用して作成された接続の名前変更、コピー、または削除ができます。

また、コネクタによって保存されたキャッシュ済みのテーブル名のリストを消去できます。

接続の名前変更、コピー、削除

メモ

データ アクセス ウィンドウでは、Windows ODBC ドライバーを使って作成された DSN 接続を名前変更、コピー、削除することはできません。これらの接続は Windows で管理できます。

- Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
- [既存の接続] タブの [ACL コネクタ] または [ACL DSN コネクタ(バンドル)] で、管理する接続にカーソルを置き、省略記号アイコン  をクリックします。
- 使用可能な場合は次のいずれかを選択します。

- コピーの作成

- コレクション名の変更
 - 接続の削除
4. 画面のプロンプトに従い、タスクを完了します。

キャッシュされているテーブル名の消去

データアクセス ウィンドウを使ってデータソースに接続できると、使用しているコネクタにより、データソースにある最初の200個のテーブルがキャッシュされます。このテーブル名のリストにより、今後、同じデータソースに今までよりも速く接続できるようになります。

データソースのスキーマ変更によりテーブルが追加、削除、または名前変更された場合は、テーブル名のキャッシュを手動で消去する必要があります。テーブル名のキャッシュとデータソースのスキーマが一致していないと、接続エラーが発生します。

キャッシュを消去してデータソースに接続できるようになると、キャッシュが、更新されたテーブル名のリストで更新されます。

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. [既存の接続] タブの [ACL コネクタ]、[ACL DSN コネクタ(バンドル)]、または [Windows DSN コネクタ] で、管理する接続にカーソルを置き、省略記号アイコン  をクリックします。
3. [キャッシュの消去] を選択します。
テーブル名のキャッシュが消去されます。

データアクセスのログファイル

2つのログファイルは、データアクセス ウィンドウで活動を記録し、データ接続ができない場合にトラブルシューティングに使用できます。

- **ServerDataAccess.log** - はデータをインポートする前のすべてのデータアクセス ウィンドウの活動とエラーを記録します

ロケーション: `C:\Users\<ユーザー アカウント>\AppData\Local\ACL\ACL for Windows\Data Access\ServerDataAccess.log`

メモ

`ServerDataAccess.log` の "Server" は Analytics がインストールされているコンピューターでローカルに実行されている Analytics のデータアクセス コンポーネントを参照します。

- **DataAccess.log** - には、インポート操作に関する情報と、

場所 `..\<Analytics プロジェクト フォルダー>\DataAccess.log`
にインポートしようとしているデータが含まれる Analytics プロジェクトに関する情報が記録されません。

データ アクセス ウィンドウの作業

データアクセスウィンドウを使用して、データをインポートして、Analytics テーブルを作成すると、データベースまたはファイルのいずれかでソース データに接続できます。

メモ

接続が失敗した場合は、2つのログファイルがトラブルシューティング時の支援となります。詳細については、「データアクセスのログファイル」前のページを参照してください。

データベースまたはクラウド データ サービスに接続する前に

データアクセス]ウィンドウを使用してデータベースまたはクラウド データ サービスに接続するときには、特定の要件または前提条件が存在します。

- **アカウント** - データベースまたはデータ サービスアカウントが必要です。アカウントは、個人アカウント、あるいは役割または御社に関連付けたアカウントにすることができます。このアカウントは Analytics アカウントとは異なります。
- **資格情報** - データベースまたはデータ サービスアカウントの有効な資格情報が必要です。必要なアカウント資格情報の種類はデータソースによって異なります。名前とパスワード、アクセストークン、一部の他の形式の資格情報を入力する必要がある場合があります。
- **事前設定** - Salesforce や Concur などのデータソースの場合、[データアクセス]ウィンドウを使用して接続する前に、クラウド データ サービス内で事前設定が必要です。

メモ

クラウド データ サービス内の接続前提条件の設定は、一般的に、御社の Salesforce 管理者または Concur 管理者など、サービスを管理する御社の責任者によって実行されます。

クラウド データ サービスに起因する接続の問題では、弊社サポートではなく、そのサービスを担当される御社内管理者にお問い合わせいただく必要があります。

データベースまたはファイルへの接続

Analytics のメインメニューから、[インポート > データベースとアプリケーション]を選択します。

メモ

データ定義ウィザードでデータとアプリケーションを選択することもできます。

既存の接続の使用

1. **既存の接続**]タブで、接続先のデータソースの接続を選択します。

ヒント

使用可能な接続のリストをフィルタリングするには、**接続のフィルタリング**]ボックスに検索文字列を入力します。名前はアルファベット順に表示されます。

接続の種類によっては、ソースデータに即時接続される場合があります。

既存の接続は、**ACL コネクタ**、**ACL DSN コネクタ**(バンドル)、**Windows DSN コネクタ**および**その他のコネクタ**の下で整理されます。

既存の接続がない場合は、セクションが表示されません。

2. すぐに接続しない場合は、次のいずれかを実行します。
 - **データ接続の設定**]パネルが開いたら、パネルの下で **接続**]をクリックします(スクロールが必要な場合があります)。
必要に応じて、接続設定を更新し、その後 **接続**]をクリックします。
 - ファイル接続のダイアログボックスが開いたら、適切なファイルへと移動し、これを選択します。

新規接続の作成

1. **新規接続**]タブを選択します。
2. 接続先のデータソースのコネクタを選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**]ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

使用可能なコネクタは、**ACL コネクタ**、**CL DSN コネクタ**(バンドル)、**Windows DSN コネクタ**および**その他のコネクタ**の下で整理されます。

3. 次のいずれかを実行します。
 - **データ接続の設定**]パネルが開いたら、接続設定を入力し、パネルの下で **保存して接続**]をクリックします(スクロールが必要な場合があります)。
デフォルトの接続名を使用するか、新しい名前を入力します。
 - **ACL DSN コネクタ**(バンドル)]セクションのコネクタで、**DSN 構成**]ダイアログボックスが開きます。
 - i. **必須フィールドを表示**]タブをクリックします。

- ii. 該当する場合は必須フィールドの値を入力し、**[Test Connection]**(接続のテスト)をクリックします。
データソースのログインページが表示されます。
 - iii. 接続詳細情報を入力し、ログインを認証します。
 - iv. 接続成功ダイアログボックスが表示されるので、**[OK]**をクリックします。
 - v. **[DSN Configuration]**ダイアログボックスで **[OK]**をクリックします。
- ファイル接続のダイアログボックスが開いたら、適切なファイルへと移動し、これを選択します。

メモ

Analytics コネクタで成功した接続は、自動的に **既存の接続** タブに表示されます。

Windows コネクタで行われた接続は、現在のデータインポートセクションでのみ保持されます。

データ接続の管理

ACL コネクタを使用して作成された接続の名前変更、コピー、または削除ができます。詳細については、「データ接続の管理」ページ 374を参照してください。

ステージング領域への1つまたは複数のテーブルの追加

1. データアクセスウィンドウで、必要に応じて **[スキーマ]** ドロップダウンリストから該当するデータベーススキーマを選択します。

メモ

一部のデータソースにはスキーマがない場合や、スキーマが1つしかない場合があります。

2. 省略可能。 **接続** パネルで、使用可能なテーブルのリストをフィルタリングするには、 **検索テーブル** ボックスに検索文字列を入力します。

リテラル検索文字列(ワイルドカードなし)の一致は、テーブル名のどの場所にも出現することがあります。検索は大文字と小文字を区別しません。

検索文字列では、1つ以上のワイルドカード文字を使用することもできます。

その他を表示

ワイルドカード	範囲	例	一致
* %	0 以上の文字	invoice-j*	<ul style="list-style-type: none"> Invoice-January Invoice-June Invoice-July
		%june	<ul style="list-style-type: none"> Invoice-June PO-June
		%invoice%	<ul style="list-style-type: none"> Invoice-June June-Invoice
? _	1 文字	invoice??	<ul style="list-style-type: none"> invoice-1 invoice-2
		j_n	<ul style="list-style-type: none"> 1月 6月

メモ

検索文字列でワイルドカードを使用する場合、一致する文字列の長さ、最初と最後の文字の長さがより厳密に評価されます。

- j_n は j で始まり n で終わる 3 文字の文字列とのみ一致します
- j*n は任意の長さの文字列と一致しますが、文字列は j から始まり n で終わる必要があります
- 対照的に、jan は任意の長さの文字列と一致し、jan は文字列のどの場所にも出現することがあります

この一致動作は、Analytics が次の方法でリテラル検索文字列 (ワイルドカードなし) を解釈する結果です。jan = *jan*

3. 省略可能。テーブルのリストの下部にスクロールし、**残りの # テーブルを表示]**をクリックします。

Analyticsはデータソースの最初の 200 テーブルを表示します。追加のテーブルが存在している場合は、**残りを表示]**リンクをクリックして、一度に最大 500 テーブルのブロックで表示できます。

メモ

リンクが表示されるには、**テーブルの検索]**ボックスが空欄でなければなりません。

4. **利用可能なテーブル]**で、テーブル名をクリックしてテーブルを **ステージング領域]**に追加します。

テーブルはアルファベット順に表示されます。テーブルを結合する予定の場合は、ステージング領域に最大で 10 個のテーブルを追加できます。SAP コネクタは現在 2 つのテーブルに制限されています。

メモ

1 度のインポートの操作で複数のテーブルを個別にインポートすることはできません。テーブルは、インポートされるようにまとめて結合される必要があります。

5. 省略可能。使用可能なテーブルのリストにシステムテーブルを追加する場合は、**システムテーブルを含む]**を選択します。

メモ

このオプションは一部のデータソースには適用されません。

テーブルの結合

複数のテーブルをステージングエリアに追加した場合は、テーブルを結合する必要があります。

テーブルの結合の詳細については、「データアクセスウィンドウのテーブルの結合」ページ 388を参照してください。Apache Drill テーブルの結合の詳細については、「Apache Drill データソースからのテーブルの結合」ページ 391を参照してください。

1. **[ステージング領域]**で、結合アイコンをクリックし、**結合の設定**にアクセスします。



2. 任意の結合のタイプをクリックします。

- 内部
- その他
- 左
- 右

メモ

Microsoft Excel および Microsoft Access コネクタを含む一部のデータコネクタは、**外部結合タイプ**をサポートしません。

3. 次の手順で共通キーフィールドを選択します。
 - a. **左の列**から、左テーブルのキーフィールドを選択します。
 - b. **右の列**から、右テーブルのキーフィールドを選択します。

ヒント

使用可能なフィールドのリストをフィルタリングするには、**左の列]**または**右の列]**ボックスに検索文字列を入力します。フィールドはアルファベット順に表示されます。

4. 省略可能。追加のキーフィールドを追加する必要がある場合は、**[キーの追加]**をクリックします。
5. **[適用]**をクリックし、結合設定を保存します。
6. 結合する各追加テーブルの結合設定を作成します。
7. 省略可能。**[インポートプレビュー]**パネルで、**[更新]**をクリックして、結合されたテーブルのプレビューを表示します。

インポートするファイルの選択

デフォルトでは、特定のフィールドを選択解除しないかぎり、テーブルのすべてのフィールドがインポートされます。

1. インポートから1つ以上のフィールドを省略するには、テーブルで **[フィールドを表示する]** ドロップダウンリストをクリックします。
2. 選択解除する場合はフィールド名をクリックします。

ヒント

ほとんどのフィールドを選択解除する場合は、**[すべて選択]** トグルをクリックし、すべてのフィールドを選択解除してから、任意のフィールドを再度選択できます。

3. **[インポート プレビュー]** パネルで、**[更新]** をクリックして、インポートに含まれるフィールドをレビューします。
4. 省略可能。暗号ハッシュ値として1つ以上のフィールドをインポートするには:
 - a. **[インポート プレビュー]** パネルで、変換する列の上部にある **[ハッシュ]** チェックボックスを選択します。

- b. 画面の下部にある **[Salt]** フィールドで、ハッシュ関数で使用する英数字文字列を入力します。

ソルト値は128文字以下です。次の文字は使用しないでください。 `()"`

ソルト値を指定しない場合、Analytics はランダム文字列を生成します。**[更新]** をクリックすると、**[Salt]** フィールドの生成された文字列が表示されます。

ハッシュ値は単方向の変換であり、フィールドのインポート後に復号化できません。

メモ

ハッシュデータの未加工値を読み取れない場合でも、データの結合または分析時に役立ちます。インポート中にACCESSDATAでハッシュ化された値を、ACLScript's HASH() 関数を使用してハッシュ化された値と比較する場合は、数値または日付時刻 Analytics フィールドを文字値に変換し、データをハッシュ化する前にスペースを切り取る必要があります。

日付時刻フィールドは次の形式を使用する必要があります。

- 日付時刻 - "YYYY-MM-DD hh:mm:ss"
- 日付 - "YYYY-MM-DD"
- 時刻 - "hh:mm:ss"

データのフィルター

デフォルトでは、1つ以上のフィルターを作成して特定のレコードを省略しない場合は、テーブルのすべてのレコードがインポートされます。

メモ

下記のフィルターオプションを両方とも使用する場合は、まず条件付きフィルターが最初に適用され、次に条件付きフィルターの結果に対し、指定した数のレコードが適用されます。

インポートするレコードの数を指定

テーブルの先頭から数えて、特定の行数までのレコードのみをインポートするように指定できます。

1. **最初の n レコードを選択**] で、テーブルの先頭から数えたインポート対象レコード数を入力します。
2. 省略可能。[**インポート プレビュー**] パネルで、**更新**] をクリックして、インポートに含まれるレコードを表示します。

ヒント

インポート対象をテーブル内のすべてのレコードに再設定するには、**最初の n レコードを選択**] に「 n 」を入力します。

条件付きフィルターを作成

1. インポート対象からレコードを条件によって除外する場合は、**フィルターを追加して結果を制限します**] をクリックして、フィルターを作成します。
2. **フィールド**] リストで、フィルタリングで使用するフィールドを選択します。

ヒント

使用可能なフィールドのリストをフィルタリングするには、**フィールド**] ボックスに検索文字列を入力します。フィールドはアルファベット順に表示されます。

メモ

テーブルを結合した場合は、結合したテーブルのいずれかからフィールドを選択できます。

3. **条件**] リストから、「である」、「等しい」または「より大きい」などの条件付き演算子を選択します。「にある」演算子を使用すると、複数のテスト値を指定できます。詳細については、「条件付き演算子「にある」の使用」 ページ 384 を参照してください。

4. 3番目のフィールドでテストする値を入力します。

メモ

論理フィールドを使用してフィルタリングしている場合は、データソースに応じてテスト値を次のいずれかにする必要があります。

- 'true' または 'false' (引用符を含む)
- 1 または 0 (1 = true、0 = false)

フィールドの実際の値のいずれかを使用したフィルタリングがエラーを返す場合は、上記の値のいずれかを試してください。

5. 省略可能。別のフィルターを追加するには、次の手順を実行します。

- a. **フィルターを追加**] をクリックします。

- b. フィルターを組み合わせる方法によって、**AND** または **OR** を選択します。

- c. 手順 2 ~ 4 を繰り返してフィルターを作成します。

フィルターをフィルターグループに追加し続け、インポートするデータの正確なセットを指定できません。

メモ

フィルターグループで複数のフィルターを組み合わせるときには、Boolean 演算子を組み合わせることができません。グループのすべてのフィルターは **AND** または **OR** を使用して組み合わせる必要があります。

6. 省略可能。フィルターグループを追加するには、次の手順を実行します。

- a. **フィルターグループを追加**] をクリックします。

- b. フィルターグループを組み合わせる方法によって、**AND** または **OR** を選択します。

- c. 手順 2 ~ 4 を繰り返してフィルターを作成します。

フィルターをフィルターグループに追加し続けるか、追加のフィルターグループを作成して、インポートするデータの正確なセットを指定できます。

メモ

各フィルターグループのフィルターが最初に評価されてから、フィルターグループが相互に比較して評価されます。

複数のフィルターグループを組み合わせるときには、Boolean 演算子を組み合わせることができません。すべてのフィルターグループは **AND** または **OR** を使用して組み合わせる必要があります。

7. 省略可能。[**インポート プレビュー**] パネルで、**更新**] をクリックして、インポートに含まれるレコードを表示します。

条件付き演算子 "にある" の使用

"にある" 演算子を使用すると、複数のテスト値を指定できます。たとえば、[City]フィールドに対し、インポート対象レコードを次の特定の都市にのみ制限する条件付きフィルターを作成できます。

```
New York、San Francisco、Dallas
```

条件付き演算子 "にある" には次の規則が適用されます。

- テスト値はカンマで区切る。テスト値にはスペースを含めることができる(上記の例を参照)。
- 1つまたは複数の一重引用符 `'` が含まれているテスト値は、二重引用符 `"` で囲みます。データコネクタによっては、一重引用符文字をエスケープ処理しなければならない場合もあります。例: `"\'abc123\'"`
- 1つまたは複数の二重引用符 `"` が含まれているテスト値は、一重引用符 `'` で囲みます。
- 次の文字のいずれかが含まれているテスト値は、二重引用符 `"` または一重引用符 `'` で囲みます:カンマ `,` または円記号 `\`
円記号の後には少なくとも1つの文字が続く必要があります。例: `"\a"` または `"\\"`
- 1つのフィルター内でテスト値を囲むのに二重引用符 `"` と一重引用符 `'` を両方使用することはおやめください。どちらか一方のみを使用してください。

最大フィールド長の調整

インポートされた文字またはメモフィールドのデフォルト最大フィールド長が短すぎるか長すぎる場合は、調整できます。

最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。

メモ

フィールド長は個別に指定できません。単一の設定はすべての文字フィールドと、インポートのすべてのテーブルのすべてのメモフィールドに適用されます。

ヒント

インポートプレビューの最初の数個の値に基づいて、フィールドを短くする場合は注意してください。値の長さが長い場合は、テーブルが切り捨てられる場合があります。

1. データアクセスウィンドウの下部で、これらのフィールドのいずれかまたは両方の文字数を増減させます。
 - 最大文字フィールド長
 - 最大メモフィールド長

2. **[インポート プレビュー]** パネルで、**[更新]** をクリックして、プレビューのフィールド長を更新します。

メモ

列のすべてのテキストを表示するには、プレビュー列をドラッグして広げる必要があります場合があります。

すべてのフィールドを文字データ型としてインポートする

すべてのフィールドを文字データとしてインポートする場合は、**[すべての文字]** を選択します。

すべてのフィールドを文字データとしてインポートすると、インポート処理が簡素化され、データを損失することなく煩わしいフィールドを Analytics で取得できます。Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。

[すべての文字] オプションは、実際に文字データ型を使用する必要がある場合に、Analytics によって数値のデータ型が自動的に割り当てられた、識別子のフィールドを持つテーブルをインポートする際に有用となります。

SQL のインポート文を編集する

SQL を理解している場合は、直接 SQL インポート文を編集できます。

SQL インポート文を編集すると、ユーザーインターフェイスでは使用できないデータインポートの要素を制御できます。たとえば、SQL インポート文のフィールド名を変更できます。

注意

SQL Mode で行った変更は、データアクセス ウィンドウでビジュアルエディターに戻ると、すべて失われます。

1. 省略可能。 **[SQL Mode]** トグル ボタンをクリックします。
2. SQL のインポート文を編集します。

メモ

SQL インポート文の本文では、ACLScript 構文 (コマンドまたは関数) を使用できません。有効な SQL 構文のみを使用する必要があります。

3. **[インポートのプレビュー]** 領域でインポートされるデータに対する更新された SQL の効果を表示するには、**[更新]** をクリックします。

インポートのプレビュー

インポート処理の任意の時点で、インポートをプレビューし、結合、フィールド選択、フィルタリング、およびフィールド長調整の効果を確認できます。

1. 省略可能。インポートのレコード数の推定と、作成される Analytics データ ファイル (.fil) のサイズを表示する場合は、**[インポート プレビュー]** 領域で、**[サイズの推定]** を選択します。

注意

[サイズの推定] オプションは注意して使用してください。大きいデータセットと特定のデータソースの場合、推定の生成はプロセッサのリソースを消費し、速度が低下する可能性があります。

2. **[インポートのプレビュー]** 領域でインポートされるデータを表示するには、**[更新]** をクリックします。

Analytics データ ファイルの保存

任意のデータセットを指定するときには、インポートされたデータを Analytics データ ファイルに保存します。

1. データ アクセス ウィンドウの下で **[保存]** をクリックします。
2. Analytics テーブルの名前を指定し、**[保存]** をクリックします。

データがインポートされ、新しいテーブルが自動的に Analytics で開きます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

Analytics テーブルとデータ ファイルの更新

データ アクセス ウィンドウを使用してインポートした Analytics テーブルと関連付けられたデータ ファイルを更新できます。テーブルを更新すると、最新のソース データでテーブルが更新されます。

ガイドライン

- **コンテンツのみが更新されます** - Analytics テーブルを更新すると、既存のフィールドのコンテンツのみが更新されます。テーブルレイアウトは更新されません。

ソースデータの構造が変更された場合 (フィールドが追加または削除された場合など) は、テーブルを更新できません。データを再インポートする必要があります。

- **開いているテーブル**- 更新するときにテーブルが開いている場合は、一時的に、テーブルのサイズの2倍に相当するディスク領域が必要です。ディスク領域が限られている場合は、更新する前にテーブルを閉じてください。
- **Analytics 12 でインポートされたテーブル**- Analytics バージョン 12 でデータ アクセス ウィンドウを使用してインポートされたテーブルは、新しいバージョンの Analytics を使用している場合でも更新できません。

これらのテーブルを更新するには、Analytics 12.5 以降を使用して再インポートしてください。

手順

1. ナビゲーターで更新する Analytics テーブルを右クリックし、**[ソース ファイルから更新]**を選択します。
2. 確認ダイアログ ボックスで、**[はい]**をクリックします。
3. **[パスワード プロンプト]**が表示される場合は、データソースのパスワードを入力し、**[OK]**をクリックします。

メモ

別のアカウントを使用してデータソースにアクセスする場合は、ユーザー名も変更できます。

4. 1 つ以上のプロンプトが表示され、変更を保存するかどうかを確認された場合は、変更を保存しない場合は、**[はい]**をクリックします。

テーブルが更新されます。

データ アクセス ウィンドウのテーブルの結合

データ アクセス ウィンドウを使用すると、1 回のインポート処理で最大 10 個のテーブルをインポートできます。

まとめてインポートするには、テーブルを結合する必要があります。1 度のインポートの操作で複数のテーブルを個別にインポートすることはできません。

メモ

Analytics で Analytics テーブルを結合することについて説明します。"テーブルの結合" ページ 960を参照してください。

このトピックでは、データ インポート処理の一部として、データ アクセス ウィンドウでテーブルを結合する方法について説明します。

テーブルの結合方法

データ アクセス ウィンドウでのテーブルの結合は、ソースデータの最大 10 個のテーブルを選択し、**ステージングエリア**に追加してから、すべてのテーブルが結合されるまで 1 度に 2 つのテーブルを結合する処理です。

共通キーフィールドを使用して最初の 2 つのテーブルを結合します。つまり、両方のテーブルにある「顧客 ID」などのデータ要素を使用します。キーフィールドに同一の値が存在する場合は 1 つの一致のみが生じます。この一致は別のテーブルのレコードを個別に結合します。

3 つ以上のテーブルを結合する場合は、2 つのテーブルの共通キーフィールドを使用して、2 番目のテーブルを 3 番目のテーブルに結合します。すべてのテーブルが結合されるまで繰り返します。

例

売掛金データを操作し、顧客の一覧、各顧客の注文、および各注文の製品の詳細が必要であるとします。

このデータを作成し、Analytics にインポートするには、ソースデータシステムから 3 つのテーブルを結合する必要があります。

- **結合 #1** - 両方のテーブルに表示される `CustID` のキーフィールドを使用して、**Customer** および **Orders** テーブルを結合します。
- **結合 #2** - 両方のテーブルに表示される `ProdID` のキーフィールドを使用して、**Order** および **Product** テーブルを結合します。

以下の図では、結合 #1 のみが完了しているため、2 番目の結合アイコンはまだ赤色です。

ステージング領域

設定の結合

内部 その他

左 右

左列 右列

`Customer`.`CustID` CustID

+ キーの追加

適用 キャンセル

ヒント

データアクセス ウィンドウでこの例の結合を試すことができます。Microsoft Access コネクタを使用し、Analytics に同梱されている次のサンプルの Microsoft Access ファイルに接続します。

..\ACL Data\Sample Data Files\Sample.mdb

結合タイプ

テーブルを結合するときには、4つの異なる結合タイプから選択できます。選択する結合タイプは、結合されたテーブルに含まれる2つの元のテーブルのレコードを制御します。

左テーブルと右テーブル

2つの元のテーブルは、選択する順序に基づいて、「左テーブル」と「右テーブル」として識別されます。

- 左テーブル- ステージング エリアに追加する最初のテーブル
- 右テーブル- ステージング エリアに追加する2番目のテーブル

複数のテーブルの結合

3つ以上のテーブルをステージングエリアに追加する場合は、左テーブルが結合する2つのテーブルの間の結合アイコンの左側になり、右テーブルが右側になります。

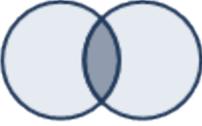
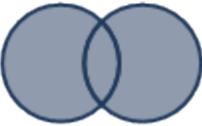
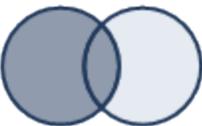
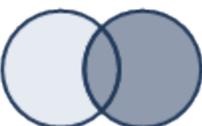
上記の例：

- 結合 #1 - **Customer** が左テーブルで、**Orders** が右テーブルです
- 結合 #2 - **Orders** が左テーブルで、**Product** が右テーブルです

参照する結合によって、**Orders** は右テーブルまたは左テーブルになることができることに注意してください。

結合後のテーブルに含まれるレコード

元のテーブルの一致したレコードのみを結合テーブルに含めるか、一致しないレコードも含めるのかを選択できます。

結合タイプ		結合後のテーブルに含まれるレコード			
		左テーブル内の一致するレコード	左テーブル内の一致しないレコード	右テーブル内の一致するレコード	右テーブル内の一致しないレコード
	内部	✓		✓	
	その他	✓	✓	✓	✓
	左	✓	✓	✓	
	右	✓		✓	✓

複数のキーフィールドを使用した結合

単一のキーフィールドが十分に一意ではなく、テーブルを正確に結合できない場合は、複数のキーフィールドを使用して2つのテーブルを結合しなければならない場合があります。

例

業者 ID で2つのテーブルを結合します。一部の業者にはロケーションがあり、各ロケーションのレコードを個別に保持したいとします。この結果を得るには、Vendor ID および Location フィールドの両方をキーフィールドとして使用します。

Vendor ID のみをキーフィールドとして使用する場合は、個別の業者のロケーションのレコードが混在します。

Location のみをキーフィールドとして使用する場合は、異なる業者のレコードが混在します。

業者 ID	場所
A-4538	バンクーバー
A-4538	バーナビー
A-4538	リッチモンド
B-2204	バンクーバー
B-2204	バーナビー

Apache Drill データソースからのテーブルの結合

データアクセスウィンドウで視覚エディターを使用すると、Apache Drill データソースから2つのテーブルのみを結合できます。

Drill データソースから3つ以上のテーブルを結合するには、**SQL モード**を使用し、括弧を使用しない結合文を作成する必要があります。

視覚エディターを使用して作成された3つ以上のテーブルの結合では、括弧が結合文内に入りますが、これは Drill からのインポートではサポートされません。

Active Directory への接続

Active Directory は Microsoft の Directory Server であり、ユーザー、グループ、コンピューターなどのオブジェクトを含む LDAP 対応 データベースを提供します。Active Directory データをインポートするには、Active Directory データ コネクターを使用します。

メモ

このコネクターを Analytics から独立して使用することはできません。コネクタードライバーの DSN は Windows の ODBC データソース アドミニストレーターを使って設定できますが、DSN 接続を本コネクターの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

Active Directory に接続するには、以下を収集する必要があります。

- Active Directory サーバーのドメイン名または IP アドレス
- 正しい接続ポート
- ユーザー名とパスワードの識別名を含む接続ユーザー アカウント

接続の前提条件の収集については、組織の Active Directory 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Active Directory のサポートにお問い合わせください。

Active Directory 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクター セクションの **新規の接続** タブから、Active Directory を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Active Directory の接続は、**既存の接続** タブに保存されます。今後、Active Directory に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Active Directoryからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	ユーザーの識別名。 パスワードとともに、このフィールドは、Active Directory サーバーで認証するために使用されます。	MYDOMAIN\test
パスワード	指定したユーザーの識別名のパスワード。 ユーザーとともに、このフィールドは、Active Directory サーバーで認証するために使用されます。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>お使いの Active Directory サーバーで匿名接続が許可される場合は、パスワードを指定しなくても接続を行うことができます。お使いのサーバーのセキュリティ設定によっては、匿名接続であっても、使用可能なテーブルをリストすることができます。ただし、匿名接続では、リストされたテーブルの一部または全部からデータを選択できない場合があります。お使いの Active Directory セキュリティ設定の詳細については、御社の管理者にお問い合わせください。</p> </div>	
サーバー	Active Directory サーバーのドメイン名または IP これには、サーバーのドメイン名または IP のみを含めればよく、LDAP:\ の部分を含める必要はありません。	10.120.1.110
ポート	Active Directory サーバーを実行する	389

設定	説明	例
	ポート。デフォルト値は389です。 このプロパティとサーバーを組み合わせ、Active Directory サーバーを指定します。	
ベース DN	結果を特定のサブツリーに制限するために使用される、識別名のベース部分。 ベース DN を指定すると、検査が必要なエントリ数を制限することで、大きいサーバーのエントリを返すときに、パフォーマンスを大幅に改善できます。	DC=myConnection,DC=com
LDAP バージョン	サーバーに接続し、サーバーと通信するために使用される LDAP バージョン。有効なオプションは、LDAP バージョン 2 および 3 で 2 と 3 です。	2
認証機構	Active Directory サーバーに接続するときに使用される認証メカニズム： <ul style="list-style-type: none"> ◦ SIMPLE(デフォルト) - デフォルトのテキスト認証がサーバーへのログインで使用されます ◦ DIGESTMD5 - より安全な DIGEST-MD5 認証が使用されます ◦ NEGOTIATE - NTLM/ネゴシエート認証が使用されます 	SIMPLE
範囲	検索範囲を制限するかどうか： <ul style="list-style-type: none"> ◦ WholeSubtree - サブツリー全体 (BaseDN とすべての子) ◦ SingleLevel - 単一レベル (BaseDN と直接の子) ◦ BaseObject - ベースオブジェクト (BaseDN のみ) <p>ヒント 範囲を制限すると、各パフォーマンスが大幅に改善されます。</p>	BaseObject

詳細設定

設定	説明	例
デフォルト DC を使用する	デフォルトドメインコントローラーに接続し、現在のユーザー資格情報を使用し	false

設定	説明	例
	て接続するために使用されます。	
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	true
鍵サイズの上 限	主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されま す。	-1
WVarchar にマッピング	文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用 できます。	true
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。	MyTable=*
大文字の ID	テーブルおよび列名を含む、すべての識別子を大文字で報告します。	false
SSL サーバー証明書	TLS/SSL を使用して接続するときにサーバーから許可される証明書。次の	C:\cert.cer

設定	説明	例
	<p>いずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サンプリント(16進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サンプリント(16進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	
拡張 SQL のサポート	<p>メモリ内クライアント側処理を有効にし、API 経由で直接サポートされるよりも、SQL 機能を強化します。</p> <ul style="list-style-type: none"> ◦ true - コネクタはできるかぎり SELECT 文処理の負荷を Active Directory に分散し、メモリのクエリの残りを処理します。この方法で、ドライバはサポートされていない述語、結合、集計を実行できます。 ◦ false - コネクタは SQL 実行を Active Directory API のサポート内容に制限します。 <p>メモ Where 節構文を使用してフィルタリングをサポートするには、この設定を false にする必要があります。</p> <p>述語の実行</p> <p>コネクタは、データソースでサポートされている節を決定してから、ソースにプッシュし、クエリを満たす行の最も小さいスーパーセットを取得します。その後、行の残りをローカルでフィルタリングします。フィルター処理はストリームされ、ドライバは非常に大きいデータセットでも効果的にフィルタリングできます。</p> <p>結合の実行</p>	false

設定	説明	例
	<p>コネクタはさまざまな手法を使用してメモリで結合します。ドライバーは、同じテーブルを複数回読み取る要件に対して、メモリ利用率をトレードオフします。</p> <p>集計の実行</p> <p>コネクタは、メモリ内の集計を処理するために必要なすべての行を取得します。</p>	

返される行のフィルタリング

Active Directory コネクタは、LDAP 検索構文と近く一致する SQL フィルタリング構文を使用します。一部のフィールドには、複数のオブジェクト属性を表す区切られたデータがあります。WHERE 節は、単一の文字列ではなく、一意の値であるかのように、これらの区切られたフィールドの各値を占める必要があります。

LDAP 検索フィルターの詳細については、[MSDN 検索フィルター構文](#)を参照してください。

ObjectCategory と ObjectClass のユーザーのフィルタリング

シナリオ

ユーザーテーブルを操作し、ObjectClass に次の属性があるレコードをインポートします。

- ユーザー
- ユーザー

また、ObjectCategory にユーザーではなく、コンピューター属性があるものにレコードを制限します。

テーブルへの接続

まず、Active Directory サーバーに接続し、ユーザーテーブルを選択します(表示されるフィールドのサブセット)。

ObjectCategory (ASCII)	ObjectClass (ASCII)	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	

レコードのフィルターリング

インポートするものにレコードを制限するには、個別フィールドとして各区切られた値を処理するフィルターを適用します。

+ 3 フィルター適用済み - 編集 ^

"User"."ObjectClass"	である	person	🗑️
AND OR			
"User"."ObjectClass"	である	user	🗑️
AND OR			
"User"."ObjectCategory"	である	Computer	🗑️

+ フィルターの追加

SQL モードを使用して、フィルターが作成する WHERE 節を検証します。

```
WHERE
(
  "User"."ObjectClass" = N'person' AND
  "User"."ObjectClass" = N'user' AND
  "User"."ObjectCategory" = N'Computer'
)
```

フィルター結果

フィルターが適用されると、テーブルには WHERE 節と一致するレコードが含まれ、テーブルをインポートします。

ObjectCategory (ASCII)	ObjectClass (ASCII)	
CN=Computer,CN=Schema,CN=Configuration,	top;person;organizationalPerson;user;computer	

Active Directory テーブルの結合

Active Directory などの LDAP 準拠データベースで使用されるデータモデルのため、SQL 結合は推奨されません。結合により、予期しない結果になる場合があります。

1 つ以上のテーブルを Active Directory データソースから結合する必要がある場合は、データアクセスウィンドウを使用して複数のテーブルをインポートしてから、Analytics で結合します。フィルターを使用して、レコード数を制限し、効率化します。

ADP への接続

ADP は、オンライン給与・人事ソリューションです。ADP データコネクターを使用して、組織の ADP データをインポートできます。

ADP データコネクターは、Analytics のバージョン 16.1 以降で利用可能です。

メモ

ADP データコネクターは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の ADP 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から ADP のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

ADP 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター(バンドル)** セクションの **新規の接続** タブから、**ADP** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
ADPへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

ADPの接続は、**[既存の接続]** タブに保存されます。今後、ADPに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。ADPからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

ADP データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、ADP に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Airtable への接続

Airtable はクラウドベースのコラボレーションプラットフォームであり、スプレッドシートの機能と関係データベースを組み合わせています。Airtable データコネクタを使用して、組織の xAirtable データをインポートできます。

メモ

Airtable データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Airtable 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Airtable のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Airtable 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Airtable を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Airtableへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Airtableの接続は、 **[既存の接続]** タブに保存されます。今後、Airtableに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Airtableからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Airtable データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Airtable に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Amazon Athena への接続

Amazon Athena は、標準の SQL を使用して Amazon S3 のデータの問い合わせができる対話型クエリサービスです。Amazon Athena データコネクタを使用して、組織の Amazon Athena データをインポートできます。

メモ

Amazon Athena データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Amazon Athena に接続するには、以下を収集する必要があります。

- Amazon Athena がクエリの結果を保存する S3 Staging Directory の場所
- AWS アカウント アクセス キー
- AWS アカウント シークレット キー
- Amazon Web Server がホストされる AWS リージョン

接続の前提条件の収集については、組織の Amazon Athena 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Amazon Athena のサポートにお問い合わせください。

Amazon Athena 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Amazon Athena を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Amazon Athenaへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Amazon Athenaの接続は、**既存の接続** タブに保存されます。今後、Amazon Athenaに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Amazon Athenaからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Amazon Athena データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **既存の接続** タブに保存されます。名前を変更する場合は、**既存の接続** タブに移動し、**[ACL DSNコネクタ]** セクションを展開して、Amazon Athena に対応する省略記号アイコン  をクリックし、**接続名の変更** を選択します。

Amazon DynamoDB への接続

Amazon DynamoDB はクラウド データ サービスです。DynamoDB データ コネクタを使用して、Amazon DynamoDB データをインポートできます。

メモ

Analytics は任意のコネクタとして DynamoDB を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実行する" ページ 2663を参照してください。

作業を開始する前に

DynamoDB に接続するには、以下を収集する必要があります。

- Amazon Web Services のホスト 地域。
- AWS アカウント アクセスキー。
- AWS アカウント シークレット キー。

接続の前提条件の収集については、組織の DynamoDB 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から DynamoDB のサポートにお問い合わせください。

DynamoDB 接続を作成します。

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、DynamoDB を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

DynamoDB の接続は、**既存の接続** タブに保存されます。今後、DynamoDB に接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。DynamoDB からデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
エンドポイント	<p>通信のエンドポイント。</p> <p>AWS に接続している場合は、AWS で DynamoDB サービスのエンドポイントを指定します。</p> <p>DynamoDB Local に接続している場合は、サーバーの IP アドレスを指定します。</p>	<p>dynamodb. [Region].amazonaws.com: [Region] は接続で使用する AWS リージョンです。</p>
DynamoDB Local	<p>AWS で DynamoDB Local サーバーまたは DynamoDB サービスのどちらに接続するかどうかを指定します。</p> <p>DynamoDB Local はクライアント側データベースであり、DynamoDB API 全体をサポートしますが、DynamoDB 自体のテーブルやデータを操作することはできません。</p>	有効
ポート	<p>DynamoDB Local サーバーがクライアント接続をリスニングするために使用する TCP ポートの番号。</p> <p>DynamoDB Local のデフォルト ポートは 8000 です。</p>	8000
地域	Amazon Web Services のホスト地域。	NORTHERNVIRGINIA
認証	<p>ドライバーで DynamoDB への接続を認証する方法を指定します。</p> <ul style="list-style-type: none"> 無効 - ドライバーは、アクセス キーとシークレット キーを使用して、DynamoDB への接続を認証します。 有効 - ドライバーは、資格情報ファイルのプロファイルを使用して、DynamoDB への接続を認証します。 	無効
資格情報ファイル	<p>資格情報ファイルの完全パスと名前。MFA 資格情報が保存されます。</p> <p>デフォルトの場所は %APPDATA%\CDData\AmazonDynamoDB Data Provider\CredentialsFile.txt です。</p>	
プロファイル名	AWS 資格情報ファイルから使用するプロファイルの名前。	
アクセス キー	AWS アカウント アクセス鍵。この値には、AWS セ	

設定	説明	例
	セキュリティ資格情報ページからアクセスできます。	
秘密鍵	AWS アカウント シークレット キー。この値には、AWS セキュリティ資格情報ページからアクセスできます。	
一時セッションの有効化	<p>ドライバーで一時資格情報を使用するかどうかを指定します。</p> <ul style="list-style-type: none"> 無効 - ドライバーは一時資格情報を使用しません。 <p>有効 - ドライバーは、一時資格情報ファイルを使用して、DynamoDB への接続を認証します。一時資格情報は、アクセス キー、シークレット キー、セッショントークンから構成されます。これは一定時間のみの有効です。</p>	
一時セッションのトークン	一時セキュリティ資格情報を使用して DynamoDB に接続するときに使用するセッショントークン。これは一定時間のみの有効です。	3600

詳細設定

設定	説明	例
再試行回数	回復可能なエラーにより要求が失敗した場合に、ドライバーが DynamoDB データベースに要求を送信する最大回数。	10
スループットの制限	<p>ドライバーが使用可能なプロビジョニングされた読み取り単位の割合。デフォルトでは、ドライバーは、プロビジョニングされたスループットの 30% まで使用できます。</p> <p>必ず、特定のテーブルを同時に使用するクライアント アプリケーション数に適した値を使用してください。たとえば、この値を 100 に設定する場合、1 つのクライアントがすべてのプロビジョニング済みスループット単位を使用するため、他のクライアントはその他のスループットが使用可能になるまでテーブルを使用できません。</p>	30
アクティブなメタデータの場所	データベースとローカルファイルのどちらからスキーマ定義を使用するのかを指定します。	データベース

設定	説明	例
データベース	DynamoDB に接続するときにドライバーで使用するスキーマ定義を含むデータベーステーブルの名前。	
ローカルファイル	DynamoDB に接続するときにドライバーで使用するスキーマ定義を含むローカルJSON ファイルの完全パス。	

データコネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データコネクタを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータコネクタを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセスウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセスウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Amazon Redshift への接続

Amazon Redshift はビジネス インテリジェンスで使用されるクラウド ウェアハウス サービスです。Amazon Redshift データをインポートするには、Amazon Redshift データコネクターを使用します。

メモ

Analytics は任意のコネクターとして Amazon Redshift を提供します。データ アクセス ウィンドウでこのコネクターを使用できない場合、一般的に、インストール中にコネクターが選択されていないことが原因です。詳細については、「任意の Analytics データコネクターと Python エンジンを実装する」ページ 2663を参照してください。

作業を開始する前に

Amazon Redshift に接続するには、以下を収集する必要があります。

- Amazon Redshift クラスターの ODBC URL からのサーバーおよびデータベース名
- データベースに接続する権限があるユーザー アカウントのユーザー名
- データベースに接続する権限があるユーザー アカウントのパスワード
- クラスターが起動したときに指定されたポート番号 (ファイアウォールがある場合はポートが開いていることを確認する)

メモ

必要な接続情報を収集するとき

- AWS 管理コンソールから ODBC URL を取得するには、クラスターのデータベースプロパティを確認します
- 接続アカウントに、Amazon Redshift 権限だけでなく、データベース権限があることを確認します

ODBC 接続の構成については、[Amazon AWS ドキュメント](#)を参照してください。

接続の前提条件の収集については、組織の Amazon Redshift 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Amazon Redshift のサポートにお問い合わせください。

Amazon Redshift 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続**] タブから、Amazon Redshift を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定**] パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続**] をクリックします。

デフォルトの**接続名**を使用するか、新しい接続名を入力します。

Amazon Redshiftの接続は、**既存の接続**] タブに保存されます。今後、Amazon Redshiftに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Amazon Redshiftからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
サーバー	Redshift クラスターのホスト名または IP アドレス。この値は AWS Management Console から取得できません。	
ポート	Amazon Redshift サーバーの TCP ポート番号。	5439
データベース	Redshift データベースの名前。 データベースを指定しない場合は、接続はユーザーのデフォルト データベースを使用します。	
ユーザー	Redshift サーバーにアクセスするためのユーザー名。 キーを使用してドライバー オプションを設定している場合は、UID がユーザー名よりも優先されます。	
パスワード	このユーザーを認証するのに使用する	

設定	説明	例
	パスワード	
SSL 認証	<p>Redshift に接続するとき使用する SSL 証明書検証モード。使用可能な値は次のとおりです。</p> <ul style="list-style-type: none"> ◦ 許可 - デフォルトでは、SSL を使用せずに接続します。サーバーで SSL 接続が必要な場合は、SSL を使用します。 ◦ 無効 - SSL を使用せずに接続します。 ◦ 優先 - 使用可能な場合は、SSL を使用して接続します。使用できない場合は、SSL を使用せずに接続します。 ◦ 必須 - 必ず SSL を使用して接続します。 ◦ verify-ca - 必ず SSL と信頼できる認証局を使用して接続します。 ◦ verify-full - 必ず SSL、信頼できる認証局、証明書と一致するサーバー名を使用して接続します。 	要求
HTTP プロキシ接続の有効化	ドライバーはプロキシ サーバー経由で IAM 認証プロセスを通過できるかどうかを指定します。	
プロキシ サーバー	IAM 認証プロセスが経由するプロキシサーバーのホスト名または IP アドレス。	
プロキシのポート	クライアント接続をリスニングするためにプロキシサーバーが使用するポート番号。	

詳細設定

設定	説明	例
Unicode の使用	ドライバーが Redshift データを Unicode として返すのか、標準 SQL タイプとして返す野かを指定します。	
ブール値の列を文字列として表示	<p>ドライバーがブール値データを返すために使用する SQL データ型を指定します。</p> <p>有効な場合、ドライバーは長さ5の SQL_VARCHAR データとしてブール値列を返します。無効な場合、長さ5の SQL_BIT データとして返します。</p>	

設定	説明	例
LongVarChar 型のテキスト	ドライバーがテキスト データを返すために使用する SQL データ型を指定します。 無効な場合、ドライバーは SQL_VARCHAR データとしてテキスト列を返します。	
最大 Varchar	VARCHAR 列の最大データ長。	255
最大 LongVarChar	LongVarChar 列の最大データ長。	8190
最大 Bytea	Bytea 列の最大データ長。	255

Amazon S3 への接続

Amazon Simple Storage Service (S3) は、Web サービス インターフェイスを使用してアクセスできるオブジェクト ストレージ サービスです。Amazon S3 データコネクタを使用して、組織の Amazon S3 データをインポートできます。

メモ

Amazon S3 データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Amazon S3 に接続するには、以下を収集する必要があります。

- AWS アカウント アクセス キー
- AWS アカウント シークレット キー
- Amazon Web Server がホストされる AWS リージョン

接続の前提条件の収集については、組織の Amazon S3 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Amazon S3 のサポートにお問い合わせください。

Amazon S3 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Amazon S3 を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Amazon S3への接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Amazon S3の接続は、**[既存の接続]** タブに保存されます。今後、Amazon S3に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Amazon S3からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Amazon S3 データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Amazon S3 に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Apache Cassandra への接続

Apache Cassandra は非 SQL データベース管理システムです。Apache Cassandra データをインポートするには、Cassandra データコネクタを使用します。

メモ

Analytics は任意のコネクタとして Cassandra を提供します。データアクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データコネクタと Python エンジンを実装する" ページ 2663 を参照してください。

作業を開始する前に

Cassandra に接続するには、以下を収集する必要があります。

- データベース サーバーのホスト名または IP アドレス
- 正しい接続ポート
- ユーザー名とパスワード(認証を使用する場合)

接続の前提条件の収集については、組織の Cassandra 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Cassandra のサポートにお問い合わせください。

Cassandra 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Cassandra を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Cassandra の接続は、**既存の接続** タブに保存されます。今後、Cassandra に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Cassandra からデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	Cassandra サーバーのホスト名または IP アドレス。	
ポート	Cassandra データベースの TCP ポート。	9042
デフォルトの鍵空間	Cassandra で接続するデフォルトのキースペース(スキーマ)。	
認証機構	Cassandra サーバーへの接続で使用する認証メカニズム。使用可能なオプション: <ul style="list-style-type: none"> ◦ 認証なし ◦ ユーザー名とパスワード 	認証なし
ユーザー名	Cassandra サーバーにアクセスするために使用するユーザー名。	
パスワード	指定したユーザー名に対応するパスワード。	

詳細設定

設定	説明	例
クエリモード	クエリを Cassandra に送信するときに使用するクエリモードを指定します。オプション: <ul style="list-style-type: none"> ◦ SQL - SQL_QUERY_MODE を使用して、SQL ですべてのクエリを実行します。 ◦ CQL - CQL_QUERY_MODE を使用して、CQL ですべてのクエリを実行します。 ◦ SQL(CQL へのフォールバックあり) - SQL_WITH_CQL_FALLBACK_QUERY_MODE を使用して、デフォルトで SQL ですべてのクエリを実行します。クエリが失敗した場合、ドライバーは CQL でクエリを実行します。 	SQL (CQL へのフォールバックあり)

データの定義とインポート

設定	説明	例
一貫性を調整可能	クエリが成功するためにクエリを処理する必要がある特定の Cassandra または Cassandra レプリカの数。	ONE
ロード バランサのポリシー	使用されるロード バランサ ポリシーを指定します。	
バイナリ型列の長さ	BLOB 列で報告するデフォルトの列長さ。	4000
文字列型列の長さ	ASCII、TEXT、VARCHAR 列で報告するデフォルトの列長さ。	4000
仮想テーブル名の区切り文字	コレクションから作成された仮想テーブルを命名するための区切り文字。仮想テーブルの名前は、元のテーブルの名前、区切り文字、コレクションの名前の順で付けられます。	_vt_
トークン認識の有効化	ロード バランシングおよびレイテンシを改善するためのトークン認識ポリシーを使用するかどうかを指定します。	
レイテンシー認識の有効化	ドライバーがレイテンシ認識アルゴリズムを使用して、低パフォーマンス ノードから負荷を分散する必要があるかどうかを指定します。	
ヌル値挿入の有効化	ドライバーが INSERT 分で指定されたすべての NULL 値を挿入する必要があるかどうかを指定します。	
大文字と小文字の区別の有効化	ドライバーがスキーマ、テーブル、列名で大文字と小目を区別するかどうかを指定します。 このオプションが有効な場合、すべてのスキーマ、テーブル、列を二重引用符 (") で囲む必要があります。	
文字列データ型での SQL_WVARCHAR の使用	テキストおよび varchar 型で SQL_WVARCHAR を使用するかどうかを指定します。	
ページングの有効化	大きい結果セットをページに分割するかどうかを指定します。	
ページあたりの行数	[ページングを有効にする] オプションを有効にした場合は、このオプションを使用して、各ページに表示する最大行数を指定します。	10000

設定	説明	例
SSL オプション	<p>ドライバーが Cassandra サーバーに接続するために SSL を使用するかどうかを指定します。オプション:</p> <ul style="list-style-type: none"> ◦ SSL なし - ドライバーは SSL を使用しません。 ◦ 単方向サーバー検証のみ - [サーバーホスト名検証を有効にする] オプションが有効な場合、クライアントは SSL を使用して Cassandra サーバーを検証します。そうでない場合、ドライバーは SSL を使用して Cassandra サーバーに接続しますが、クライアントとサーバーは相互に検証しません。 ◦ 双方向サーバーおよびクライアント検証 - [サーバーホスト名検証を有効にする] オプションが有効な場合、クライアントと Cassandra サーバーは SSL を使用して相互に検証します。そうでない場合、ドライバーは SSL を使用して Cassandra サーバーに接続しますが、クライアントとサーバーは相互に検証しません。 	SSL なし
サーバーのホスト名検証の有効化	ドライバーがサーバーのホスト名を SSL 証明書のホスト名と照合する必要があるかどうかを指定します。	
Ssltrustedcertspath	サーバーを検証するための証明書を含む .pem ファイルへの完全パス。	
クライアント側の証明書	クライアントを検証するための証明書を含む .pem ファイルへの完全パス。	
クライアント側の秘密鍵	クライアントを検証するために使用される秘密鍵を含むファイルへの完全パス。	
鍵ファイルのパスワード	[クライアント側秘密鍵] フィールドで指定された秘密鍵ファイルのパスワード。	

Cassandra の問い合わせ

Apache Cassandra 設計の利点の 1 つは、非正規化されたデータを少ない数のテーブルに格納できることです。セット、リスト、マップなどのネストされたデータ構造を利用することで、トランザクションが簡易化されます。ただし、Analytics はこのデータ型へのアクセスをサポートしません。コレクション(セット、リスト、マップ)内のデータを仮想テーブルに再正規化することで、コネクタを使用して、ユーザーは直接データを操作できますが、Cassandra の非正規化された形式のデータの保存は残します。

テーブルにコレクション列が含まれる場合は、初めてテーブルが問い合わせられるときに、コネクタは次の仮想テーブルを作成します。

- コレクション列を除く、実際のテーブルと同じデータを含む「基本」テーブル。
- 各コレクション列の仮想テーブル。ネストされたデータを展開します。

仮想テーブルは実際のテーブルのデータを参照し、コネクタが非正規化されたデータにアクセスできるようにします。仮想テーブルを問い合わせることで、ODBC 経由で Cassandra コレクションの内容にアクセスできます。

基本テーブルと仮想テーブルは、データベースに存在するテーブルのリストに追加テーブルとして表示されます。基本テーブルは表される実際のテーブルと同じ名前を使用します。コレクションを表す仮想テーブル名は実際のテーブル名、区切り文字(デフォルトは `_vt_`)、列の名前を使用して設定されます。

例

ExampleTable は Cassandra データベーステーブルであり、整数の主キー列 `pk_int`、リスト列、マップ列、およびセット列 (`StringSet`) を含みます。

コレクションを含むソーステーブル

pk_int	一覧	Map	StringSet
1	["1","2","3"]	{"S1": "a", "S2": "b"}	{"a", "b", "c"}
3	["100","101","102","105"]	{"S1": "t"}	{"a","e"}

コネクタは複数の仮想テーブルを生成し、この単一テーブルを表します。最初の仮想テーブルは基本テーブルです。

基本テーブル

pk_int
1
3

基本テーブルには、コレクションを除く元のデータベーステーブルと同じデータを含みます。これはこのテーブルから省略され、他の仮想テーブルで展開されます。

次のテーブルは、`List`、`Map`、`StringSet` 列からデータを再正規化する仮想テーブルを示します。

一覧

pk_int	List#index	List#value
1	0	1
1	1	2
1	2	3
3	0	100
3	1	101
3	2	102
3	3	105

Map

pk_int	Map#key	Map#value
1	S1	a
1	S2	十億
3	S1	兆

StringSet

pk_int	StringSet#value
1	a
1	十億
1	c
3	a
3	e

仮想テーブルの外部キー列は実際のテーブルの主キー列を参照し、仮想テーブル行が対応する実際のテーブル行を示します。列名が #index または #key で終わる列は、元のリストまたはマップ内のデータの位置を示します。列名が #value で終わる列は、コレクションから展開されたデータを含んでいます。

データコネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データコネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータコネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Apache Cassandra データコネクターの変更内容

Apache Cassandra データコネクターに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	変更
14.2	本コネクターでは、Apache Cassandra 2.0 への接続がサポートされなくなりました。接続は、Apache Cassandra 2.1、2.2、3.0 に対してのみ行われます。

Apache Drill への接続

Drill は Big Data アプリケーションから発生する半構造化されたデータに対して高性能分析を実行するための Apache のオープンソース SQL クエリエンジンです。Apache Drill データをインポートするには、Apache Drill データコネクタを使用します。

メモ

Analytics は任意のコネクタとして Drill を提供します。データアクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データコネクタと Python エンジンをインストールする」ページ 2663 を参照してください。

作業を開始する前に

Drill に接続するには、以下を収集する必要があります。

- 正しい認証タイプ(基本またはなし)
- ユーザー名とパスワード(認証を使用する場合)
- 正しい接続タイプ(Direct to Drillbit 埋め込みモードまたは Zookeeper Quorum 分散モード)
- 分散モードを使用している場合は Drillbit または各ノードのホストおよびポート

分散モードでは、ホストおよびポート情報をカンマ区切りの形式で **Quorum]** フィールドに入力する必要があります。

```
<ホスト名/IP アドレス> : <ポート番号>, <ホスト名/IP アドレス> : <ポート番号>, .  
...
```

- 分散モードを使用している場合は Drillbit クラスターの名前

接続の前提条件の収集については、組織の Apache Drill 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Apache Drill のサポートにお問い合わせください。

高度な接続プロパティについては、Apache Drill オンラインヘルプの *Windows* での *ODBC* の構成を参照してください。

Drill 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Apache Drillを選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの**接続名**を使用するか、新しい接続名を入力します。

Apache Drillの接続は、**既存の接続** タブに保存されます。今後、Apache Drillに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Apache Drillからデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
接続の種類	ドライバー接続タイプを指定します。オプション: <ul style="list-style-type: none"> ◦ 直接 Drillbit へ - ドライバーは単一の Drill サーバーに接続します。 ◦ Zookeeper Quorum - ドライバーは ZooKeeper クラスターに接続します。 	直接 Drillbit へ
クォーラム	ZooKeeper クラスターのサーバーを指定します。カンマ(,)を使用して複数のサーバーを区切ります。	
クラスター ID	ドライバーが接続する ZooKeeper クラスターの名前。	drillbits1
ホスト	Drill サーバーのホスト名または IP アドレス。	localhost
ポート	Drill サーバーがクライアント接続をリス	31010

設定	説明	例
	ニングするために使用する TCP ポート。	
認証のタイプ	<p>ドライバーで Drill への接続を認証する方法を指定します。オプション:</p> <ul style="list-style-type: none"> ○ 認証なし - ドライバーは Drill への接続を認証しません。 ○ 基本認証 - ドライバーは、ユーザー名とパスワードを使用して、接続を認証します。 	認証なし
ユーザー	Drill サーバーで認証するために使用されるユーザー名。	
パスワード	Drill サーバーで認証するためのパスワード。	
カタログ	すべてのスキーマ/データベースが整理される総合カタログの名前。このカタログ名は、SQL_DATABASE_NAME または CURRENT CATALOG の値として使用されます。	DRILL
デフォルトのスキーマ	スキーマがクエリで明示的に指定されていないときに使用する、データベーススキーマの名前。他のスキーマでクエリを発行するには、明示的にクエリを指定します。	
非同期の無効化	ドライバーが非同期クエリをサポートするかどうかを指定します。	

詳細設定

設定	説明	例
高度なプロパティ	ドライバーを構成するセミコロン (;) を使用して各詳細プロパティを区切り、括弧 ({}) を使用して接続文字列のすべての詳細プロパティを囲みません。	<pre>CastAnyToVarchar=true; HandshakeTimeout=5; QueryTimeout=180; TimestampTZDisplayTimezone=local; ExcludedSchemas=sys, INFORMATION_SCHEMA; NumberOfPrefetchBuffers=5</pre>

Apache HBase への接続

Apache HBase は Hadoop データベースであり、分散型の拡張可能な大規模データストアです。Apache HBase データコネクタを使用して、HBase データをインポートできます。

メモ

Analytics は任意のコネクタとして HBase を提供します。データアクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データコネクタと Python エンジンを実装する」ページ 2663 を参照してください。

作業を開始する前に

HBase に接続するには、以下を収集する必要があります。

- データベース サーバーのホスト名または IP アドレス
- 正しい接続ポート
- ユーザー名とパスワード(認証を使用する場合)

接続の前提条件の収集については、組織の HBase 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から HBase のサポートにお問い合わせください。

HBase 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、HBase を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

HBase の接続は、**既存の接続** タブに保存されます。今後、HBase に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。HBase からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	HBase サーバーのホスト名または IP アドレス。	
ポート	Apache HBase サーバーのポート。	9090
認証のタイプ	HBase サーバーへの接続で使用する認証メカニズム。オプション: <ul style="list-style-type: none"> ○ 認証なし - ドライバーは接続を認証しません。 ○ 基本認証 - ドライバーは、HBase ユーザー名とパスワードを使用して、接続を認証します。 	認証なし
ユーザー	HBase インスタンスにアクセスするためのユーザー名。	
パスワード	HBase インスタンスにアクセスするためのユーザー名に対応するパスワード。	

詳細設定

設定	説明	例
フェッチあたりの最大行数	要求ごとにクエリが返す行の最大数。	4096
スキーマ定義の行数上限	スキーマを生成するときにドライバーが抽出する行数。	1024

Apache Hive への接続

Apache Hive はクラウド データ サービスです。Apache Hive データ コネクタを使用して、Hive データをインポートできます。

メモ

Analytics は任意のコネクタとして Hive を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データ コネクタと Python エンジンを実装する」ページ 2663を参照してください。

作業を開始する前に

Hive に接続するには、以下を収集する必要があります。

- ユーザー名とパスワード
- 正しい接続ポート
- 使用される認証スキーム
- サーバーのホスト名または IP アドレス
- サーバーと通信するための転送モード
- 読み取りアクセス許可

接続の前提条件の収集については、組織の Hive 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Hive のサポートにお問い合わせください。

Hive 接続の作成

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL コネクタ セクションの 新規の接続** タブから、**Hive** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Hive の接続は、**既存の接続** タブに保存されます。今後、Hive に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Hiveからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
Hive サーバーのタイプ	接続先の Hive サーバー インスタンスを指定します。	Hive サーバー 2
サービス検出モード	Hive Server サービスが検出される方法を指定します。オプション： <ul style="list-style-type: none"> サービス検出なし - ドライバーは、検出 サービスを使用して Hive に接続します。 ZooKeeper - ドライバーは、ZooKeeper サービス経由で Hive Server サービスを検出します。 	サービス検出なし
ホスト	Hive サーバーのホスト名または IP アドレス。	
ポート	Hive サーバー インスタンスに接続するためのポート。	10000
データベース	スキーマがクエリで明示的に指定されていないときに使用する、データベーススキーマの名前。	
ZooKeeper の名前空間	Hive Server 2 znodes に対して ZooKeeper で構成される名前空間。	
認証機構	使用する認証メカニズムを指定します。オプション： <ul style="list-style-type: none"> 認証なし Kerberos ユーザー名 ユーザー名とパスワード Windows Azure HDInsight Service 	認証なし
領域	Hive Server 2 ホストの領域。	
ホストの FQDN	Hive サーバー ホストの完全修飾ドメイン名。	_HOST

設定	説明	例
サービス名	Hive サーバーの Kerberos サービスプリンシパル名。	
ユーザー名	Hive サーバーで認証するためのユーザー名。	
パスワード	Hive サーバーで認証するためのユーザー名のパスワード。	
委任 UID	接続の認証済みユーザーではなく、ドライバーがすべての Hive オペレーションを委任する必要がある委任されたユーザーのユーザー ID。	
Thrift による転送	Thrift レイヤーで使用する転送プロトコルを指定します。オプション: <ul style="list-style-type: none"> ○ バイナリ ○ SASL ○ HTTP 	バイナリ

詳細設定

設定	説明	例
SSL の有効化	クライアントが SSL 暗号化接続を使用して Hive サーバーと通信するかどうかを指定します。	
列名とホスト名の不一致を許可	CA が発行した SSL 証明書名が Hive サーバーのホスト名と一致するかどうかを指定します。	
自己署名サーバー証明書の許可	この証明書が信頼できる証明書のリストにない場合でも、ドライバーが自己署名証明書を使用する Hive サーバーへの接続を許可するかどうかを指定します。	
信頼できる証明書	SSL を使用するときサーバーを検証するための、信頼できる CA 証明書を含む .pem ファイルへの完全パス。	
双方向 SSL	双方向 SSL を有効にするかどうかを指定します。	
クライアントの証明書ファイル	クライアントの SSL 証明書を含む .pem ファイルへの完全パス。	

設定	説明	例
クライアントの秘密鍵ファイル	クライアントの SSL 秘密鍵を含む .pem ファイルへの完全パス。	
クライアントの秘密鍵のパスワード	クライアント秘密鍵 フィールドで指定された秘密鍵ファイルのパスワード。	
ネイティブ クエリの使用	ドライバーがネイティブ HiveQL クエリを使用するかどうかを指定します。このオプションが選択されていない場合、ドライバーはアプリケーションで発行されたクエリを HiveQL の同等の形式に変換します。	
Fast SQLPrepare	ドライバーが SQLExecute へのクエリ実行を遅延するかどうかを指定します。	
ドライバー設定を優先	ドライバー全体の構成設定が接続および DNS 設定よりも優先されるかどうかを指定します。	
非同期実行を使用	クエリを同期で実行するか非同期で実行するかどうかを指定します。	
非同期実行のポーリング間隔	クエリ実行状況をポーリングする間隔 (ミリ秒)。	100
クエリでテーブルを取得	ドライバーが SHOW TABLES クエリを使用して、データベースからテーブル名を取得するかどうかを指定します。無効な場合、ドライバーは GetTables Thrift API 呼び出しを使用します。	
Unicode の SQL 文字型	文字列データ型で返される SQL タイプを指定します。有効な場合、ドライバーは SQL_WVARCHAR(STRING および VARCHAR 列)と SQL_WCHAR(CHAR 列)を返します。	
システム テーブルの表示	ドライバーが SQLTables や SQLColumns などのカタログ関数呼び出しで hive_system テーブルを返すかどうかを指定します。	
SSPI のみを使用	ドライバーが SSPI プラグインまたは MIT Kerberos を使用した Kerberos 認証を処理するかどうかを指定します。	
セッションの自動回復が無効です	既存のセッションが有効ではないときに、ドライバーで自動的に新しいセッションを開くかどうかを指定します。	

設定	説明	例
ブロックあたりフェッチされる行数	一度にクエリが返す行の最大数。	10000
文字列型列のデフォルト長	STRING 列に含まれる最大文字数。	255
バイナリ型列の長さ	BINARY 列の最大データ長さ。	32767
小数列の尺度	数値データ型の小数点以下の桁数。	10
ソケットのタイムアウト	終了するまでにオペレーションがアイドル状態である秒数。	60
HTTP のパス	Hive サーバーに対応する部分 URL。	

Hive 接続フィールド

列の一意の名前

データアクセス ウィンドウ経由で行われる Hive 接続では、デフォルトで 0 に設定される `EnableUniqueColumnName` 接続文字列パラメーターが使用されます。接続時に正しい列名が取得されることを保証するために、このパラメーターの値は 0 にする必要があります。

データアクセス ウィンドウを使用するのではなく DSN を使用して Hive 接続を作成する場合は、デフォルトでこの値が 1 に設定されます。接続が動作するには、Windows リポジトリで 0 に変更する必要があります。

メモ

13.1 より古いバージョンの ACL で確立された DSN 接続を使用するスクリプトは、バージョン 13.1 にアップグレードした後に動作し続けます。

Apache Phoenix への接続

Apache Phoenix は、Apache HBase によるオンラインランザクシオン処理をサポートするリレーショナルデータベースエンジンです。Apache Phoenix データコネクタを使用して、組織の Apache Phoenix データをインポートできます。

メモ

Apache Phoenix データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Apache Phoenix 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Apache Phoenix のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Apache Phoenix の接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Apache Phoenix を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Apache Phoenixへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Apache Phoenixの接続は、 **[既存の接続]** タブに保存されます。今後、Apache Phoenixに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Apache Phoenixからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Apache Phoenix データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Apache Phoenix に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Apache Spark への接続

Apache Spark はクラウド データ サービスです。Apache Spark データ コネクタを使用して、Spark データをインポートできます。

メモ

Analytics は任意のコネクタとして Spark を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データ コネクタと Python エンジンを実装する」ページ 2663を参照してください。

作業を開始する前に

Spark に接続するには、以下を収集する必要があります。

- ユーザー名とパスワード
- 正しい接続ポート
- 使用される認証スキーム
- サーバーのホスト名または IP アドレス
- サーバーと通信するための転送モード
- 読み取りアクセス許可

接続の前提条件の収集については、組織の Spark 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Spark のサポートにお問い合わせください。

Spark 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、**Spark**を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Sparkの接続は、**既存の接続** タブに保存されます。今後、Sparkに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Sparkからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
Spark Server タイプ	接続先の Spark サーバー インスタンスを指定します。	SharkServer2
ホスト	Spark サーバーのホスト名または IP アドレス。	
ポート	Spark サーバー インスタンスに接続するためのポート。	10000
データベース	スキーマがクエリで明示的に指定されていないときに使用する、データベーススキーマの名前。	デフォルト
認証機構	使用する認証メカニズムを指定します。オプション： <ul style="list-style-type: none"> ○ 認証なし ○ Kerberos ○ ユーザー名 ○ ユーザー名とパスワード ○ Windows Azure HDInsight Emulator ○ Windows Azure HDInsight Service 	認証なし
領域	Spark Thrift サーバー ホストの領域。	
ホストの FQDN	Spark Thrift サーバー ホストの完全修飾ドメイン名。	_HOST
サービス名	Spark サーバーの Kerberos サービスプリンシパル名。	
ユーザー名	Spark サーバーで認証するためのユーザー名。	
パスワード	Spark サーバーで認証するためのユーザー名のパスワード。	
委任 UID	接続の認証済みユーザーではなく、ド	

設定	説明	例
	ライバーがすべての Spark オペレーションを委任する必要がある委任されたユーザーのユーザー ID。	
Thrift による転送	Thrift レイヤーで使用する転送プロトコルを指定します。オプション: <ul style="list-style-type: none"> バイナリ SASL HTTP 	バイナリ

詳細設定

設定	説明	例
SSL の有効化	クライアントが SSL 暗号化接続を使用して Spark サーバーと通信するかどうかを指定します。	
列名とホスト名の不一致を許可	CA が発行した SSL 証明書名が Spark サーバーのホスト名と一致するかどうかを指定します。	
自己署名サーバー証明書の許可	この証明書が信頼できる証明書のリストにない場合でも、ドライバーが自己署名証明書を使用する Spark サーバーへの接続を許可するかどうかを指定します。	
信頼できる証明書	SSL を使用するときサーバーを検証するための、信頼できる CA 証明書を含む .pem ファイルへの完全パス。	
双方向 SSL	双方向 SSL を有効にするかどうかを指定します。	
クライアントの証明書ファイル	クライアントの SSL 証明書を含む .pem ファイルへの完全パス。	
クライアントの秘密鍵ファイル	クライアントの SSL 秘密鍵を含む .pem ファイルへの完全パス。	
クライアントの秘密鍵のパスワード	クライアント秘密鍵 フィールドで指定された秘密鍵ファイルのパスワード。	
ネイティブ クエリの使用	ドライバーがネイティブ HiveQL クエリを使用するかどうかを指定します。このオプションが選択されていない場合、ドライバーはアプリケーションで発行されたク	

設定	説明	例
	エリを HiveQL の同等の形式に変換します。	
Fast SQLPrepare	ドライバーが SQLExecute へのクエリ実行を遅延するかどうかを指定します。	
ドライバー設定を優先	ドライバー全体の構成設定が接続および DNS 設定よりも優先されるかどうかを指定します。	
非同期実行を使用	クエリを同期で実行するか非同期で実行するかどうかを指定します。	
非同期実行のポーリング間隔	クエリ実行状況をポーリングする間隔 (ミリ秒)。	100
クエリでテーブルを取得	ドライバーが SHOW TABLES クエリを使用して、データベースからテーブル名を取得するかどうかを指定します。無効な場合、ドライバーは GetTables Thrift API 呼び出しを使用します。	1
Unicode の SQL 文字型	文字列データ型で返される SQL タイプを指定します。有効な場合、ドライバーは SQL_WVARCHAR (STRING および VARCHAR 列) と SQL_WCHAR (CHAR 列) を返します。	
システムテーブルの表示	ドライバーが SQLTables や SQLColumns などのカタログ関数呼び出しで spark_system テーブルを返すかどうかを指定します。	
SSPI のみを使用	ドライバーが SSPI プラグインまたは MIT Kerberos を使用した Kerberos 認証を処理するかどうかを指定します。	
セッションの自動回復が無効です	既存のセッションが有効ではないときに、ドライバーで自動的に新しいセッションを開くかどうかを指定します。	
ブロックあたりフェッチされる行数	一度にクエリが返す行の最大数。	10000
文字列型列のデフォルト長	STRING 列に含まれる最大文字数。	255
バイナリ型列の長さ	BINARY 列の最大データ長さ。	32767
小数列の尺度	数値データ型の小数点以下の桁数。	10
ソケットのタイムアウト	終了するまでにオペレーションがアイドル状態である秒数。	60

設定	説明	例
HTTP のパス	Spark サーバーに対応する部分 URL。	/spark

Spark 接続フィールド

列の一意の名前

データアクセスウィンドウ経由で行われるSpark接続では、デフォルトで0に設定される `EnableUniqueColumnName` 接続文字列パラメーターが使用されます。接続時に正しい列名が取得されることを保証するために、このパラメーターの値は0にする必要があります。

データアクセスウィンドウを使用するのではなくDSNを使用してSpark接続を作成する場合は、デフォルトでこの値が1に設定されます。接続が動作するには、Windows リポジトリで0に変更する必要があります。

メモ

13.1 より古いバージョンのACLで確立されたDSN接続を使用するスクリプトは、バージョン13.1にアップグレードした後に動作し続けます。

AWS Data Management への接続

AWS Data Management は Amazon Web Services を管理するための Web ベースのインターフェイスです。AWS Data Management データコネクタを使用して、AWS Data Management データをインポートできます。

メモ

AWS Data Management データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Amazon Data Management に接続するには、以下を収集する必要があります。

- AWS アカウント アクセスキー
- AWS アカウント シークレット キー
- Amazon Web Server がホストされる AWS リージョン

接続の前提条件の収集については、組織の AWS Data Management 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から AWS Data Management のサポートにお問い合わせください。

AWS Data Management 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続**] タブから、**AWS Data Management** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
AWS Data Managementへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

AWS Data Managementの接続は、 **[既存の接続]** タブに保存されます。今後、AWS Data Managementに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。AWS Data Managementからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

AWS Data Management データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSNコネクタ]** セクションを展開して、AWS Data Management に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Azure Analysis Services への接続

Azure Analysis Services はデータモデリング、データ分析、レポートのためのクラウドプラットフォームです。Azure Analysis Services データコネクタを使用して、Azure Analysis Services データをインポートできます。

メモ

Azure Analysis Services データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Azure Analysis Services 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Azure Analysis Services のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Azure Analysis Services 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Azure Analysis Services を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Azure Analysis Servicesへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Azure Analysis Servicesの接続は、 **[既存の接続]** タブに保存されます。今後、Azure Analysis Servicesに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Azure Analysis Servicesからデータをインポートする方法については、「データアクセスウィンドウの作業」 ページ 376を参照してください。

接続名の変更

Azure Analysis Services データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSNコネクタ]** セクションを展開して、Azure Analysis Services に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Azure Data Catalog への接続

Azure Data Catalog はデータ ディスカバリおよびコラボレーション データ注釈向けのクラウド サービスです。Azure Data Catalog データ コネクタを使用して、Azure Data Catalog データをインポートできます。

メモ

Azure Data Catalog データ コネクタは当社のデータ アクセスおよび接続 パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Azure Data Catalog 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Azure Data Catalog のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Azure Data Catalog 接続を作成する

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Azure Data Catalog** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Azure Data Catalogへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Azure Data Catalogの接続は、**[既存の接続]** タブに保存されます。今後、Azure Data Catalogに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Azure Data Catalogからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Azure Data Catalog データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSNコネクタ]** セクションを展開して、Azure Data Catalog に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Azure Data Lake Storage への接続

Azure Data Lake Storage はビッグデータ向けのクラウドベースのファイルストレージおよび管理設備です。Azure Data Lake Storage データコネクタを使用して、組織の Azure データをインポートできます。

メモ

Azure Data Lake Storage データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Azure Data Lake Storage 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Azure Data Lake Storage のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Azure Data Lake Storage 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Azure Data Lake Storage を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Azure Data Lake Storageへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Azure Data Lake Storageの接続は、 **[既存の接続]** タブに保存されます。今後、Azure Data Lake Storageに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Azure Data Lake Storageからデータをインポートする方法については、「データアクセスウィンドウの作業」 ページ 376を参照してください。

接続名の変更

Azure Data Lake Storage データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Azure Data Lake Storage に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Azure Data Management への接続

Azure Data Management は Azure データを管理するためのオプションを提供します。Azure Data Management データコネクタを使用して、組織の Azure データをインポートできます。

メモ

Azure Data Management データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Azure Data Management に接続するには、Azure Data Management サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Azure Data Management 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Azure Data Management のサポートにお問い合わせください。

Azure Data Management 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Azure Data Management** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Azure Data Managementへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Azure Data Managementの接続は、 **[既存の接続]** タブに保存されます。今後、Azure Data Managementに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Azure Data Managementからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Azure Data Management データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクター]** セクションを展開して、Azure Data Management に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Azure Table Storage への接続

Azure Table Storage はクラウドベースの NoSQL データストアであり、大量の構造化された非リレーショナルデータを保存できます。Azure Table Storage データコネクタを使用して、組織の Azure Table Storage データをインポートできます。

メモ

Azure Table Storage データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Azure Table Storage に接続するには、Azure Table Storage サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Azure Table Storage 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Azure Table Storage のサポートにお問い合わせください。

Azure Table Storage を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Azure Table Storage** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Azure Table Storageへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Azure Table Storageの接続は、**[既存の接続]** タブに保存されます。今後、Azure Table Storageに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Azure Table Storageからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Azure Table Storage データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Azure Table Storage に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Basecamp への接続

Basecamp は、オンラインプロジェクト管理プラットフォームです。Basecamp データコネクタを使用して、組織の Basecamp データをインポートできます。

メモ

Basecamp データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Basecamp 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Basecamp のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Basecamp 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Basecamp** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Basecampへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Basecampの接続は、**[既存の接続]** タブに保存されます。今後、Basecampに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Basecampからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Basecamp データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Basecamp に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Box への接続

Box はクラウドベースのコンテンツ管理サービスであり、ビジネスでのファイル共有とコラボレーションを可能にします。Box データコネクタを使用して、組織の Box データをインポートできます。

メモ

Box データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Box に接続するには、Box サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Box 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Box のサポートにお問い合わせください。

Box 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Box** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN Configuration(DSN 構成)]** ダイアログボックスで **[Test Connection(接続のテスト)]** をクリックします。

Box のログインページが表示されます。

4. ログインするには

- Box アカウント資格情報がある場合は、資格情報を入力します。
- 組織で SSO を使用している場合は、**[Use Single Sign On (SSO)(シングルサインオン (SSO) を使用)]** をクリックします。

5. **[Authorize(許可)]** をクリックします。

SSO を使用している場合は、認証資格情報を入力してログインします。

- 表示されるページで、**Grant access to Box(Box へのアクセスを付与)**をクリックします。
ページが正常に許可されます。
- 接続成功ダイアログボックスが表示されるので、**OK**をクリックします。
Box への接続が正常に確立されます。
- DSN 構成**ダイアログボックスで **OK**をクリックします。

Box の接続は、**Box** という名前で **既存の接続** タブに保存されます。今後、保存したこの接続を使用して Box に再接続できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Box からのデータのインポートに関するヘルプについては、"データアクセスウィンドウの作業" ページ 376を参照してください。

メモ

多要素認証 (MFA) を使用して Box に接続しているときには、約 1 時間後に認証トークンが期限切れになり、既存の接続とインポートが失敗します。認証エラーが発生すると、既存の接続は動作しません。さらに操作を続けるには、Box への新しい接続を作成する必要があります。

接続名の変更

Box データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **既存の接続** タブに保存されます。名前を変更する場合は、**既存の接続** タブに移動し、**ACL DSN コネクター** セクションを展開して、Box に対応する省略記号アイコン  をクリックし、**接続名の変更** を選択します。

Cloudera Impala への接続

Cloudera Impala はクラウド データ サービスです。Cloudera Impala データ コネクタを使用して、Impala データをインポートできます。

メモ

Analytics は任意のコネクタとして Impala を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実装する" ページ 2663を参照してください。

作業を開始する前に

Impala に接続するには、以下を収集する必要があります。

- ユーザー名
- パスワード
- 読み取りアクセス許可

接続の前提条件の収集については、組織の Impala 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Impala のサポートにお問い合わせください。

Impala 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Impala を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Impala の接続は、**既存の接続** タブに保存されます。今後、Impala に接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Impala からデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	Impala サーバーのホスト名または IP アドレス。	
ポート	Impala サーバー インスタンスに接続するためのポート。	
データベース	デフォルトで使用する Impala データベースの名前。	
認証機構	使用する認証メカニズム。使用可能なオプション: <ul style="list-style-type: none"> ○ 認証なし ○ Kerberos ○ SASL ユーザー名 ○ ユーザー名とパスワード 	認証なし
領域	Impala ホストの領域。	
ホストの FQDN	Impala サーバー ホストの完全修飾ドメイン名。	_HOST
サービス名	Impala サーバーの Kerberos サービスプリンシパル名。	Impala
ユーザー名	Impala サーバーにアクセスするためのユーザー名。	
パスワード	Impala サーバーへのアクセスを認証するためのパスワード。	
転送バッファ サイズ	ネットワークから暗号化されていないデータをバッファリングするためにメモリに予約するバイト数。	1000
簡易認証およびセキュリティ層 (SASL) を使用	ドライバーで認証を処理するために SASL を使用するかどうかを指定します。	
委任 UID	このオプションでユーザー ID が指定されると、Impala ドライバーはすべての Impala に対する処理を、接続で認証されたユーザーではなく、指定したユーザーに委任します。	

詳細設定

設定	説明	例
SSLの有効化	クライアントがSSL暗号化接続を使用してImpalaサーバーと通信するかどうかを指定します。	
列名とホスト名の不一致を許可	CAが発行したSSL証明書名がImpalaサーバーのホスト名と一致するかどうかを指定します。	
自己署名サーバー証明書の許可	ドライバーが自己署名証明書を使用するImpalaサーバーへの接続を許可するかどうかを指定します。	
信頼できる証明書	SSLを使用するときにサーバーを検証するための、信頼できるCA証明書を含む.pemファイルへの完全パス。	
ネイティブクエリの使用	ドライバーがネイティブImpala SQLクエリを使用するかどうかを指定します。このオプションが選択されていない場合、ドライバーはアプリケーションで発行されたクエリをImpala SQLの同等の形式に変換します。アプリケーションがImpalaアウェアで、既にImpala SQLを発行している場合、このオプションを憂苦尾にして、クエリ変換の負荷が大きくなることを回避してください。	
取引シミュレーションの有効化	ドライバーがトランザクションをシミュレートするかどうかを指定します。無効にすると、トランザクション文を含むクエリを実行しようとした場合に、ドライバーでエラーが返されます。	
SQLでUnicode型を使用	文字列データ型で返されるSQLタイプを指定します。有効な場合、ドライバーはSQL_WVARCHAR (STRING および VARCHAR 列) と SQL_WCHAR (CHAR 列) を返します。	
ブロックあたりフェッチされる行数	一度にクエリが返す行の最大数。	10000
ソケットのタイムアウト	要求がタイムアウトし、エラーメッセージが返される前に、TCPソケットがサーバーからの応答を待機する秒数。 0に設定すると、TCPソケットで要求のタイムアウトが発生しません。	30
文字列型列の長さ	STRING列に含まれる最大文字数。	32767

データ コネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサード パーティ ベンダーが加えた変更により、1 つまたは複数のデータ コネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータ アクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データ ソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データ アクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データ ソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データ インポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

CockroachDB への接続

CockroachDB は、分散型 SQL データベースです。CockroachDB データコネクタを使用すると、組織の CockroachDB データをインポートできます。

メモ

CockroachDB データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の CockroachDB 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から CockroachDB のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

CockroachDB 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、CockroachDB を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
CockroachDBへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

CockroachDBの接続は、 **[既存の接続]** タブに保存されます。今後、CockroachDBに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。CockroachDBからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

CockroachDB データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、CockroachDB に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Concur への接続

Concur はクラウド ベースの旅費/経費管理 サービスです。組織の Concur データをインポートするには、Concur データ コネクタを使用します。

メモ

Analytics は任意のコネクタとして Concur を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実装する" ページ 2663を参照してください。

作業を開始する前に

Concur の資格情報

Concur に接続するには、以下が必要です。

- Concur アクセストークン

Concur アクセストークンがない場合は、以下を収集する必要があります。

- Concur のユーザー名およびパスワード
- Concur クライアント ID
- Concur API サーバー ホスト名 (デフォルトは "www.concursolutions.com")

これらの資格情報で、[データ アクセス] ウィンドウの [トークンの取得] オプションを使用して、接続の認証に必要なアクセストークンを生成できます。

メモ

取得するときに、トークンの有効期限を書き留めておいてください。Concur 認証トークンは 1 年後に期限切れになります。

接続の前提条件の収集については、組織の Concur 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Concur のサポートにお問い合わせください。

Concur クライアント ID を取得する

お客様に Concur アクセストークンがない場合は、お客様、または御社の Concur 管理者が Concur クライアント ID を取得する必要があります。クライアント ID を使ってアクセストークンを生成することができます。

クライアント ID を取得するには、[Register Partner Application] オプションを使って Concur Web サービスの管理領域に表示します。

Concur Web サービスでは、クライアント ID は以下のいずれかの名前と呼ばれています。

- Application Authorization Key(アプリケーション権限の承認キー)
- Consumer Key(コンシューマキー)

メモ

Concur Web サービスに [Register Partner Application] オプションが表示されない場合は、あなたまたはあなたの Concur 管理者が Concur のサポートに問い合わせる必要があります。必要な Concur ライセンスが御社にない可能性もあります。

Concur 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Concur を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Concur の接続は、**既存の接続** タブに保存されます。今後、Concur に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Concur からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	Concur サーバーのホスト名。	https://concur solutions.com
OAuth クライアント ID	クライアント ID は、OAuth 承認サーバーにアプリケーションを登録するときに割り当てられます。	

設定	説明	例
ユーザー名	Concur への認証に使用するユーザーアカウント	
パスワード	ユーザーを Concur で認証するのに使用するパスワード。	
アクセストークン	認証されたユーザーが Concur サービスで保護されたリソースにアクセスするためのアクセストークン。	

詳細設定

設定	説明	例
ダブルバッファリングの有効化	ダブルバッファリングを使用して、アダプターが Concur データを取得するかどうかを指定します。ただし、ダブルバッファリングが有効なときには、アダプターのメモリおよびリソース消費量が多くなる場合があります。	
HTTPS の使用	データソースエンドポイントが HTTPS で暗号化されるかどうかを指定します。	
ユーザー パラメーター	/api/v3.0/expense エンドポイントグループの下にあるエンドポイントのテーブルを問い合わせるときの、ユーザークエリパラメーターで使用する値。	すべての
Windows プロキシ設定を使用する	システムプロキシ設定を使用するかどうかを指定します。カスタムプロキシ設定を使用するには、このオプションを無効にする必要があります。このオプションに設定された値は他のプロキシ設定よりも優先されます。	
プロキシのホスト	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。	206.174.193.118
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ ユーザー	ProxyServer プロキシで認証するためのユーザー名。	user@domain
プロキシのパスワード	ProxyServer プロキシでプロキシ ユーザーを認証するためのパスワード。	

Concur でアクセスできるデータ

データ アクセス ウィンドウにより、Concur のデータのサブセットにアクセスできます。データには、次にリストアップした Concur API からアクセスすることもできます。

メモ

以下の一覧にない API は、データ アクセス ウィンドウからアクセスできません。各エンドポイントに存在するデータについては、[Concur の API のドキュメント](#)を参照してください。

Concur では、従来のリレーショナルテーブルおよび列に適用されるデータ型/構造ルールに従わないデータは、構造体に保存されます。そのため、このようなデータはリレーショナル形式にマッピングしておく必要があります。これを行うため、コネクタは Concur データを ODBC 互換形式にマッピングします。

モジュール	API のエンドポイント
共通	リスト
	リストの項目
	場所
経費	エントリ
	経費フォーム
	経費グループの構成
	明細
	簡易経費
	レポート
洞察	機会
領収書画像	領収書画像
出張	旅行

複数ユーザーの経費データへのアクセス

メモ

複数ユーザーの費用データにアクセスするには、Concur に接続するために使用する Concur アカウントに適切な権限が必要です。

Concur アカウント権限のヘルプが必要な場合、御社の Concur 管理者に連絡してください。管理者から支援を得られない場合は、あなたまたは管理者から Concur のサポートにお問い合わせください。

費用モジュール内の複数の Concur アカウントのデータにアクセスするには、**詳細オプション**]にある、オプションの **ユーザーパラメーター**]フィールドを "ALL" に設定します。

また、このフィールドに特定の Concur ユーザーのユーザー名を入力して、そのユーザーの費用データを表示することもできます。

SAP Concur への接続

SAP Concur はクラウド ベースの旅費/経費管理サービスです。組織の SAP Concur データをインポートするには、SAP Concur データ コネクターを使用できます。

メモ

SAP Concur データ コネクターは当社のデータ アクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の SAP Concur 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SAP Concur のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

SAP Concur 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、SAP Concur を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
SAP Concurへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

SAP Concurの接続は、 **[既存の接続]** タブに保存されます。今後、SAP Concurに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。SAP Concurからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

SAP Concur データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクター]** セクションを展開して、SAP Concur に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Couchbase への接続

Couchbase は非 SQL 文書指向 データベースです。Couchbase データをインポートするには、Couchbase データコネクタを使用します。

メモ

Analytics は任意のコネクタとして Couchbase を提供します。データアクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データコネクタと Python エンジンを実装する」ページ 2663を参照してください。

作業を開始する前に

Couchbase に接続するには、以下を収集する必要があります。

- クエリとインデックス サービスが有効な Couchbase インスタンスの 1 つ以上のノード
- データベース サーバーのホスト名
- 正しい接続ポート
- 選択した認証方法の該当する資格情報

Couchbase インスタンスで認証が必要な場合、1 つ以上のバケットの名前とパスワードを指定する JSON 文字列または JSON ファイルを提供する必要があります。詳細については、「JSON 認証形式」ページ 473を参照してください。

- 有効な SSL 証明書 (SSL 経由で接続する場合)

接続の前提条件の収集については、組織の Couchbase 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Couchbase のサポートにお問い合わせください。

Couchbase 接続を作成する

- Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
- ACL コネクタ セクションの **新規の接続** タブから、Couchbase を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

- データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Couchbaseの接続は、**既存の接続**タブに保存されます。今後、Couchbaseに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Couchbaseからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

設定	説明	例
サーバー	Couchbase サーバーのホスト名または IP アドレス。この値は HTTP または HTTPS URL に設定できます。	couchbase-server.com
ポート	Couchbase サーバーを実行するポート。	8093
認証機構	Couchbase サーバーに接続するときに使用する認証メカニズム。 <ul style="list-style-type: none"> ○ 認証なし - ドライバーは接続を認証しません。 ○ 資格情報を使用 - ドライバーは資格情報文字列で指定された名前とパスワードを使用して接続を認証します。 ○ 資格情報ファイルを使用 - ドライバーは資格情報ファイルで指定された名前とパスワードを使用して接続を認証します。 	
資格情報	Couchbase サーバー インスタンスで認証するために、1 つ以上のユーザーまたはバケットの名前とパスワードを指定する JSON 文字列。資格情報文字列は次の形式を使用する必要があります。 <pre>[{"user": "[UserName1]", "pass": "[Password1]"}, {"user": "[UserName2]", "pass": "[Password2]"}]</pre>	
資格情報ファイル	Couchbase サーバー インスタンスで認証するための資格情報を含む JSON ファイルへの完全パス。 <p>JSON ファイルには、複数のユーザーまたはバケットの名前とパスワードを含める必要があります。</p>	
SSL の有効化	Couchbase サーバーに接続するときに	false

設定	説明	例
	SSLを使用するかどうかを指定します。	
SSL 証明書	<p>SSLを有効にする]を True に設定するときに、サーバーから許可される証明書。</p> <p>次のいずれかを使用できます。</p> <ul style="list-style-type: none"> 完全な PEM 証明書 証明書を含むローカルファイルへのパス 公開鍵 MD5 または SHA1 サムプリント(スペースまたはコロンで 16 進数値を区切ることができます) <p>コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer

詳細設定

設定	説明	例
クエリモード	<p>クエリを Couchbase サーバーに送信するときに使用するクエリモードを指定します。</p> <ul style="list-style-type: none"> SQL-ドライバーは SQL ですべてのクエリを実行します。 N1QL-ドライバーは N1QL ですべてのクエリを実行します。 	SQL
一貫性	<p>インデックス スキャン中に施行するデータ一貫性のレベル。このプロパティを次の値のいずれかに設定します。</p> <ul style="list-style-type: none"> NOT_BOUNDED AT_PLUS REQUEST_PLUS STATEMENT_PLUS 	REQUEST_PLUS
ロード バランサの有効化	<p>ドライバーが Couchbase クラスターのノード間でロード バランシングとフェールオーバーをサポートするかどうかを指定します。</p>	
サンプル サイズ	<p><code>SchemaMapOperation</code> プロパティの生成中にデータの構造を検出するためにドライバーが抽出する文書の数。</p>	100

設定	説明	例
型名リスト	文書タイプを指定するためにバケットが使用する属性のカンマ区切りのリスト。各リスト項目は、バッククォート(` `)で囲まれた、コロンの(:)、バッククォート(` `)で囲まれた属性名でなければなりません。	<code>product:` `type`,` `store:` `type`,` `customer:` `jsonType`,` `sales:` `type`</code>

Couchbase の問い合わせ

JSON 認証形式

```
[{"user" : "userName1", "pass" : "passWord1"},
{"user" : "userName2", "pass" : "passWord2"}]
```

SQL クエリとN1QL API 呼び出し

コネクタは Couchbase Server に対して標準の SQL クエリを使用し、標準 SQL-92 クエリを同等の N1QL クライアント API 呼び出しに変換します。この変換により、Couchbase Server インスタンスに対して標準クエリを実行できます。クエリを完全に変換できない場合は、クエリの変換された部分が1つ以上の N1QL クエリとして処理のために Couchbase Server インスタンスに渡され、クエリの変換されない部分がコネクタによって処理されます。

メモ

データ構造の名前は、大文字と小文字を区別するため、クエリのテーブル、列、バケットなどの構造の大文字と小文字がデータベースの構造と一致することを確認する必要があります。

スキーマ定義

Couchbase は従来の関係テーブルおよび列と比べ、データ型と構造の異なるルールに従うデータを保存できます。Couchbase データはバケットと文書に整理され、ネストされた配列または異なる型の要素の配列を含むことができます。このデータは関係形式にマッピングされる必要があります。これを実現するために、コネクタは Couchbase データを ODBC 互換形式にマッピングするスキーマ定義を生成します。

必要なスキーマ定義がないデータベースに接続するときには、コネクタは次を実行して自動的に生成します。

1. データベースで特定された各文書型に対して、コネクタは複数の文書からデータを抽出し、データの構造を検出します。

2. コネクタはタイプに基づいてすべての文書をコレクションに整理し、スキーマ定義の部分としてこれらのコレクションを保存します。スキーマ定義を使用すると、ドライバーはテーブルとしてコレクションを公開します。
3. データベースで検出された各配列に対して、コネクタは仮想テーブルを生成して、データを展開し、スキーマの一部としてこれらの仮想テーブルを保存します。スキーマを使用して、ドライバーは標準テーブルとして仮想テーブルを公開します。
4. コネクタは各列の Couchbase データ型を定義し、各 Couchbase データ型を、最大数の値を最適な方法で表すことができる SQL データ型にマッピングします。

基本テーブル

基本テーブルは Couchbase 文書のコレクションのデータを表します。文書は行として表示され、配列ではないすべての属性は列として表示されます。各基本テーブルでは、コネクタは主キー PK を作成します。これは、各行の元になる Couchbase 文書を特定します。

コネクタでは、基本テーブルの名前は表される文書型です。Couchbase では、基本テーブルの名前はデータの元となるバケットです。

仮想テーブル

仮想テーブルは配列のサポートを提供します。各仮想テーブルには 1 つの配列のデータがあり、テーブルの各行は配列の要素を表します。要素に配列が含まれる場合、コネクタは必要に応じて追加の仮想テーブルを作成し、ネストされたデータを展開します。

各仮想テーブルで、コネクタは、配列の元になる文書を特定し、関連付けられた基本テーブルから列を参照する主キー列名を作成します。また、コネクタはインデックス列(名前サフィックス `_IDX`)を作成し、配列内の要素の位置を示します。

例

次の例は、Couchbase データベース名 `ExampleDatabase` にコネクタが接続する基本テーブルと仮想テーブルを示します。これには 2 つの文書 `Customer_123221` と `Order_221354` があります。

`Customer_123221` 文書は `Customer` 型であり、次の属性を含みます。`SavedAddresses` 属性は配列です。

```
{
  "Type": "Customer",
  "Name": "John Doe",
  "SavedAddresses": ["123 Main St.", "456 1st Ave"]
}
```

Order_221354 文書は Order 型であり、次の属性を含みます。**CreditCard**属性はオブジェクトであり、**Items** 属性はオブジェクトの配列です。

```
{
  "Type": "Order",
  "CustomerID": "Customer_123221",
  "CreditCard": {
    "Type": "Visa",
    "CardNumber": "4111 1111 1111 1111",
    "Expiry": "12/12",
    "CVN": "123"
  },
  "Items": [
    {"ItemID": 89123, "Quantity": 1},
    {"ItemID": 92312, "Quantity": 5}
  ]
}
```

Analytics が **ExampleDatabase** に接続し、スキーマを生成するときには、コネクタは各文書タイプのコレクションを作成し、これらのコレクションを次に示す 2 つの基本テーブルとして公開します。

基本テーブル Customer

PK	名前
"Customer_123221"	John Doe

基本テーブル順序

PK	顧客 ID	CreditCard _Type	CreditCard _Number	CreditCard _Expiry	CreditCard _CVN
"Order_221354"	"Customer_123221"	"Visa"	"4111 1111 1111 1111"	"12/12"	"123"

Customer_123221 文書の **SavedAddresses** 配列と **Order_221354** 文書の **Items** 配列はこれらの基本テーブルには表示されません。代わりに、コネクタは各配列の仮想テーブルを生成します。

SavedAddresses テーブル

PK	SavedAddresses_IDX	SavedAddresses
"Customer_123221"	0	"123 Main St."
"Customer_123221"	1	"456 1st Ave"

Items テーブル

PK	Items_IDX	ItemID	Quantity
"Order_221354"	0	89123	1
"Order_221354"	1	92312	5

DigitalOcean への接続

DigitalOcean はクラウド インフラストラクチャプロバイダーです。DigitalOcean データ コネクターを使用して、組織の DigitalOcean データをインポートできます。

メモ

DigitalOcean データ コネクターは当社のデータ アクセスおよび接続 パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の DigitalOcean 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から DigitalOcean のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

DigitalOcean 接続を作成する

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、**DigitalOcean** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
DigitalOceanへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

DigitalOceanの接続は、 **[既存の接続]** タブに保存されます。今後、DigitalOceanに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。DigitalOceanからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

DigitalOcean データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、DigitalOcean に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

DocuSign への接続

DocuSign は電子契約管理ツールです。DocuSign データコネクタを使用して、組織の DocuSign データをインポートできます。

メモ

DocuSign データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

DocuSign に接続するには、DocuSign サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の DocuSign 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から DocuSign のサポートにお問い合わせください。

DocuSign 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**DocuSign** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。

接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。

7. ダイアログボックスが表示されるので、**OK**]をクリックします。

DocuSignへの接続が正常に確立されます。

8. **DSN 構成**]ダイアログボックスで **OK**]をクリックします。

DocuSignの接続は、**既存の接続**]タブに保存されます。今後、DocuSignに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。DocuSignからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

DocuSign データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**既存の接続**]タブに保存されます。名前を変更する場合は、**既存の接続**]タブに移動し、**ACL DSN コネクター**]セクションを展開して、DocuSign に対応する省略記号アイコン

 をクリックし、**接続名の変更**]を選択します。

Dropbox への接続

Dropbox はクラウドベースのファイルホスティング サービスです。Dropbox データ コネクタを使用して、組織の BDropbox データをインポートできます。

メモ

Dropbox データ コネクタは当社のデータ アクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Dropbox 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dropbox のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Dropbox 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Dropbox を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Dropboxへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dropboxの接続は、 **[既存の接続]** タブに保存されます。今後、Dropboxに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Dropboxからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Dropbox データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Dropbox に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Dynamics CRM への接続

Microsoft Dynamics CRM は顧客関係管理 (CRM) システムです。Dynamics CRM データコネクタを使用して、組織の Dynamics CRM データをインポートできます。

メモ

Dynamics CRM データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics CRM に接続するには、次の情報を収集する必要があります。

- 組織のルート URL
- ユーザー資格情報
- 接続している Dynamics CRM サーバーの種類

接続の前提条件の収集については、組織の Dynamics CRM 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics CRM のサポートにお問い合わせください。

Dynamics CRM 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、Dynamics CRM を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Dynamics CRMへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics CRMの接続は、**[既存の接続]** タブに保存されます。今後、Dynamics CRMに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Dynamics CRMからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Dynamics CRM データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Dynamics CRM に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Dynamics GP への接続

Microsoft Dynamics GP は、財務、棚卸資産、業務を管理するためのビジネス会計ソフトウェアです。Dynamics GP データコネクタを使用して、組織の Dynamics GP データをインポートできます。

メモ

Dynamics GP データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics GP に接続するには、次の情報を収集する必要があります。

- 組織の一意の識別子
- Dynamics GP サーバーの URL
- Dynamics GP サーバーに接続するためのユーザー資格情報

接続の前提条件の収集については、組織の Dynamics GP 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics GP のサポートにお問い合わせください。

Dynamics GP 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Dynamics GP を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Dynamics GPへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics GPの接続は、**[既存の接続]** タブに保存されます。今後、Dynamics GPに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Dynamics GPからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Dynamics GP データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSNコネクタ]** セクションを展開して、Dynamics GP に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Dynamics NAV への接続

Microsoft Dynamics NAV は企業資源計画 (ERP) システムです。Dynamics NAV データコネクターを使用して、組織の Dynamics NAV データをインポートできます。

メモ

Dynamics NAV データコネクターは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics NAV に接続するには、次の情報を収集する必要があります。

- Dynamics NAV サーバーのルート URL
- Dynamics NAV サーバーのインスタンス
- Dynamics NAV サーバーに接続するためのユーザー資格情報
- サーバーに接続するために使用される認証スキーム

接続の前提条件の収集については、組織の Dynamics NAV 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics NAV のサポートにお問い合わせください。

Dynamics NAV 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター (バンドル)** セクションの **新規の接続**] タブから、**Dynamics NAV** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Dynamics NAVへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics NAVの接続は、**[既存の接続]** タブに保存されます。今後、Dynamics NAVに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Dynamics NAVからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Dynamics NAV データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Dynamics NAV に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Dynamics 365 Business Central への接続

Microsoft Dynamics 365 Business Central は企業資源計画 (ERP) システムです。Dynamics 365 Business Central データコネクタを使用して、組織の Dynamics 365 Business Central データをインポートできます。

メモ

Dynamics 365 Business Central データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics 365 Business Central に接続するには、次の情報を収集する必要があります。

- Dynamics 365 Business Central で使用されている会社の名前
- Dynamics 365 Business Central 組織への URL
- データにアクセスするための Microsoft Online テナント

接続の前提条件の収集については、組織の Dynamics 365 Business Central 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics 365 Business Central のサポートにお問い合わせください。

Dynamics 365 Business Central 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ (バンドル)** セクションの **新規の接続** タブから、**Dynamics 365 Business Central** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。

4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Dynamics 365 Business Centralへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics 365 Business Centralの接続は、 **[既存の接続]** タブに保存されます。今後、Dynamics 365 Business Centralに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Dynamics 365 Business Centralからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Dynamics 365 Business Central データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Dynamics 365

Business Central に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Dynamics 365 Finance and Operations への接続

Microsoft Dynamics 365 Finance and Operations はクラウドベースの企業資源計画 (ERP) システムです。Dynamics 365 Finance and Operations データコネクタを使用して、組織の Dynamics 365 Finance and Operations データをインポートできます。

メモ

Dynamics 365 Finance and Operations データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics 365 Finance and Operations に接続するには、次の情報を収集する必要があります。

- Dynamics 365 Business Central 組織への URL
- データにアクセスするための Microsoft Online テナント
- すべての会社またはデフォルトの会社からデータをインポートするかどうか

接続の前提条件の収集については、組織の Dynamics 365 Finance and Operations 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics 365 Finance and Operations のサポートにお問い合わせください。

Dynamics 365 Finance and Operations 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、Dynamics 365 Finance and Operations を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ポップアップに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。

4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Dynamics 365 Finance and Operationsへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics 365 Finance and Operationsの接続は、 **[既存の接続]** タブに保存されます。今後、Dynamics 365 Finance and Operationsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Dynamics 365 Finance and Operationsからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Dynamics 365 Finance and Operations データ コネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Dynamics 365 Finance and Operations に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Dynamics 365 Sales への接続

Microsoft Dynamics 365 Sales は組織の営業部門向けの顧客関係管理 (CRM) ソリューションです。Dynamics 365 Sales データコネクターを使用して、組織の Dynamics 365 Sales データをインポートできます。

メモ

Dynamics 365 Sales データコネクターは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Dynamics 365 Sales に接続するには、次の情報を収集する必要があります。

- Dynamics 365 Business Central 組織への URL
- データにアクセスするための Microsoft Online テナント

接続の前提条件の収集については、組織の Dynamics 365 Sales 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Dynamics 365 Sales のサポートにお問い合わせください。

Dynamics 365 Sales 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター (バンドル)** セクションの **新規の接続**] タブから、**Dynamics 365 Sales** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Dynamics 365 Salesへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Dynamics 365 Salesの接続は、**[既存の接続]** タブに保存されます。今後、Dynamics 365 Salesに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Dynamics 365 Salesからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Dynamics 365 Sales データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、Dynamics 365 Sales に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Edgar Online への接続

Edgar Online は、株式、投資信託、およびその他の上場資産の金融データおよび財務報告書を作成および配布するソリューションです。Edgar Online データコネクタを使用して、組織の Edgar Online データをインポートできます。

メモ

Edgar Online データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Edgar Online に接続するには、Edgar Online アカウントに接続するために、現在認証されているユーザーの AppKey を収集する必要があります。

接続の前提条件の収集については、組織の Edgar Online 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Edgar Online のサポートにお問い合わせください。

Edgar Online 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Edgar Online** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Edgar Onlineへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Edgar Onlineの接続は、**[既存の接続]** タブに保存されます。今後、Edgar Onlineに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Edgar Onlineからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Edgar Online データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、Edgar Online に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Elasticsearch への接続

Elasticsearch は、データストアとアナリティクスエンジンを備えた検索プラットフォームです。Elasticsearch データコネクタを使用すると、組織の Elasticsearch データをインポートできます。

メモ

Elasticsearch データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Elasticsearch 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Elasticsearch のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Elasticsearch 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Elasticsearch** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Elasticsearchへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Elasticsearchの接続は、 **[既存の接続]** タブに保存されます。今後、Elasticsearchに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Elasticsearchからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Elasticsearch データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Elasticsearch に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

電子メールへの接続

電子メールデータコネクターを使用して、標準メールプロトコルの IMAP または POP を使用している単一のアカウントの電子メールメッセージをインポートします。電子メールサーバーに接続するときには、各テーブル名はサーバーのメールボックスフォルダーを表し、各レコードは電子メールメッセージを表します。

メモ

このコネクターを Analytics から独立して使用することはできません。コネクタードライバーの DSN は Windows の ODBC データソースアドミニストレーターを使って設定できますが、DSN 接続を本コネクターの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

このコネクターはメールサーバーにのみ格納された単一のアカウントの電子メールメッセージを取得します。一部の電子メールクライアントに含まれる、チャットや ToDo などの機能には接続しません。

作業を開始する前に

電子メールに接続するには、以下を収集する必要があります。

- 電子メールサーバーのドメイン名または IP アドレス
- 正しい接続ポート
- ユーザー名とパスワードを含む接続ユーザーアカウント

メモ

お使いの電子メールサーバーでは IMAP プロトコルまたは POP プロトコルをご利用いただく必要があります。

接続の前提条件の収集については、組織の電子メール管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から電子メールのサポートにお問い合わせください。

電子メール接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクターセクションの [新規の接続] タブから、電子メールを選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、[接続のフィルタリング] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

3. **データ接続の設定**]パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続**]をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

電子メールの接続は、**既存の接続**]タブに保存されます。今後、電子メールに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。電子メールからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
プロトコル	接続先の電子メールサーバーの種類： <ul style="list-style-type: none"> ◦ IMAP ◦ POP 	IMAP
ユーザー	認証するために使用される電子メールアカウントのユーザー。パスワードとともに、このフィールドは電子メールサーバーで認証するために使用されます。	recipient@example.com
パスワード	認証するために使用される電子メールアカウントのパスワード。ユーザーとともに、このフィールドは電子メールサーバーで認証するために使用されます。	
サーバー	メールサーバーの名前または IP アドレス。 このプロパティは、メールサーバーの IP アドレスまたはドメイン名を指定します。接続を試行する前に設定する必要があり、接続中には変更できません。	imap.gmail.com
ポート	メールサーバーのポート。デフォルト値： <ul style="list-style-type: none"> ◦ IMAP - 143(非 SSL) または 993 (SSL) ◦ POP - 110(非 SSL) または 995 (SSL) 接続を実行するには、有効なポート番号(1 ~ 65535 の値)が必要です。プロパティは接続を試行する前に設定する必要があり、接続中には変更できません。	993

詳細設定

設定	説明	例
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	true
電子メール サービス	<p>使用している電子メール サービスの名前。</p> <p>操作しているサービスの IMAP 接続を最適化します。次のいずれかを使用できます。</p> <ul style="list-style-type: none"> ○ AOL ○ Gmail ○ Outlook ○ Yahoo ○ その他 <p>メモ</p> <p>電子メールコネクタを使用すると、Outlook 365 アカウントのタスクまたはカレンダー メールボックスの件名フィールドの内容を取得できません。Outlook 365 アカウントに接続している場合は、Exchange コネクタを使用することを検討してください。詳細については以下を参照 "Exchange への接続" ページ 514</p>	その他
メッセージを含む	<p>メッセージ本文コンテンツと添付ファイルデータを含めるかどうか。</p> <p>注意</p> <p>この設定はパフォーマンスに影響するため、クエリで多数のレコードを処理する場合にはタイムアウトになる可能性があります。</p>	false
HTML	メッセージ本文が HTML かテキストかどうかを決定します。	true
キープアライブ	インスタンス全体で接続をアライブにたままにするかどうかを決定します。	true

設定	説明	例
リスト メールボックス	すべてのメールボックスを一覧表示するか、サブスクライブされた IMAP メールボックスのみを一覧表示するかどうか。 IMAP のみ: <ul style="list-style-type: none"> すべて サブスクライブ済み 	すべて
鍵 サイズの上限	主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクターが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。	-1
WVarchar にマッピング	文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。	true
最大項目	返す最大項目数。 デフォルト値は -1 です。この値はすべての項目が返されることを保証します。	-1
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。	MyTable=*
SSL サーバー証明書	TLS/SSL を使用して接続するときにサーバーから許可される証明書。次の	C:\cert.cer

設定	説明	例
	<p>いずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サンプリント(16進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サンプリント(16進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	
拡張 SQL のサポート	<p>メモリ内クライアント側処理を有効にし、API 経由で直接サポートされるよりも、SQL 機能を強化します。</p> <ul style="list-style-type: none"> ◦ true - コネクタはできるかぎり SELECT 文処理の負荷を IMAP に分散し、メモリのクエリの残りを処理します。この方法で、ドライバーはサポートされていない述語、結合、集計を実行できます。 ◦ false - コネクタは SQL 実行を IMAP API のサポート内容に制限します。 <p>述語の実行</p> <p>コネクタは、データソースでサポートされている節を決定してから、ソースにプッシュし、クエリを満たす行の最も小さいスーパーセットを取得します。その後、行の残りをローカルでフィルタリングします。フィルター処理はストリームされ、ドライバーは非常に大きいデータセットでも効果的にフィルタリングできます。</p> <p>結合の実行</p> <p>コネクタはさまざまな手法を使用してメモリで結合します。ドライバーは、同じテーブルを複数回読み取る要件に対して、メモリ利用率をトレードオフします。</p> <p>集計の実行</p>	

設定	説明	例
	コネクタは、メモリ内の集計を処理するために必要なすべての行を取得します。	
UID モード	true の場合、デフォルトの一時 ID の代わりに、永久メッセージ ID が使用されます。	false
大文字の ID	テーブルおよび列名を含む、すべての識別子を大文字で報告します。	false
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	システムプロキシ設定を使用するかどうか	true

設定	説明	例
	<p>かを指定します。ProxyAutoDetect を FALSE に設定すると、カスタム プロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。</p>	
プロキシ ユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。Windows または Kerberos 認証を使用している場合は、このプロパティを次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	john_doe@example.com
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合があります。</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p>	206.174.193.115

設定	説明	例
	デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetectをfalseに設定します。	
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	ProxyServer プロキシに接続するときに使用する SSL タイプ: <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではありません。 ◦ TUNNEL - 接続はトンネルプロキシ経由です。プロキシ サーバーはリモート ホストへの接続を開き、トラフィックはプロキシ経由で前後します。 	AUTO

メッセージの本文および添付ファイルの取得

デフォルトでは、メッセージの本文(および添付ファイル)が返されるのは、テーブルから1つのレコードを選択した場合のみです。複数のレコードが返される場合には、これらのフィールドは空白のままになります。

1つのクエリで複数のレコードのメッセージ本文を取得したい場合は、**[高度な設定]の[コンテンツを含める]**オプションを設定する必要があります。これらのフィールドを返すことは多大なリソースを消費するため、多数のレコードについてメッセージ本文を返すとパフォーマンスに影響します。メッセージの本文または添付ファイルを調べる必要がある場合は、他のフィールドを使って、詳しく分析したいメッセージの内容を把握してください。その後で、これらのメッセージを個別にクエリして、メッセージの本文と添付ファイルを調べます。

データ コネクタの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクタを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1つまたは複数のデータ コネクタを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセス ウィンドウの **既存の接続** タブで、接続名の横にある  **>キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

電子メールデータコネクターの変更内容

電子メールデータコネクターに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	変更
14.2	<p>既存のフィールド、To と From には、電子メールアドレスのみが表示されるようになりました。</p> <p>新しいフィールド、FullTo と FullFrom には、電子メールアドレスと電子メールエイリアスが表示されるようになりました。</p>
15.0	<p>コネクターの 最大項目数 フィールドのデフォルト値は -1 です。この値が指定されているときには、コネクターはインポート中にすべての項目を返します。</p> <p>最大項目数 が 100 または他の値に設定されている場合に、15.0 にアップグレードした後は、コネクターが指定された数のレコードからの項目のみを返します。</p> <p>前のバージョンの Analytics で ACCESSDATA コマンドを使用してすべての項目を返していた場合は、15.0 にアップグレードするときに、スクリプトを開き、maxitems の値を -1 に更新して、スクリプトを再実行します。</p>

Epicor ERP への接続

Epicor ERP は企業資源計画 (ERP) ソリューションです。Epicor ERP データコネクタを使用して、組織の Epicor ERP データをインポートできます。

メモ

Epicor ERP データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Epicor ERP に接続するには、次の情報を収集する必要があります。

- インストールされている ERP アプリケーションの名前
- ERP インスタンスがホストされているサーバーの URL
- ERP インスタンスを認証するためのユーザー資格情報
- データを取得するサービス

接続の前提条件の収集については、組織の Epicor ERP 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Epicor ERP のサポートにお問い合わせください。

Epicor ERP 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Epicor ERP を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Epicor ERPへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Epicor ERPの接続は、 **[既存の接続]** タブに保存されます。今後、Epicor ERPに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Epicor ERPからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Epicor ERP データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Epicor ERP に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Exact Online への接続

Exact Online はクラウドベースの会計および顧客関係管理 (CRM) ソリューションです。Exact Online データコネクタを使用して、組織の Exact Online データをインポートできます。

メモ

Exact Online データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

Exact Online に接続するには、次の情報を収集する必要があります。

- 接続している Exact Online サーバーのリージョン
- OAuth 承認サーバーにアプリケーションを登録するときに割り当てられるクライアント ID
- OAuth 承認サーバーにアプリケーションを登録するときに割り当てられるクライアント シークレット

接続の前提条件の収集については、組織の Exact Online 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Exact Online のサポートにお問い合わせください。

Exact Online 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Exact Online** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Exact Onlineへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Exact Onlineの接続は、 **[既存の接続]** タブに保存されます。今後、Exact Onlineに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Exact Onlineからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Exact Online データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Exact Online に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Excel Online への接続

Excel Online は Web 版の Microsoft Excel スプレッドシート プログラムです。Excel Online データコネクタを使用して、組織の Excel Online データをインポートできます

メモ

Excel Online データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Excel Online 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Excel Online のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Excel Online 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Excel Online** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Excel Onlineへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Excel Onlineの接続は、**[既存の接続]** タブに保存されます。今後、Excel Onlineに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Excel Onlineからデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376を参照してください。

接続名の変更

Excel Online データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Excel Online に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Exchange への接続

Microsoft Exchange の電子メールとカレンダー サーバーからデータをインポートするには、Exchange データコネクタを使用します。単一の Exchange アカウントからデータをインポートできます。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタドライバの DSN は Windows の ODBC データソースアドミニストレーターを使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

Exchange に接続するには、以下を収集する必要があります。

- Exchange サーバーのドメイン名または IP アドレス(Exchange Web サービス URL)
- 接続先の Exchange プラットフォームのバージョン
- ユーザー名とパスワードを含む接続ユーザー アカウント

接続の前提条件の収集については、組織の Exchange 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Exchange のサポートにお問い合わせください。

Exchange 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Exchange を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Exchange の接続は、**既存の接続** タブに保存されます。今後、Exchange に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Exchange からデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	認証するために使用される Exchange アカウントのユーザー。パスワードとともに、このフィールドはサーバーで認証するために使用されます。	recipient@example.com
パスワード	認証するために使用される Exchange アカウントのパスワード。ユーザーとともに、このフィールドはサーバーで認証するために使用されます。	
サーバー	接続先の Exchange サーバーのアドレス。 これは Exchange Web サービス URL に設定してください。Exchange Online の場合、 https://outlook.office365.com/EWS/Exchange.asmx に設定します。	https://outlook.office365.com/EWS/Exchange.asmx
プラットフォーム	Exchange サーバーに関連付けられたプラットフォーム。	Exchange_Online

詳細設定

設定	説明	例
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	true
認証スキーム	認証で使用されるスキーム。 パスワードとユーザーとともに、このフィールドは、サーバーで認証するために使用されます。BASIC はデフォルトのオプションです。次のオプションを使用して、認証スキームを選択します。 <ul style="list-style-type: none"> ◦ NTLM - Windows 資格情報で認証します。 ◦ BASIC -HTTP BASIC 認証を使用します。 	BASIC

設定	説明	例
	<p>メモ</p> <p>Microsoft は、2020 年 10 月に、Exchange Web Services の基本認証を廃止することを発表しました。代替認証スキームを使用することを検討してください。</p> <ul style="list-style-type: none"> ○ DIGEST - HTTP DIGEST 認証を使用します。 ○ NEGOTIATE - は認証メカニズムをサーバーとネゴシエートします。AuthScheme を NEGOTIATE に設定すると、Kerberos 認証を使用します。 ○ KERBEROSDELEGATION - Kerberos プロトコル経由で委任を使用します。なりすますアカウントのユーザーとパスワードを設定します。 	
なりすましタイプ	<p>Exchange サイトに要求を送信している間になりすましで使用する識別子のタイプ。</p> <ul style="list-style-type: none"> ○ PrincipalName - なりすますために使用するアカウントのユーザープリンシパル名 (UPN) を表します。これはユーザー アカウントが存在するドメインの UPN です。 ○ SID - なりすましで使用するアカウントのセキュリティ ID (SID) のセキュリティ説明定義言語 (SDDL) を表します ○ PrimarySmtpAddress - Exchange なりすましで使用するアカウントの主簡易メール転送プロトコル (SMTP) を表します。主 SMTP アドレスが指定されている場合、Active Directory ディレクトリサービス ルックアップを使用して、ユーザーの SID を使用します。使用可能な場合は、SID または UPN を使用することをお勧めします ○ SmtpAddress - Exchange なりすましで使用するアカウントの簡易メール転送プロトコル (SMTP) を表します。SMTP アドレスが指定されている場合、Active Directory ルックアップを使用して、ユーザーの SID を使用します。使用可能な場合は、SID または UPN を使用することをお勧めします 	PrincipalName

設定	説明	例
なりすましユーザー	Exchange サイトに要求を送信している間になりすましユーザー。	
コンテンツを含める	返されるすべてのレコードにメッセージの本文内容を含めるかどうか。 <div style="border-left: 2px solid red; padding-left: 10px; margin-left: 20px;"> <p>注意</p> <p>この設定はパフォーマンスに影響するため、クエリで多数のレコードを処理する場合にはタイムアウトになる可能性があります。</p> </div>	false
鍵サイズの上 限	主キー列の最大長。サイズを0に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列がSQL_LONGVARCHARとして返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHARではなく、SQL_LONGVARCHARとして報告されます。	-1
WVarchar にマッピング	文字列タイプをSQL_VARCHARではなく、SQL_WVARCHARにマッピングするかどうかを制御します。デフォルトで設定されます。 文字列をSQL_WVARCHARにマッピングし、さまざまな国際文字セット格納するため、MapToWVarcharはデフォルトでtrueに設定されています。falseに設定すると、SQL_VARCHARを使用できます。	true
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できないEntity Frameworkで特に有用です。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3"の形式です。"*"文字を使用すると、すべてのテーブルとすべての列を含めることができます。	MyTable=*

設定	説明	例
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サムプリント(16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サムプリント(16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
拡張 SQL のサポート	<p>メモリ内クライアント側処理を有効にし、API 経由で直接サポートされるよりも、SQL 機能を強化します。</p> <ul style="list-style-type: none"> ◦ true - コネクターはできるかぎり SELECT 文処理の負荷を IMAP に分散し、メモリのクエリの残りを処理します。この方法で、ドライバーはサポートされていない述語、結合、集計を実行できます。 ◦ false - コネクターは SQL 実行を IMAP API のサポート内容に制限します。 <p>述語の実行</p> <p>コネクターは、データソースでサポートされている節を決定してから、ソースにプッシュし、クエリを満たす行の最も小さいスーパーセットを取得します。その後、行の残りをローカルでフィルタリングします。フィルター処理はストリームされ、ドライバーは非常に大きいデータセットでも効果的にフィルタリングできません。</p> <p>結合の実行</p> <p>コネクターはさまざまな手法を使用してメモリで結合します。ドライバーは、同じテーブルを複数回読み取る要件に対して、メモリ利用率をトレードオフします。</p>	

設定	説明	例
	<p>集計の実行</p> <p>コネクタは、メモリ内の集計を処理するために必要なすべての行を取得します。</p>	
大文字の ID	テーブルおよび列名を含む、すべての識別子を大文字で報告します。	false
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタムプロキシ	true

設定	説明	例
	設定を使用します。これは他のプロキシ設定よりも優先されます。	
プロキシ ユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。Windows または Kerberos 認証を使用している場合は、このプロパティを次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	john_doe@example.com
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使</p>	206.174.193.115

設定	説明	例
	用する場合は、ProxyAutoDetect を false に設定します。	
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	ProxyServer プロキシに接続するとき使用する SSL タイプ: <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではありません。 ◦ TUNNEL - 接続はトンネルプロキシ経由です。プロキシサーバーはリモートホストへの接続を開き、トラフィックはプロキシ経由で前後します。 	AUTO

ユーザーアカウントになりすます

Exchange の *ApplicationImpersonation* ロールと必要な権限を持つユーザーは、以下の2つの方法で他ユーザーのメールボックスのデータを読み取ることができます。

- **なりすましタイプ**]および **なりすましユーザー**]フィールドを使って接続を設定する
- SQL モードで WHERE 句の中で **ImpersonationUser** フィルターを使用する

```
SELECT * FROM "Exchange"."DRAFTS" "DRAFTS" WHERE
ImpersonationUser='user@example.com'
```

メッセージ本文を取得する

デフォルトでは、メッセージ本文が返されるのは、テーブルから1つのレコードを選択した場合のみです。複数のレコードが返される場合には、これらのフィールドは空白のままになります。

1つのクエリで複数のレコードのメッセージ本文を取得したい場合は、**高度な設定**]の **コンテンツを含める**]オプションを設定する必要があります。

メモ

このコネクタでは、添付ファイルの名前をリストすることはできますが、添付ファイルの内容にアクセスすることはできません。メッセージ本文の内容にのみアクセスできます。

メッセージ本文を返すことは多大なリソースを消費するため、多数のレコードについてメッセージ本文を返すとパフォーマンスに影響します。メッセージ本文を調べる必要がある場合は、他のフィールドを使って、詳しく分析したいメッセージの内容を把握してください。その後で、これらのメッセージを個別にクエリして、各メッセージのメッセージ本文を調べます。

フィルターリングの制限事項

次のフィルター条件およびフィールドの組み合わせはサポートされていません。

フィールド	サポートされていない演算子
すべての日付時刻フィールド	<ul style="list-style-type: none"> 日付(=)
SenderName	<ul style="list-style-type: none"> である(=) で始まる(LIKE "%値") 含む(LIKE "%値%")
SenderEmailAddress	
FromName	

データ コネクタの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクタを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータ コネクタを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータ アクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データ ソースに接続してインポートを再実行する前に、コネクタのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データ アクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Exchange データコネクタの変更内容

Exchange データコネクタに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	Exchange のテーブル	変更
14.2	該当なし	<p>コネクタにおける Exchange スキーマの名前が <code>Exchange</code> から <code>EWS</code> に変更されました。</p> <p>ACCESSDATA コマンドで必要な更新の例：</p> <ul style="list-style-type: none"> ◦ 旧 - <code>SELECT * FROM "Exchange"."Calendar" "Calendar"</code> ◦ 新 - <code>SELECT * FROM "EWS"."Calendar" "Calendar"</code>
	カレンダー	<p>フィールドの名前変更：</p> <ul style="list-style-type: none"> ◦ <code>Recurrence_DayOfWeek</code> が <code>Recurrence_DaysOfWeek</code> に名前変更されました <p>追加されたフィールド：</p> <ul style="list-style-type: none"> ◦ <code>ModifiedOccurrences_Aggregate</code> ◦ <code>DeletedOccurrences_Aggregate</code> <p>データ型が変更されました：</p> <ul style="list-style-type: none"> ◦ <code>Recurrence_StartDate</code> が日付型から日付時刻型に変更されました ◦ <code>Recurrence_EndDate</code> が日付型から日付時刻型に変更されました
	受信箱と送信済み項目	<p>フィールドの名前変更：</p> <ul style="list-style-type: none"> ◦ <code>ToRecipients_Name</code> が <code>ToRecipients_Names</code> に名前変更されました ◦ <code>ToRecipients_EmailAddress</code> が <code>ToRecipients_EmailAddresses</code> に名前変更されました ◦ <code>ToRecipients_ItemId</code> が <code>ToRecipients_ItemIds</code> に名前変更されました ◦ <code>CcRecipients_Name</code> が <code>CcRecipients_Names</code> に名前変更されました ◦ <code>CcRecipients_EmailAddress</code> が <code>CcRecipients_EmailAddresses</code> に名前変更されました ◦ <code>CcRecipients_ItemId</code> が <code>CcRecipients_ItemIds</code> に名前変更されました ◦ <code>BccRecipients_Name</code> が <code>BccRecipients_Names</code> に名前変更されました ◦ <code>BccRecipients_EmailAddress</code> が <code>BccRecipients_EmailAddresses</code> に名前変更されました ◦ <code>BccRecipients_ItemId</code> が <code>BccRecipients_ItemIds</code> に名前変更されました
	タスク	<p>フィールドの名前変更：</p> <ul style="list-style-type: none"> ◦ <code>Recurrence_NumberOfOccurrences</code> が <code>Recurrence_NumberOfOccurrences</code> に名前変更されました

Google Analytics への接続

Google Analytics は Web サイトトラフィックの追跡、分析、レポートのための Web アナリティクスプラットフォームです。Google Analytics データコネクタを使用して、組織の Google Analytics データをインポートできます。

メモ

Google Analytics データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Google Analytics 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google Analytics のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Google Analytics 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Google Analytics** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Google Analyticsへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Google Analyticsの接続は、**[既存の接続]** タブに保存されます。今後、Google Analyticsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Google Analyticsからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Google Analytics データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Google Analytics に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Google BigQuery への接続

Google BigQuery はクラウド データ サービスです。Google BigQuery データ コネクタを使用して、BigQuery データをインポートできます。

メモ

Analytics は任意のコネクタとして Google BigQuery を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実装する" ページ 2663を参照してください。

作業を開始する前に

Google BigQuery に接続するには、以下を収集する必要があります。

- 認証詳細
- カタログ

接続の前提条件の収集については、組織の Google BigQuery 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google BigQuery のサポートにお問い合わせください。

BigQuery 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Google BigQuery を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Google BigQuery の接続は、**既存の接続** タブに保存されます。今後、Google BigQuery に接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステー징領域** が表示されるので、データのインポートを開始できます。Google BigQuery からデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
OAuth 機構	<p>ドライバーを認証する認証メカニズムを指定します。</p> <ul style="list-style-type: none"> ○ ユーザー認証 - ドライバーは Google ユーザー アカウント 経由でユーザーを認証します。 ○ サービス認証 - ドライバーは Google サービス アカウント 経由でサービスを認証します。 <p>ユーザー認証]を使用するときには、ログイン]をクリックして、Google サインイン]ページを表示します。</p>	ユーザー認証
確認コード	リフレッシュトークンを生成するために Google から取得するコード。	
リフレッシュトークン	BigQuery へのアクセスを承認するために Google から取得するリフレッシュトークン。リフレッシュトークンは、確認コードを入力した後に自動的に生成されます。	
電子メール	サービス認証を使用するときには、サービス アカウント の電子メール ID を入力します。	
キー ファイルのパス	サービス認証を構成するときには、このオプションを、サービス アカウント 認証 電子メール アドレスを認証するために使用される .p12 または .json 鍵ファイルへの完全パスに設定します。	
カタログ (プロジェクト)	Google BigQuery プロジェクト の名前。	

詳細設定

設定	説明	例
プロキシのホスト	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用し	

設定	説明	例
	て、HTTP プロキシで認証できます。	
プロキシのポート	プロキシ サービスが実行されている TCP ポート。 デフォルト値は 80 です。	80
プロキシのユーザー名	プロキシ サーバーで認証するために使用されるユーザー名。	
プロキシのパスワード	プロキシ サーバーで認証するためのパスワード。	
1 秒あたりの最大要求数 (0=無制限)	1 秒あたりの最大要求数を指定します。 スロットリングなしで 1 秒あたりの要求数を無制限にするには、0 を入力します。	0
ブロックあたりの行数	各データ要求でドライバーが取得する必要がある最大行数。	16384
文字列型列のデフォルト長	STRING 列に含まれる最大文字数。	65536
大きな結果セットを収容するデータセット名	大きい結果セットの場合に一時テーブルに格納するために使用する BigQuery データセットの ID。 大きい結果セットのサポートを有効にする場合にのみ、このオプションの値を指定します。 大きい結果セットを許可]オプションを選択した場合にのみ、このフィールドが有効です。	_odbc_temp_tables
一時テーブルの有効期間 (ms)	一時テーブルが期限切れになるまでの時間(秒)。テーブルの有効期限を設定しない場合は、値 0 を指定します。	3600000
言語の方言	ドライバーがクエリを実行するときに、標準 SQL 構文を使用するか、レガシー BigQuery SQL 構文を使用するかを指定します。 <ul style="list-style-type: none"> ◦ 有効 - 標準 SQL 構文 ◦ 無効 - レガシー BigQuery SQL 構文 	
SQLPrepare メタデータとレガシー SQL を両方有効化 (低速)	このクエリで BigQuery のレガシー SQL ダイアレクトを使用するかどうかを指定します。	

設定	説明	例
大きな結果セットを許可する	レガシー SQL を使用するとき、28MB を超える結果をクエリするかどうかを指定します。	

データコネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データコネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータコネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセスウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセスウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Google Cloud Storage への接続

Google Cloud Storage は、Google Cloud Platform 上のデータを管理するための Web ベースのファイルストレージサービスです。Google Cloud Storage データコネクタを使用して、組織の Google Cloud Storage データをインポートできます。

メモ

Google Cloud Storage データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Google Cloud Storage 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google Cloud Storage のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Google Cloud Storage の接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Google Cloud Storage を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Google Cloud Storageへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Google Cloud Storageの接続は、 **[既存の接続]** タブに保存されます。今後、Google Cloud Storageに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Google Cloud Storageからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Google Cloud Storage データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Google Cloud Storage に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Google Contacts への接続

Google Contacts は Web ベースの連絡先管理ツールです。Google Contacts データコネクターを使用して、組織の Google Contacts データをインポートできます。

メモ

Google Contacts データコネクターは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Google Contacts 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google Contacts のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Google Contacts 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター(バンドル)** セクションの **新規の接続** タブから、**Google Contacts** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Google Contactsへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Google Contactsの接続は、**[既存の接続]** タブに保存されます。今後、Google Contactsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Google Contactsからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Google Contacts データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Google Contacts に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Google Drive への接続

Google Drive はクラウドベースのファイルストレージ サービスです。Google Drive データ コネクターを使用して、組織の Google Drive データをインポートできます。

メモ

Google Drive データ コネクターは当社のデータ アクセスおよび接続 パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Google Drive 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google Drive のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Google Drive 接続を作成する

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、**Google Drive** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Google Driveへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Google Driveの接続は、 **[既存の接続]** タブに保存されます。今後、Google Driveに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Google Driveからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Google Drive データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクター]** セクションを展開して、Google Drive に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Google Sheets への接続

Google Sheets は Web ベースのスプレッドシート プログラムです。Google Sheets データ コネクターを使用して、組織の Google Sheets データをインポートできます。

メモ

Google Sheets データ コネクターは当社のデータ アクセスおよび接続 パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Google Sheets 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Google Sheets のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Google Sheets 接続を作成する

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、**Google Sheets** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Google Sheetsへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Google Sheetsの接続は、**[既存の接続]** タブに保存されます。今後、Google Sheetsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Google Sheetsからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Google Sheets データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Google Sheets に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

IBM Cloudant への接続

IBM Cloudant は、オープンソースの Apache CouchDB をベースとした分散型データベースです。IBM Cloudant データコネクタを使用して、組織の IBM Cloudant データをインポートできます。

メモ

IBM Cloudant データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の IBM Cloudant 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から IBM Cloudant のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

IBM Cloudant 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、IBM Cloudant を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
IBM Cloudantへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

IBM Cloudantの接続は、 **[既存の接続]** タブに保存されます。今後、IBM Cloudantに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。IBM Cloudantからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

IBM Cloudant データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、IBM Cloudant に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Jira への接続

Jira は、ソフトウェア問題追跡およびプロジェクト管理を行うための、クラウド ベースまたはサーバー ベースのプラットフォームです。御社の Jira データにアクセスするには、Jira データ コネクタを使用してください。

作業を開始する前に

Jira 資格情報

Jira に接続するには、以下を収集する必要があります。

- Jira ユーザー名
- Jira API トークン(クラウド アカウント)、または Jira のパスワード(サーバー インスタンス)
- 御社の Jira プラットフォームのホスト名

接続の前提条件の収集については、組織の Jira 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Jira のサポートにお問い合わせください。

API トークンの取得

Jira のクラウド アカウントに接続するには、アカウント認証のために API トークンが必要になります。トークンを生成するには、Atlassian のアカウントにログインし、[API tokens] ページに移動して **[Create API token]** をクリックします。生成されたトークンをコピーして、それを **[Data Connection Settings]** パネルの **[API Token]** に入力します。

Jira の "Issues" テーブル

Jira の "Issues" テーブルには、組織で作成した多くのカスタムフィールドを追加できます。このため、Issues テーブルは Jira API により動的に取り扱われ、ユーザーが同テーブルに接続するたびに同テーブルに含まれるフィールドが組み立てられます。Jira 管理者が Issues テーブルのカスタムフィールドを変更すると、特定のフィールド選択を想定している Analytics スクリプトが失敗します。

この問題を解決するには、Analytics のデータ アクセス ウィンドウを使って Jira のインポートを手動で行います。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。Issues テーブルへの変更と一致するように、スクリプト本体のフィールド指定を更新します。

Jira 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Jira を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Jira の接続は、**既存の接続** タブに保存されます。今後、Jira に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Jira からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	Jira クラウド アカウントのホスト名、または Jira サーバー インスタンス。	https://your_org.atlassian.net
インスタンス タイプ	御社によって使用されている Jira プラットフォームのタイプ: クラウド ベースまたはオンプレミス サーバー	クラウド
ユーザー	Jira への認証に使用するユーザー アカウント	admin_1@your_org.com
API トークン	Jira のクラウド アカウントの場合: 現在認証しようとしているユーザーの Jira API トークン	
パスワード	Jira のサーバー インスタンスの場合: ユーザーの認証に使用する Jira パスワード	
カスタム フィールドを含める	Jira テーブルの標準フィールドのほか	true

設定	説明	例
	に、御社によって追加されたカスタムフィールドを含めます。	

詳細設定

設定	説明	例
鍵サイズの上限	主キー列の最大長。サイズを0に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。	-1
WVarchar にマッピング	文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。	true
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。	MyTable=*
SSL サーバー証明書	TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。 <ul style="list-style-type: none"> 完全 PEM 証明書 	C:\cert.cer

設定	説明	例
	<ul style="list-style-type: none"> ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サムプリント (16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サムプリント (16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	
日付時刻を GMT に変換	<p>インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。</p>	true
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> </div> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC -HTTP BASIC 認証を実行します。 	BASIC

設定	説明	例
	<ul style="list-style-type: none"> ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシ ユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、このプロパティを次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	john_doe@example.com
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を</p>	

設定	説明	例
	Windows パスワードに設定します。 Kerberos 認証を完了するためにこれらが必要な場合もあります。	
プロキシ サーバー	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。 HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。 デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	ProxyServer プロキシに接続するときに使用する SSL タイプ: <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではありません。 ◦ TUNNEL - 接続はトンネルプロキシ経由です。プロキシサーバーはリモート ホストへの接続を開き、トラフィックはプロキシ経由で前後します。 	AUTO

データ コネクタの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクタを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサード パーティ ベンダーが加えた変更により、1 つまたは複数のデータ コネクタを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータ アクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクタのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセスウィンドウの **既存の接続** タブで、接続名の横にある  **>キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Jira データコネクタの変更内容

Jira データコネクタに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	Jira のテーブル	追加されたフィールド	削除されたフィールド
14.2	Attachments	AuthorAccountId	AuthorKey ContentEncoded FilePath Name
	Boards		FilterId (このフィールドを参照する、スクリプトとして作成された既存の Jira インポートは引き続き動作しますが、このフィールドはもはや存在していません)。
	Comments	AuthorAccountId UpdateAuthorAccountId	AuthorKey AuthorName UpdateAuthorKey UpdateAuthorName
	IssueChangeLogs	AuthorAccountId AuthorDisplayName	AuthorKey
	Projects	LeadAccountId	LeadKey LeadName Recent

データの定義とインポート

Analytics のバージョン	Jira のテーブル	追加されたフィールド	削除されたフィールド
	SecurityLevels	SecuritySchemeld	Link
	Users	AccountId	Key Name IncludeInactive
	Watchers	AccountId	Key Name
	Worklogs	AuthorAccountId UpdateAuthorAccountId	AuthorName UpdateAuthorName

JSON サービスへの接続

JavaScript Object Notation(JSON) は、属性と値のペアと配列データ型から成るデータオブジェクトを送信するのに使用される標準的なファイル形式です。JSON は Web サービスとの通信によく使用される形式です。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタードライバの DSN は Windows の **ODBC データソースアドミニストレータ** を使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

メモ

接続で認証が必要な場合は、ACL Robotics の Professional Edition で JSON コネクタを使用できません。

作業を開始する前に

JSON サービスに接続するには、問い合わせ対象となるファイルの構造を判別すると共に、お使いのファイルが行が含まれる表形式の構造に分割するのに必要な JSONPath または XPath の構文を作成する必要があります。各接続について 1 つのテーブルをインポートできます。

接続の前提条件の収集については、御社の JSON サービスの管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から接続先となるシステムのサポート サービスにお問い合わせください。

認証が必要な JSON 接続

接続する JSON サービスでトークンまたはユーザー資格情報での認証が必要な場合は、Windows **ODBC Data Source Administrator** でデータ接続 (DSN) を作成する必要があります。認証情報を含む DNS を作成した後、データアクセス ウィンドウで選択できます。

JSON 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、JSON を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

JSONの接続は、**既存の接続** タブに保存されます。今後、JSONに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。JSONからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
テーブル	<p>JSONドキュメント内の配列要素のJSONPath(JSONドキュメントを複数の行に分割するのに使用されます)。</p> <p>これは、配列要素のJSONPath(またはXPathの構文)を指定するもので、JSONドキュメントを複数の行に分割するのに使用されます。</p> <p>このプロパティは、スキーマファイル(RSDファイル)が存在しない場合にスキーマ定義を生成するのに使用されます。</p>	\$. <値>

設定	説明	例
	<p>メモ</p> <p>JSONPathとXPathは標準化されたクエリ形式です。この構文とキーワードの詳細については、オンラインのリソースを参照してください。</p>	
JSON の場所	JSON リソースの Uniform Resource Identifier(URI) または絶対ファイルパス	http://sample.example.net/examples.json

詳細設定

設定	説明	例
鍵サイズの上限	<p>主キー列の最大長。サイズを0に接続すると、キー長が元の長さに戻ります。</p> <p>このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。</p>	255
Long Varchar にマッピング	<p>列が SQL_LONGVARCHAR として返されるかどうかを制御します。</p> <p>この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。</p>	-1
WVarchar にマッピング	<p>文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p> <p>文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。</p>	MyTable=*

設定	説明	例
	この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。	
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サムプリント (16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サムプリント (16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	true
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p>	BASIC

設定	説明	例
	<p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetectをfalseに設定し、ProxyServerとProxyPortもfalseに設定する必要があります。認証するには、ProxyAuthSchemeを設定し、必要に応じて、ProxyUserとProxyPasswordを設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLMまたはKerberosトークンを取得します。 ◦ PROPRIETARY - NTLMまたはKerberos tokenを生成しません。HTTP要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetectをFALSEに設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシ ユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、このプロパティを次</p>	john_doe@example.com

設定	説明	例
	<p>の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。</p>	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ:</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではあ 	AUTO

設定	説明	例
	<p>りません。</p> <ul style="list-style-type: none">○ TUNNEL - 接続はトンネルプロキシ経由です。プロキシサーバーはリモートホストへの接続を開き、トラフィックはプロキシ経由で前後します。	

Kintone への接続

Kintone は、チームのデータ、ワークフロー、コミュニケーションを一元化するワークプレイスプラットフォームです。Kintone データコネクタを使用して、組織の Kintone データをインポートできます。

メモ

Kintone データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Kintone 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Kintone のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Kintone 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Kintone を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Kintone への接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Kintone の接続は、 **[既存の接続]** タブに保存されます。今後、Kintone に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Kintone からデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376 を参照してください。

接続名の変更

Kintone データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Kintone に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

LDAP への接続

LDAP(Lightweight Directory Access Protocol: ライト ウェイト ディレクトリアクセス プロトコル) は、IP (Internet Protocol: インターネット プロトコル) ネット ワーク上で分散型ディレクトリ情報サービスをアクセス、管理するための、業界標準のアプリケーション プロトコルです。LDAP コネクターは、ODBC 接続をサポートする任意のアプリケーションから直接、LDAP データにリアルタイムでアクセスするのに使用します。

メモ

このコネクターを Analytics から独立して使用することはできません。コネクタードライバーの DSN は Windows の ODBC データソース アドミニストレーターを使って設定できますが、DSN 接続を本コネクターの [Windows DSN configuration(Windows DSN 設定)] ダイアログ 内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

LDAP データに接続するには、以下を収集する必要があります。

- LDAP サーバーのドメイン名または IP アドレス
- 正しい接続ポート
- ユーザー名とパスワードの識別名を含む接続ユーザー アカウント

接続の前提条件の収集については、御社の LDAP 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から LDAP システムのアプリケーション サポートにお問い合わせください。

LDAP 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクター セクションの **新規の接続**] タブから、LDAP を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット 順に表示されます。

3. **データ接続の設定**] パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続**] をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

LDAP の接続は、**既存の接続**] タブに保存されます。今後、LDAP に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。LDAPからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	ユーザーの識別名。 このフィールドとパスワードを組み合わせることで、LDAP サーバーで認証を行います。	MYDOMAIN\test
パスワード	指定したユーザーの識別名のパスワード。 このフィールドとユーザーを組み合わせることで、LDAP サーバーで認証を行います。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>お使いのLDAP サーバーで匿名接続が許可される場合は、パスワードを指定しなくても接続を行うことができます。お使いのサーバーのセキュリティ設定によっては、匿名接続であっても、使用可能なテーブルをリストすることができます。ただし、匿名接続では、リストされたテーブルの一部または全部からデータを選択できない場合があります。お使いのLDAP セキュリティ設定の詳細については、御社の管理者にお問い合わせください。</p> </div>	
サーバー	LDAP サーバーのドメイン名または IP これには、サーバーのドメイン名または IP のみを含めればよく、LDAP:\ の部分を含める必要はありません。	10.120.1.110
ポート	LDAP サーバーを実行するポート。デフォルト値は 389 です。 このプロパティとサーバーを組み合わせ	389

設定	説明	例
	<p>ることで、LDAP サーバーを指定します。</p>	
ベース DN	<p>結果を特定のサブツリーに制限するために使用される、識別名のベース部分。</p> <p>ベース DN を指定すると、検査が必要なエントリ数を制限することで、大きいサーバーのエントリを返すときに、パフォーマンスを大幅に改善できます。</p>	DC=myConnection,DC=com
LDAP バージョン	<p>サーバーに接続し、サーバーと通信するために使用される LDAP バージョン。有効なオプションは、LDAP バージョン 2 および 3 で 2 と 3 です。</p> <p>このコネクタは、RFC 1777、2251 などの LDAP RFC で指定されている、標準の LDAP クライアントです。</p>	2
認証機構	<p>LDAP サーバーに接続するときに使用される認証メカニズム:</p> <ul style="list-style-type: none"> ◦ SIMPLE (デフォルト) - デフォルトのテキスト認証がサーバーへのログインで使用されます ◦ DIGESTMD5 - より安全な DIGEST-MD5 認証が使用されます ◦ NEGOTIATE - NTLM/ネゴシエート認証が使用されます 	SIMPLE
範囲	<p>検索範囲を制限するかどうか:</p> <ul style="list-style-type: none"> ◦ WholeSubtree - サブツリー全体 (BaseDN とすべての子) ◦ SingleLevel - 単一レベル (BaseDN と直接の子) ◦ BaseObject - ベースオブジェクト (BaseDN のみ) <p>ヒント 範囲を制限すると、各パフォーマンスが大幅に改善されます。</p>	BaseObject

詳細設定

設定	説明	例
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	true
鍵サイズの上限	主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。	-1
WVarchar にマッピング	文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。	true
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。	MyTable=*
大文字の ID	テーブルおよび列名を含む、すべての識別子を大文字で報告します。	false

設定	説明	例
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> 完全 PEM 証明書 証明書を含むローカルファイルへのパス 公開鍵 MD5 サムプリント(16 進数値はスペースまたはコロン区切りにすることができます) SHA1 サムプリント(16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
拡張 SQL のサポート	<p>メモリ内クライアント側処理を有効にし、API 経由で直接サポートされるよりも、SQL 機能を強化します。</p> <ul style="list-style-type: none"> true - コネクターはできるかぎり SELECT 文処理の負荷を LDAP サーバーに分散し、メモリのクエリの残りを処理します。この方法で、ドライバーはサポートされていない述語、結合、集計を実行できます。 false - コネクターは SQL 実行を LDAP API のサポート内容に制限します。 <p>メモ</p> <p>Where 節構文を使用してフィルタリングをサポートするには、この設定を false にする必要があります。</p> <p>述語の実行</p> <p>コネクターは、データソースでサポートされている節を決定してから、ソースにプッシュし、クエリを満たす行の最も小さいスーパーセットを取得します。その後、行の残りをローカルでフィルタリングします。フィルター処理はストリームされ、ドライバーは非常に大きいデータセットでも効果的にフィルタリングできます。</p>	false

設定	説明	例
	<p>結合の実行</p> <p>コネクタはさまざまな手法を使用してメモリで結合します。ドライバーは、同じテーブルを複数回読み取る要件に対して、メモリ利用率をトレードオフします。</p> <p>集計の実行</p> <p>コネクタは、メモリ内の集計を処理するために必要なすべての行を取得します。</p>	

返される行のフィルタリング

このコネクタでは、LDAP 検索構文と密接に連携した SQL フィルタリング構文が使用されます。一部のフィールドには、複数のオブジェクト属性を表す区切られたデータがあります。WHERE 節は、単一の文字列ではなく、一意の値であるかのように、これらの区切られたフィールドの各値を占める必要があります。

ObjectCategory と ObjectClass のユーザーのフィルタリング

シナリオ

ユーザーテーブルを操作し、ObjectClass に次の属性があるレコードをインポートします。

- ユーザー
- ユーザー

また、ObjectCategory にユーザーではなく、コンピューター属性があるものにレコードを制限します。

テーブルへの接続

まず、LDAP サーバーに接続し、ユーザーテーブルを選択します(表示されるフィールドのサブセット)。

ObjectCategory (ASCII)	ObjectClass (ASCII)	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	
CN=Computer,CN=Schema,CN=Configuration,...	top;person;organizationalPerson;user;computer	
CN=Person,CN=Schema,CN=Configuration,DC...	top;person;organizationalPerson;user	

レコードのフィルターリング

インポートするものにレコードを制限するには、個別フィールドとして各区切られた値を処理するフィルターを適用します。

+ 3 フィルター適用済み - 編集 ^

"User"."ObjectClass"	である	person	🗑️
AND OR			
"User"."ObjectClass"	である	user	🗑️
AND OR			
"User"."ObjectCategory"	である	Computer	🗑️
+ フィルターの追加			

SQL モードを使用して、フィルターが作成する WHERE 節を検証します。

```
WHERE
(
  "User"."ObjectClass" = N'person' AND
  "User"."ObjectClass" = N'user' AND
  "User"."ObjectCategory" = N'Computer'
)
```

フィルター結果

フィルターが適用されると、テーブルには WHERE 節と一致するレコードが含まれ、テーブルをインポートします。

ObjectCategory (ASCII)	ObjectClass (ASCII)	
CN=Computer,CN=Schema,CN=Configuration,	top;person;organizationalPerson;user;computer	

LDAP テーブル同士の結合

LDAP 準拠データベースで使用されるデータモデルのため、SQL 結合は推奨されません。結合により、予期しない結果になる場合があります。

1 つ以上のテーブルを LDAP データソースから結合する必要がある場合は、データアクセス ウィンドウを使用して複数のテーブルをインポートしてから、Analytics で結合します。フィルターを使用して、レコード数を制限し、効率化します。

LinkedIn への接続

LinkedIn はビジネスプロフェッショナル向けのプロフェッショナル ネットワーキング Web サイトです。LinkedIn データコネクタを使用して、組織の LinkedIn データをインポートできます。

メモ

LinkedIn データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

LinkedIn に接続するには、次の情報を収集する必要があります。

- 現在のログインユーザーの会社 ID
- アクセストークン用にユーザーが必要な特定のスコープ

接続の前提条件の収集については、組織の LinkedIn 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から LinkedIn のサポートにお問い合わせください。

LinkedIn 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **[新規の接続]** タブから、**LinkedIn** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**[接続のフィルタリング]** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **[必須フィールドを表示]** または **[基本]** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]**をクリックします。
LinkedInへの接続が正常に確立されます。
8. **[DSN 構成]**ダイアログボックスで **[OK]**をクリックします。

LinkedInの接続は、**[既存の接続]**タブに保存されます。今後、LinkedInに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。LinkedInからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

LinkedIn データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]**タブに保存されます。名前を変更する場合は、**[既存の接続]**タブに移動し、**[ACL DSN コネクタ]**セクションを展開して、LinkedIn に対応する省略記号アイコン  をクリックし、**[接続名の変更]**を選択します。

Marketo への接続

Marketo はマーケティング自動化プラットフォームです。Marketo REST API エンドポイントを指定し、コネクタを使用して、Marketo システムからデータをインポートします。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタドライバの DSN は Windows の ODBC データソースアドミニストレータを使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

Marketo に接続するには、アクセスしようとするエンドポイントの OAuth 資格情報を構成および取得する必要があります。OAuth 資格情報の取得については、[Marketo 認証ドキュメント](#)を参照してください。

メモ

Marketo の API サービスは、1 日の要求数とシミュレーション要求数が制限されます。これらの制限で問題が発生する場合は、Marketo 管理者または Marketo サポートまでお問い合わせください。

接続の前提条件の収集については、組織の Marketo 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Marketo のサポートにお問い合わせください。

Marketo 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Marketo を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Marketo の接続は、**既存の接続** タブに保存されます。今後、Marketo に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Marketoからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
OAuth クライアント Id	クライアント ID は、Marketo の OAuth 承認サーバーにアプリケーションを登録するときに割り当てられます。	xvz1evFS4wEEPTGEFPHBog
OAuth クライアント シークレット	クライアント シークレットは、Marketo の OAuth 承認サーバーにアプリケーションを登録するときに割り当てられます。	L8qq9PZyRg6ieKGEKhZolGC0vJWLw8iEJ88DRdyOg
Rest エンドポイント	REST Web サービス エンドポイントの URL は、Marketo Web サイトの Admin ページで、Marketo によって提供されています。	https://064-CCJ-768.mktorest.com/rest

詳細設定

設定	説明	例
鍵サイズの上 限	主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクターが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_	-1

設定	説明	例
	LONGVARCHAR として報告され ます。	
WVarchar にマッピング	<p>文字列タイプを SQL_VARCHAR では なく、SQL_WVARCHAR にマッピングす るかどうかを制御します。デフォルトで設 定されます。</p> <p>文字列を SQL_WVARCHAR にマッピ ングし、さまざまな国際文字セット 格納 するため、MapToWVarchar はデフォルト で true に設定されています。false に 設定すると、SQL_VARCHAR を使用 できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかど うかを指定します。</p> <p>この設定は、テーブル列ではない場合 に、擬似列の値を設定できない Entity Framework で特に有用です。</p> <p>この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文 字を使用すると、すべてのテーブルとす べての列を含めることができます。</p>	MyTable=*
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときに サーバーから許可される証明書。次の いずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカル ファイルへの パス ◦ 公開鍵 ◦ MD5 サンプリント (16 進数値はス ペースまたはコロン区切りにするこ とができます) ◦ SHA1 サンプリント (16 進数値はス ペースまたはコロン区切りにするこ とができます) <p>TLS/SSL 接続を使用している場合 は、このプロパティを使用して、サーバー から許可される TLS/SSL 証明書を指 定できます。コンピューターから信頼さ れない他の認定書はすべて拒否されま す。</p>	C:\cert.cer
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するた めに使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキ シで認証するために使用する認証タイ</p>	BASIC

設定	説明	例
	<p>ブを指定します。</p> <p>メモ コネクターはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p>	john_doe@example.com

設定	説明	例
	<p>Windows または Kerberos 認証を使用している場合は、このプロパティを次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。</p>	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ:</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効 	AUTO

設定	説明	例
	<p>です。</p> <ul style="list-style-type: none">◦ NEVER - 接続は SSL 有効ではありません。◦ TUNNEL - 接続はトンネルプロキシ経由です。プロキシ サーバーはリモート ホストへの接続を開き、トラフィックはプロキシ経由で前後します。	

MarkLogic への接続

MarkLogic は、データ統合のためのクラウドプラットフォームです。MarkLogic データ コネクタを使用すると、組織の MarkLogic データをインポートできます。

メモ

MarkLogic データ コネクタは当社のデータ アクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の MarkLogic 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から MarkLogic のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

MarkLogic 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**MarkLogic** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
MarkLogicへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

MarkLogicの接続は、 **[既存の接続]** タブに保存されます。今後、MarkLogicに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。MarkLogicからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

MarkLogic データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、MarkLogic に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Microsoft Access への接続

Microsoft Access はデータベース管理システムです。Microsoft Access データコネクタを使用して、Microsoft Access データをインポートします。

メモ

Microsoft Access データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Microsoft Access 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Microsoft Access のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Microsoft Access 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Microsoft Access** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Microsoft Accessへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Microsoft Accessの接続は、 **[既存の接続]** タブに保存されます。今後、Microsoft Accessに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Microsoft Accessからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Microsoft Access データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Microsoft Access に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Microsoft OneDrive への接続

Microsoft OneDrive は Web ベースのファイルホスティング サービスです。Microsoft OneDrive データコネクタを使用して、Microsoft OneDrive データをインポートできます。

メモ

Microsoft OneDrive データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Microsoft OneDrive 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Microsoft OneDrive のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Microsoft OneDrive 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Microsoft OneDrive** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Microsoft OneDriveへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Microsoft OneDriveの接続は、 **[既存の接続]** タブに保存されます。今後、Microsoft OneDriveに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Microsoft OneDriveからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Microsoft OneDrive データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Microsoft OneDrive に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Microsoft SQL Server への接続

Microsoft SQL Server は広く使用されているリレーショナルデータベース管理システムです。SQL Server データコネクタを使用して、SQL Server データをインポートできます。

メモ

Analytics は任意のコネクタとして SQL Server を提供します。データアクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データコネクタと Python エンジンを実装する」ページ 2663 を参照してください。

作業を開始する前に

SQL Server に接続するには、以下を収集する必要があります。

- データベース サーバーのホスト名
- 正しい接続ポート
- サービスプリンシパル名 (SPN) (統合 Windows 認証を使用する場合)
- ユーザー名とパスワード (標準認証を使用する場合)
- 接続先のスキーマとテーブルの読み取りアクセス

接続の前提条件の収集については、組織の SQL Server 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SQL Server のサポートにお問い合わせください。

SQL Server 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、SQL Server を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

SQL Server の接続は、**既存の接続** タブに保存されます。今後、SQL Server に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SQL Serverからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

設定	説明	例
サーバー	SQL Server インスタンスの名前またはネットワークアドレス。	
ポート	SQL Server データベースをホストするサーバーのポート。	1433
データベース	指定したサーバーで実行されているSQL Server データベースの名前。	
暗号化	<p>ドライバーがサーバーへのTLS/SSL接続のネゴシエーションを試行するかどうかを設定します。</p> <p>デフォルトでは、ドライバーは、システムの信頼できる証明書ストアに対して、サーバー証明書を確認します。</p> <p>別の証明書を指定するには、SSLServerCertを設定します。</p>	false
CA 証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。</p> <p>次のいずれかを使用できます。</p> <ul style="list-style-type: none"> ◦ 完全な PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 または SHA1 サムプリント(スペースまたはコロンで 16 進数値を区切ることができます) <p>コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	cacerts.pem
サーバー証明書を信頼する	<p>Windows Integrated Security で SQL Server を認証するかどうかを指定します。</p> <p>True に設定すると、Windows 認証を実行するために Windows ID が使用されます。</p> <p>False に設定すると、SQL Server 認証が使用されます。</p>	false

設定	説明	例
セキュリティ接続を使用する	ドライバーが Kerberos プロトコルを使用して SQL Server への接続を認証するかどうかを指定します。	true
サーバーの SPN	SQL Server インスタンスのサービスプリンシパル名。	
ユーザー	SQL Server で認証するために指定されたユーザー名。	
パスワード	指定したユーザーを SQL Server で認証するためのパスワード。	

Microsoft Teams への接続

Microsoft Teams は、ビジネスコミュニケーションプラットフォームです。Microsoft Teams データ コネクタを使用して、組織の Microsoft Teams データをインポートできます。

メモ

Microsoft Teams データ コネクタは当社のデータ アクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Microsoft Teams 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Microsoft Teams のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Microsoft Teams 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Microsoft Teams** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Microsoft Teamsへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Microsoft Teamsの接続は、**[既存の接続]** タブに保存されます。今後、Microsoft Teamsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Microsoft Teamsからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Microsoft Teams データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSNコネクタ]** セクションを展開して、Microsoft Teams に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

MongoDB への接続

MongoDB はクラウド データ サービスです。MongoDB データ コネクタを使用して、MongoDB データをインポートできます。

メモ

Analytics は任意のコネクタとして MongoDB を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実装する" ページ 2663を参照してください。

作業を開始する前に

MongoDB に接続するには、以下を収集する必要があります。

- MongoDB データベースをホストするサーバーのサーバー名または IP アドレス
- 正しい接続ポート
- MongoDB データベースの名前

接続の前提条件の収集については、組織の MongoDB 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から MongoDB のサポートにお問い合わせください。

MongoDB 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、MongoDB を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

MongoDB の接続は、**既存の接続** タブに保存されます。今後、MongoDB に接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。MongoDB からデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
サーバー	MongoDB データベースをホストするサーバーのホスト名または IP アドレス	
ポート	MongoDB サーバーがクライアント接続をリスニングするために使用する TCP ポートの番号。 デフォルト値は 27017 です。	27017
データベース	MongoDB データベースの名前。	
レプリカセットへの接続	ドライバーが MongoDB 実装のレプリカセットにアクセスできるかどうかを指定します。 <ul style="list-style-type: none"> 無効 - ドライバーはレプリカセットにアクセスできません。 有効 - ドライバーは MongoDB 実装のレプリカセットにアクセスできます。 	無効
レプリカセット名	ドライバーがアクセスするレプリカセットの名前。	
セカンダリサーバー	レプリカセットに接続するときに使用するサーバーのカンマ区切りのリスト。 サーバーがクライアント接続をリスニングするために使用している TCP ポートを指定するには、コロン(:)とポート番号をサーバー名または IP アドレスの後ろに付けます。	
認証機構	MongoDB が接続を認証するために使用する認証メカニズム。 <ul style="list-style-type: none"> 認証なし - ドライバーは接続を認証しません。 MongoDB ユーザー名とパスワード - SCRAM-SHA-1 プロトコルを使用して認証します。これは MongoDB で使用されるデフォルトの認証プロトコルです。 Kerberos - Kerberos プロトコルを使用して認証します。 	認証なし

設定	説明	例
	<ul style="list-style-type: none"> LDAP - LDAP プロトコルを使用して認証します。 	
サービス名	MongoDB サーバーの Kerberos サービスプリンシパル名。	mongodb
認証のタイプ	<p>認証用の MongoDB データベースの名前。</p> <p>認証データベースがデータを取得するためのデータベースとは異なる場合にのみ、この値が必須です。</p>	admin
ユーザー名	MongoDB で認証するためのユーザー名。	
パスワード	MongoDB で認証するためのパスワード。	

詳細設定

設定	説明	例
SSL の有効化	ドライバーがサーバーに接続するために SSL を使用するかどうかを指定します。	無効
自己署名証明書の許可	サーバーからの自己署名 SSL 証明書を許可するかどうかを指定します。	無効
PEM 鍵ファイル	クライアントを検証するための証明書を含む .pem ファイルへの完全パス。	
PEM 鍵のパスワード	[PEM 鍵ファイル] フィールドで指定されたクライアント証明書ファイルのパスワード。	
証明局のファイル	サーバーを検証するために使用する .pem ファイルの完全パス。	
証明局のディレクトリ	サーバーを検証するための .pem ファイルを含むディレクトリの完全パス。この設定により、ドライバーは SSL 検証で複数の .pem ファイルにアクセスできます。	
証明書失効リスト ファイル	取り消された証明書のリストを含む .pem ファイルの完全パス。	
ダブルバッファリングの有効化	ダブルバッファリングを使用してデータを	有効

設定	説明	例
	取得するかどうかを指定します。 MongoDB ドライバーは、SELECT オペレーション中に、ダブルバッファリングを使用して、ドライバーのパフォーマンスを改善することができます。	
文字列を SQL_WVARCHAR として公開	文字列データ型が SQL_WVARCHAR または SQL_VARCHAR のどちらにマッピングされるかどうかを指定します。	有効
バイナリを SQL_LONGVARIABLE として公開	SQL_LONGVARIABLE または SQL_VARIABLE のどちらの型としてバイナリ列を返すかどうかを指定します。	有効
継承の有効化	ドライバーが仮想テーブル間の結合を最適化し、フィルタリングおよび集計最適化を処理のために MongoDB データベースに渡すかどうかを指定します。	有効
ブロックあたりにフェッチできるドキュメント数	一度にクエリが返すドキュメントの最大数。ダブルバッファリングが有効なときに使用されるバッファサイズもこの設定で決まります。 デフォルト値は 4096 です。	4096
文字列型の列のサイズ	STRING 列に含まれる最大文字数。 デフォルト値は 255 です。	255
バイナリ型の列のサイズ	バイナリ型の列の最大データ長さ。 デフォルト値は 32767 です。	32767
メタデータ機構	ドライバーがスキーマ定義を検索する場所を指定します。 <ul style="list-style-type: none"> データベース - MongoDB データベースからスキーマ定義を読み込みます。 ローカルファイル - [ローカルファイル] フィールドで指定された JSON ファイルからスキーマ定義を読み込みます。 	データベース
ローカルファイル	MongoDB に接続するときにドライバーで使用するスキーマ定義を含むローカル JSON ファイルの完全パス。	
サンプリング手法	一時スキーマ定義を生成するときにデータをサンプリングする方法を指定します。	前向き

データの定義とインポート

設定	説明	例
	<ul style="list-style-type: none"> ◦ 前方 - データベースの最初のレコードから順にデータをサンプリングします。 ◦ 後方 - データベースの最後のレコードから逆順にデータをサンプリングします。 	
サンプリングするドキュメント数 (全ドキュメントの場合は0)	<p>一時スキーマ定義を生成するためにサンプリングできるレコードの最大数。</p> <p>このオプションが0に設定されると、データベースのすべてのドキュメントがサンプリングされます。</p> <p>デフォルト値は100です。</p>	100
サンプリングステップ数	<p>データベースをスキャンして一時スキーマ定義を生成するときに、レコードをサンプリングする間隔。たとえば、このオプションを2に設定すると、データベースのレコードが2件ごとにサンプリングされます。</p> <p>デフォルト値は1です。</p>	1
書き戻しバッチサイズ (1 から 1000)	<p>書き込みオペレーション中に一度に処理できるドキュメントの最大数。</p> <p>デフォルト値は500です。</p>	500
書き込み確認	<p>書き込みオペレーションの成功を報告するために書き込みオペレーションを確認する必要がある、プライマリおよびセカンダリサーバーの合計数。</p> <p>このオプションが0に設定されると、書き込みオペレーションを確認する必要がありません。</p> <p>デフォルト値は1です。</p>	1
書き込み確認のタイムアウト	<p>オペレーションが失敗したことを報告する前に、セカンダリサーバーが書き込みオペレーションを確認までにドライバーが待機する最大秒数。</p> <p>このオプションが0に設定されると、ドライバーはタイムアウトしません。代わりに、ドライバーはすべてのセカンダリサーバーが書き込みオペレーションを確認するまで待機し、その後にオペレーションが成功したことを報告します。</p> <p>デフォルト値は0です。</p>	0
書き込み確認ありでの記録対象書き	書き込みオペレーションを確認する前	無効

設定	説明	例
込み数	に、書き込みオペレーションのデータをジャーナルにコミットする必要があるかどうかを指定します。	

データコネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データコネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータコネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセスウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセスウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

MongoDB データコネクターの変更

MongoDB データコネクターに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	変更
14.2	このコネクターでは、MongoDB 3.0 および 3.2 への接続がサポートされなくなりました。MongoDB 3.4、3.6、4.0 には接続することができます。

MySQL への接続

MySQL は一般的なオープンソース関係データベース管理システムです。組織の MySQL データをインポートするには、MySQL データコネクタを使用します。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタドライバの DSN は Windows の ODBC データソースアドミニストレータを使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

MySQL に接続するには、以下を収集する必要があります。

- データベース サーバーのホスト名
- 正しい接続ポート
- ユーザー名とパスワード(標準認証を使用する場合)
- 接続先のスキーマとテーブルの読み取りアクセス

接続の前提条件の収集については、組織の MySQL 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から MySQL のサポートにお問い合わせください。

MySQL 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、MySQL を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

MySQL の接続は、**既存の接続** タブに保存されます。今後、MySQL に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。MySQLからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
サーバー	MySQL データベースをホストするサーバーのホスト名またはIP。	192.168.0.1
ポート	MySQL データベースをホストするサーバーのポート。	3306
データベース	MySQL サーバーに接続するときに接続先となるデフォルト データベースの名前。	
ユーザー	MySQL に接続するときに認証するユーザー。	
パスワード	指定したユーザーを MySQL サーバーで認証するためのパスワード。	

詳細設定

設定	説明	例
統合セキュリティ	Windows Integrated Security で MySQL サーバーを認証するかどうかを指定します。	false
鍵サイズの上限	主キー列の最大長。サイズを0に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されま	-1

設定	説明	例
	す。	
WVarchar にマッピング	<p>文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p> <p>文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。</p>	MyTable=*
クエリパススルー	このオプションは、クエリをそのまま直接 MySQL サーバーに渡します。	true
大文字の ID	このプロパティは、テーブルおよび列名を含む、すべての識別子を報告します。	false

NetSuite への接続

NetSuite は Oracle が提供するクラウドベースの企業資源計画 (ERP) サービスです。NetSuite データコネクタを使用して、組織の NetSuite データをインポートできます。

メモ

NetSuite データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

NetSuite に接続するには、次の情報を収集する必要があります。

- NetSuite に関連付けられた会社アカウント
- ログインするためのユーザー資格情報
- OAuth アクセストークンと資格情報
- 使用するスキーマの種類

接続の前提条件の収集については、組織の NetSuite 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から NetSuite のサポートにお問い合わせください。

NetSuite 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**NetSuite** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
NetSuiteへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

NetSuiteの接続は、 **[既存の接続]** タブに保存されます。今後、NetSuiteに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。NetSuiteからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

NetSuite データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、NetSuite に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

OData への接続

OData はオープン REST ベースのプロトコルで、データの問い合わせと更新を実行します。OData データコネクタを使用して、組織の OData データをインポートできます。

メモ

OData データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

OData に接続するには、次の情報を収集する必要があります。

- 組織または OData サービスファイルのルート URL
- OData アカウントに接続するための資格情報
- 使用される認証スキーム
- データにアクセスするための Microsoft Online テナント
- 認証するための Azure Active リソース

接続の前提条件の収集については、組織の OData 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から OData のサポートにお問い合わせください。

OData 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、OData を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
ODataへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

ODataの接続は、 **[既存の接続]** タブに保存されます。今後、ODataに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。ODataからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

OData データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、OData に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Odoo への接続

Odoo は業務管理ソフトウェアスイートです。Odoo データ コネクタを使用して、組織の Odoo データをインポートできます。

メモ

Odoo データ コネクタは当社のデータ アクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Odoo 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Odoo のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Odoo 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ (バンドル)** セクションの **新規の接続** タブから、**Odoo** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Odooへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Odooの接続は、 **[既存の接続]** タブに保存されます。今後、Odooに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Odooからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Odoo データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Odoo に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Open Exchange Rates への接続

Open Exchange Rates は、200 を超える各国通貨およびデジタル通貨のリアルタイム データとデータ履歴を提供する外国為替 (FOREX) レート サービスです。信頼性の高い複数の情報源のデータがアルゴリズムによってブレンドされ、追跡されるので、完全性が保証されます。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタドライバの DSN は Windows の ODBC データソース アドミニストレータを使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration (Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

Open Exchange Rates のデータに接続するには、Open Exchange Rates に登録してそのアプリ ID を取得する必要があります。アプリ ID を取得するには、openexchangerates.org にサインアップします。

接続の前提条件の収集については、組織の Open Exchange Rates 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Open Exchange Rates のサポートにお問い合わせください。

Open Exchange Rates 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Open Exchange Rates を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Open Exchange Rates の接続は、**既存の接続** タブに保存されます。今後、Open Exchange Rates に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Open Exchange Rates からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
アプリケーション Id	Open Exchange Rates に登録したときに取得するアプリ ID。 この値は、Open Exchange Rates のダッシュボードに表示されます。	881BDCBDB7FAABF047A6178DF4956172

詳細設定

設定	説明	例
鍵サイズの上 限	主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。 このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。	255
Long Varchar にマッピング	列が SQL_LONGVARCHAR として返されるかどうかを制御します。 この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。	-1
WVarchar にマッピング	文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。	true
擬似列	擬似列を列としてテーブルに含めるかどうかを指定します。 この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity	MyTable=*

設定	説明	例
	<p>Framework で特に有用です。</p> <p>この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。</p>	
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サンプリント (16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サンプリント (16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
日付時刻を GMT に変換	<p>インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。</p>	true
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p>	BASIC

設定	説明	例
	<p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetectをfalseに設定し、ProxyServerとProxyPortもfalseに設定する必要があります。認証するには、ProxyAuthSchemeを設定し、必要に応じて、ProxyUserとProxyPasswordを設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLMまたはKerberosトークンを取得します。 ◦ PROPRIETARY - NTLMまたはKerberos tokenを生成しません。HTTP要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetectをFALSEに設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、このプロパティを次</p>	john_doe@example.com

設定	説明	例
	<p>の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。</p>	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ:</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではあ 	AUTO

設定	説明	例
	<p>りません。</p> <ul style="list-style-type: none">○ TUNNEL - 接続はトンネルプロキシ経由です。プロキシサーバーはリモートホストへの接続を開き、トラフィックはプロキシ経由で前後します。	

Oracle への接続

Oracle は広く使用されているリレーショナル データベース管理システム (RDBMS) です。Oracle データ コネクタを使用して、会社のオンプレミス Oracle データベースからデータをインポートできます。

メモ

Oracle データ コネクタは、Oracle Cloud または Oracle Fusion データ ソースからのデータのインポートをサポートしません。

ACL for Windows は任意のコネクタとして Oracle を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データ コネクタと Python エンジンを実装する」ページ 2663を参照してください。

作業を開始する前に

Oracle の資格情報

Oracle に接続するには、以下を収集する必要があります。

- Oracle のユーザー名 およびパスワード
- データベースのホスト名または IP アドレスおよびサービス名
- データベース内のテーブルへの読み取りアクセス許可

接続の前提条件の収集については、組織の Oracle 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Oracle のサポートにお問い合わせください。

Oracle Instant Client

このコネクタを使用するには、Analytics と同じコンピューターに Oracle Instant Client をインストールする必要があります。Instant Client が何ビット版であるかと、お使いのオペレーティングシステムが何ビット版であるかが一致する必要があります。32 ビット版の Oracle Instant Client を 64 ビット マシンにインストールしても、接続が失敗するためです。

Oracle 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続**] タブから、Oracle を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定**] パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続**] をクリックします。

デフォルトの**接続名**を使用するか、新しい接続名を入力します。

Oracle の接続は、**既存の接続**] タブに保存されます。今後、Oracle に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Oracle からデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
サーバー	Oracle データベースをホストするサーバーのホスト名または IP。	localhost
ポート	Oracle データベースをホストするサーバーに接続するためのポート。	1521
サービス名	Oracle データベースのサービス名。	XE
ユーザー	Oracle データベースで認証するためのユーザー ID。	SYSTEM
パスワード	指定したユーザーを Oracle データベースで認証するためのパスワード。	

詳細設定

設定	説明	例
閲覧可能なスキーマ	参照可能なデータベースオブジェクトツリーを制限する、スキーマのカンマ区切りのリストを提供します。	BrowsableSchemas=SYSTEM,SYS

Oracle Eloqua への接続

Oracle Eloqua はサービスとしてのマーケティングオートメーションソフトウェア(SaaS)プラットフォームです。Oracle Eloqua データコネクタを使用して、組織の Oracle Eloqua データをインポートできます。

メモ

Oracle Eloqua データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

Oracle Eloqua に接続するには、次の情報を収集する必要があります。

- Eloqua アカウントを使用する会社
- Eloqua アカウントに接続するための資格情報

接続の前提条件の収集については、組織の Oracle Eloqua 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Oracle Eloqua のサポートにお問い合わせください。

Oracle Eloqua 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション]を選択します。
2. ACL DSN コネクタ(バンドル)セクションの **新規の接続** タブから、Oracle Eloqua を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Oracle Eloquaへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Oracle Eloquaの接続は、**[既存の接続]** タブに保存されます。今後、Oracle Eloquaに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Oracle Eloquaからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Oracle Eloqua データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Oracle Eloqua に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Oracle HCM Cloud への接続

Oracle HCM Cloud は、クラウドベースの人的資本管理 (Human Capital Management: HCM) ソリューションです。Oracle HCM Cloud データコネクタを使用して、組織の Oracle HCM Cloud データをインポートできます。

Oracle HCM Cloud データコネクタは、Analytics のバージョン 16.1 以降で利用可能です。

メモ

Oracle HCM Cloud データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Oracle HCM Cloud 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Oracle HCM Cloud のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Oracle HCM Cloud の接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、Oracle HCM Cloud を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示]** または **基本]** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **テスト接続]** をクリックせずに **OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**OK]** をクリックします。
Oracle HCM Cloudへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **OK]** をクリックします。

Oracle HCM Cloudの接続は、**既存の接続]** タブに保存されます。今後、Oracle HCM Cloudに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Oracle HCM Cloudからデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376を参照してください。

接続名の変更

Oracle HCM Cloud データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **既存の接続]** タブに保存されます。名前を変更する場合は、**既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Oracle HCM Cloud に対応する省略記号アイコン  をクリックし、**接続名の変更]** を選択します。

Oracle Sales Cloud への接続

Oracle Sales Cloud は顧客関係管理 (CRM) ソリューションです。Oracle Sales Cloud データコネクタを使用して、組織の Oracle Sales Cloud データをインポートできます。

メモ

Oracle Sales Cloud データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Oracle Sales Cloud に接続するには、次の情報を収集する必要があります。

- Oracle Sales Cloud サーバーの URL
- サーバーの Oracle Sales Cloud アカウントに接続するための資格情報

接続の前提条件の収集については、組織の Oracle Sales Cloud 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Oracle Sales Cloud のサポートにお問い合わせください。

Oracle Sales Cloud 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、Oracle Sales Cloud を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Oracle Sales Cloudへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Oracle Sales Cloudの接続は、**[既存の接続]** タブに保存されます。今後、Oracle Sales Cloudに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Oracle Sales Cloudからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Oracle Sales Cloud データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Oracle Sales Cloud に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Parquet への接続

Apache Parquet は大量の複雑なデータ向けに設計されたオープンソース型の列ベースのデータストレージ形式です。Parquet データコネクタを使用して、組織の Parquet データをインポートできます。

メモ

Parquet データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Parquet 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Parquet のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Parquet 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Parquet** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Parquetへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Parquetの接続は、 **[既存の接続]** タブに保存されます。今後、Parquetに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Parquetからデータをインポートする方法については、「データアクセスウィンドウの作業」 ページ 376を参照してください。

接続名の変更

Parquet データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Parquet に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Presto への接続

Presto はオープンソース SQL クエリエンジンであり、さまざまなデータソースに対して対話型のアナリティクスクエリを実行します。Presto データコネクタを使用して、Presto データをインポートできます。

作業を開始する前に

Presto に接続するには、以下を収集する必要があります。

- ユーザー名とパスワード
- サーバーのホスト名または IP アドレス
- 正しい接続ポート
- 正しい認証スキーム

接続の前提条件の収集については、組織の Presto 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Presto のサポートにお問い合わせください。

Presto 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Presto を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Presto の接続は、**既存の接続** タブに保存されます。今後、Presto に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Presto からデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	Presto サーバーで認証する Presto ユーザー アカウント。	
パスワード	ユーザーをサーバーで認証するためのパスワード。	
サーバー	Presto REST サーバーのホスト名または IP アドレス。	
ポート	Presto REST サーバーのポート。	8080
カタログ	サーバーに対するすべての要求で使用 するカタログの名前。 この値が設定されていない場合、ドライ バーは Presto サーバーから使用可能 なカタログを取得します。	
認証スキーム	ドライバーが使用する認証スキームを 指定します。 使用可能なオプションは次のとおりで す。 <ul style="list-style-type: none"> ◦ なし - ドライバーは接続を認証しま せん。 ◦ LDAP - ドライバーは LDAP を使用 して接続を認証します。 ◦ Kerberos - ドライバーは Kerberos を使用して接続を認証します。 	
Kerberos KDC	ユーザーを認証するために使用される Kerberos Key Distribution Center (KDC) サービス。 Kerberos KDC が指定されていない場 合、ドライバーは KRB5 構成ファイルま たはドメイン名とホストから自動的にこ のプロパティを検出しようとします。	
Kerberos 領域	Kerberos Key Distribution Service (KDC) でユーザーを認証するために使 用される Kerberos 領域。	

設定	説明	例
Kerberos SPN	Kerberosドメインコントローラーのサービスプリンシパル名 (SPN)。 Kerberosドメインコントローラーで SPN が認証している URL と同じでない場合、このプロパティを使用して、SPN を設定できます。	
Kerberos Keytab ファイル	Kerberos プリンシパルと暗号化された鍵の組み合わせを含む Keytab ファイル。	
SSL を使用する	SSL を有効にするかどうかを指定します。	

詳細設定

設定	説明	例
鍵サイズの上限	主キー列の最大長。	255
Long Varchar にマッピング	このプロパティは、列が SQL_ LONGVARCHAR として返されるかどうかを制御します。	-1
WVarchar にマッピング	このプロパティは、文字列タイプを SQL_ VARCHAR ではなく、SQL_ WVARCHAR にマッピングするかどうかを制御します。	
擬似列	この値は、擬似列を列としてテーブルに含めるかどうかを指定します。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。	
SSL サーバー証明書	SSL を使用して接続するときにサーバーから許可される必要がある証明書。 次のいずれかを使用できます。 <ul style="list-style-type: none"> ◦ 完全な PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 または SHA1 サンプリント (スペースまたはコロンで 16 進数値を区切ることができます) コンピューターから信頼されない他の認	

設定	説明	例
	定書はすべて拒否されます。	
クエリパススルー	このオプションを有効にすると、すべてのクエリが直接 Presto に渡されます。	
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。このオプションが無効な場合、日付時刻値は、Analytics を実行するシステムのオペレーティングシステムタイムゾーンに変換されます。	
プロキシ認証スキーム	<p>プロキシ サーバーで認証するために使用する認証スキームを指定します。使用可能なオプションは次のとおりです。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に設定されたプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos トークンを生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。このオプションに設定された値は他のプロキシ設定よりも優先されます。このため、カスタムプロキシ設定を使用する場合は、このオプションを無効にしてください。	
プロキシ ユーザー	プロキシ サーバーで認証するためのユーザー名。	
プロキシのパスワード	プロキシ サーバーでプロキシ ユーザーを認証するためのパスワード。	
プロキシ サーバー	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。	
プロキシのポート	プロキシ サーバーが実行されている TCP ポート。	
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ。使用可能なオプションは次のとおりです。</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場 	AUTO

設定	説明	例
	<p>合、TUNNEL オプションを使用します。URL が HTTP URL の場合、コンポーネントは NEVER オプションを使用します。</p> <ul style="list-style-type: none"> ○ ALWAYS - 接続は常に SSL が有効です。 ○ NEVER - 接続は SSL が有効ではありません。 ○ TUNNEL - 接続はトンネルプロキシ経由です。 <p>このオプションは、プロキシ サーバーの値を入力するときのみ有効です。</p>	

Qualys への接続

Qualys はクラウドベースのセキュリティおよびコンプライアンス ソリューション スイートです。Qualys データコネクターを使用して、Qualys データをインポートできます。

作業を開始する前に

Qualys に接続するには、以下を収集する必要があります。

- Qualys サーバーのホスト名または IP アドレス
- ユーザー名とパスワード
- Qualys からデータをインポートするためのインポート モード
- スキャンされる資産
- 特定のスキャンの一意のスキャン参照 ID

接続の前提条件の収集については、組織の Qualys 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Qualys のサポートにお問い合わせください。

Qualys 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクターセクションの **新規の接続** タブから、Qualys を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Qualys の接続は、**既存の接続** タブに保存されます。今後、Qualys に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Qualys からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
サーバー	Qualys サーバーのホスト名または IP アドレス。	
ユーザー	Qualys サーバーで認証する Qualys ユーザー アカウント。	
パスワード	ユーザーをサーバーで認証するためのパスワード。	
モード	Qualys からデータをインポートするときに使用するインポート モードを指定します。 オプション: <ul style="list-style-type: none"> ◦ 資産グループ別に脆弱性をインポート - Qualys アカウントに関連付けられたすべての考えられる資産グループと脆弱性結果からデータをインポートできます。 ◦ KnowledgeBase をインポート - 特定の日付範囲内に修正または公開された脆弱性でフィルタリングされたナレッジベース データをインポートできます。 ◦ スキャン参照 ID 別にスキャンをインポート - Qualys アカウントに関連付けられたスキャン ID を指定し、そのスキャン結果を Analytics にインポートできます。 	資産グループ別に脆弱性をインポート
資産	データをスキャンしてインポートする資産を指定できます。資産を選択するには、 [サーバー] 、 [ユーザー] 、 [パスワード] フィールドの値を入力し、 [モード] フィールドで 資産グループ別に脆弱性をインポート を選択します。 資産のリストを取得し、複数の資産を選択するには、 資産を選択 ボタンをクリックする必要があります。CTRL キーを使用できます。	
フィルター	デフォルトまたはカスタム日付フィルター	過去 7 日間

設定	説明	例
	<p>を選択できます。</p> <p>オプション:</p> <ul style="list-style-type: none"> ◦ 過去 7 日間 ◦ 過去 14 日間 ◦ 過去 30 日 ◦ 過去 90 日間 ◦ 処理されたデータ 	
次の後に処理	<p>指定した日付の後に処理されたデータをインポートできます。</p> <p>このフィールドは、[フィルター]フィールドで 処理日 が選択されているときにのみ有効です。</p>	2019-08-17
次の前に処理	<p>指定した日付の前に処理されたデータをインポートできます。</p> <p>このフィールドは、[フィルター]フィールドで 処理日 が選択されているときに有効です。</p>	
フィルター条件	<p>公開日または変更日でデータをフィルタリングします。</p> <p>このフィールドは、[モード]フィールドで KnowledgeBase をインポート が選択されているときに有効です。</p>	公開日
前	<p>出力をフィルタリングして、特定の日付の前に公開または変更された脆弱性を表示します。</p>	
後	<p>出力をフィルタリングして、特定の日付の後に公開または変更された脆弱性を表示します。</p>	
スキャン参照 ID	<p>インポートするスキャンを指定できます。スキャンを選択するには、[サーバー]、[ユーザー]、[パスワード]フィールドの値を入力し、[モード]フィールドで スキャン参照 ID 別にスキャンをインポート を選択します。</p> <p>使用可能なスキャンのリストを取得するには、スキャンを選択 ボタンをクリックする必要があります。</p>	

詳細設定

設定	説明	例
プロキシを有効にする	プロキシ サーバーを使用するかどうかを指定します。	
プロキシ ユーザー	プロキシ サーバーで認証するためのユーザー名。	john_doe@example.com
プロキシのパスワード	プロキシ サーバーでプロキシ ユーザーを認証するためのパスワード。	
プロキシ サーバー	プロキシ サーバーのホスト名またはIPアドレス。	206.174.193.116
プロキシのポート	プロキシ サーバーが実行されているポート。	

QuickBooks への接続

QuickBooks はオンライン会計ソフトウェアであり、請求書の作成、費用の管理、取引詳細情報の分析が可能になります。QuickBooks データコネクタを使用して、組織の QuickBooks データをインポートできます。

メモ

QuickBooks データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

QuickBooks に接続するには、QuickBooks アカウントに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の QuickBooks 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から QuickBooks のサポートにお問い合わせください。

QuickBooks 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、QuickBooks を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **Test Connection** (接続のテスト) をクリックします。

メモ

Test Connection (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**テスト接続** をクリックせずに **OK** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]**をクリックします。
QuickBooksへの接続が正常に確立されます。
8. **[DSN 構成]**ダイアログボックスで **[OK]**をクリックします。

QuickBooksの接続は、**[既存の接続]**タブに保存されます。今後、QuickBooksに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。QuickBooksからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

QuickBooks データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]**タブに保存されます。名前を変更する場合は、**[既存の接続]**タブに移動し、**[ACL DSN コネクター]**セクションを展開して、QuickBooks に対応する省略記号アイコン  をクリックし、**[接続名の変更]**を選択します。

QuickBooks Online への接続

QuickBooks Online は中小企業向けのクラウドベースの会計ソリューションです。QuickBooks Online データコネクターを使用して、組織の QuickBooks Online データをインポートできます。

メモ

QuickBooks Online データコネクターは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

QuickBooks Online に接続するには、次の情報を収集する必要があります。

- サンドボックス アカウントを使用しているかどうか
- OAuth 承認サーバーにアプリケーションを登録するときに割り当てられるクライアント ID およびクライアント シークレット

接続の前提条件の収集については、組織の QuickBooks Online 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から QuickBooks Online のサポートにお問い合わせください。

QuickBooks Online 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター(バンドル)** セクションの **新規の接続**] タブから、**QuickBooks Online** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
QuickBooks Onlineへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

QuickBooks Onlineの接続は、 **[既存の接続]** タブに保存されます。今後、QuickBooks Onlineに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。QuickBooks Onlineからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

QuickBooks Online データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSNコネクタ]** セクションを展開して、QuickBooks Online に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

QuickBooks POS への接続

QuickBooks Point of Sale (POS) は中小企業向けの POS システムであり、ユーザーは売上、顧客を追跡し、在庫を管理できます。QuickBooks POS データコネクタを使用して、組織の QuickBooks POS データをインポートできます。

メモ

QuickBooks POS データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

QuickBooks POS に接続するには、次の情報を収集する必要があります。

- リモート コネクタの URL
- リモート コネクタに接続するための資格情報

接続の前提条件の収集については、組織の QuickBooks POS 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から QuickBooks POS のサポートにお問い合わせください。

QuickBooks POS 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続**] タブから、**QuickBooks POS** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
QuickBooks POSへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

QuickBooks POSの接続は、**[既存の接続]** タブに保存されます。今後、QuickBooks POSに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。QuickBooks POSからデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376を参照してください。

接続名の変更

QuickBooks POS データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、QuickBooks POS に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

REST サービスへの接続

REpresentational State Transfer (REST) は、Web 上のコンピューターシステム間の通信を標準化するためのアーキテクチャスタイル。Rest-compliant または RESTful システムは簡単に相互通信できます。

GET メソッドを使用するエンドポイントをヒットすると、RESTful システムから Analytics にデータを読み込むことができます。データをリソースに送信するために設計された POST や PUT 等の他のタイプの HTTP 接続はサポートされていません。

注意

Analytics 15 では、REST コネクタが強化され、その他の認証方法がサポートされるようになりました。また、対応する接続フィールドも更新されます。ただし、アップグレードすると、前のバージョンのスクリプトは動作しません。Analytics 15 バージョンにアップグレードする場合は、コネクタを再構成し、RESTful システムに接続する必要があります。ロボットまたは Analytics Exchange エンジンで前のバージョンの REST コネクタ スクリプトを実行している場合は、Analytics 15 にアップグレードした後に、再構成したスクリプトをアップロードする必要があります。

作業を開始する前に

Analytics を RESTful システムに接続するには、次の項目が必要です。

- 接続する RESTful システムの URI エンドポイント。
- そのシステムの接続資格情報(必要な場合)。場合によっては、ユーザー名とパスワードです。一部のシステムは OAuth などのより実質的な資格情報を使用します。必要な資格情報がわからない場合は、御社の REST サービスの管理者にお問い合わせください。管理者から支援を得られない場合は、接続するシステムのサポート サービスにお問い合わせください。

REST 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの [新規の接続] タブから、REST を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、[接続のフィルタリング] ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. [データ接続の設定] パネルが開いたら、接続設定を入力し、パネルの下部にある [保存して接続] をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

RESTの接続は、**既存の接続**タブに保存されます。今後、RESTに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。RESTからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
URI	RESTful リソースの Uniform Resource Identifier(URI) または絶対ファイルパス	https://jsonplaceholder.typicode.com/users/
書式	データがXML または JSON 形式かどうかを指定します。	JSON
承認タイプ	<p>認証で使用されるスキーム。使用可能な認証タイプは次のとおりです。</p> <ul style="list-style-type: none"> ○ 認証なし - ドライバーは認証を使用しません。 ○ API キー - ドライバーは API キーを使用して認証します。 ○ ベアラートークン - ドライバーはベアラートークンを使用して認証します。 ○ 基本認証 - ドライバーはユーザー名とパスワードによる基本認証を使用します。 ○ OAuth 1.0 - ドライバーは OAuth 1.0 を使用して認証します。 ○ OAuth 2.0 - ドライバーは OAuth 2.0 を使用して認証します。 	認証なし
キー	<p>認証する API を指定します。</p> <p>承認タイプ フィールドで API キー を選択すると、このフィールドが有効になります。</p>	
値	このフィールドの値は、サーバーで認証するために入力 API キーとともに使用されま	

設定	説明	例
	す。 承認タイプ フィールドで API キー を選択すると、このフィールドが有効になります。	
追加先	要求ヘッダーまたはクエリパラメーターで API キーと値のペアを追加するかどうかを指定します。	ヘッダー
トークン	サーバーで認証するためのベアラートークンを指定します。トークンは、要求ヘッダーに含まれる JSON Web Token (JWT) などのアクセスキーにすることができます。 承認タイプ フィールドで ベアラートークン を選択すると、このフィールドが有効になります。	
ユーザー	REST データソースに接続するために使用されるユーザー名。 承認タイプ フィールドで 基本認証 を選択すると、このフィールドが有効になります。	kgibbons
パスワード	REST データソースに接続するために使用されるユーザー名のパスワード。 承認タイプ フィールドで 基本認証 を選択すると、このフィールドが有効になります。	
OAuth 承認 URL	OAuth サービスの承認 URL。	https://login.example.com/services/oauth2/authorize
OAuth アクセストークン	OAuth アクセストークンを取得する URL。OAuth 1.0 では、この URL で承認された要求トークンがアクセストークンと交換されます。	https://login.example.com/services/oauth2/access
OAuth 要求トークン URL	要求トークンを取得する URL。	https://login.example.com/services/oauth2/token
OAuth クライアント Id	クライアント ID は、OAuth 承認サーバーに REST データソースを登録するときに割り当	ZYDPLLBWSK3MVQJSIYHB1OR2JXCY0X2C5UJ2QAR2MAAIT5Q

データの定義とインポート

設定	説明	例
	てられます。	
OAuth クライアントシークレット	クライアントシークレットは、OAuth 承認サーバーに REST データソースを登録するときに割り当てられます。	
OAuth 更新トークン URL	OAuth トークンを更新する URL。OAuth 2.0 では、古いアクセストークンの有効期限が切れたときに、この URL で、更新トークンが新しいアクセストークンと交換されます。	https://login.example.com/services/oauth2/refresh
OAuth 付与タイプ	OAuth フローの付与タイプ。使用可能なオプションは次のとおりです。 <ul style="list-style-type: none">◦ CODE◦ CLIENT◦ PASSWORD	CODE
URL コールバック	認証時に戻る OAuth コールバック URL。この値は、アプリ設定で指定するコールバック URL と一致する必要があります。	https://www.example.com/api/billing/123

詳細設定

設定	説明	例
カスタム ヘッダー	<p>このプロパティは、Content-Type や From など、他のプロパティから作成された HTTP 要求ヘッダーに追加される HTTP ヘッダーの文字列に設定することができます。</p> <p>ヘッダーの形式は、HTTP 仕様の規定に従い、"header: value" にする必要があります。各ヘッダーは独自の行に入ります。</p> <p>このオプションは慎重に使用してください。無効なヘッダーが含まれる場合、HTTP 要求が失敗することがあります。このプロパティは、専用または非標準 API と統合するために、微調整する際に有用です。</p>	<p>Content-Type: text/html; charset=utf-8</p> <p>Connection: keep-alive</p>

設定	説明	例
カスタム URL パラメーター	要求に含まれるカスタム クエリ文字列。パラメーターはクエリ文字列としてエンコードされる必要があります。クエリ文字列の値は URL でエンコードされる必要があります。	field1=value1&field2=value2&field3=value3
アクセス鍵	AWS アカウント アクセス鍵。この値には、AWS セキュリティ資格情報ページからアクセスできます。	AKIAIOSFODNN7EXAMPLE
秘密鍵	AWS アカウント シークレット キー。この値には、AWS セキュリティ資格情報ページからアクセスできます。	wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
地域	Amazon Web Services のホスト地域。	NORTHERNCALIFORNIA
その他	<p>特定の使用例でのみ必要な非表示のプロパティ。通常、何も入力する必要がありません。セミコロン区切りで複数のプロパティを指定します。</p> <p>キャッシュ構成</p> <ul style="list-style-type: none"> ◦ CachePartial=True - 列のサブセットのみをキャッシュに保存します。これはクエリで指定できます。 ◦ QueryPassthrough=True - 指定されたクエリを、ドライバーの SQL パーサーを使用する代わりに、キャッシュデータベースに渡します。 <p>統合と形式</p> <ul style="list-style-type: none"> ◦ SupportAccessLinkedMode - Access のリンクモードでは、ほとんどのデータソースが複数 ID クエリをサポートしないため、一般的に、常にキャッシュを使用することをお勧めします。ただし、Access でドライバーを使用せず、リンクモードで使用する場合、このプロパティを False に設定し、特定のテーブルに対する SELECT * クエリのキャッシュの使用を回避する必要があります。 ◦ DefaultColumnSize - データソースがメタデータで列長を提供しないときに、文字列フィールドのデフォルト長を設定します。デフォルト値は 2000 です。 ◦ ConvertDateTimeToGMT - コンピューターのローカル時刻の代わりに、日付時刻値を GMT (UTC) 	CachePartial=True; QueryPassthrough=True

設定	説明	例
	<p>に変換するかどうか。</p> <ul style="list-style-type: none"> ◦ RecordToFile=filename - 基本のソケット データ転送を指定されたファイルに記録します。 <p>OAuth プロパティ</p> <ul style="list-style-type: none"> ◦ InitiateOAuth - このプロパティを設定すると、接続するときに、プロセスを開始して、OAuth アクセストークンを取得または更新します。次のオプションを使用できます。 <ul style="list-style-type: none"> • OFF - OAuth フローが完全に自分によって処理されることを示します。 OAuthAccessToken は認証が必要です。 • GETANDREFRESH - OAuth フロー全体がドライバーによって処理されることを示します。現在、トークンが存在しない場合、ブラウザー経由で確認することで取得します。トークンが存在する場合は、必要に応じて更新されます。 • REFRESH - ドライバーが OAuthAccessToken の更新のみを処理することを示します。ブラウザー経由で認証するようにドライバーによって確認されることはありません。最初に OAuthAccessToken と OAuthRefreshToken の取得を処理する必要があります。 ◦ OAuthSettingsLocation - InitiateOAuth が true に設定されているときに、OAuth 値が保存される設定ファイルの場所。InitiateOAuth が有効な場合、ドライバーは OAuth 値を設定ファイルに保存し、OAuth 接続プロパティを手動で入力するようにユーザーに要求することを回避します。OAuth セッションが期限切れになると、InitiateOAuth が設定されていれば、ドライバーは自動的に新しいアクセストークンを取得します。InitiateOAuth が有効で、OAuthSettingsLocation が定義されていない場合、ドライバーはデフォルト設定ファイルの 	

設定	説明	例
	<p>%AppData%\CData\REST Data Provider\OAuthSettings.txt (Windows)を使用します。macOS では、このファイルは ~/Library/Application Support/CData/REST Data Provider/OAuthSettings.txt にあります。Linux では、~/cdata/.config です。</p> <ul style="list-style-type: none"> ○ OAuthAccessToken - OAuthAccessToken は、認証処理の一部として OAuth サーバーから取得します。アクセストークンには、サーバー依存のタイムアウトがあり、要求間で再利用できます。アクセストークンは、ユーザー名とパスワードの代わりに使用されます。アクセストークンは、サーバーで課すことによって、資格情報を保護します。 ○ OAuthAccessTokenSecret - OAuthAccessTokenSecret は、認証処理の一部として、OAuth サーバーから取得されます。OAuthAccessToken を使用し、タイムアウトするまで、複数の要求を使用できます。 ○ OAuthRefreshToken - OAuthRefreshToken プロパティは、OAuth 認証を使用するときに、OAuthAccessToken を更新するために使用されます。 	
データモデル	<p>JSON または XML ドキュメントを解析し、メタデータを生成するときに使用するデータモデルを指定します。</p> <ul style="list-style-type: none"> ○ ドキュメント - ドキュメントごとの行を表す単一テーブルが返されます。このデータモデルでは、ネストされたドキュメント(オブジェクト配列)は平坦化されず、集計として返されます。XPath 値が明示的に指定されていない場合、ドライバーは検出された最上位のドキュメント(オブジェクト配列)を XPath として特定し、使用します。 ○ Flattened Documents - JSON または XML ファイルの使用可能なドキュメントの JOIN を表す単一テーブルが返されます。このデータモデルでは、ネストされた XPath 	ドキュメント

データの定義とインポート

設定	説明	例
	<p>値は SQL JOIN と同じ方法で機能します。また、ネストされた兄弟 XPath 値(同じ高さの子パス)は、SQL CROSS JOIN として処理されます。明示的に指定されない場合、ドライバーは、JSON または XML ファイルを解析し、使用可能なドキュメント(ネストされたドキュメントを含む)を特定して、使用可能な XPath 値を特定します。</p> <ul style="list-style-type: none"> ◦ 関係 - 複数のテーブルが返されます。各 XPath の 1 つが指定されます。このデータモデルでは、ネストされたドキュメント(オブジェクト配列)が、関係テーブルおよび主キー、外部キーとして返されます。明示的に指定されない場合、ドライバーは、JSON または XML ファイルを解析し、使用可能なドキュメント(ネストされたドキュメントを含む)を特定して、使用可能な XPath 値を特定します。 	
データソース	このプロパティは、REST リソースの場所の URI を指定します。	s3://remotePath/file.json
平坦化された配列	<p>デフォルトでは、配列でネストされたオブジェクトは、JSON または XML の文字列として返されます。</p> <p>FlattenArrays プロパティを使用して、ネストされた配列の要素を固有の列に平坦化できます。FlattenArrays を、ネストされた配列から返す要素数に設定します。</p> <p>FlattenArrays を -1 に設定すると、ネストされた配列のすべての要素を平坦化します。</p>	1
平坦化されたオブジェクト	FlattenObjects を true に設定すると、オブジェクト プロパティを固有の列に平坦化します。そうでない場合、配列でネストされたオブジェクトは、JSON または XML の文字列として返されます。	true
JSON 形式	JSON ドキュメントの形式を指定します。	JSON
生成されたスキーマファイル	<p>解析されたドキュメントからスキーマファイル(RSD)を生成するかどうかを指定します。</p> <ul style="list-style-type: none"> ◦ Never - スキーマファイル(RSD) 	Never

設定	説明	例
	<p>は生成されません。</p> <ul style="list-style-type: none"> ○ OnUse - テーブルのスキーマファイル (RSD) が存在しない場合、初めてテーブルが参照される時に、スキーマファイル (RSD) が生成されます。 ○ OnStart - 現在スキーマファイル (RSD) がないテーブルに対して、接続時に、スキーマファイル (RSD) が生成されます。 <p>このプロパティは、Format、URI、XPath、および Location とともに使用されます。</p>	
XPath	<p>XML/JSONドキュメント内の同じ高さで繰り返す要素の Path (JSONドキュメントを複数の行に分割するのに使用されます)。</p> <p>複数のパスは、セミコロン区切りのリストで指定できます。DataModel 設定では、テーブルを作成し、データを表示するために XPath 値がどのように使用されるのかを構成できます。</p>	\$.store.book[0].title
Kerberos Keytab ファイル	Kerberos プリンシパルと暗号化された鍵の組み合わせを含む Keytab ファイル。	/path_to_keytab_file/filename.keytab
Kerberos SPN	Kerberos ドメインコントローラーでサービスプリンシパル名 (SPN) が認証している URL と同じでない場合、このプロパティを使用して、SPN を設定します。	HTTP/TimeOffWebPortal
行スキャン深さ	<p>テーブルの列を動的に決定するときにスキャンする行数。スキーマ (RSD) ファイルがテーブルで使用できないとき (GenerateSchemaFiles を使用する など) には、列が動的に決定されます。値が大きいと、要求が長くなりますが、より正確になります。</p> <p>この値を 0 (ゼロ) に設定すると、文書全体が解析されます。</p>	100
SSL サーバー証明書	<p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p> <p>指定されていない場合は、コンピューター</p>	<ul style="list-style-type: none"> ○ -----BEGIN CERTIFICATE----- MIICHCCAe4CAQAwDQYJKoZIhvc.....Qw== ----- END CERTIFICATE----- ○ C:\cert.cer ○ -----BEGIN RSA PUBLIC KEY----- MIGfMA0GCSq.....AQAB -----END RSA PUBLIC KEY----- ○ ecadbda5a1529c58a1e9e09828d70e4

データの定義とインポート

設定	説明	例
	<p>ターが信頼するすべての証明書が許可されます。</p> <p>**'を使用すると、すべての証明書を許可することを示します(セキュリティの問題のため推奨されません)。</p>	<ul style="list-style-type: none"> 34a929226ae0819f2ec14b4a3d904f801cbb150d
鍵サイズの上限	<p>主キー列の最大長。</p> <p>一部のODBCツールでは、主キー列の長さが特定の値を超えることができません。このプロパティでは、ODBCドライバーが主キー列のすべての報告された列を無効にします。Microsoft Access Linked データソースとしてODBCドライバーを使用するときには、特に注意してください。</p> <p>LimitKeySize を 0 に接続すると、キー長が元の長さに戻ります。</p>	255
Long Varchar にマッピング	<p>このプロパティは、列が SQL_LONGVARCHAR として返されるかどうかを制御します。</p> <p>一部のアプリケーションでは、特定の文字数よりも大きいすべてのテキストデータを SQL_LONGVARCHAR として報告する必要があります。これを使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。</p>	-1
WVarchar にマッピング	<p>このプロパティは、文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。</p> <p>アスタリスク(*)文字を使用すると、すべてのテーブルとすべての列を含めることができます。例:*=*</p>	Table1=Column1, Table1=Column2, Table2=Column3
大文字の ID	<p>すべての ID を大文字で報告します。これは Oracle データベースのデフォルトであるため、Oracle Database</p>	false

設定	説明	例
	Gateway などの Oracle ツールとの統合に適しています。	
プロキシ認証スキーム	<p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <p>ドライバーはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NONE ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ NTLM ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシユーザー	ProxyServer プロキシで認証するために使用されるユーザー名。	jgibbons
プロキシのパスワード	ProxyServer プロキシで認証するために使用されるパスワード。	UsaPhone897Batteries!Tokyo
プロキシサーバー	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。	192.168.1.100
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80

設定	説明	例
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ。</p> <ul style="list-style-type: none">◦ AUTO -デフォルト設定。URL が HTTPS URL の場合、TUNNEL オプションを使用します。URL が HTTP URL の場合、コンポーネントは NEVER オプションを使用します。◦ ALWAYS -接続は常に SSL 有効です。◦ NEVER -接続は SSL 有効ではありません。◦ TUNNEL -接続はトンネルプロキシ経由です。プロキシサーバーはリモートホストへの接続を開き、トラフィックはプロキシ経由で前後します。	AUTO
プロキシ例外	<p>プロキシサーバー経由での接続から除外されるホストまたは IP のセミコロン区切りのリスト。</p> <p>デフォルトでシステムプロキシ設定を使用します。追加の構成は必要ありません。この接続のプロキシ例外を明示的に構成する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort を構成する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p>	172.16.254.1;192.0.2.1

Rsam に接続する

Rsam では、結果を保存し、必要なときに取得できる、保存された検索を作成することができます。これらの保存された検索を Analytics をインポートできます。このデータを Analytics に取り込むと、それを準備、分析し、リザルトに移動して、データに関するストーリーボードと視覚化を作成および共有することができます。

インポートする前に

データを Analytics に取り込む前に、次の要件および考慮事項に注意してください。

まず Rsam を統合する

ユーザーが Rsam コネクタを使用できるようにするには、Rsam 管理者が Rsam 統合手順を実行しておく必要があります。詳細については、[Rsam と Diligent HighBond および ACL Robotics との統合](#)を参照してください。

Rsam バージョン要件

御社が使用する Rsam Cloud のバージョンは 9.2.2210 以降でなければなりません。前のバージョンまたはオンプレミス エディションの Rsam の場合、Analytics は Rsam 内のデータにアクセスできません。

アクセス許可

ユーザー名とパスワードで認証するか、API キーで認証するかどうかに関係なく、Analytics で使用できる保存された検索は、Rsam で使用できる保存された検索と同じです。アクセスは Rsam の権限とロールに基づきます。インポートするために必要なデータにアクセスできない場合は、アクセス対象を調整する権限を唯一持っている御社の Rsam 管理者にお問い合わせください。

Rsam で保存された検索を設定する

続行する前に、Analytics に取り込む保存された検索が正しく設定されていることを確認します。この手順のヘルプについては、Rsam のエンドユーザー ヘルプおよび管理者ヘルプを参照してください。

- 必要な保存された検索を作成するか、アクセスできることを確認します。Analytics または HighBond に取り込まれている保存された検索は、その目的でのみ作成されている必要があります。既存の保存された検索を使用する場合は、コピーを作成し、そのコピーを使用します。これにより、別のユーザーが検索を修正したり、プラットフォーム間でデータ転送を自動化するために作成したスクリプトを誤って破損させたりできないことが保証されます。

- Analyticsに取り込む各検索で、[HighBondプラットフォーム]オプションが選択されていることを確認します。
- インポートが「平坦」になるように、保存された検索のグループ化を削除します。レコードをグループ化する場合は、データをAnalyticsまたはHighBondに取り込んだ後に視覚化およびストーリーボードを作成するときに、そのようにします。
- インポートする前に、Rsamデータを完璧にしておく必要はありません。Analyticsでは、インポートするデータをフィルター、クリーンアップ、結合、標準化できます。ただし、一般的に、使用可能な最善のデータを使用し、変換を最小化することをお勧めします。少し時間をかけ、Rsamの保存された検索が必要なすべてのデータを生成し、分析に関係のないデータを省略することを保証できます。このようにすると、データ準備プロセスがより簡単になります。

Analytics が列名を決定する方法

Rsamのオブジェクトとレコードには、表示名と管理名があります。一般的には、Rsamユーザーには、そのプラットフォーム全体で表示名が表示されます。管理名はRsam管理者のみがRsam環境を保守するために使用する非表示のフィールドです。

デフォルトでは、Analyticsはフィールド名としてRsamの表示名を取得します。これはほとんどのユーザーが表示するために使用する項目であるためです。ただし、Analyticsには厳密なフィールド名要件がありません。スペースと特殊文字はAnalyticsではアンダースコアに変換されます。つまり、Analyticsの列名は、Rsamの保存された検索の列名とは正確に一致しない可能性があります。一般的に、これは問題にはなりません、Analyticsでの表示内容が変わる場合があります。

- 異なる物理フィールド名を使用する場合は、AnalyticsがRsamで作成するSQLクエリを変更し、独自のフィールド名を指定できます。ただし、Analyticsフィールド名要件は適用されます。将来、更新されたデータをインポートする場合は、このクエリを追跡する必要があります。
- デフォルトのフィールド名を使用し、実際にAnalyticsで表示される列ヘッダーを変更する場合は、Analyticsの実際のフィールド名には影響せずに、[列表示名](#)をわかりやすい名前に変更できます。

Rsam 接続の作成

1. Analyticsのメインメニューから、[インポート > データベースとアプリケーション]を選択します。
2. ACLコネクタセクションの **新規の接続** タブから、Rsamを選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Rsamの接続は、**既存の接続** タブに保存されます。今後、Rsamに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Rsamからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	接続している Rsam インスタンス。	https://training02.rsam.com/
API アプリ名	Rsam API アプリ名。デフォルトでは、これは「rsam_ap」ですが、Rsam で変更できます。	rsam_api
認証方法	Rsam に接続する際に、ユーザー名とパスワードを使用しているか、API キーを使用しているかどうか。	パスワード
ユーザー	ユーザー名とパスワードを使用して Rsam で認証している場合は、ここでユーザー名を入力します。	jgibbons
パスワード	ユーザー名とパスワードを使用して Rsam で認証している場合は、ここでパスワードを入力します。	UsaPhone897Batteries!Tokyo
API キー	API キーを使用して Rsam で認証している場合は、ここに入力します。API キーは各ユーザーで固有ですが、まだない場合は、Rsam 管理者が Rsam で生成する必要があります。	99141fae-4c41-4abd-ade2-469f7d0151a4

詳細設定

設定	説明	例
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。	True

次の手順

Analytics に Rsam データがあると、そのデータを準備して、分析するためのさまざまなオプションがあります。

- **準備** - Analytics のツールを使用して、データをクリーンアップおよび標準化して、他のソース内のデータと結合し、分析用に準備します。お手持ちのデータから、より包括的なストーリーを引き出す方法について考えましょう。
 - 御社の IT 部門が置かれているリスク状況についてさらに学習しますか?
 - 組織の ERP システムには、組織の Rsam 業者管理プログラムで管理されていない業者がいますか?
 - 組織の HR データベースには、Rsam のポリシー管理キャンペーンで管理されていない従業員がいますか?
- 以上のような質問に答えるには、お手持ちのデータを収集して準備してください。詳細については、"分析用のデータの準備" ページ 865を参照してください。
- **分析** - Analytics コマンドおよび関数を使用して、データを分析します。詳細については、"データの分析" ページ 1151を参照してください。
- **エクスポート** - HighBond のリザルト アプリを含む他の形式に、データをエクスポートできます。詳細については、"データのエクスポート" ページ 210と "HighBond のリザルト アプリへの例外のエクスポート" ページ 216を参照してください。
- **自動化** - スクリプトを作成し、HighBond の [ロボット アプリ](#)を使用して、データのインポート、集計、リザルト へのエクスポートなどの繰り返しタスクを自動化できます。時間と共に Rsam(およびその他) のデータが変化しても、HighBond のストーリーボードにもそれらの変化が自動的に反映されるようになっています。
- **視覚化** - データをリザルトにエクスポートしたら、リザルトを使ってわかりやすい [視覚化](#)を作成します。視覚化は、[ストーリーボード](#)に変換して、社内の他のユーザーと簡単に共有することができます。

RSS/ATOM への接続

Really Simple Syndication(RSS) および Atom は、Web サイト上でニュースや記事を公開するために使用される XML ベースの書式/フィード言語です。RSS/ATOM データコネクタを使用して、RSS/ATOM データをインポートできます。

作業を開始する前に

RSS/ATOM に接続するには、以下を収集する必要があります。

- フィードタイプが RSS または Atom かどうか
- フィードの URI

接続の前提条件の収集については、組織の RSS/ATOM 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から RSS/ATOM のサポートにお問い合わせください。

RSS/ATOM 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、RSS/ATOM を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

RSS/ATOM の接続は、**既存の接続** タブに保存されます。今後、RSS/ATOM に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。RSS/ATOM からデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
種類	フィードの種類を指定します。使用可能なオプションは次のとおりです。 <ul style="list-style-type: none"> ◦ RSS ◦ ATOM 	RSS
URI	フィード リソースの場所の Uniform Resource Identifier (URI)。 これは HTTP ソースまたはファイルにすることができます。	

詳細設定

設定	説明	例
認証スキーム	HTTP 認証で使用されるスキームを指定します。使用可能なオプションは次のとおりです。 <ul style="list-style-type: none"> ◦ NTLM - Windows 資格情報で認証します。 ◦ BASIC HTTP 基本認証を使用します。 ◦ DIGEST HTTP DIGEST 認証を使用します。 ◦ NONE 匿名認証を使用します。たとえば、公開サイトへのアクセスの目的です。 ◦ KERBEROSDELEGATION Kerberos プロトコル経由で委任を使用します。なりすますアカウントのユーザーとパスワードを設定します。 	NONE
認証トークン	HTTP 要求の有効期間を特定するために認証で使用されるトークン。 接続で設定された認証トークンは、要求変数「@authtoken」として HTTP サーバーで投稿されます。	
ユーザー	接続を認証するためのユーザー名。	
パスワード	接続を認証するためのユーザー名のパスワード。	

設定	説明	例
	スワード。	
鍵サイズの上限	主キー列の最大長。	255
Long Varchar にマッピング	このプロパティは、列が SQL_ LONGVARCHAR として返されるかどうかを制御します。	-1
WVarchar にマッピング	このプロパティは、文字列タイプを SQL_ VARCHAR ではなく、SQL_ WVARCHAR にマッピングするかどうかを制御します。	
擬似列	この値は、擬似列を列としてテーブルに含めるかどうかを指定します。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。	
SSL サーバー証明書	SSL を使用して接続するときにサーバーから許可される必要がある証明書。 次のいずれかを使用できます。 <ul style="list-style-type: none"> ◦ 完全な PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 または SHA1 サムプリント(スペースまたはコロンで 16 進数値を区切ることができます) コンピューターから信頼されない他の認定書はすべて拒否されます。	
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。このオプションが無効な場合、日付時刻値は、Analytics を実行するシステムのオペレーティングシステムタイムゾーンに変換されます。	
プロキシ認証スキーム	プロキシサーバーで認証するために使用する認証スキームを指定します。使用可能なオプションは次のとおりです。 <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ NONE - ドライバーは接続を認証しません。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に設定された 	BASIC

設定	説明	例
	<p>プロトコルに基づき、NTLM または Kerberos トークンを取得します。</p> <ul style="list-style-type: none"> ◦ PROPRIETARY - NTLM または Kerberos トークンを生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。このオプションに設定された値は他のプロキシ設定よりも優先されます。このため、カスタムプロキシ設定を使用する場合は、このオプションを無効にしてください。	
プロキシユーザー	プロキシサーバーで認証するためのユーザー名。	
プロキシのパスワード	プロキシサーバーでプロキシユーザーを認証するためのパスワード。	
プロキシサーバー	HTTPトラフィックをルーティングするプロキシのホスト名またはIPアドレス。	
プロキシのポート	プロキシサーバーが実行されているTCPポート。	
プロキシSSLタイプ	<p>ProxyServerプロキシに接続するときに使用するSSLタイプ。使用可能なオプションは次のとおりです。</p> <ul style="list-style-type: none"> ◦ AUTO - URLがHTTPS URLの場合、TUNNELオプションを使用します。URLがHTTP URLの場合、コンポーネントはNEVERオプションを使用します。 ◦ ALWAYS - 接続は常にSSLが有効です。 ◦ NEVER - 接続はSSLが有効ではありません。 ◦ TUNNEL - 接続はトンネルプロキシ経由です。 <p>このオプションは、プロキシサーバーの値を入力するときのみ有効です。</p>	AUTO

Sage 50 UK への接続

Sage 50 UK は会計管理ソフトウェアであり、財務データの処理、外貨での取引、請求書および顧客の管理で使用されます。Sage 50 UK データコネクタを使用して、組織の Sage 50 UK データをインポートできます。

メモ

Sage 50 UK データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Sage 50 UK に接続するには、次の情報を収集する必要があります。

- Sage 50 UK SData サービスの URL
- Sage 50 UK アカウントに接続するための資格情報
- 使用される認証スキーム

接続の前提条件の収集については、組織の Sage 50 UK 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Sage 50 UK のサポートにお問い合わせください。

Sage 50 UK 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Sage 50 UK を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
Sage 50 UKへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Sage 50 UKの接続は、**[既存の接続]** タブに保存されます。今後、Sage 50 UKに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Sage 50 UKからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

Sage 50 UK データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、Sage 50 UK に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Sage Cloud Accounting への接続

Sage Cloud Accounting はクラウドベースの会計ソフトウェアであり、支払い、請求書、給与、および税金を管理します。Sage Cloud Accounting データコネクタを使用して、組織の Sage Cloud Accounting データをインポートできます。

メモ

Sage Cloud Accounting データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

Sage Cloud Accounting に接続するには、Sage Cloud Accounting サーバーの接続詳細情報を収集する必要があります。

接続の前提条件の収集については、組織の Sage Cloud Accounting 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Sage Cloud Accounting のサポートにお問い合わせください。

Sage Cloud Accounting 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション]を選択します。
2. ACL DSN コネクタ(バンドル)セクションの **新規の接続** タブから、Sage Cloud Accounting を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Sage Cloud Accountingへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Sage Cloud Accountingの接続は、 **[既存の接続]** タブに保存されます。今後、Sage Cloud Accountingに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Sage Cloud Accountingからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Sage Cloud Accounting データ コネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Sage Cloud Accounting に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Sage Intacct への接続

Sage Intacct は中小企業向けのクラウドベースの財務管理および会計ソフトウェアです。Sage Intacct データコネクタを使用して、組織の Sage Intacct データをインポートできます。

メモ

Sage Intacct データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Sage Intacct に接続するには、次の情報を収集する必要があります。

- Sage Intacct で認証するユーザーの会社 ID
- Sage Intacct に接続するための URL
- Sage Intacct アカウントに接続するための資格情報
- Sage Intacct に割り当てられた Web Services SenderID とパスワード

接続の前提条件の収集については、組織の Sage Intacct 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Sage Intacct のサポートにお問い合わせください。

Sage Intacct 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Sage Intacct を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Sage Intacctへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Sage Intacctの接続は、 **[既存の接続]** タブに保存されます。今後、Sage Intacctに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Sage Intacctからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Sage Intacct データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Sage Intacct に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Salesforce への接続

Salesforce.com はクラウドのカスタマーリレーションシップマネジメント (CRM) プラットフォームです。Salesforce データコネクタを使用して、Salesforce データをインポートできます。

メモ

Analytics は任意のコネクタとして Salesforce を提供します。データアクセスウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、「任意の Analytics データコネクタと Python エンジンを実装する」ページ 2663 を参照してください。

作業を開始する前に

Salesforce の資格情報

Salesforce に接続するには、以下を収集する必要があります。

- Salesforce のユーザー名
- Salesforce のパスワード
- セキュリティトークン

メモ

セキュリティトークンを必要とする接続もあれば、必要としない接続もあります。セキュリティトークンは、それがないと接続が失敗する場合に限って必要です。

接続の前提条件の収集については、組織の Salesforce 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Salesforce のサポートにお問い合わせください。

Salesforce の API およびエディション

このコネクタでは、Salesforce の API を使ってデータにアクセスするため、あなたまたは社内の Salesforce 管理者は、あなたが接続を行う前に、Salesforce を使って会社およびユーザーアカウントが API にアクセスできるようにしておく必要があります。また、あなたは以下の Salesforce エディションのいずれかを持っている必要があります。

- Developer Edition
- Professional Edition
- Enterprise Edition
- Unlimited Edition

メモ

あなたが Salesforce の既存のユーザーで、アカウントをこれらのエディションのいずれかにアップグレードしたい場合は、Salesforce の顧客担当者にお問い合わせください。

Salesforce 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Salesforce を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ポップアップに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Salesforce の接続は、**既存の接続** タブに保存されます。今後、Salesforce に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Salesforce からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	サーバーで認証する Salesforce アカウントのユーザー名。	
パスワード	Salesforce サーバーで認証するために使用される Salesforce アカウントのパスワード。	
セキュリティトークン	Salesforce アカウントへのアクセスを認証するセキュリティトークン。 セキュリティトークンを取得するには、プロファイル情報に移動し、セキュリティトークンをリセットします。パスワードがリセットされた場合は、セキュリティトークンもリセットする必要があります。	

設定	説明	例
プロキシ サーバー	HTTPトラフィックをルーティングするプロキシのホスト名またはIPアドレス。 ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。	
プロキシのポート	HTTPトラフィックをリダイレクトする、HTTP プロキシが実行されているポート。	80
プロキシ ユーザー	ProxyServer プロキシで認証するために使用されるユーザー名。	
プロキシのパスワード	ProxyServer プロキシでプロキシ ユーザーを認証するためのパスワード。	

詳細設定

設定	説明	例
サンドボックスの URL への接続を有効化	Salesforce サンドボックス アカウントに接続するかどうかを指定します。	false
サンドボックスの URL		
SSL サーバー証明書	TLS/SSL を使用して接続するときにサーバーから許可される証明書。 次のいずれかを使用できます。 <ul style="list-style-type: none"> 完全な PEM 証明書 証明書を含むローカルファイルへのパス 公開鍵 MD5 または SHA1 サンプリント(スペースまたはコロンで 16 進数値を区切ることができます) <p>コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	
WVarchar にマッピング	文字列型を SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを指定します。デフォルトで設定されます。 文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に	true

データの定義とインポート

設定	説明	例
	設定すると、SQL_VARCHARを使用できます。	

SAP への接続

SAP はエンタープライズ ビジネス スイートであり、広範囲のビジネス プロセスを管理します。組織の SAP データをインポートするには、ACL コネクターを使用します。

メモ

SAP コネクター、および必要に応じて SNC(Secure Network Communications) および SSO(シングルサインオン) を設定するには、適切なレベルの技術的な知識を有する作業員が必要です。詳細については、"ACLコネクター(SAP) の設定" ページ 672を参照してください。

SAP コネクターを使用するには、基本的な Analytics サブスクリプションの他に、追加のサブスクリプション資格が必要です。必要なサブスクリプション資格がない場合は、SAP に接続できません。

SAP コネクター サブスクリプションについては、担当者に問い合わせてください。

互換性がある SAP システム

ACL コネクター(SAP) は次の SAP システムで使用できます。

- S/4HANA
- SAP HANA を含むすべてのサポートされているデータベース プラットフォームで実行されている次のシステムのすべての強化レベル
 - SAP ERP 6.0 (ECC 6.0)
 - SAP CRM 7.0
 - SAP SRM 7.0
 - SAP SCM 7.0
 - SAP EWM 7.0

SAP 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクター セクションの **新規の接続**] タブから、SAP を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

3. **データ接続の設定**] パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接**

続]をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

SAPの接続は、**既存の接続**]タブに保存されます。今後、SAPに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SAPからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続設定

メモ

SAPコネクターの必要な設定については、組織のSAP Basis 管理者にお問い合わせください。

SAP システムと Secure Network Communications (SNC)

組織のSAPシステムでは、Secure Network Communications (SNC) が有効な場合があります。これにより、SAP の異なるコンポーネント間の暗号化された通信と、シングルサインオンを使用したユーザー認証が可能になります。

SNC はSAPシステムのソフトウェアレイヤーであり、外部セキュリティ製品と統合することで、基本SAPセキュリティを拡張することができます。

シングルサインオンの有無に関係なく、SNCを使用している場合は、ACL Connector for SAPで追加の設定が必要です。

基本設定

設定	説明	例
あらかじめ読み込まれたSAPシステム 省略可能	次のSAP GUI 構成ファイルのいずれかで指定された接続設定のSAPシステムの名前: <ul style="list-style-type: none"> ◦ <code>SAPUILandscape.xml</code> ◦ <code>saplogon.ini</code> あらかじめ読み込まれたSAPシステムを選択する場合、SAP GUI 構成ファイルの値により、さまざまな他のSAP接続設定が自動的に入力されます。 SAP GUI 構成ファイルのデフォルトの場所: <code>C:\Users\ username >\AppData\Roaming\SAP\Common</code>	PHR - Production Human Resources

設定	説明	例
	両方の構成ファイルが <code>.\SAP\Common</code> フォルダーにある場合、 <code>SAPUILandscape.xml</code> が優先されます。	
サーバー	SAP システムのホスト名または IP アドレス。	<ul style="list-style-type: none"> ◦ phr-1.example.com ◦ 52.202.133.148
クライアント	SAP システム内のクライアントを特定する 3 桁のコード。 クライアントは、SAP システムのサブパーティションです。	800
言語	SAP システムの言語。	EN
サーバータイプ	SAP システムでのサーバーの構成： <ul style="list-style-type: none"> ◦ NORMAL - 単一のサーバー ◦ LOAD BALANCE - サーバーのロードバランスグループ 	<ul style="list-style-type: none"> ◦ NORMAL ◦ LOAD BALANCE
インスタンス番号 [サーバータイプ = NORMAL]	SAP システムの 2 桁のインスタンス番号。 インスタンス番号は、SAP コネクターと SAP システム間の送受信通信で使用されるポート番号の最後の 2 桁です。 詳細については、「SAP ポート構成」ページ 675 を参照してください。	01
ログオングループ [サーバータイプ = LOAD BALANCE]	SAP システムのサーバーのロードバランスグループの名前。	ACL Production
システム Id [サーバータイプ = LOAD BALANCE]	SAP システムの 3 文字の ID。	PHR
認証方法	SAP システムで使用されるユーザー認証の方法： <ul style="list-style-type: none"> ◦ ユーザー名とパスワード <ul style="list-style-type: none"> • 標準 SAP 認証方法 • 安全なネットワーク通信 (SNC) は無効です ◦ シングルサインオン <ul style="list-style-type: none"> • SNC とシングルサインオン ◦ SNC(シングルサインオンなし) <ul style="list-style-type: none"> • SNC(シングルサインオンなし) 	
認証方法 = ユーザー名とパスワード		

データの定義とインポート

設定	説明	例
ユーザー	ユーザーの SAP ユーザー ID。	
パスワード	ユーザーの SAP パスワード。	
認証方法 = SNC(シングルサインオンなし)		
SNC ユーザー名	ユーザーの SAP SNC ユーザー ID。	
SNC パスワード	ユーザーの SAP SNC パスワード。	
認証方法 = シングルサインオン または 認証方法 = SNC(シングルサインオンなし)		
パートナー名	通信パートナー(ターゲット名)の SNC 名。	<p>"p:CN=sap01.host1, OU=Administration, O=myCompany, C=US"</p> <p>p:CN=SAPService@myCompany.com</p> <p>メモ スペースがある場合は、SNC 名を引用符で囲みます。</p>
保護の品質	<p>Secure Network Communications (SNC) によって適用されるセキュリティ保護のレベル:</p> <ul style="list-style-type: none"> ○ 認証 - 通信パートナーの検証。追加のデータ保護はありません。 ○ 完全性 - 転送中のデータの変更または操作を検出します。(認証を含む) ○ プライバシー - 転送中のメッセージの暗号化。SNC によって提供されるセキュリティ保護の最大レベル。(完全性と認証を含む) ○ 使用可能な最大 - SAP システムに接続すると、使用可能な最高レベルのセキュリティ保護が使用されます。 	
メタデータ機構	<p>SAP システムで使用される認証プロトコル:</p> <ul style="list-style-type: none"> ○ Kerberos ○ NTLM ○ 個人 	
メカニズム = 個人		
ライブラリパス	外部セキュリティ製品の暗号化ライブラ	C:\SNC_Cryptolib\sapcrypto.dll

設定	説明	例
	りのネットワークパス。	"C:\Security Product\library.dll" <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>スペースがある場合は、ネットワークパスを引用符で囲みます。</p> </div>

詳細設定

設定	説明	例
通貨換算を適用 省略可能	Analytics への SAP データのインポート中に通貨フィールドの自動換算を有効にします。 詳細については、"SAP データのインポート中の通貨の換算" ページ 670 を参照してください。	
要求ごとの最大レコード数 省略可能	クエリの各部分の最大レコード数を指定します。 5000 ~ 1,000,000 の数値を指定できます。ほとんどの場合、デフォルトの 15,000 レコードを使用してください。 コネクターが SAP システムを照会するときには、すべてのレコードが返されるまで、複数の要求が作成され、バッチでレコードが返されます。バッチが完全なテーブルになります。 低速ネットワークでは、要求ごとのレコード数を増やすと、コネクターの再接続回数が減り、SAP コネクターのパフォーマンスが改善される場合があります。	100000
変数文字列長 省略可能	データ型 STRING または RAWSTRING の SAP フィールドの開始からインポートされる文字数。 これらの型のフィールドには、大量のシステム情報が最後の文字として存在することがあります。この設定を使用して、インポートされる文字数を制限することができます。	6 STRING フィールドの最初の 6 文字にのみ関心がある情報があることを知っています。

設定	説明	例
	この設定を省略する場合、デフォルトの256文字が使用されます。	
一時作業ディレクトリ	<p>一時的にSAPデータを格納するディレクトリのパス。</p> <p>SAPコネクタがデータを受信するときに、一時的にローカル保存されます。大きいテーブルをデフォルトの保存場所であるAnalyticsプロジェクトフォルダーに格納できない場合は、オーバーフローエラーになる可能性があります。この設定では、大きいファイルの十分なスペースがある別の一時保存場所を指定できます。</p> <p>デフォルトの入力済み設定を使用する場合、データはAnalyticsプロジェクトフォルダーに一時的に保存されます。</p>	D:\SAP_temp_storage
SAP ルーター 省略可能	<p>SAPシステムがSAPルーターを使用している場合はルーターの接続文字列。</p> <p>接続文字列の一般的な形式： /H/hostname/S/portnumber/W/password(使用される場合)</p>	<p>/H/saprouter.mycompany.com/S/3299</p> <p>/H/saprouter.mycompany.com/S/3299/W/password</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>H、S、およびWは大文字でなければなりません。SAPルーターの接続文字列の特定の形式は異なる場合があります。</p> </div>
RFC 汎用モジュール 省略可能	<p>SAPコネクタ - の基本となるRFC汎用モジュール。</p> <p>このフィールドは空欄にします。値はバックグラウンドで自動的に入力されます。</p> <p>将来モジュール名が変更される場合のためにフィールドが存在します。新しい名前は手動で指定する必要があります。</p>	

SAP テーブルの検索

SAP への新しい接続を作成した後、Analytics は、データアクセス ウィンドウの **使用可能なテーブル** リストに、200 件の共通 SAP テーブルを一覧表示します。リストをスクロールし、テーブル名をクリックして、テーブルを **ステージング領域** に追加します。

使用可能なテーブルリストの検索

リストをスクロールする代わりに、テーブル名またはテーブル説明を検索する方が簡単な場合があります。リストの上の **テーブルの検索** ボックスに検索文字列を入力できます。検索文字列には、リテラル文字とワイルドカード文字の両方を使用できます。

検索はリスト以外も対象とし、テーブル名のローカル キャッシュに保存された表示されない名前も検索します。

使用可能なテーブル の検索の詳細については、"ステージング領域への1つまたは複数のテーブルの追加" ページ 378を参照してください。

検索したテーブルが見つからない場合

検索したテーブルが **使用可能なテーブル** リストまたはテーブル名 キャッシュに見つからない場合、SAP テーブルを検索し、追加のテーブル名を取得できます。

1. **テーブルの検索** ボックスに検索文字列を入力します。
テーブルが見つからない場合、**データベースのテーブル名を検索** ボタンが表示されます。
2. **データベースのテーブル名を検索** をクリックします。

検索文字列と一致する SAP システムのすべてのテーブル名が **使用可能なテーブル** リストに追加されます。テーブル名はテーブル名 キャッシュにも追加されるため、将来すぐに使用できます。

メモ

SAP データベースの検索は、短縮 SAP テーブル名 (ANAT、BNKA など) のみが対象です。テーブル説明は検索されません。

ヒント

使用可能なテーブル リストまたはテーブル名 キャッシュから選択したテーブル名を削除することはできません。テーブルのリストをクリーンアップするには、SAP 接続を削除して再作成します。リストは共通 SAP テーブルのみを含むように戻ります。

SAP テーブルの結合

データ アクセス ウィンドウの **ステージング領域** で SAP テーブルを結合できます。SAP テーブルの結合には次の制限があります。

- 最大 10 個のテーブルを結合できる他の Analytics コネクタと異なり、結合できるテーブルは 8 個です。
- 内部結合および左結合はサポートされます。外部結合および右結合はサポートされません。

詳細については、"データ アクセス ウィンドウのテーブルの結合" ページ 388を参照してください。

キーフィールド アイコン

キーフィールド アイコンは、**ステージング領域**に追加された SAP テーブルのキーフィールドを特定します。キーフィールド アイコンの目的は、SAP テーブルを結合するために使用できるフィールドを特定することです。

メモ

SAP データを Analytics にインポートすると、キーフィールド指定が失われ、キーフィールドは非キーフィールドとは異なる方法で処理されません。

ANAT			
<input type="checkbox"/>	名前	説明	種類
<input type="checkbox"/>	ANLAR	Asset types	CHAR
<input type="checkbox"/>	MANDT	Client	CLNT
<input type="checkbox"/>	SPRAS	Language Key	LANG
<input type="checkbox"/>	TXK50	Asset description	CHAR
<input type="checkbox"/>	XLTXID	Indicator: Long text active	CHAR
へ フィールドの非表示			

SAP データ型

ステージング領域に追加された SAP テーブルのフィールド名に表示される型は SAP データ型です。他の Analytics コネクタには Analytics データ型が表示されます。SAP データ型はインポート時に Analytics データ型に変換されます。

SAP システムの言語

[**データ接続設定**] パネルでは、接続している SAP システムの言語を指定できます。

Analytics でサポートされている言語

SAP システムの言語が Analytics でサポートされている言語のいずれかである場合、**使用可能なテーブル**] リストには、サポートされている言語でテーブル説明が表示されます。

Analytics でサポートされている言語：

英語	ドイツ語	スペイン語	フランス語
中国語	日本語	ポルトガル語	

Analytics でサポートされている言語

SAP システムの言語が Analytics でサポートされている言語のいずれかである場合、**使用可能なテーブル]**リストには、英語でテーブル説明の初期リストが表示されます。続いて SAP システムを検索して取得されたテーブルがシステムの言語で表示されます。結果として、**使用可能なテーブル]**リストのテーブル説明は、英語と SAP システムの言語で表示されます。

メモ

短縮 SAP テーブル名 (ANAT、BNKA など) はすべての言語の標準です。

SAPデータのインポート中の通貨の換算

Analytics にデータをインポート中に SAP データの金額を自動的に換算できます。

通貨換算設定の有効化

1. SAP 接続を作成、または既存の SAP 接続を開きます。
詳細については、「SAP 接続の作成」ページ 662を参照してください。
2. **データ接続の設定]**パネルで、**詳細オプションの表示]**をクリックします。
3. **通貨換算を適用]**を選択します。
4. その他の必要な接続設定を指定して、**保存して接続]**または **接続]**をクリックします。

データテーブルと通貨キーの結合

1. 通貨データを含む SAP テーブルを**ステージング領域**に追加します。
ステージング領域のテーブルでの作業の詳細については、「データアクセス ウィンドウの作業」ページ 376を参照してください。
2. 省略可能。インポートする必要がない通貨フィールドがテーブルにあることが事前にわかっている場合、選択解除します。
3. **インポートのプレビュー]**パネルで、**更新]**をクリックします。

SAP テーブルに換算が必要な通貨データを含む選択済みのフィールドがある場合、通貨キーを含む別のテーブルにこのテーブルを結合することを指示するメッセージが表示されます。例：

テーブル 'BSEG' のフィールド 'DMBTR' では、通貨の桁数調整要のフィールド 'WAERS' を含むテーブル 'T001' と結合する必要があります

4. メッセージの指示に従って、2 番目のテーブルを追加し、結合を作成します。

重要

結合を作成するときに**左結合**を選択します。

ステージング領域の結合テーブルの詳細については、"データ アクセス ウィンドウのテーブルの結合" ページ 388を参照してください。

5. **[インポートのプレビュー]** パネルで、もういちど **[更新]** をクリックします。
SAPテーブルに換算が必要な通貨データを含む別のフィールドがある場合、結合の指示を含む別のメッセージが表示されます。
6. 指示に従って新しい結合を作成します。
7. **[更新]** をクリックしてもメッセージが表示されなくなり、結合されたデータが表示されるまでこのプロセスを繰り返します。
8. SAPのインポートのその他の必要な構成を実行し、データを Analytics に保存します。
詳細については、"データ アクセス ウィンドウの作業" ページ 376を参照してください。

ACLコネクタ(SAP) の設定

ACLコネクタ(SAP) を使用して SAP データをインポートするには、いくつかの設定タスクを完了する必要があります。

SAPシステムで次の操作を実行します。

- SAP システム用の SAP コネクタアドオンのインストール
- SAPコネクタユーザーの SAP 承認の構成
- 1 つ以上の SAP サーバーでポートを構成

ACLコネクタ(SAP) を使用するローカルコンピューターまたはネットワークサーバーで次の操作を実行します。

- SAP GUI for Windows のインストール
- SAP 接続情報と資格情報の構成

HighBond のロボット アプリで SAP コネクタを使用できますが、組織でオンプレミスのロボット エージェントを使用している場合のみです。SAP コネクタは、クラウドベースのロボット エージェントで使用するようには構成できません。

メモ

SAP コネクタ、および必要に応じて SNC(Secure Network Communications) および SSO(シングルサインオン) を設定するには、適切なレベルの技術的な知識を有する作業が必要です。

SAP コネクタを使用するには、基本的な Analytics サブスクリプションの他に、追加のサブスクリプション資格が必要です。必要なサブスクリプション資格がない場合は、SAP に接続できません。

SAP コネクタ サブスクリプションについては、担当者に問い合わせてください。

互換性がある SAP システム

ACL コネクタ(SAP) は次の SAP システムで使用できます。

- S/4HANA
- SAP HANA を含むすべてのサポートされているデータベース プラットフォームで実行されている次のシステムのすべての強化レベル
 - SAP ERP 6.0 (ECC 6.0)
 - SAP CRM 7.0
 - SAP SRM 7.0
 - SAP SCM 7.0
 - SAP EWM 7.0

互換性がある SAP SPAM/SAINT バージョン

SAP システム用 SAP アドオンのインストールでは、必要なバージョンの SAP SPAM/SAINT は 0053 以上です。

SAP SPAM は Support Package Manager です。SAP SAINT は Add-On Installation Tool です。

SAP システム用の SAP コネクタ アドオンのインストール

SAP Basis Administrator は、SAP システムの SAP コネクタ アドオンをインストールする必要があります。SAP コネクタが SAP システムと通信するには、アドオンが必要です。SAP コネクタ サブスクリプションのユーザーは Launchpad からアドオン ファイルをダウンロードできます (www.highbond.com)。

SAP Front End を使用して SAP コネクタ アドオンをインストールします。SAP アドオンをインストールするための互換性がある SAP Basis リリースは 700 以上です。

1. HighBond (www.highbond.com) にサインインします。
2. Launchpad の [ソース] の下にある **ダウンロード** をクリックします。
3. **ACL for Windows**] トップタブの **Data Connector for SAP ERP**] の下で **バージョン 1.4 のダウンロード** をクリックします。

ダウンロードした .zip ファイルには次の .sar ファイルが含まれています。

- **DABEXP_SAPCONN_AOI_140.sar** - SAP コネクタ アドオンをインストールするためのインストーラー ファイル。
 - **DABEXP_SAPCONN_AOU_140.sar** - 既存の SAP コネクタ アドオンをバージョン 1.0 または 1.1 からアップグレードするためのアップグレード ファイル。
4. SAP システムで、**SAINT** トランザクションを使用して、**Add-On Installation Tool** にアクセスします。
 5. **Add-On Installation Tool** を使用して、SAP アドオン インストール パッケージをインストールまたはアップグレードします。

SAP 承認

メモ

SAP 承認は SAP セキュリティ管理者によって付与される必要があります。

SAP システムに接続し、データを抽出するには、SAP コネクタのユーザーは、次の SAP アクセスおよび承認が必要です。

- SAP システムに接続するための SAP ユーザー ID およびパスワード。
- SAP テーブル承認を含む、特定の SAP 承認オブジェクトおよび承認

SAP 承認オブジェクト

SAP コネクタの使用には、次の特定の SAP 承認が必要です。

メモ

ユーザーへの SAP 承認の割り当ての詳細については、SAP セキュリティドキュメントを参照してください。

承認クラス	承認オブジェクト	フィールド	値	詳細
AAAB アプリケーション横断型承認オブジェクト	S_RFC RFC アクセスの承認チェック	ACTVT	16(承認実行)	デスクトップコンピューターなどのリモート ロケーションから SAP システムの汎用モジュールを実行するユーザーの能力を制御します。
		RFC_NAME	/SDF/RI_CRM CMON RFC1 SYST /DABEXP/DAB_FUGR	
		RFC_TYPE	FUGR(汎用グループ)	
		あるいは、SAP NW 7.0 EHP 2 (SAP Basis 702) の時点では、RFC_TYPE を FUNC(汎用モジュール) に設定できます。 FUNC では、汎用グループ経由ではなく、直接汎用モジュールを承認する必要があります。		
		ACTVT	16(承認実行)	
		RFC_NAME	/SDF/CMO_GET_INSTNO CMO_GET_INSTNO RFC_GET_FUNCTION_INTERFACE RFC_GET_NAMETAB RFCPING /DABEXP/RFC_SAPCONNECTOR	
RFC_TYPE	FUNC(汎用モジュール)			

承認クラス	承認オブジェクト	フィールド	値	詳細
BC_A Basis: Administration	S_TABU_ DIS	ACTVT	03 (表示)	特定の SAP テーブルグループへのユーザーのアクセスを制御します。
		DICBERCLS (承認グループ)	*	
	S_TABU_ NAM	ACTVT	03 (表示)	個別の SAP テーブルへのユーザーのアクセスを制御します。
		TABLE (テーブル名)	*	
<p>メモ</p> <p>SAP コネクタのユーザーは、分析を実行するために、アクセスする必要がある SAP テーブルに対する権限の承認が割り当てられている必要があります。</p> <p>たとえば、勘定元帳監査を実行するユーザーは、勘定元帳テーブルに対する権限の承認が必要です。</p> <p>御社独自のビジネスプロセスにより、テーブルに対する権限の承認が必要なユーザーと必要な権限の承認を規定します。SAP セキュリティ管理者と連携し、ユーザーが必要なアクセスの適切なレベルを決定します。</p>				

SAP ポート構成

SAP コネクタで接続する各 SAP サーバーで、送受信通信で次の TCP/IP ポートを開く必要があります。

ポート名	ポート番号	コメント
ディスパッチャー	32<NN>	SAP GUI で使用
ゲートウェイ	33<NN>	RFC 通信で使用
メッセージサーバー	36<NN>	
<p>メモ</p> <p><NN> は SAP システムのインスタンス番号です。SAP システム番号が 10 の場合、ポート 3210、3310、および 3610 を開く必要があります。</p>		

SAP GUI for Windows のインストール

SAP GUI for Windows バージョン 7.60 以上を、ACL コネクタ (SAP) を使用するローカルコンピューターまたはネットワークサーバーにインストールする必要があります。SAP GUI では、SAP コネクタがリモートで SAP システムにアクセスできます。

Analytics ユーザー	SAP GUI は Analytics と同じコンピューターにインストールする必要があります。
ロボット ユーザー	SAP GUI をオンプレミスロボット エージェントがあるサーバーにインストールする必要があります。

SAP 接続情報と資格情報

通常、SAP システムの接続情報は SAP Basis 管理者が管理します。必要な接続情報を提供するとともに SAP Logon ウィンドウでの必要な接続情報を設定するよう、SAP Basis 管理者に依頼してください。

管理者から支援を得られない場合は、あなたまたは管理者から SAP のサポートにお問い合わせください。

SAP に接続するには、自分または管理者が以下を収集する必要があります。

- 正しいサーバータイプ(標準、またはロードバランス)
- インスタンス番号(標準サーバータイプ)
- ログオングループとシステム ID(ロードバランスサーバータイプ)
- SAP システムのホスト名または IP アドレス
- SAP のユーザー名およびパスワード
- クライアント番号
- SAP システムのテーブルへの読み取りアクセス許可

SAP システムと Secure Network Communications (SNC)

組織の SAP システムでは、Secure Network Communications (SNC) が有効な場合があります。これにより、SAP の異なるコンポーネント間の暗号化された通信と、シングルサインオンを使用したユーザー認証が可能になります。

SNC は SAP システムのソフトウェアレイヤーであり、外部セキュリティ製品と統合することで、基本 SAP セキュリティを拡張することができます。

シングルサインオンの有無に関係なく、SNC を使用している場合は、ACL Connector for SAP で追加の設定が必要です。詳細については、「接続設定」ページ 663を参照してください。

SAP 接続エラー

エラー番号	エラーコード	説明
0	不明	SAP への接続、またはデータのインポートを完了できません。不明な

エラー番号	エラーコード	説明
		エラーが発生しました。
1	SapDriverInitializationError	SAP に接続できません。必要なドライバー構成ファイルが見つからない可能性があります。
2	SapDriverLicenseInvalidError	SAP に接続できません。ドライバーライセンスの問題があります。
3	ConnectionPropertiesParserValueMustNotBeNullOrWhitespaceError	SAP に接続できません。必要な接続パラメーターが見つかりません。
4	ConnectionPropertiesParserLibraryTypeOutOfRangeError	ライブラリパラメーターの値が有効ではありません。有効な値: ClassicRfc, NetweaverRfc
5	ConnectionPropertiesParserRfcFunctionTypeOutOfRangeError	RFC 関数パラメーターの値が有効ではありません。有効な値: Standard, Extended, Three
6	ConnectionPropertiesParserServerDetailInfoAmbiguousError	SAP に接続できません。次のプロパティの1つ以上に指定された値が正しくありません: InstanceNumber, LogonGroup, SID。
7	ConnectionPropertiesParserVariableStringLengthNotAnIntegerError	接続パラメーターの変数文字列長は整数である必要があります。
8	SapConnectionInfoValidationClientIsNullOrWhitespaceError	クライアント接続パラメーターの値が見つかりません。
9	SapConnectionInfoValidationClientLongerThanThreeCharactersError	クライアント接続パラメーターの値が3文字を超えています。
10	SapConnectionInfoValidationLanguageIsNullOrWhitespaceError	言語接続パラメーターの値が見つかりません。

エラー番号	エラーコード	説明
11	SapConnectionInfoValidationLanguageIsLongerThanTwoCharactersError	言語接続パラメーターの値が2文字を超えています。
12	SapConnectionInfoValidationUsernameIsNullOrEmptyError	ユーザー名接続パラメーターの値が見つかりません。
13	SapConnectionInfoValidationPasswordIsNullOrEmptyError	パスワード接続パラメーターの値が見つかりません。
14	SapConnectionInfoValidationTestDownloadTableIsNullOrEmptyError	テストダウンロードテーブル接続パラメーターの値が見つかりません。
15	SapConnectionInfoValidationRfcFunctionModuleIsNullOrEmptyError	RFC関数モジュール接続パラメーターの値が見つかりません。
16	SapConnectionInfoValidationSapConnectionServerInfoIsNullError	SAPに接続できません。SAPサーバープロパティが見つかりません(内部ドライバーエラー)。
17	SapConnectionInfoValidationVariableStringLengthMustBeGreaterZeroError	SAPに接続できません。変数文字列長が0より大きくなければなりません(内部ドライバーエラー)。
18	SapConnectionInfoValidationServerIsNullOrEmptyError	サーバー接続パラメーターの値が見つかりません。
19	SapConnectionInfoValidationServerDetailsIsNullError	SAPに接続できません。SAPサーバー詳細プロパティが見つかりません(内部ドライバーエラー)。
20	SapConnectionInfoValidationInstanceNumberNullOrEmptyError	インスタンス番号接続パラメーターの値が見つかりません。
21	SapConnectionInfoValidationInstanceNumberMustHaveTwoDigitsError	インスタンス番号接続パラメーターの値は2桁である必要があります。

データの定義とインポート

エラー番号	エラーコード	説明
22	SapConnectionInfoValidationLogonGroupNullOrWhitespaceError	ログオングループ接続パラメーターの値が見つかりません。
23	SapConnectionInfoValidationSidNullOrWhitespaceError	SID 接続パラメーターの値が見つかりません。
24	SapConnectionInfoValidationSidLongerThanThreeCharactersError	SID 接続パラメーターの値が3文字を超えています。
25	SapDriverErpError	SID サーバーへの接続が失われました。不明なエラーが発生しました。
26	SapDriverTableNotFoundError	テーブルが見つかりません。
27	SapDriverTableStructureNotExportableError	テーブルをインポートできません。データがありません。
28	SapDriverAppendStructureNotExportableError	テーブルをインポートできません。データがありません。
29	SapDriverGetTablesDownloadError	SAP テーブルのダウンロードを完了できませんでした。
30	SapDriverSqlParsingError	SQL 構文が有効ではない可能性があります。データのインポートを実行できません。エラー原因を特定できません。
31	SapDriverSqlParsingNoSqlScriptFragmentError	SQL 文を指定する必要があります。
32	SapDriverSqlParsingExactlyOneBatchAllowedError	1 SQL のみが許可されます。
33	SapDriverSqlParsingExactlyOneSelectStatementAllowedError	1 SELECT 文のみが許可されます。
34	SapDriverSqlParsingFragmentNotAllowedError	SQL 文の一部が許可されないか、現在の位

エラー番号	エラーコード	説明
		置では許可されません。たとえば、UPDATE または DELETE 文は許可されません。
36	SapDriverSqlParsingJoinTypeNotAllowedError	指定された JOIN タイプは許可されません。INNER JOIN または LEFT JOIN を指定してください。
37	SapDriverSqlParsingFirstJoinValueNotATableError	JOIN で指定された最初の値がテーブルではありません。
38	SapDriverSqlParsingOnlyOneJoinAllowedError	1 JOIN のみが許可されます。
39	SapDriverSqlParsingSecondJoinValueNotATableError	JOIN で指定された 2 番目の値がテーブルではありません。
40	SapDriverSqlParsingNoJoinConditionError	JOIN 条件を空にすることはできません。
41	SapDriverSqlParsingJoinConditionOringNotAllowedError	OR と JOIN 条件の使用は許可されません。
42	SapDriverSqlParsingJoinConditionNotAllowedError	指定された JOIN 条件のタイプは許可されません。
43	SapDriverSqlParsingJoinOperatorMustBeEqualsOnlyError	JOIN 演算子は等しいのみにすることができます。
44	SapDriverSqlParsingJoinSameFieldError	フィールドをそれ自体に JOIN できません。
45	SapDriverSqlParsingOnlyExplicitJoinsAllowedError	明示的な JOINS のみが許可されます。
46	SapDriverSqlParsingHasNoSelectStatementError	SELECT 文を指定する必要があります。
47	SapDriverSqlParsingHasNoQuerySpecificationError	SQL 構文が無効である可能性があります。クエリ指定が必要です。

データの定義とインポート

エラー番号	エラーコード	説明
48	SapDriverSqlParsingSqlFieldMustBeQualifiedWithTableError	フィールドはテーブル名またはテーブルエイリアスで修飾する必要があります。
49	SapDriverSqlParsingSqlSelectStarFieldMustOnlyBeQualifiedWithTableNameError	SELECT * はテーブル名またはテーブルエイリアスで修飾する必要があります。
50	SapDriverSqlParsingSqlFieldMustOnlyBeQualifiedWithTableNameError	フィールドはテーブル名またはテーブルエイリアスでのみ修飾できます。
51	SapDriverSqlParsingSqlTableMustHaveExactlyOneIdentifierError	テーブル名は修飾しないでください。
52	SapDriverSqlParsingSqlQueryeMustHaveAtLeastOneFieldError	SELECT 文には1つ以上のフィールドを含める必要があります。
53	SapDriverSqlParsingSqlQueryeMustHaveAtLeastOneTableError	SELECT 文には1つ以上のテーブルを含める必要があります。
54	SapDriverSqlParsingSqlEnhancedQuerySqlTableNotMatchingAnySapTableError	フィールドを修飾するテーブルがSAPシステムから受信されたテーブルと一致しません。
55	SapDriverSqlParsingSqlFieldQualifierNotMatchingAnyTableError	フィールドを修飾するテーブルがSELECT文で指定されたテーブルと一致しません。
56	SapDriverSqlParsingSqlSelectStarFieldQualifierNotMatchingAnyTableError	* フィールドを修飾するテーブルがSELECT文で指定されたテーブルと一致しません。
57	SapDriverSqlParsingSqlFieldNotInSapTableError	SELECT 文で指定されたフィールドがSAPデータに存在しません。
58	SapDriverSqlParsingSqlFieldNotExportableError	SELECT 文で指定されたフィールドをSAPシステムでインポートできません。

エラー番号	エラーコード	説明
59	SapDriverSqlParsingSqlFieldNotFilterableError	WHERE 節で指定されたフィールドは SAP システムでフィルタリングできません。
60	SapDriverSqlParsingTopRowFilterMustHaveAbsolutValueError	SELECT TOP 文は絶対値を指定する必要があります。
61	SapDriverSqlParsingTopRowFilterWithTiesNotAllowedError	SELECT TOP 文はタイなしで使用する必要があります。
63	SapDriverSqlParsingTopRowFilterMustHaveIntegerValueError	SELECT TOP 文は整数値を指定する必要があります。
64	SapDriverSqlParsingTopRowFilterIntegerValueMustBeGreaterZeroError	SELECT TOP 文はゼロより大きい値を指定する必要があります。
68	SapDriverSqlParsingTopRowFilterFromClauseMustHaveInnerQuerySpecification-Error	SELECT TOP 文の FROM 節にはクエリ指定を含める必要があります。
69	SapDriverSqlParsingOringFilterValuesOnlyForSameFieldAndClauseError	異なるフィールドは OR 条件で指定できません。同じフィールドは OR 条件で使用する必要があります、同じフィールド演算子を使用する必要があります。
70	SapDriverSqlParsingFilterExpressionNotAllowedError	フィールドをフィルタリングするために WHERE 節で指定された式は許可されません。
71	SapDriverSqlParsingFilterValueLiteralNotAllowedError	フィールドをフィルタリングするために WHERE 節で指定されたリテラルは許可されません。
72	SapDriverSqlParsingFilterComparisonTypeNotAllowedError	フィールドをフィルタリングするために WHERE 節で指定された比較演算子は許可されません。
73	SapDriverPackageCreationFilterTypeNotSupportedError	フィルタータイプがサ

データの定義とインポート

エラー番号	エラーコード	説明
		ポートされていません。
74	SapDriverTableCreationMaximumRecordLimitReachedError	選択したフィールド数は、SAP RFC(SAP Remote Function Call) でサポートされている最大数を超過しています。
75	SapDriverExecuteDownloadError	SAP テーブルデータのインポート中に不明なエラーが発生しました。
76	SapDriverSearchTablesDownloadError	SAP テーブルの検索中に不明なエラーが発生しました。
77	SapDriverSqlParsingNoFromClauseError	SELECT 文には FROM 句を含める必要があります。
78	SapDriverSqlParsingNoTableReferenceInFromClauseError	SELECT 文の FROM 句はテーブルを指定する必要があります。
79	SapDriverSqlParsingExpectedTopRowForPreviewError	外部 SELECT 文は TOP 句が必要です。
80	SapDriverSqlParsingPreviewStatmentMustNotHaveWhereClauseError	外部 SELECT 文は WHERE 節が必要です。
81	SapDriverSqlParsingPreviewStatmentMustHaveUnqualifiedStarQualifierOnlyError	外部 SELECT 文は * 修飾子のみが必要です。
82	SapDriverProjectPathNotFoundError	指定された一時作業ディレクトリが存在しません。

SAP ByDesign への接続

SAP Business ByDesign はクラウドベースの企業資源計画 (ERP) システムです。組織の SAP ByDesign データをインポートするには、SAP ByDesign データコネクタを使用できます。

メモ

SAP ByDesign データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

SAP ByDesign に接続するには、以下を収集する必要があります。

- システムホスト名の URL
- SAP ByDesign アカウントに接続するための資格情報

接続の前提条件の収集については、組織の SAP ByDesign 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SAP ByDesign のサポートにお問い合わせください。

SAP ByDesign 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、SAP ByDesign を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
SAP ByDesignへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

SAP ByDesignの接続は、**[既存の接続]** タブに保存されます。今後、SAP ByDesignに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SAP ByDesignからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

SAP ByDesign データ コネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、SAP ByDesign に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

SAP Hybris Cloud for Customer への接続

SAP Hybris Cloud for Customer (C4C) はクラウドベースの顧客関係管理 (CRM) システムです。組織の SAP Hybris C4C データをインポートするには、SAP Hybris C4C データコネクタを使用できます。

メモ

SAP Hybris C4C データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

SAP Hybris C4C に接続するには、以下を収集する必要があります。

- SAP Hybris C4C アカウントに接続するための資格情報
- 接続を確立するテナント
- 接続する SAP Hybris C4C URL

接続の前提条件の収集については、組織の SAP Hybris C4C 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SAP Hybris C4C のサポートにお問い合わせください。

SAP Hybris C4C 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ (バンドル) セクションの **新規の接続** タブから、SAP Hybris C4C を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
SAP Hybris C4Cへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

SAP Hybris C4Cの接続は、 **[既存の接続]** タブに保存されます。今後、SAP Hybris C4Cに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。SAP Hybris C4Cからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

SAP Hybris C4C データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、SAP Hybris C4C に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

SAP SuccessFactors への接続

SAP SuccessFactors はクラウドベースの人事管理システムです。組織の SAP SuccessFactors データをインポートするには、SAP SuccessFactors データコネクタを使用できます。

メモ

SAP SuccessFactors データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

SAP SuccessFactors に接続するには、以下を収集する必要があります。

- SAP SuccessFactors サーバーの URL
- SAP SuccessFactors アカウントに接続するための資格情報
- 会社の一意の識別子

接続の前提条件の収集については、組織の SAP SuccessFactors 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SAP SuccessFactors のサポートにお問い合わせください。

SAP SuccessFactors 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**SAP SuccessFactors** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
SAP SuccessFactorsへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

SAP SuccessFactorsの接続は、**[既存の接続]** タブに保存されます。今後、SAP SuccessFactorsに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SAP SuccessFactorsからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

SAP SuccessFactors データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、SAP SuccessFactors に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

ServiceNow に接続する

ServiceNow は IT 管理ソリューションのクラウド プロバイダーです。ServiceNow データをインポートするには、ServiceNow データ コネクターを使用します。

メモ

このコネクターを Analytics から独立して使用することはできません。コネクタードライバーの DSN は Windows の ODBC データソース アドミニストレーターを使って設定できますが、DSN 接続を本コネクターの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

ServiceNow に接続するには、以下を収集する必要があります。

- ユーザー名とパスワードを含む接続ユーザー アカウント
- テーブルの取得元となるインスタンス
- クライアント ID とクライアント シークレット OAuth 資格情報は、ServiceNow の OAuth 承認サーバーでアプリケーションを登録するときに割り当てられます。

ヒント

アプリケーションの登録と OAuth 資格情報の詳細については、「ServiceNow ドキュメントの「OAuth の設定」を検索してください。

接続の前提条件の収集については、組織の ServiceNow 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から ServiceNow のサポートにお問い合わせください。

ServiceNow 接続を作成する

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクターセクションの **新規の接続**] タブから、ServiceNow を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

3. **データ接続の設定**] パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続**] をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

ServiceNowの接続は、**既存の接続** タブに保存されます。今後、ServiceNowに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。ServiceNowからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	ServiceNow への認証に使用するユーザー アカウント	admin_1
パスワード	このユーザーを認証するのに使用するパスワード	
インスタンス	テーブルの取得元となる ServiceNow インスタンス	staging16387
OAuth クライアント Id	<p>クライアント ID は、ServiceNow の OAuth 承認サーバーでアプリケーションを登録するときに割り当てられます。</p> <p>OAuth では、アプリケーションを登録する必要があります。登録の一部として、クライアント ID(別称:コンシューマキー) およびクライアント シークレットを受け取ります。接続を行うには、これら両方を指定する必要があります。</p>	356a825803610300ef0662490d237522
OAuth クライアント シークレット	<p>クライアント シークレットは、ServiceNow の OAuth 承認サーバーでアプリケーションを登録するときに割り当てられます。</p> <p>OAuth では、アプリケーションを登録する必要があります。登録の一部として、クライアント ID(別称:コンシューマキー) およびクライアント シークレットを受け取ります。接続を行うには、これら両方を指定する必要があります。</p>	secretTest

詳細設定

設定	説明	例
日付時刻を GMT に変換	<p>インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。</p> <p>メモ この方法を選択する場合に、日付 (=) 演算子を使って日付時刻フィールドに適用するフィルターを使用するには、そのフィルターの条件の日付時刻値を入力する際、'Z' 形式の指定子(例: 2017-01-01 12:38:54Z)を使用する必要があります。</p>	true
鍵サイズの上限	<p>主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。</p> <p>このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。</p>	255
Long Varchar にマッピング	<p>列が SQL_LONGVARCHAR として返されるかどうかを制御します。</p> <p>この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。</p>	-1
WVarchar にマッピング	<p>文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p> <p>文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この設定は、テーブル列ではない場合</p>	MyTable=*

設定	説明	例
	<p>に、擬似列の値を設定できない Entity Framework で特に有用です。</p> <p>この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。</p>	
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サンプリント (16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サンプリント (16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetect を false に設定し、ProxyServer と ProxyPort も false に設定する必要があります。認証するには、ProxyAuthScheme を設定し、必要に応じて、ProxyUser と ProxyPassword を設定します。</p> </div>	BASIC

設定	説明	例
	<p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC -HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	<p>システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。</p>	true
プロキシ ユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、このプロパティを次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	john_doe@example.com
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追</p>	

設定	説明	例
	<p>加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。</p>	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ:</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではありません。 ◦ TUNNEL - 接続はトンネルプロキシ経由です。プロキシサーバーはリモートホストへの接続を開き、トラフィックはプロキシ経由で前後します。 	AUTO

データ コネクタの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクタを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたはODBCドライバーのサードパーティベンダーが加えた変更により、1つまたは複数のデータコネクタを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセスウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクタのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセスウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたはODBCドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

ServiceNow データコネクタの変更内容

ServiceNow データコネクタに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	変更
14.2	ServiceNow の日付フィールドが、YYYY-MM-DD の書式(長さ 10 文字)を使って、ServiceNow データコネクタの日付データ型にマッピングされるようになりました。 旧バージョンでは、ServiceNow の日付フィールドは、YYYY-MM-DD hh:mm:ss の書式(長さ 19 文字)を使って、ServiceNow データコネクタの日付時刻データ型にマッピングされていました。インポートされるデータの時刻部分は空になっていて、"00:00:00" と表されていました。

SFTP への接続

SFTP は Secure Shell Protocol (SSH) プロトコル上で実行される安全なファイル転送プロトコルです。SFTP データコネクタを使用して、SFTP データをインポートできます。

メモ

SFTP データコネクタを使用すると、拡張子が .xlsx と csv のファイルタイプのみをインポートできます。このコネクタのスクリプト サポートは制限されている場合もあります。

作業を開始する前に

SFTP に接続するには、以下を収集する必要があります。

- SFTP サーバーに接続するための資格情報
- SFTP サーバーのホスト名または IP アドレス。
- 正しい接続ポート
- SFTP サーバーのパス

接続の前提条件の収集については、組織の SFTP 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SFTP のサポートにお問い合わせください。

SFTP 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、SFTP を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの**接続名**を使用するか、新しい接続名を入力します。

SFTP の接続は、**既存の接続** タブに保存されます。今後、SFTP に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。SFTP からデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ユーザー	SFTP サーバーで認証するユーザーアカウント。	
パスワード	ユーザーを SFTP サーバーで認証するためのパスワード。	
ホスト	SFTP サーバーを実行するインスタンスのホスト名または IP アドレス。	https://sftp.abc.com/
ポート	SFTP サーバーのポート。	443
パス	SFTP サーバーの現在の作業ディレクトリ。作業ディレクトリを変更するには、絶対ディレクトリパス、またはこのプロパティの既存の値に関する相対パスに設定します。	

詳細設定

設定	説明	例
鍵サイズの上限	主キー列の最大長。	255

設定	説明	例
Long Varchar にマッピング	このプロパティは、列が SQL_LONGVARCHAR として返されるかどうかを制御します。	-1
WVarchar にマッピング	このプロパティは、文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。	
擬似列	この値は、擬似列を列としてテーブルに含めるかどうかを指定します。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。	
SSL サーバー証明書	SSL を使用して接続するときにサーバーから許可される必要がある証明書。 次のいずれかを使用できます。 <ul style="list-style-type: none"> ◦ 完全な PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 または SHA1 サンプリント(スペースまたはコロンで 16 進数値を区切ることができます) <p>コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	

SharePoint への接続

Microsoft SharePoint は、組織のコンテンツとアプリケーションを共有、管理するための、Web ベースの共同作業プラットフォームです。SharePoint データをインポートするには、SharePoint データコネクターを使用します。

インポートする前に

表形式データのみ

Analytics はタブ形式でのみ SharePoint データを問い合わせることができます。次の項目は、表形式ではないため、Analytics から問い合わせることができません。

- 添付ファイル
- FileVersions
- GetValidTerms
- アクセス許可
- ビュー

多要素認証はサポートされていません

SharePoint データコネクターは、多要素認証が必要な SharePoint アカウントのデータにはアクセスできません。

既知の問題：日付時刻フィールドによるフィルターリング

日付時刻を GMT に変換 オプションをオンにしてデータをインポートした場合は、日付時刻フィールドによってフィルターリングすることはできません。日付時刻フィールドによってフィルターリングしたい場合は、インポートを行う前に、このオプションをオフにしてください。

SharePoint 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Sharepoint を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Sharepointの接続は、**既存の接続** タブに保存されます。今後、Sharepointに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Sharepointからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続設定

基本設定

設定	説明	例
接続名	Analytics でこの接続に付ける名前。	Sharepoint
ホスト	接続する SharePoint サイト。 URL をサイト コレクションに設定し、すべてのネストされたサブサイトで、すべてのリストと文書を処理します。そのサイトでのみリストと文書を処理するには、特定のサイトへの URL を設定します。	https://teams.example.com/teamA
ユーザー	SharePoint ユーザー名。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>御社で AD FS、OneLogin、または OKTA などの SSO ツールを使用して SharePoint にアクセスするには、SharePoint 資格情報でなく SSO 資格情報を入力して、Analytics 経由で接続します。</p> </div>	jgibbons
パスワード	SharePoint パスワード。	MyStrongPasswordExample

設定	説明	例
	<p>メモ 御社で AD FS、OneLogin、または OKTA などの SSO ツールを使用して SharePoint にアクセスするには、SharePoint 資格情報でなく SSO 資格情報を入力して、Analytics 経由で接続します。</p>	
Share Point 版	使用中の SharePoint のエディション。SharePoint Online または SharePoint On-Premise のいずれか。	Sharepoint オンライン
SSO 使用	<p>true に設定すると、シングルサインオン (SSO) は、ユーザーおよびパスワード経由で指定されたアカウントを使用して、SharePoint Online と認証するために必要です。Active Directory Federation Services (AD FS)、OneLogin、および OKTA SSO もサポートされます。</p> <p>SSO ドメインで構成されたドメインがユーザーのドメインと異なる場合は、SSO ドメインが必要になる場合があります。</p> <p>SSO は SharePoint Online を使用するときのみ適用されます。SSO はオンプレミスバージョンの SharePoint ではサポートされていません。</p>	false
認証スキーム	<p>パスワードとユーザーとともに、このフィールドは、サーバーで認証するために使用されます。NTLM はデフォルトのオプションです。次のオプションを使用して、認証スキームを選択します。</p> <ul style="list-style-type: none"> ○ NTLM - これを設定すると、Windows 資格情報で認証します。 ○ NEGOTIATE - Auth Scheme が NEGOTIATE に設定されている場合、サーバーとの認証メカニズムをネゴシエーションします。Auth Scheme を NEGOTIATE に設定すると、Kerberos 認証を使用します。 ○ KERBEROSDELEGATION - これを設定して、Kerberos プロトコル経由で委任を使用します。なりすますアカウントのユーザーとパスワードを設定します。 ○ NONE - これを設定すると、匿名認証を使用します。たとえば、公開サイトへのアクセスの目的です。 ○ FORMS - SharePoint インスタンスが Web フォーム経由で ○ DIGEST - HTTP DIGEST 認証を使用します。 ○ BASIC - これを設定すると、HTTP 基本認証を使用します。 	NTLM

詳細設定

設定	説明	例
SSOドメイン	<p>このプロパティは、シングルサインオンを使用するとき (SSO の使用が true) のときにのみ適用されます。ユーザーのドメイン (user@myssodomain.com など) が SSO サービスで構成されたドメイン (user@myssodomain.com など) と異なる場合にのみ適用されます。</p> <p>AD FS、OneLogin、または OKTA SSO を使用するときには、プロパティが必要になる場合があります。</p>	myssodomain.com
日付時刻を GMT に変換	<p>インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。</p>	true
鍵サイズの上 限	<p>一部の ODBC ツール (Microsoft Access など) では、主キー列の長さが特定の値を超えることができません。このプロパティでは、ODBC ドライバーが主キー列のすべての報告された列を無効にします。Microsoft Access Linked データソースとして ODBC ドライバーを使用するときには、特に注意してください。</p> <p>鍵サイズの上限を 0 に接続すると、キー長が元の長さに戻ります。</p>	255
Long Varchar にマッピング	<p>このプロパティは、列が SQL_LONGVARCHAR として返されるかどうかを制御します。</p> <p>一部のアプリケーションでは、特定の文字数よりも大きいすべてのテキスト データを SQL_LONGVARCHAR として報告する必要があります。これを使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。</p>	-1
WVarchar に マッピング	<p>このプロパティは、文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p> <p>文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、Map To WVarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この設定は、テーブル列ではない場合に、擬似列</p>	<ul style="list-style-type: none"> ◦ Table1=Column1, Table2=Column3 ◦ *=*

設定	説明	例
	<p>の値を設定できない Entity Framework で特に有用です。</p> <p>アスタリスク(*)文字を使用すると、すべてのテーブルとすべての列を含めることができます。</p>	
大文字の ID	<p>このプロパティを設定すると、テーブルおよび列名を含む、すべての識別子を報告します。これは Oracle データベースのデフォルトであるため、Oracle Database Gateway などの Oracle ツールとの統合に適しています。たとえば、このプロパティを使用して、ID を引用することを回避できます。</p>	false
プロキシ認証スキーム	<p>プロキシ サーバーで認証するために使用する認証タイプ。</p> <p>この値は、プロキシ サーバーおよびプロキシ ポートで指定された HTTP プロキシで認証するために使用する認証タイプを指定します。</p> <p>デフォルトでシステム プロキシ設定を使用します。追加の構成は必要ありません。別のプロキシに接続する場合は、Proxy Server および Proxy Port の他に、Proxy Auto Detect を false に設定する必要があります。認証するには、Proxy Authentication Scheme を設定し、必要に応じて、Proxy User および Proxy Password を設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC -HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLM または Kerberos トークンを取得します。 ◦ PROPRIETARY - NTLM または Kerberos token を生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	<p>システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetect を FALSE に設定すると、カスタム プロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。</p> <p>デフォルトでは、ドライバーはシステム HTTP プロキシを使用します。別のプロキシに接続する場合は、これを false に設定します。</p> <p>HTTP プロキシに接続するには、プロキシ サーバーを参照してください。</p>	false
プロキシ ユーザー	<p>プロキシ サーバーで認証するために使用されるユーザー名。</p> <p>Proxy User および Proxy Password オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p>	<p>jjibbons@example.com</p>

データの定義とインポート

設定	説明	例
	<p>Proxy AuthScheme で使用可能な認証タイプのいずれかを選択できます。</p> <p>HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、次の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	
プロキシのパスワード	<p>プロキシ サーバーで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシサーバーで認証するために使用されます。HTTP プロキシを指定するには、Proxy Server と Proxy Port を設定できます。認証タイプを指定するには、Proxy Auth Scheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、Proxy User と Proxy Password を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、Proxy User と Proxy Password を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシに接続する場合は、Proxy AutoDetect を false に設定します。</p>	MyStrongPasswordExample
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、Proxy Auto Detect を false に設定します。</p>	127.168.192.10
プロキシのポート	<p>HTTP トラフィックをリダイレクトする、HTTP プロキシが実行されているポート。プロキシ サーバーで HTTP プロキシを指定します。</p>	80
プロキシ SSL タイプ	<p>このプロパティは、ProxyServer で指定された HTTP プロキシへの接続用に SSL を使用するタイミングを決定します。値は、AUTO、ALWAYS、NEVER、または TUNNEL にすることができます。</p> <p>Auto -デフォルト設定。URL が HTTPS URL の場合、TUNNEL オプションを使用します。URL が HTTP URL の場合、コンポーネントは NEVER オブ</p>	自動

設定	説明	例
	<p>ションを使用します。</p> <p>Always - 接続は常に SSL 有効です。</p> <p>Never - 接続は SSL 有効ではありません。</p> <p>Tunnel - 接続はトンネルプロキシ経由です。プロキシサーバーはリモート ホストへの接続を開き、トラフィックはプロキシ経由で前後します。</p>	
プロキシ例外	<p>プロキシサーバー経由での接続から除外されるホストまたは IP のセミコロン区切りのリスト。</p> <p>このプロパティで定義されたアドレスを除くすべてのアドレスで、プロキシサーバーが使用されます。</p>	127.168.189.10; 127.168.188.11

ShipStation への接続

ShipStation は、e コマースの出荷ソリューションです。ShipStation データコネクタを使用すると、組織の ShipStation データをインポートできます。

メモ

ShipStation データコネクタは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の ShipStation 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から ShipStation のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

ShipStation 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、ShipStation を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
ShipStationへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

ShipStationの接続は、**[既存の接続]** タブに保存されます。今後、ShipStationに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。ShipStationからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

ShipStation データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、ShipStation に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Slack への接続

Slack は、複数のデバイスとプラットフォームで使用できるチームや職場向けのメッセージング アプリです。Slack データコネクタを使用して、組織の Slack データをインポートできます。

メモ

Slack データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

Slack に接続するには、Slack サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Slack 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Slack のサポートにお問い合わせください。

Slack 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. **ACL DSN コネクタ(バンドル)セクションの 新規の接続**]タブから、**Slack**を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**]ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN Configuration(DSN 構成)]** ダイアログボックスで **[Test Connection(接続のテスト)]** をクリックします。

Slack のログインページが表示されます。

4. ログインするには

- Slack アカウント 資格情報がある場合は、資格情報を入力します。
- 組織で SSO を使用している場合は、ボタンをクリックして、SSO でサインインします。

5. **続行]** をクリックします。

SSO を使用している場合は、認証資格情報を入力してログインします。

6. 接続成功ダイアログボックスが表示されるので、**[OK]**をクリックします。

Slack への接続が正常に確立されます。

7. **[DSN 構成]**ダイアログボックスで **[OK]**をクリックします。

Slack の接続は、**Slack** という名前で **[既存の接続]** タブに保存されます。今後、保存したこの接続を使用して Slack に再接続できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Slack からデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376 を参照してください。

接続名の変更

Slack データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、Slack に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Snowflake への接続

Snowflake はクラウド ベースのデータウェアハウス プラットフォームです。Snowflake データ コネクタを使用 して、組織の Snowflake データをインポート できます。

メモ

Snowflake データ コネクタは当社のデータ パートナーである CData によって提供されて います。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを 参照してください。

作業を開始する前に

Snowflake に接続するには、以下を収集する必要があります。

- Snowflake データベースに接続するための資格情報
- Snowflake ウェアハウスの名前
- Snowflake データベースの URL

接続の前提条件の収集については、組織の Snowflake 管理者に支援を仰いでください。管理者から 支援を得られない場合は、あなたまたは管理者から Snowflake のサポートにお問い合わせください。

Snowflake 接続の作成

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Snowflake** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボッ クスに検索文字列を入力します。コネクタはアルファベット 順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Snowflakeへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Snowflakeの接続は、 **[既存の接続]** タブに保存されます。今後、Snowflakeに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Snowflakeからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Snowflake データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクター]** セクションを展開して、Snowflake に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Splunk への接続

Splunk はセキュリティ情報およびイベント管理ソリューション(SIEM) です。Splunk データ コネクターを使用して、組織の Splunk データをインポートできます。

メモ

Splunk データ コネクターは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Splunk に接続するには、以下を収集する必要があります。

- Splunk アカウントに接続するための資格情報
- Splunk アカウントにアクセスするための URL

接続の前提条件の収集については、組織の Splunk 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Splunk のサポートにお問い合わせください。

Splunk 接続の作成

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクター(バンドル) セクションの **[新規の接続]** タブから、**Splunk** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**[接続のフィルタリング]** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **[必須フィールドを表示]** または **[基本]** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**OK**]をクリックします。
Splunkへの接続が正常に確立されます。
8. **DSN 構成**]ダイアログボックスで **OK**]をクリックします。

Splunkの接続は、**既存の接続**]タブに保存されます。今後、Splunkに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Splunkからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

Splunk データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **既存の接続**]タブに保存されます。名前を変更する場合は、**既存の接続**]タブに移動し、**ACL DSN コネクター**]セクションを展開して、Splunk に対応する省略記号アイコン  をクリックし、**接続名の変更**]を選択します。

Square への接続

Square は財務およびマーチャント サービス アグリゲーターです。Square データ コネクターを使用して、組織の Square データをインポートできます。

メモ

Square データ コネクターは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Square に接続するには、Square サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Square 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Square のサポートにお問い合わせください。

Square 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクター(バンドル) セクションの 新規の接続**] タブから、**Square** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**テスト接続**] をクリックせずに **OK**] をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。

7. ダイアログボックスが表示されるので、**OK**]をクリックします。

Squareへの接続が正常に確立されます。

8. **DSN 構成**]ダイアログボックスで **OK**]をクリックします。

Squareの接続は、**既存の接続**]タブに保存されます。今後、Squareに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Squareからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Square データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **既存の接続**]タブに保存されます。名前を変更する場合は、**既存の接続**]タブに移動し、**ACL DSN コネクター**]セクションを展開して、Square に対応する省略記号アイコン  をクリックし、**接続名の変更**]を選択します。

Stripe への接続

Stripe は電子商取引 Web サイトおよびモバイルアプリケーション向けのオンライン決済インフラストラクチャサービスを提供します。Stripe データコネクタを使用して、組織の Stripe データをインポートできます。

メモ

Stripe データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Stripe に接続するには、Stripe サーバーに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の Stripe 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Stripe のサポートにお問い合わせください。

Stripe 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、**Stripe** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。

接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。

7. ダイアログボックスが表示されるので、**OK**]をクリックします。

Stripeへの接続が正常に確立されます。

8. **DSN 構成**]ダイアログボックスで **OK**]をクリックします。

Stripeの接続は、**既存の接続**]タブに保存されます。今後、Stripeに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。Stripeからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Stripe データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **既存の接続**]タブに保存されます。名前を変更する場合は、**既存の接続**]タブに移動し、**ACL DSN コネクタ**]セクションを展開して、Stripe に対応する省略記号アイコン  をクリックし、**接続名の変更**]を選択します。

SugarCRM への接続

SugarCRM はオープンソース顧客関係管理 (CRM) ソフトウェアです。SugarCRM データ コネクタを使用して、組織の SugarCRM データをインポートできます。

メモ

SugarCRM データ コネクタは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

SugarCRM に接続するには、以下を収集する必要があります。

- インストールされている SugarCRM の API バージョン
- SugarCRM アカウントの URL
- SugarCRM アカウントに接続するための資格情報

接続の前提条件の収集については、組織の SugarCRM 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SugarCRM のサポートにお問い合わせください。

SugarCRM 接続の作成

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**SugarCRM** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
SugarCRMへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

SugarCRMの接続は、**[既存の接続]** タブに保存されます。今後、SugarCRMに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SugarCRMからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

SugarCRM データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、SugarCRM に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

SuiteCRM への接続

SuiteCRM は、オープンソースのカスタマーリレーションシップマネジメント (Customer Relationship Management: CRM) ソリューションです。SuiteCRM データコネクターを使用して、組織の SuiteCRM データをインポートできます。

メモ

SuiteCRM データコネクターは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の SuiteCRM 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SuiteCRM のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

SuiteCRM 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクター(バンドル) セクションの **[新規の接続]** タブから、**SuiteCRM** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**[接続のフィルタリング]** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **[必須フィールドを表示]** または **[基本]** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
SuiteCRMへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

SuiteCRMの接続は、**[既存の接続]** タブに保存されます。今後、SuiteCRMに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。SuiteCRMからデータをインポートする方法については、「データアクセスウィンドウの作業」ページ 376を参照してください。

接続名の変更

SuiteCRM データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、SuiteCRM に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

SurveyMonkey への接続

SurveyMonkey クラウドベースのオンライン調査ツールです。SurveyMonkey データ コネクターを使用して、組織の SurveyMonkey データをインポートできます。

メモ

SurveyMonkey データ コネクターは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

SurveyMonkey に接続するには、SurveyMonkey アカウントに接続するための資格情報を収集する必要があります。

接続の前提条件の収集については、組織の SurveyMonkey 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から SurveyMonkey のサポートにお問い合わせください。

SurveyMonkey 接続を作成する

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、**SurveyMonkey** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
SurveyMonkeyへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログ ボックスで **[OK]** をクリックします。

SurveyMonkeyの接続は、 **[既存の接続]** タブに保存されます。今後、SurveyMonkeyに接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。SurveyMonkeyからデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

SurveyMonkey データ コネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、SurveyMonkey に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Sybase への接続

Sybase はデータベース管理システム(DBMS) です。Sybase データ コネクタを使用して、組織の Sybase データをインポートできます。

メモ

Sybase データ コネクタは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

Sybase に接続するには、以下を収集する必要があります。

- Sybase データベースに接続するための資格情報
- Sybase データベースのポート
- Sybase データベースを実行しているサーバーの名前
- Sybase データベースの名前

接続の前提条件の収集については、組織の Sybase 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Sybase のサポートにお問い合わせください。

Sybase 接続を作成する

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**Sybase** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Sybaseへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Sybaseの接続は、 **[既存の接続]** タブに保存されます。今後、Sybaseに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Sybaseからデータをインポートする方法については、"データアクセス ウィンドウの作業" ページ 376を参照してください。

接続名の変更

Sybase データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクター]** セクションを展開して、Sybase に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Sybase IQ への接続

Sybase IQ はデータウェアハウスおよびビッグデータ管理向けの関係データベースサーバーです。Sybase IQ データコネクタを使用して、組織の Sybase IQ データをインポートできます。

メモ

Sybase IQ データコネクタは当社のデータパートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

Sybase IQ に接続するには、以下を収集する必要があります。

- Sybase IQ アカウントに接続するための資格情報
- Sybase IQ データベースのポート
- Sybase IQ または SAP SQL Anywhere データベースを実行しているサーバーの名前
- Sybase IQ または SAP SQL Anywhere データベースの名前

接続の前提条件の収集については、組織の Sybase IQ 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Sybase IQ のサポートにお問い合わせください。

Sybase IQ 接続を作成する

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクタ(バンドル) セクションの **新規の接続** タブから、Sybase IQ を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Sybase IQへの接続が正常に確立されます。
8. **[DSN構成]** ダイアログボックスで **[OK]** をクリックします。

Sybase IQの接続は、 **[既存の接続]** タブに保存されます。今後、Sybase IQに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Sybase IQからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Sybase IQ データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Sybase IQ に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

Tenable.sc への接続

Tenable.sc はネットワークおよび IT インフラストラクチャ向けのセキュリティおよび脆弱性管理ソリューションです。Tenable.sc データコネクタを使用して、Tenable.sc データをインポートできます。

作業を開始する前に

Tenable.sc に接続するには、以下を収集する必要があります。

- Tenable.sc インスタンスのホスト名
- Tenable.sc インスタンスの正しい接続ポート
- インスタンスに接続するための正しい認証方法
- インスタンスに接続するためのユーザー資格情報

接続の前提条件の収集については、組織の Tenable.sc 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Tenable.sc のサポートにお問い合わせください。

Tenable.sc 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタセクションの **新規の接続** タブから、Tenable.sc を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Tenable.sc の接続は、**既存の接続** タブに保存されます。今後、Tenable.sc に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Tenable.sc からデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
ホスト	Tenable.sc を実行するインスタンスのホスト名または IP アドレス。	https://sc.tenable.online/
ポート	Tenable.sc インスタンスのポート。	443
認証方法	<p>Tenable.sc インスタンスに接続するための認証方法。オプション:</p> <ul style="list-style-type: none"> API キー - アクセスキーとシークレットキーを使用して認証します。 ユーザー名/パスワード - ユーザー名とパスワードを使用して認証します。 <p>メモ API キー承認を使用するには、Tenable.sc 5.13.x 以降のバージョンが必要です。</p>	API キー

設定	説明	例
ユーザー	Tenable.sc インスタンスで認証するための Tenable.sc ユーザー アカウント。	
パスワード	ユーザーを Tenable.sc インスタンスで認証するためのパスワード。	
アクセス鍵	Tenable.sc インスタンスで認証するための API アクセスキー。	
秘密鍵	Tenable.sc インスタンスで認証するための API シークレット キー。	

詳細設定

設定	説明	例
鍵サイズの上 限	主キー列の最大長。	255
Long Varchar にマッピング	このプロパティは、列が SQL_LONGVARCHAR として返されるかどうかを制御します。	-1
WVarchar にマッピング	このプロパティは、文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。	
擬似列	この値は、擬似列を列としてテーブルに含めるかどうかを指定します。 この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。	
SSL サーバー証明書	SSL を使用して接続するときにサーバーから許可される必要がある証明書。 次のいずれかを使用できます。 <ul style="list-style-type: none"> 完全な PEM 証明書 証明書を含むローカルファイルへの 	

設定	説明	例
	<p>パス</p> <ul style="list-style-type: none"> 公開鍵 MD5 または SHA1 サムプリント (スペースまたはコロンで 16 進数値を区切ることができます) <p>コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	
日付時刻を GMT に変換	インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。このオプションが無効な場合、日付時刻値は、Analytics を実行するシステムのオペレーティングシステムタイムゾーンに変換されます。	
プロキシ認証スキーム	<p>プロキシ サーバーで認証するために使用する認証スキームを指定します。使用可能なオプションは次のとおりです。</p> <ul style="list-style-type: none"> BASIC - HTTP BASIC 認証を実行します。 DIGEST - HTTP DIGEST 認証を実行します。 NEGOTIATE - 認証に設定されたプロトコルに基づき、NTLM または Kerberos トークンを取得します。 PROPRIETARY - NTLM または Kerberos トークンを生成しません。HTTP 要求の承認ヘッダーでこのトークンを指定する必要があります。 	BASIC
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。このオプションに設定された値は他のプロキシ設定よりも優先されます。このため、カスタムプロキシ設定を使用する場合は、このオプションを無効にしてください。	
プロキシ ユーザー	プロキシ サーバーで認証するためのユーザー名。	
プロキシのパスワード	プロキシ サーバーでプロキシ ユーザーを認証するためのパスワード。	
プロキシ サーバー	HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。	
プロキシのポート	プロキシ サーバーが実行されている TCP ポート。	
プロキシ SSL タイプ	ProxyServer プロキシに接続するときに	AUTO

設定	説明	例
	<p>使用する SSL タイプ。使用可能なオプションは次のとおりです。</p> <ul style="list-style-type: none"> ○ AUTO - URL が HTTPS URL の場合、TUNNEL オプションを使用します。URL が HTTP URL の場合、コンポーネントは NEVER オプションを使用します。 ○ ALWAYS - 接続は常に SSL が有効です。 ○ NEVER - 接続は SSL が有効ではありません。 ○ TUNNEL - 接続はトンネルプロキシ経由です。 <p>このオプションは、プロキシ サーバーの値を入力するときのみ有効です。</p>	

Teradata への接続

Teradata はクラウド データ サービスです。Teradata データ コネクタを使用して、Teradata データをインポートできます。

メモ

Analytics は任意のコネクタとして Teradata を提供します。データ アクセス ウィンドウでこのコネクタを使用できない場合、一般的に、インストール中にコネクタが選択されていないことが原因です。詳細については、"任意の Analytics データ コネクタと Python エンジンを実装する" ページ 2663を参照してください。

作業を開始する前に

Teradata に接続するには、以下の情報を収集する必要があります。

- Teradata データベース インスタンスの名前または IP アドレス
- ユーザー名
- パスワード

接続の前提条件の収集については、組織の Teradata 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Teradata のサポートにお問い合わせください。

Teradata 接続の作成

1. Analytics のメイン メニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL コネクタ セクションの **新規の接続** タブから、Teradata を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

3. **データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの接続名を使用するか、新しい接続名を入力します。

Teradata の接続は、**既存の接続** タブに保存されます。今後、Teradata に接続するには、保存したこの接続を使用できます。

接続が確立されると、データ アクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Teradata からデータをインポートする方法については、"データ アクセス ウィンドウの作業" ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
名前または IP アドレス	Teradata データベース インスタンスの完全修飾ドメイン名または IP アドレス。	
統合セキュリティの使用	<p>ドライバーがシングルサインオン(SSO)または従来のサインオン(CSO)のどちらかを使用して接続を認証するかどうかを指定します。</p> <ul style="list-style-type: none"> 有効 - ドライバーは SSO を使用し、クライアント コンピューターのユーザー情報から取得された Teradata データベース認証資格情報を使用して接続を認証します。 無効 - ドライバーは CSO を使用し、Teradata データベース認証資格情報を入力するようにユーザーに要求します。 	無効
メタデータ機構	<p>データベースへの接続を認証するためにドライバーで使用するメカニズム:</p> <ul style="list-style-type: none"> KRBS - Kerberos プロトコルを使用します。アプリケーションでは、ユーザー名とパスワードが提供されます。 LDAP - LDAP プロトコルを使用します。アプリケーションでは、ユーザー名とパスワードが提供されます。 SPNEGO - Windows .NET クライアントから Teradata データベースにログオンするユーザーで Kerberos 認証をサポートします。 TD2 - Teradata 2 メカニズムを使用します。この場合、ユーザーは Teradata データベースユーザー名とパスワードを入力する必要があります。 TDNEGO - Teradata Negotiating で自動的に選択されるメカニズムを使用します(シングルサインオンを含む)。 	LDAP
ユーザー名	指定した認証メカニズムで Teradata データベース接続を認証するユーザー名。	

設定	説明	例
パスワード	データベースにアクセスするためのパスワード。	
テラデータ ウォレットの有効化	ドライバーが Teradata Wallet 参照文字列を使用して接続を認証するかどうかを指定します。	
デフォルトのデータベース	Teradata 接続が開いているときにアクセスするデータベースの名前。	
アカウントの文字列	個別のユーザー アカウントを識別し、特定のユーザー ID に関連付けられています。	
セッションの文字セット	セッションで使用する文字セット(文字エンコーディング)。デフォルト値は ASCII です。	ASCII

詳細設定

設定	説明	例
ヘルプ データベースなし	ヘルプ データベースを使用するかどうかを指定します。 <ul style="list-style-type: none"> 有効 - SQLTables でワイルドカード文字が使用されていないときに、SELECT 文が使用されます。 無効 - HELP DATABASE コマンド菓子用されます。 	無効
検索パターンの無視	アンダースコア(_)とパーセント記号(%)文字が標準の文字として解釈されるか、検索ワイルドカードとして解釈されるかどうかを指定します。	
レガシーパーサーの有効化	レガシーパーサーが有効かどうかを指定します。	
エラーイベントの記録	エラーイベントをログに出力するかどうかを指定します。	
小数点の記号として地域設定を使用	小数点記号で地域設定を使用するか、地域設定ではなくピリオド(.)を使用するかどうかを指定します。	有効
データ暗号化の有効化	データベースとのすべての通信が暗号化されるか、認証情報のみが暗号化されるかどうかを指定します。	

データの定義とインポート

設定	説明	例
拡張ステートメント情報の有効化	データベースで使用可能なときに拡張ステートメント情報が使用されるかどうかを指定します。 このオプションが有効な場合は、ODBC API 関数 SQLDescribeParam がサポートされます。	
セッションのモード	データベースでのセッション中にドライバーが使用するセッションモードを指定します。 <ul style="list-style-type: none"> ◦ ANSI ◦ Teradata ◦ システム デフォルト デフォルト値は "システム デフォルト" です。	ANSI
応答バッファの最大サイズ	SQL 要求の応答バッファの最大サイズ(キロバイト)。 デフォルト値は 65536 です。	65536
TDMST のポート番号	Teradata データベースにアクセスするために使用されるポート番号。 デフォルト値は 1025 です。	1025
変換 DLL 名	Teradata サーバーとドライバーとの間で転送されるすべてのデータを変換するための関数を含む .dll ファイルへの完全パス。	
変換オプション	変換 DLL ファイルで使用されるオプション。	
ログインのタイムアウト	データベースにログインするときにドライバーが応答を待機する秒数。この時間を過ぎると、オペレーションがキャンセルされます。 デフォルト値は 20 です。	20
データソースの DNS エントリ	データソースの DNS で定義されたエントリ数。	
TCP_NODELAY の使用	TCP が小さいパケットをただちに送信するか、単一の大きいパケットになるまで送信を待機するかどうかを指定します。 <ul style="list-style-type: none"> ◦ 有効 - TCP はただちに小さいパケットを送信します。このオプションでは送信の遅延を回避できますが、ネットワークトラフィックが増える可能性 	有効

設定	説明	例
	<p>があります。</p> <ul style="list-style-type: none"> 無効 - TCP は小さいパケットを収集し、単一の大きいパケットにまとめます。このオプションではネットワークトラフィックが減りますが、送信の遅延が生じる可能性があります。 	
カタログ名としてのヌル値の使用	ドライバーが Catalog Name パラメーターを NULL に設定するかどうかを指定します。	
先読み機能の有効化	現在のメッセージの処理中に、次の応答メッセージを要求するかどうかを指定します。	
システム呼び出しの再試行 (EINTR)	EINTR エラーが発生したときに、ドライバーがソケット システム呼び出しを再試行するか、SQL_ERROR を返すかどうかを指定します。	
TIMESTAMP パラメーターとしての DATE データの使用	SQL_TIMESTAMP および SQL_C_TIMESTAMP として返されるパラメーターで DATE データが送信されるかどうかを指定します。	
2.x アプリケーションでのカスタム カタログモードの使用	このオプションが有効な場合、非対応の検索パターンを使用する ODBC 2.x アプリケーションの後方互換性が確保されます。	
SQL_TIMESTAMP の CREATE_PARAMS 列に空文字列を返す	SQL_TIMESTAMP データで SQLGetTypeInfo を呼び出すときに、空の文字列を返すか、CREATE_PARAMS 列の特定の値を返すかどうかを指定します。	
最大値を返す CHAR/VARCHAR Length as 32K (長さが 32K の CHAR/VARCHAR)	データで SQLGetTypeInfo を呼び出すときに、COLUMN_SIZE 列のハードコード値を返すかどうかを指定します。	

データ コネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサード パーティ ベンダーが加えた変更により、1 つまたは複数のデータ コネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去**] を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Twitter への接続

Twitter のライブ データを接続し、ツイート、フォロワー、メッセージ、検索などにアクセスします。コネクタは **アプリケーションのみの認証** を使用するため、Twitter でアプリケーションを登録し、必要な資格情報を取得する必要があります。

メモ

このコネクタを Analytics から独立して使用することはできません。コネクタドライバの DSN は Windows の **ODBC データソースアドミニストレータ** を使って設定できますが、DSN 接続を本コネクタの [Windows DSN configuration(Windows DSN 設定)] ダイアログ内でなく、Analytics 内でテストする必要があります。

作業を開始する前に

Twitter に接続するには、アプリケーションを登録し、[Twitter Application Management](#) ポータルを使用して、Twitter から OAuth 資格情報を取得する必要があります。

接続の前提条件の収集については、組織の Twitter 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Twitter のサポートにお問い合わせください。

Twitter 接続の作成

- Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
- ACL コネクタ セクションの **新規の接続** タブから、Twitter を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

- データ接続の設定** パネルが開いたら、接続設定を入力し、パネルの下部にある **保存して接続** をクリックします。

デフォルトの **接続名** を使用するか、新しい接続名を入力します。

Twitter の接続は、**既存の接続** タブに保存されます。今後、Twitter に接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Twitter からデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376 を参照してください。

接続設定

基本設定

設定	説明	例
OAuth クライアント ID	<p>クライアント ID は、Twitter の OAuth 承認サーバーでアプリケーションを登録するときに割り当てられます。</p> <p>OAuth では、アプリケーションを登録する必要があります。登録の一部として、クライアント ID(別称:コンシューマキー)およびクライアントシークレットを受け取ります。OAuthClientId と OAuthClientSecret の両方を、OAuth サーバーに接続するように設定されません。</p>	xvz1evFS4wEEPTGEFPHBog
OAuth クライアント シークレット	<p>クライアントシークレットは、Twitter の OAuth 承認サーバーにアプリケーションを登録するときに割り当てられます。</p> <p>OAuth では、アプリケーションを登録する必要があります。登録の一部として、クライアント ID(別称:コンシューマキー)およびクライアントシークレットを受け取ります。OAuthClientId と OAuthClientSecret の両方を、OAuth サーバーに接続するように設定されません。</p>	L8qq9PZyRg6ieKGEKhZolGC0vJWLw8iEJ88DRdyOg
OAuth アクセストークン	<p>OAuth を使用して接続するためのアクセストークン。</p> <p>OAuthAccessToken プロパティは、OAuth を使用して接続するために使用されます。</p> <p>OAuthAccessToken は、認証処理の一部として OAuth サーバーから取得します。サーバー依存タイムアウトであり、要求間で再</p>	xvz1evFS4wEEPTGEFPHBog:L8qq9PZyRg6ieKGEKhZolGC0vJWLw8iEJ88-DRdyOg

設定	説明	例
	<p>利用できます。</p> <p>アクセストークンは、ユーザー名とパスワードの代わりに使用されます。アクセストークンは、サーバーで課すことによって、資格情報を保護します。</p>	
OAuth アクセストークンシークレット	<p>OAuth を使用して接続するための OAuth アクセストークンシークレット。</p> <p>OAuthAccessTokenSecret プロパティは、OAuth を使用して接続および認証するために使用されます。OAuthAccessTokenSecret は、認証処理の一部として、OAuth サーバーから取得されます。</p> <p>OAuthAccessToken を使用し、タイムアウトするまで、複数の要求を使用できます。</p>	Ewy4p5VygSPOUI1rhbCIVZcLnu05Y23Md22F0AKQYh1Hg

詳細設定

設定	説明	例
日付時刻を GMT に変換	<p>インポート時に日付時刻フィールドを GMT タイムゾーンに変換します。日付時刻値は、false の場合には、Analytics を実行するコンピューターのオペレーティングシステムタイムゾーンに変換されます。</p>	true
鍵サイズの上限	<p>主キー列の最大長。サイズを 0 に接続すると、キー長が元の長さに戻ります。</p> <p>このプロパティでは、コネクタが主キー列のすべての報告された列を無効にします。</p>	255
Long Varchar にマッピング	<p>列が SQL_LONGVARCHAR として返されるかどうかを制御します。</p> <p>この設定を使用すると、指定されたサイズよりも大きい列をマッピングし、SQL_VARCHAR ではなく、SQL_LONGVARCHAR として報告されます。</p>	-1

設定	説明	例
WVvarchar にマッピング	<p>文字列タイプを SQL_VARCHAR ではなく、SQL_WVARCHAR にマッピングするかどうかを制御します。デフォルトで設定されます。</p> <p>文字列を SQL_WVARCHAR にマッピングし、さまざまな国際文字セット格納するため、MapToWVvarchar はデフォルトで true に設定されています。false に設定すると、SQL_VARCHAR を使用できます。</p>	true
擬似列	<p>擬似列を列としてテーブルに含めるかどうかを指定します。</p> <p>この設定は、テーブル列ではない場合に、擬似列の値を設定できない Entity Framework で特に有用です。</p> <p>この接続設定の値は、 "Table1=Column1, Table1=Column2, Table2=Column3" の形式です。"*" 文字を使用すると、すべてのテーブルとすべての列を含めることができます。</p>	MyTable=*
大文字の ID	テーブルおよび列名を含む、すべての識別子を大文字で報告します。	false
SSL サーバー証明書	<p>TLS/SSL を使用して接続するときにサーバーから許可される証明書。次のいずれかを指定できます。</p> <ul style="list-style-type: none"> ◦ 完全 PEM 証明書 ◦ 証明書を含むローカルファイルへのパス ◦ 公開鍵 ◦ MD5 サンプリント(16 進数値はスペースまたはコロン区切りにすることができます) ◦ SHA1 サンプリント(16 進数値はスペースまたはコロン区切りにすることができます) <p>TLS/SSL 接続を使用している場合は、このプロパティを使用して、サーバーから許可される TLS/SSL 証明書を指定できます。コンピューターから信頼されない他の認定書はすべて拒否されます。</p>	C:\cert.cer
拡張 SQL のサポート	<p>メモリ内クライアント側処理を有効にし、API 経由で直接サポートされるよりも、SQL 機能を強化します。</p> <ul style="list-style-type: none"> ◦ true - コネクターはできるかぎり SELECT 文処理の負荷を IMAP に 	

設定	説明	例
	<p>分散し、メモリのクエリの残りを処理します。この方法で、ドライバーはサポートされていない述語、結合、集計を実行できます。</p> <ul style="list-style-type: none"> ◦ false -コネクタはSQL 実行をIMAP API のサポート内容に制限します。 <p>述語の実行</p> <p>コネクタは、データソースでサポートされている節を決定してから、ソースにプッシュし、クエリを満たす行の最も小さいスーパーセットを取得します。その後、行の残りをローカルでフィルタリングします。フィルター処理はストリームされ、ドライバーは非常に大きいデータセットでも効果的にフィルタリングできます。</p> <p>結合の実行</p> <p>コネクタはさまざまな手法を使用してメモリで結合します。ドライバーは、同じテーブルを複数回読み取る要件に対して、メモリ利用率をトレードオフします。</p> <p>集計の実行</p> <p>コネクタは、メモリ内の集計を処理するために必要なすべての行を取得します。</p>	
プロキシ認証スキーム	<p>ProxyServer プロキシで認証するために使用する認証タイプ。</p> <p>この値は、ProxyServer および ProxyPort で指定されたHTTP プロキシで認証するために使用する認証タイプを指定します。</p>	BASIC

設定	説明	例
	<p>メモ</p> <p>コネクタはデフォルトでシステムプロキシ設定を使用し、さらなる構成は必要ありません。別のプロキシに接続する場合は、ProxyAutoDetectをfalseに設定し、ProxyServerとProxyPortもfalseに設定する必要があります。認証するには、ProxyAuthSchemeを設定し、必要に応じて、ProxyUserとProxyPasswordを設定します。</p> <p>認証タイプは次のいずれかにすることができます。</p> <ul style="list-style-type: none"> ◦ BASIC - HTTP BASIC 認証を実行します。 ◦ DIGEST - HTTP DIGEST 認証を実行します。 ◦ NEGOTIATE - 認証に適したプロトコルに基づき、NTLMまたはKerberosトークンを取得します。 ◦ PROPRIETARY - NTLMまたはKerberos tokenを生成しません。HTTP要求の承認ヘッダーでこのトークンを指定する必要があります。 	
プロキシ自動検出	システムプロキシ設定を使用するかどうかを指定します。ProxyAutoDetectをFALSEに設定すると、カスタムプロキシ設定を使用します。これは他のプロキシ設定よりも優先されます。	true
プロキシユーザー	<p>ProxyServer プロキシで認証するために使用されるユーザー名。</p> <p>ProxyUser および ProxyPassword オプションは、ProxyServer で指定された HTTP プロキシに対して接続および認証するために使用されます。</p> <p>ProxyAuthScheme で使用可能な認証タイプのいずれかを選択できます。HTTP 認証を使用している場合は、これを HTTP プロキシで認識されるユーザーのユーザー名に設定します。</p> <p>Windows または Kerberos 認証を使用している場合は、このプロパティを次</p>	john_doe@example.com

設定	説明	例
	<p>の形式のいずれかのユーザー名に設定します。</p> <ul style="list-style-type: none"> ◦ user@domain ◦ domain\user 	
プロキシのパスワード	<p>ProxyServer プロキシで認証するために使用されるパスワード。</p> <p>このプロパティは、NTLM (Windows)、Kerberos、または HTTP 認証をサポートする HTTP プロキシ サーバーで認証するために使用されます。HTTP プロキシを指定するには、ProxyServer と ProxyPort を設定できます。認証タイプを指定するには、ProxyAuthScheme を設定します。</p> <p>HTTP 認証を使用している場合は、追加で、ProxyUser と ProxyPassword を HTTP プロキシに設定します。</p> <p>NTLM 認証を使用している場合は、ProxyUser と ProxyPassword を Windows パスワードに設定します。Kerberos 認証を完了するためにこれらが必要な場合もあります</p>	
プロキシ サーバー	<p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。</p> <p>HTTP トラフィックをルーティングするプロキシのホスト名または IP アドレス。ドライバーは、HTTP、Windows (NTLM)、または Kerberos 認証タイプを使用して、HTTP プロキシで認証できます。</p> <p>デフォルトでは、ドライバーはシステムプロキシを使用します。別のプロキシを使用する場合は、ProxyAutoDetect を false に設定します。</p>	206.174.193.115
プロキシのポート	ProxyServer プロキシが実行されている TCP ポート。	80
プロキシ SSL タイプ	<p>ProxyServer プロキシに接続するときに使用する SSL タイプ:</p> <ul style="list-style-type: none"> ◦ AUTO - URL が HTTPS URL の場合、コネクタは TUNNEL オプションを使用します。URL が HTTP URL の場合、コネクタは NEVER オプション(デフォルト)を使用します。 ◦ ALWAYS - 接続は常に SSL 有効です。 ◦ NEVER - 接続は SSL 有効ではあ 	AUTO

設定	説明	例
	<p>りません。</p> <ul style="list-style-type: none"> ○ TUNNEL - 接続はトンネルプロキシ経由です。プロキシ サーバーはリモート ホストへの接続を開き、トラフィックはプロキシ経由で前後します。 	

ストリーミング テーブル

TweetStream テーブルなど、継続的に更新されるデータをキャプチャするテーブルはクエリしないでください。ストリーミング テーブルは、履歴データのアーカイブではなく、リアルタイムのアクティビティが格納されるだけです。これらのテーブルで確立される接続は開いたままになるため、ご使用アカウントでのAPIの更新頻度制限を超えてしまう可能性があります。

データ コネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データ ソースまたは ODBC ドライバーのサード パーティ ベンダーが加えた変更により、1 つまたは複数のデータ コネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータ アクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データ ソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データ アクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去**] を選択します。

- **フィールド指定の更新** - データ ソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

Twitter データ コネクタの変更内容

Twitter データ コネクタに加えられた具体的な変更内容を次に挙げます。

Analytics のバージョン	変更
14.2	DirectMessagesSent および DirectMessagesReceived テーブルが削除され、新しいテーブル、DirectMessages に取って代わられました。
	Tweets テーブルの IdLong フィールドのデータ型が、long から string に変更されました。

UPS への接続

UPS はグローバル配送および物流会社であり、レポート ツール、データベース、カスタム アプリケーションからライブ追跡 データを提供しています。UPS データ コネクタを使用して、組織の UPS データをインポートできます。

メモ

UPS データ コネクタは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

UPS に接続するには、以下を収集する必要があります。

- UPS への要求を処理するサーバー
- UPS サーバーに接続するためのアクセス キー
- UPS サーバーに接続するための資格情報
- UPS アカウント番号

接続の前提条件の収集については、組織の UPS 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から UPS のサポートにお問い合わせください。

UPS 接続の作成

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]**を選択します。
2. **ACL DSN コネクタ(バンドル)** セクションの **新規の接続** タブから、**UPS** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **[DSN 構成]** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
UPSへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

UPSの接続は、**[既存の接続]** タブに保存されます。今後、UPSに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。UPSからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

UPS データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、UPS に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

USPS への接続

United States Postal Service(USPS)は配送および郵便サービスであり、レポート ツール、データベース、カスタムアプリケーションからライブ追跡データを提供しています。USPS データコネクターを使用して、組織のUSPS データをインポートできます。

メモ

USPS データコネクターは当社のデータパートナーであるCDataによって提供されています。接続フィールドについては、[CData の Web サイト](#)で提供されているドキュメントを参照してください。

作業を開始する前に

USPS に接続するには、以下を収集する必要があります。

- 郵便プロバイダー
- 要求がサンドボックス サーバーまたはプロダクション サーバーのどちらから送信されるかどうか
- 荷送人のアカウント番号
- Stamps サーバーで認証するための Stamps ユーザー ID
- サーバーで認証するためのパスワード

接続の前提条件の収集については、組織のUSPS 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者からUSPS のサポートにお問い合わせください。

USPS 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション]を選択します。
2. ACL DSN コネクター(バンドル)セクションの **新規の接続** タブから、**USPS**を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング**ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
USPSへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

USPSの接続は、**[既存の接続]** タブに保存されます。今後、USPSに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。USPSからデータをインポートする方法については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

接続名の変更

USPS データコネクターを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクター]** セクションを展開して、USPS に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

Workday への接続

Workday は、クラウドベースの企業管理システムです。Workday データコネクターを使用して、組織の Workday データをインポートできます。

メモ

Workday データコネクターは当社のデータアクセスおよび接続パートナーである CData Software によって提供されています。個別の接続フィールドの詳細については、次のリンクから CData ドキュメントをご覧ください。

作業を開始する前に

接続処理を開始する前に、CData ドキュメントの [Windows DNS 構成](#) セクションを確認し、ユーザー名、パスワード、トークン、またはキー値といった接続に必要な情報を準備してください。

接続の前提条件の収集については、組織の Workday 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から Workday のサポートにお問い合わせください。

接続プロファイル

基本的な接続と認証だけでなく、データソース接続の追加プロパティを構成できます。詳細については、CData ドキュメントの [Connection Properties](#) を参照してください。

Workday 接続の作成

1. Analytics のメインメニューから、[インポート > データベースとアプリケーション] を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、Workday を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。 **[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、 **[OK]** をクリックします。
Workdayへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

Workdayの接続は、 **[既存の接続]** タブに保存されます。今後、Workdayに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて **ステージング領域** が表示されるので、データのインポートを開始できます。Workdayからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

Workday データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも **[既存の接続]** タブに保存されます。名前を変更する場合は、 **[既存の接続]** タブに移動し、 **[ACL DSN コネクタ]** セクションを展開して、Workday に対応する省略記号アイコン  をクリックし、 **[接続名の変更]** を選択します。

xBase への接続

xBase は、書式設定された大量のデータを構造化された形式で保存するための、複雑なデータおよびノート ファイルです。xBase データ コネクタを使用して、組織の xBase データをインポートできます。

メモ

xBase データ コネクタは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

xBase に接続するには、xBase アカウントへのデータ ソース パスを収集する必要があります。

接続の前提条件の収集については、組織の xBase 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から xBase のサポートにお問い合わせください。

xBase 接続を作成する

1. Analytics のメイン メニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. **ACL DSN コネクタ(バンドル) セクションの 新規の接続**] タブから、**xBase** を選択します。

ヒント

使用可能なコネクタのリストをフィルタリングするには、**接続のフィルタリング**] ボックスに検索文字列を入力します。コネクタはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成**] ダイアログボックスで **必須フィールドを表示**] または **基本**] をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。
5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**テスト接続**] をクリックせずに **OK**] をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。

7. ダイアログボックスが表示されるので、**[OK]**をクリックします。

xBaseへの接続が正常に確立されます。

8. **[DSN 構成]**ダイアログボックスで **[OK]**をクリックします。

xBaseの接続は、**[既存の接続]**タブに保存されます。今後、xBaseに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセスウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。xBaseからデータをインポートする方法については、"データアクセスウィンドウの作業" ページ 376を参照してください。

接続名の変更

xBase データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前で **[既存の接続]**タブに保存されます。名前を変更する場合は、**[既存の接続]**タブに移動し、**[ACL DSN コネクタ]**セクションを展開して、xBase に対応する省略記号アイコン  をクリックし、**[接続名の変更]**を選択します。

ZenDesk への接続

ZenDesk はマルチチャネル顧客サポート プラットフォームです。ZenDesk データ コネクターを使用して、組織の ZenDesk データをインポートできます。

メモ

ZenDesk データ コネクターは当社のデータ パートナーである CData によって提供されています。接続フィールドについては、[CData の Web サイト](#) で提供されているドキュメントを参照してください。

作業を開始する前に

ZenDesk に接続するには、次の情報を収集する必要があります。

- ZenDesk サポート URL
- ZenDesk アカウントで認証するための資格情報
- ZenDesk 増分 API を使用するかどうか
- ZenDesk で認証するユーザーの API トークン

接続の前提条件の収集については、組織の ZenDesk 管理者に支援を仰いでください。管理者から支援を得られない場合は、あなたまたは管理者から ZenDesk のサポートにお問い合わせください。

ZenDesk 接続の作成

1. Analytics のメインメニューから、**[インポート > データベースとアプリケーション]** を選択します。
2. ACL DSN コネクター(バンドル) セクションの **新規の接続** タブから、**ZenDesk** を選択します。

ヒント

使用可能なコネクターのリストをフィルタリングするには、**接続のフィルタリング** ボックスに検索文字列を入力します。コネクターはアルファベット順に表示されます。

[DSN Configuration] ダイアログボックスが表示されます。

3. **DSN 構成** ダイアログボックスで **必須フィールドを表示** または **基本** をクリックします。
4. 必須フィールドまたは基本フィールドがある場合は、値を入力します。

5. **[Test Connection]** (接続のテスト) をクリックします。

メモ

[Test Connection] (接続のテスト) ボタンをクリックすると、指定された接続と認証詳細情報が正しいかどうかを検証します。**[テスト接続]** をクリックせずに **[OK]** をクリックすると、接続詳細情報が検証されずに保存され、指定した値が正しくない場合は後から動作しない可能性があります。

6. ログインページが表示されたら、必要な接続または認証の詳細を入力します。
接続が成功すると、ダイアログボックスが開き、成功メッセージが表示されます。
7. ダイアログボックスが表示されるので、**[OK]** をクリックします。
ZenDeskへの接続が正常に確立されます。
8. **[DSN 構成]** ダイアログボックスで **[OK]** をクリックします。

ZenDeskの接続は、**[既存の接続]** タブに保存されます。今後、ZenDeskに接続するには、保存したこの接続を使用できます。

接続が確立されると、データアクセス ウィンドウが開いて**ステージング領域**が表示されるので、データのインポートを開始できます。ZenDeskからデータをインポートする方法については、「データアクセス ウィンドウの作業」 ページ 376を参照してください。

接続名の変更

ZenDesk データコネクタを作成するときには、任意の接続名を指定することはできません。接続はデータソースと同じ名前でも**[既存の接続]** タブに保存されます。名前を変更する場合は、**[既存の接続]** タブに移動し、**[ACL DSN コネクタ]** セクションを展開して、ZenDesk に対応する省略記号アイコン  をクリックし、**[接続名の変更]** を選択します。

HighBond プロジェクト データのインポート

Analytics テーブルを作成するには、HighBond プロジェクトで操作する権限があるプロジェクトからデータをインポートします。

Project Manager からは、HighBond インスタンスのすべてのアクティブなプロジェクトに含まれるすべてのテキスト情報を集合的に含むさまざまなタイプのテーブルをインポートできます。

当該の情報を活用する方法

プロジェクトからインポートされた情報を使用して、さまざまな種類の分析を実行できます。例：

- あるプロジェクトの未完了の情報を見つける方法として、キーフィールドでインポートされたテーブルを結合する
詳細については、"プロジェクトからインポートされたテーブルの結合" 見開きページを参照してください。
- Tableau や Microsoft Power BI などのさまざまなレポート ツールでの使用を目的に、Analytics から情報をエクスポートする
詳細については、"サードパーティレポート アプリケーションからの Analytics への接続" ページ 1405 を参照してください。

レコード長の制限

Analytics にインポートできる最大レコード長は 32,767 文字または 16,383 Unicode 文字です。

最大値を超えるデータをプロジェクトからインポートしようとすると、エラーメッセージが表示され、インポートが失敗します。インポートから大きい説明文フィールドを削除し、レコード長を削減してください。

もう一つの方法は、Analytics のコマンドラインで `IMPORT GRCPROJECT` コマンドを使用してインポートを実行することです。このコマンドを使用するとき、`CHARMAX` パラメーターを含めることができます。これを使用すると、インポートするフィールドごとに最大文字数を指定できます。詳細については、"IMPORT GRCPROJECT コマンド" ページ 1892を参照してください。

手順

メモ

HighBond に接続するときには、パスワードを指定する必要がある場合があります。詳細については、「パスワード要件」次のページを参照してください。

1. [インポート]> [HighBond]> [プロジェクト]を選択します。
2. [インポートするデータの選択]ダイアログボックスで、組織フォルダーをダブルクリックします。
3. インポートするテーブルのフォルダーをダブルクリックします。

メモ

一度にAnalytics にインポートできるテーブルは1つだけです。共通キーフィールドがある場合は、プロジェクトからインポートされたテーブルを後から結合できます。

4. インポートするフィールドを選択するか、**すべて選択**]を選択し、テーブル全体をインポートして**OK**]をクリックします。
データがプロジェクトからインポートされます。インポートされたテーブルの使用性を改善するには、HTML マークアップがあるリッチテキスト フィールドが最後に配置されます。
5. **データファイルを別名で保存**]ダイアログボックスで、Analytics データファイルのファイル名を入力し、必要であれば、そのファイルを保存する場所を変更して、**保存**]をクリックします。
6. Analytics プロジェクトに追加する Analytics テーブルの名前について、デフォルトの名前をそのまま用いるか、または別の名前を入力して、**OK**]をクリックします。

メモ

Analytics のテーブル名は、最長で64文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

プロジェクトからインポートされたテーブルの結合

プロジェクト アプリから Analytics にテーブルをインポートした後、テーブルを結合し、作業していたプロジェクトのテキスト情報の一部またはすべてを作成できます。

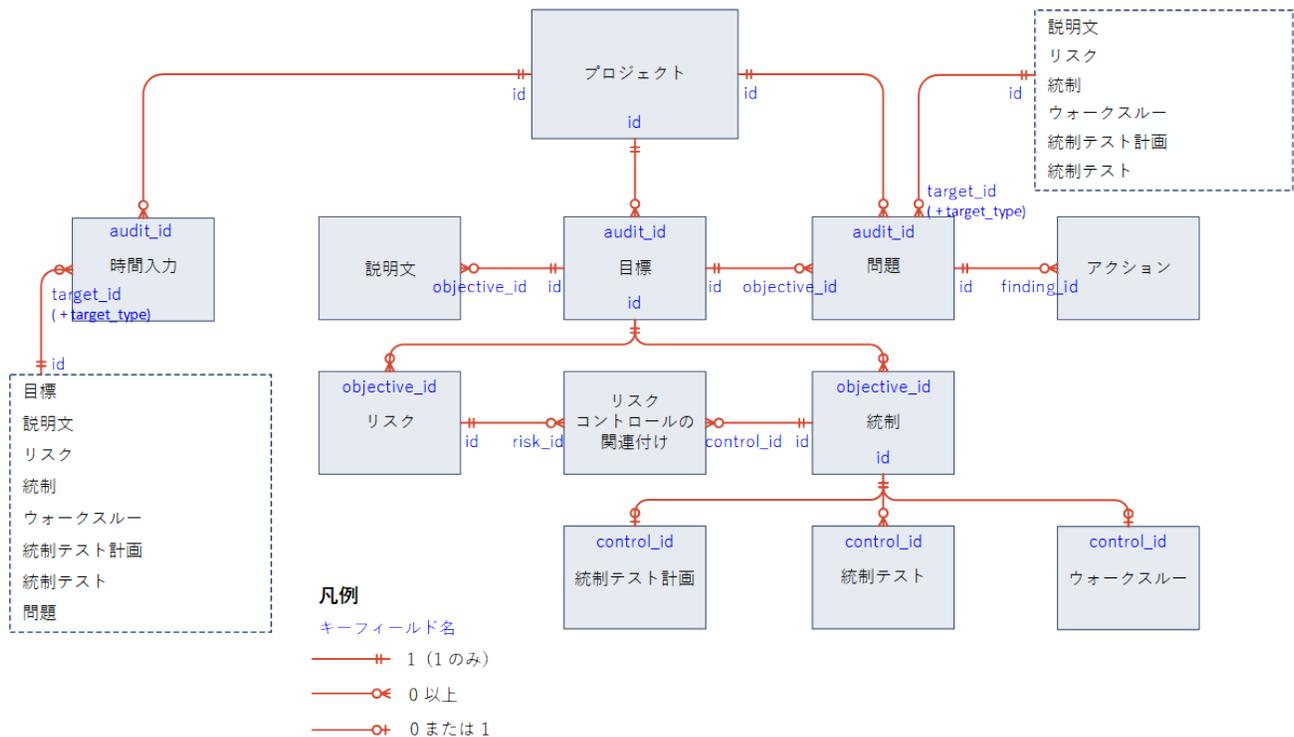
以下の図は、プロジェクトのさまざまなテーブルが関連付けられる方法を示し、インポートが完了した後にテーブルを結合するために使用できる共通キーフィールドを提供します。

問題テーブルおよび時間入力テーブルの `target_type` フィールドはキーフィールドではありません。このフィールドは、キーフィールドではなく、個々の問題または時間入力に関連付ける特定のテーブルの種類を指定するものです。`target_type` フィールドを使用して、結合をフィルタリングし、関心があるテーブル関係のみが含まれます。

ヒント

Analytics で結合を作成するときには、`id` を主キーにし、`<テーブル>_id` を副キーにします。

すべての副レコードまたは多対多結合を含む結合タイプを使用します。詳細については、「テーブルの結合」ページ 960を参照してください。



パスワード要件

パスワードが必要ではない

Analytics のコピーをアクティベーションするためにオンライン アクティベーションを使用した場合は、HighBond からインポートするパスワードを指定する必要がありません。パスワードは自動的に作成され、コンピューターに保存されたアクティベーション情報に基づいて HighBond に送信されます。

パスワードが必要

次のどちらの状況でも、HighBond からインポートするためにパスワードを指定する必要はありません。

- オフラインアクティベーションを使用して Analytics のコピーをアクティブ化した場合
- スクリプトを使用して HighBond からインポートし、ロボットでスクリプトを実行した場合

必須のパスワード値は、HighBond アクセストークンです。

HighBond アクセストークンを取得

注意

アカウント パスワードのようにアクセストークンを保護してください。HighBond アカウント 固有の情報が含まれます。アクセストークンを共有しないでください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール>HighBond アクセストークン**]を選択します。
- **スクリプト エディター**内を右クリックし、 **挿入 > HighBond トークン**]を選択します。

API アクセストークンを管理します]ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理**]ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する - トークン**]列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する - トークンを作成 > Analytics**]をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー**]をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログ ボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているデータアクセスとパスワード定義方法に応じて、次のいずれかを実行します。

Analytics ユーザーインターフェイス

コピーしたトークンを、HighBond に手動でアクセスする際に表示されるパスワードプロンプトに貼り付けます。

Analytics スクリプト

- **PASSWORD コマンド** - コピーしたトークンをスクリプト実行中に表示されるパスワードプロンプトに貼り付けます。
 - **SET PASSWORD コマンド** - コピーしたトークンを、スクリプト内の SET PASSWORD コマンド構文の適切な位置に貼り付けます。
5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。
- 新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
- 詳細については、[アクセストークンの作成と管理](#)を参照してください。

HighBond リザルト データのインポート

Analytics テーブルを作成するには、HighBond リザルトで操作する権限があるコレクションからデータをインポートします。コレクションの個別の統制テストからデータテーブルまたは解釈をインポートできます。

メモ

リザルトでは、統制テストは「データアナリティクス」と言います。

当該の情報を活用する方法

リザルトからデータを Analytics にインポートする機能により、例外に対して二次的またはフォローアップ分析を実行できます。分析を実行した後、往復処理でデータをリザルトにエクスポートできます。詳細については、「HighBond のリザルト アプリへの例外のエクスポート」ページ 216を参照してください。

リザルト データを往復するときのフィールド名の考慮事項

リザルトと Analytics 間でデータを往復する場合は、リザルト テーブルのすべてのフィールド名がより厳しい Analytics フィールド名要件を満たすことを確認する必要があります。そうでない場合、Analytics とリザルト データが一致しないおそれがあります。

たとえば、リザルト フィールド名の特殊文字は、Analytics にインポートされるときに自動的にアンダースコアに変換されます。これは、フィールド名がリザルトの元の名前と一致しないことを意味します。後から Analytics データをリザルトの元のテーブルにエクスポートする場合は、フィールドが正しく一致しません。

往復するデータでこの問題を回避するには、CSV または Excel ファイルからリザルトにデータをアップロードする前に、以下の Analytics フィールド名要件を満たしていることを確認します。

- 特殊文字またはスペースがない
- 数字で始まらない
- 英数文字またはアンダースコア文字(_)のみを含む

レコード長の制限

Analytics にインポートできる最大レコード長は 32,767 文字または 16,383 Unicode 文字です。

最大値を超えるデータをリザルトからインポートしようとすると、エラーメッセージが表示され、インポートが失敗します。インポートから大きい説明文フィールドを削除し、レコード長を削減してください。

もう一つの方法は、Analytics のコマンドラインで `IMPORT GRCRESULTS` コマンドを使用してインポートを実行することです。このコマンドを使用するとき、`CHARMAX` パラメーターを含めることができます。これを使用すると、インポートするフィールドごとに最大文字数を指定できます。詳細については、「IMPORT GRCRESULTS コマンド」ページ 1900を参照してください。

手順

メモ

HighBond に接続するときには、パスワードを指定する必要がある場合があります。詳細については、「パスワード要件」見開きページを参照してください。

1. [インポート] > [HighBond] > [リザルト] を選択します。
2. [インポートするデータの選択] ダイアログボックスで、組織フォルダーをダブルクリックします。
3. 適切な統制テストに移動します。

メモ

次の場合にデータにアクセスするために必要な権限がありません

- 統制テストを含むコレクションが表示されません
- メッセージ解釈のリストの取得エラーが統制テストにアクセスしようとする则表示されます。

詳細については、[リザルトでのロールと権限の割り当てを参照してください](#)。

アクセス許可の支援が必要な場合は、社内の HighBond アカウント管理者、またはリザルトの管理者にお問い合わせください。

4. Analytics にインポートする次のテーブルのいずれかを選択します。

- 解釈 (存在する場合)
- 監査証跡
- コメント
- 結果テーブル (テーブルフィールド選択フォルダーをダブルクリック)

メモ

一度に Analytics にインポートできるテーブルは 1 つだけです。

ヒント

レコード ID をキーフィールドとして、Analytics で結果テーブル、監査証跡、コメントを結合できます。結合で主テーブルとして結果テーブルを使用します。

5. 結果テーブルをインポートする場合、次のカテゴリからインポートする個別のフィールドを選択します。

- ユーザーが生成し、システムが生成したワークフロー情報を含む **メタデータ** - フィールド
- **追加** - コレクションと統制テスト名、およびレコード ID

Analytics の結果テーブルを結合する場合、必ずレコード ID を選択します。

- リザルトにインポートされたデータまたはリザルト アンケートの回答を含む **データ** - フィールド
- **すべて選択** - はテーブル全体をインポートします

6. **OK]**をクリックします。
データがリザルト からインポートされます。
7. **データ ファイルを別名で保存]**ダイアログ ボックスで、Analytics データ ファイルのファイル名を入力し、必要であれば、そのファイルを保存する場所を変更して、**保存]**をクリックします。
8. Analytics プロジェクトに追加する Analytics テーブルの名前について、デフォルト の名前をそのまま用いるか、または別の名前を入力して、**OK]**をクリックします。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

パスワード要件

パスワードが必要ではない

Analytics のコピーをアクティベーションするためにオンライン アクティベーションを使用した場合は、HighBond からインポートするパスワードを指定する必要がありません。パスワードは自動的に作成され、コンピューターに保存されたアクティベーション情報に基づいて HighBond に送信されます。

パスワードが必要

次のどちらの状況でも、HighBond からインポートするためにパスワードを指定する必要はありません。

- オフラインアクティベーションを使用して Analytics のコピーをアクティブ化した場合
- スクリプトを使用して HighBond からインポートし、ロボットでスクリプトを実行した場合

必須のパスワード値は、HighBond アクセストークンです。

HighBond アクセストークンを取得

注意

アカウント パスワードのようにアクセストークンを保護してください。HighBond アカウント 固有の情報が含まれます。アクセストークンを共有しないでください。

1. 次のいずれかを実行します。
 - Analytics メインメニューで **ツール>HighBond アクセストークン]**を選択します。
 - **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン]**を選択します。

API アクセストークンを管理します] ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する** - **トークン** 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - **トークンを作成 > Analytics** をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー** をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているデータアクセスとパスワード定義方法に応じて、次のいずれかを実行します。

Analytics ユーザーインターフェイス

コピーしたトークンを、HighBond に手動でアクセスする際に表示されるパスワードプロンプトに貼り付けます。

Analytics スクリプト

- **PASSWORD コマンド** - コピーしたトークンをスクリプト実行中に表示されるパスワードプロンプトに貼り付けます。
- **SET PASSWORD コマンド** - コピーしたトークンを、スクリプト内の SET PASSWORD コマンド構文の適切な位置に貼り付けます。

5. Launchpad で、トークンが含まれているダイアログボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[アクセストークンの作成と管理](#)を参照してください。

テーブルレイアウトを使用したデータの構造化

データを Analytics にインポートするか、Analytics でデータに接続するときには、データを読み取り、分析を実行するために、未加工データを構造化する必要があります。Analytics の構造の仕組みは**テーブルレイアウト**と呼ばれます。Analytics は、Analytics プロジェクトで、すべてのテーブルのテーブルレイアウトを自動的に生成します。

テーブルレイアウトは、ソース データ ファイルまたはソース ロケーションの未加工データを記述および定義するメタデータのコレクションです。テーブルレイアウトがないと、ソース データは、識別されない大量の数字と文字のように表示されます。テーブルレイアウトには順序があります。

テーブルレイアウトのメタデータ要素

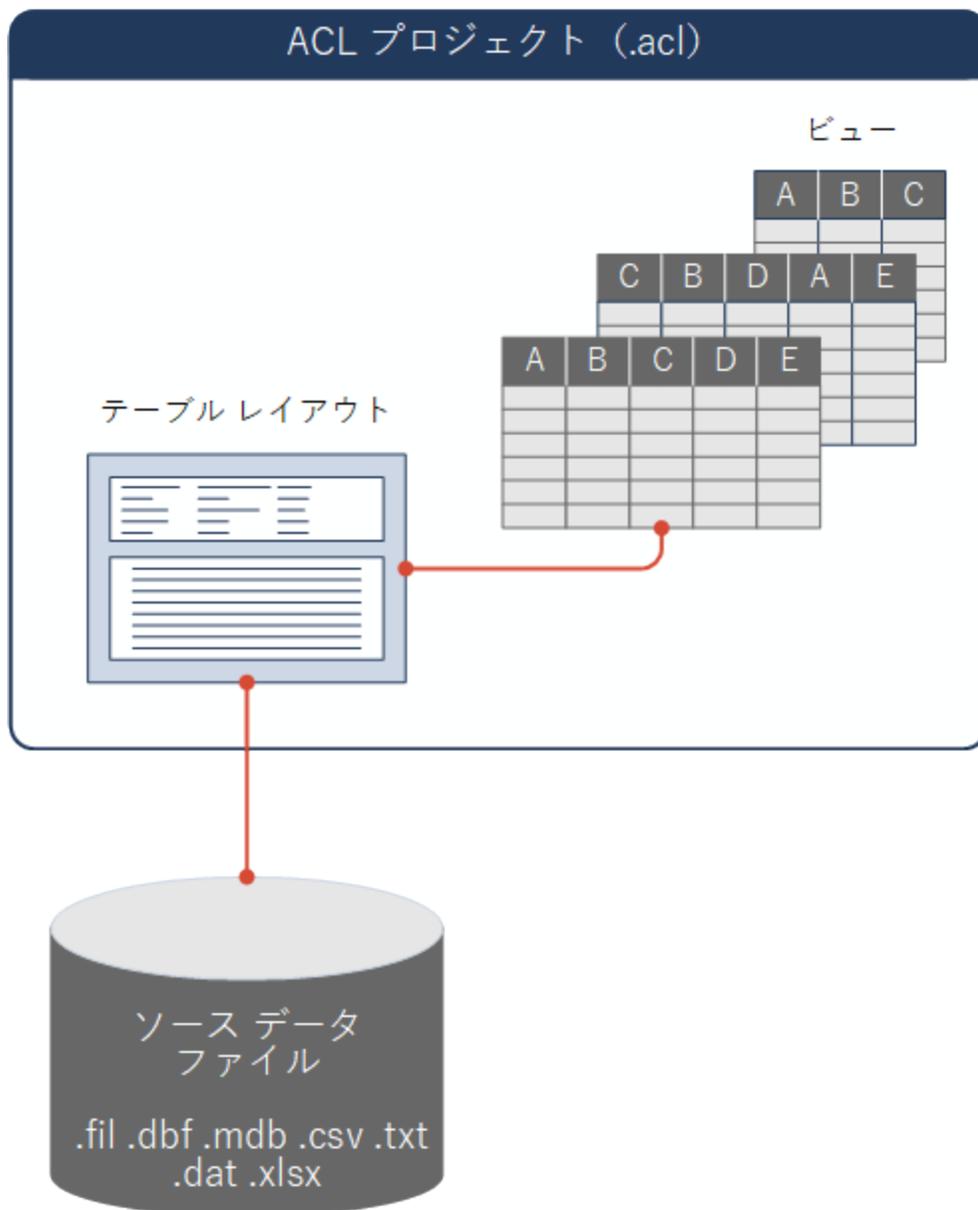
テーブルレイアウトは、次を含む、多数のメタデータを指定します。

- **名前** - フィールド名 - 物理フィールド名と、ビューの対応する列の表示名
- **型** - フィールドのデータ カテゴリとデータ型
- **物理特性** - フィールドの開始位置とバイト長
- **形式** - 数値フィールドの小数点桁数などの形式の詳細と、日付時刻フィールドの日付形式
- **接続** - ソース データの場所

テーブルレイアウトと Analytics テーブルに関連付ける方法

以下の図は Analytics テーブルの他のコンポーネントに関連したテーブルレイアウトを示しています。

テーブルレイアウトが一般にどのように Analytics テーブルに関連しているかについての情報は、"Analytics テーブルの構造" ページ 118を参照してください。



テーブルレイアウトとソース データ ファイル

Metaphor_Trans_All テーブルのテーブルレイアウト (ACL_Demo.ac1 サンプルデータプロジェクト) は以下に表示され、その後に関連付けられたソース データ ファイル Metaphor_Trans_All.fil が表示されます。

テーブルレイアウトがソース データを編成し、多数のフィールドを定義する方法を明確に確認できます。

テーブルレイアウト

Metaphor_Trans_All [C:\ACL_DATA\Sample Data Files\Metaphor_Trans_All.fil]

テーブルレイアウト オプション フィールド/式の編集 新しいデータ フィルター の追加

すべてのフィールド

名前	見出し	開始	カテゴリ	長さ	小数点以下の桁数	データ型	if テスト	静的
CARDNUM	CARDNUM	1	C	36	0	ASCII		
CREDLIM	CREDLIM	37	N	12	0	NUMERIC		
CUSTNO	CUSTNO	49	C	6	0	ASCII		
EXPDT	EXPDT	55	D	10	0	DATE		
FINCHG	FINCHG	65	N	10	2	NUMERIC		
MINPYMTDUE	MINPYMTDUE	75	N	10	2	NUMERIC		
NEWBAL	NEWBAL	85	N	10	2	NUMERIC		
PASTDUEAMT	PASTDUEAMT	95	N	15	2	NUMERIC		
PMTDUEEDT	PMTDUEEDT	110	D	10	0	DATE		
PREVBAL	PREVBAL	120	N	12	2	NUMERIC		
RATE	RATE	132	N	9	2	NUMERIC		
STMTDT	STMTDT	141	D	10	0	DATE		

ASCII |10... |20... |30... |40... |50... |60... |70... |80... |90... |100... |110... |

1	8590122497663807					0	10.00	37.23	0.0004/2	
2	8590-1222-8196- 4011					800081246503/01/2004	14.68	30.00	929.79	10.0005/0
3	8590120784984566					690005159310/01/2004	113.20	129.00	6408.12	0.0005/0
4	8590124253621744					720025040204/01/2004	101.41	0.00	5891.59	0.0005/1
5	8590125999743363					900077808802/01/2004	0.00	0.00	384.95	0.0005/1
6	8590120716753180					800077808801/01/2005	0.50	20.79	85.20	10.7905/0
7	8590128947747852					1000025040206/01/2004	0.00	1.31	1.31	0.0005/1
8	8590122720558982					570005159301/01/2005	0.00	10.00	32.35	0.0005/1
9	8590128676326319					730077808809/01/2005	0.00	0.00	0.00	0.0004/3
10	8590124781270125					800077808803/01/2004	14.46	0.00	1280.69	0.0005/0
11	8590121762084715					600005159301/01/2005	0.00	0.00	-0.02	0.0005/2
12	8590129593164703					900005159309/01/2005	0.00	10.00	19.95	0.0005/2
13	8590127307204051					900087803510/01/2004	0.00	10.00	455.12	0.0005/0

16 進

ソース データ ファイル

テーブルレイアウトのダイアログ ボックス

テーブルレイアウト ダイアログ ボックス(上記)を使用すると、テーブルレイアウトを定義および編集することができます。このダイアログ ボックスは、**編集 > テーブルレイアウト**]を選択すれば、いつでも利用できます。このダイアログ ボックスには3つのタブが含まれており、ここでテーブルレイアウトに対するさまざまな作業が行えます。

タブ	説明
テーブルレイアウト オプション	このタブでは、レコード長やテーブルレイアウトに関連付けられているデータソースなど、テーブルレイアウトの一般的なプロパティを設定したり、そのテーブルレイアウトに関する情報をノートとして追加したりします。
フィールド/式の編集	このタブでは、テーブルレイアウトに対してフィールドの作成、編集または削除を行います。物理データフィールドと演算フィールドのどちらに対しても作業できます。
新しいデータフィルターの追加	このタブでは、データフィルターを定義します。このフィルターは、データソースから含めるデータ、または定義しているレコードから除外するデータを指定する規則です。 テーブルレイアウトのデータフィルターは、Analytics のビューでのフィルターとは異なり、通常は データ定義ウィザード で使用可能なオプションを使ってデータソースを定義できなかった場合にのみ必要です。

テーブルレイアウトを使った作業

Analyticsによって自動的に生成されるテーブルレイアウトが必要なすべての情報を備えている場合、修正せずに使用できます。ただし、テーブルレイアウトの編集が必要になる場合があります。

名前の変更、コピー、インポート、エクスポートなど、テーブルレイアウトに関連する他の処理を実行することもできます。

テーブルレイアウトの編集

場合によっては自動的に作成されたテーブルレイアウトを編集するか、あるいは手作業でテーブルレイアウトを作成する必要があるかもしれません。テーブルレイアウトのフィールドやレコードは、必要に応じて追加、削除または変更することができます。

テーブルレイアウトを編集する理由

次の理由でテーブルレイアウトを編集できます。

- **フィールドの追加または削除** -データフィールド、演算フィールド、またはデータフィルターを追加または削除する場合。
- **デフォルト フィールド 定義の修正** -Analytics が作成したフィールド定義で、データ型、数値または日付時刻フィールドの書式、またはその他のフィールド プロパティを変更する場合。
- **重複フィールドの定義** -重複フィールドを定義する場合。

重複フィールドは、レコード内の1バイト以上のバイト位置を共有します。たとえば、レコード内のあるフィールドの先頭から6バイトを日付フィールド (YYMMDD など) として定義し、その月を表す2バイト部分 (MM など) を月フィールドとして定義できます。2つのフィールドは重複します。

- **エラーの修正** -**データ定義ウィザード**でテーブルを定義した後に判明したエラーを修正する場合。
たとえば、データの表示が不適切な場合は、レコード長やスキップする長さを変更する必要があります。
- **テーブルレイアウトを手動で作成する** - **[オプション] ダイアログ ボックスの [フラット ファイルを手動で定義する]** オプションを選択した場合。

このオプションによって、Analytics のデフォルトの動作が変わります。データソースとしてフラット ファイルを選択して、**データ定義ウィザード**でその他の基本的な情報を提供すると、**テーブルレイアウト** ダイアログ ボックスが表示されるので、Analytics テーブル用にレコードとフィールドを手動で定義する必要があります。

テーブルレイアウトの名前を変更する

テーブルレイアウトの名前を変更することができます。テーブルレイアウトの名前を変更しても、関連するデータファイルの名前は変更されません。テーブルが現在開いている場合は、テーブルレイアウトの名前

を変更することはできません。

メモ

Analytics スクリプトで参照されているテーブルレイアウトの名前を変更する場合は、スクリプト内にあるそのテーブルへの参照もすべて更新する必要があります。そうしないと、そのスクリプトは実行時に失敗します。

方法の説明

1. 名前を変更したいテーブルが開いている場合は、テーブルを閉じます。
2. **総覧**]タブで、テーブルレイアウトを右クリックして **名前の変更**]を選択します。
3. テーブルレイアウトの新しい名前を入力し、**Enter** キーを押します。

メモ

テーブルレイアウトの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

テーブルレイアウトをコピーする

テーブルレイアウトをコピーして、同一のテーブルレイアウトを同じ Analytics データファイルまたはデータソースと関連付け、その後、コピーしたテーブルレイアウトを変更することができます。テーブルレイアウトをコピーして変更する方が、新しいテーブルレイアウトを一から作成するよりも簡単な場合があります。

同じデータファイルまたはデータソースに複数のテーブルレイアウトを関連付けると、同じデータに対して異なるフィールドを定義したり、フィールドのさまざまなサブセットを作成したり、異なる演算フィールドを作成したりすることなどができます。これは、大きなテーブルを操作する場合に役立ちます。すべてのフィールド定義が単一のテーブルレイアウトに含まれていたら、大きなテーブルは扱いにくいでしょう。

注意

同じ Analytics ファイルに複数のテーブルレイアウトを関連付けている場合は注意してください。**オプション**]ダイアログボックスで **テーブルと一緒にデータファイルを削除する**]をオンにした場合は、いずれかのテーブルレイアウトを削除すると、同時にデータファイルも削除されます。つまり、その他のテーブルレイアウトのデータも使用できなくなるということです。

データファイルはすぐに削除されます。Windows のごみ箱には送られません。

方法の説明

1. **総覧**]タブでコピーするテーブルレイアウトを右クリックし、**コピー**]を選択します。
2. プロジェクト フォルダーを右クリックし、**貼り付け**]を選択します。
3. フィールド定義をコピーするか共有するかを尋ねるメッセージが表示されたら、**共有**]をクリックする特別な理由がない限り、**コピー**]をクリックします。

メモ

メッセージは、**オプション]**ダイアログボックスで **テーブルレイアウトを共有しない]** が選択されていない場合に表示されます。詳細については、「テーブルレイアウトの共有」下を参照してください。

テーブルレイアウトがコピーされ、名前の末尾に増分数字が付けられます。たとえば、「Table2」となります。必要に応じて、テーブルレイアウトの名前を変更することができます。

テーブルレイアウトの共有

テーブルレイアウトを共有して、同じテーブルレイアウトと2つ以上の Analytics データ ファイルを関連付けることができます。一般的には、データファイルごとに個別のテーブルレイアウトを保持する必要があります。しかし、1つのテーブルレイアウトを共有すれば、複数のデータファイルに対するテーブルレイアウトの一元管理が可能となり、作業の手間を省くことができます。

詳細については、「テーブルオプション」 ページ 125の**テーブルレイアウトを共有しない**を参照してください。

メモ

テーブルレイアウトを共有するのは、テーブルレイアウトをコピーして、データファイルを共有することと同じではありません。

- テーブルレイアウトを共有する場合は、単一のテーブルレイアウトが2つ以上のデータファイルに関連付けられます。
- データファイルを共有する場合は、2つ以上のテーブルレイアウトが単一のデータファイルに関連付けられます。

方法の説明

1. **オプション]**ダイアログボックスの **テーブル]** タブ(**ツール > オプション > テーブル]**) で、 **テーブルレイアウトを共有しない]** を必ずオフにします。
2. 次のいずれかを実行します。
 - 抽出や並べ替えなど、ソーステーブルと同じレコード構造を持つ新しいデータファイルに結果を出力する、Analytics 操作を実行します。
結果テーブルとソーステーブルは現在、同じテーブルレイアウトを共有しています。
 - テーブルレイアウトをコピーして貼り付け、表示される確認ダイアログボックスで **共有]** をクリックします。
3. テーブルレイアウトをコピーした場合は、以下を実行します。
 - a. 新しいテーブルレイアウトを右クリックして、 **新しいソースデータへのリンク]** をクリックします。
 - b. **ファイルの場所の選択]** ダイアログボックスが表示された場合、データファイルまたはデータソースが存在する場所を選択して **OK]** をクリックします。
ローカルまたはネットワークの場所の場合は **クライアント]** を選択し、Analytics サーバー上の場所の場合は **サーバー]** とサーバープロファイルを選択します。

- c. **[ファイルの選択]**ダイアログボックスで、新たにリンクするデータファイルまたはデータソースを探して選択し、**[開く]**をクリックします。

詳細については、「Analytics テーブルのデータソースの変更」ページ 784を参照してください。

4. 共有テーブルレイアウトの作成が完了したら、**[オプション]**ダイアログボックスで **[テーブルレイアウトを共有しない]**をオンにして、後続の操作で共有テーブルレイアウトを不注意に作成しないようにします。

別の Analytics プロジェクトからテーブルレイアウトをコピーする

ある Analytics プロジェクトから別のプロジェクトへテーブルレイアウトをコピーすることができます。これにより、テーブルレイアウトを一から作成するのではなく、テーブルレイアウトと、それに含まれるフィールド定義を再利用することが可能になります。作業の手間を省いたり、テーブルレイアウトを再利用したり、また他の Analytics ユーザーと共有したりできることに加え、整合性も保証されます。同時に単一のテーブルレイアウト、または複数のテーブルレイアウトをコピーできます。

テーブルレイアウトに特定の Analytics データファイル(.fil)との関連が指定されており、コピー先の Analytics プロジェクトを含んでいるフォルダーに同じ名前のデータファイルが存在する場合には、コピー先フォルダー内のそのデータファイルとコピーしたテーブルレイアウトは自動的に関連付けられます。コピー先フォルダーに同じ名前のデータファイルが存在しない場合は、コピーしたテーブルレイアウトを新しいデータソースにリンクする必要があります。

メモ

コピーしたテーブルレイアウトとそれに関連付けられるデータファイルは一致しなければなりません。つまり、データファイル内のデータの構造は、テーブルレイアウトによって指定されるフィールド定義と一致している必要があります。

データ構造は、データファイルに含まれるデータ要素(フィールド)、フィールド数やフィールドの並び順、各フィールドのデータ型や長さを示します。テーブルレイアウトとデータファイルが一致していない場合は、雑然としたデータになったり、データが欠落する結果となります。

方法の説明

1. コピーされたテーブルレイアウトを格納するプロジェクトを開きます。
2. **ナビゲーターの [総覧]**タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**[別のプロジェクトからコピー > テーブル]**を選択します。

Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。

3. **[プロジェクト ファイルの場所]**ダイアログボックスで、テーブルレイアウトのコピー元となる Analytics プロジェクトを探して選択し、**[開く]**をクリックします。
4. **[インポート]**ダイアログボックスで、次のいずれかの作業を行って、1 つ以上のテーブルレイアウトを **インポート先 <プロジェクト名 >**]リストに追加します。

- テーブルレイアウトをダブルクリックする。
複数のテーブルレイアウトを

- Ctrl キーを押しながらクリックし、右矢印ボタンをクリックする。
- **すべて追加]**をクリックして、すべてのテーブルレイアウトを追加する。

インポート先 < プロジェクト名 >]リストからテーブルレイアウトを削除するには、個々のテーブルレイアウトをダブルクリックするか、複数のテーブルレイアウトを Ctrl キーを押しながらクリックした後に左矢印ボタンをクリックするか、または **すべてクリア]**をクリックします。

5. **[OK]**をクリックして、コピー先のプロジェクトにテーブルレイアウトをコピーします。
プロジェクト内に同じ名前のテーブルレイアウトが既に存在する場合は、コピーしたテーブルレイアウト名の末尾に増分数字が付けられます。
6. コピーしたテーブルレイアウトを新しいデータソースにリンクする必要がある場合は、"Analytics テーブルのデータソースの変更" ページ 784を参照してください。

テーブルレイアウトをインポートする

個別の .layout ファイルとして Analytics プロジェクトの外部に存在するテーブルレイアウトをインポートすることができます。これにより、テーブルレイアウトを一から作成するのではなく、テーブルレイアウトと、それに含まれるフィールド定義を再利用することが可能になります。作業の手間を省いたり、テーブルレイアウトを再利用したり、また他の Analytics ユーザーと共有したりできることに加え、整合性も保証されます。一度にインポートできるテーブルレイアウトは 1 つだけです。

テーブルレイアウトに特定の Analytics データファイル (.fil) との関連が指定されており、プロジェクトを含んでいるフォルダーに同じ名前のデータファイルが存在する場合には、フォルダー内のそのデータファイルとインポートしたテーブルレイアウトは自動的に関連付けられます。プロジェクト フォルダーに同じ名前のデータファイルが存在しない場合は、インポートしたテーブルレイアウトを新しいデータソースにリンクする必要があります。

メモ

インポートしたテーブルレイアウトとそれに関連付けられるデータファイルは一致しなければなりません。つまり、データファイル内のデータの構造は、テーブルレイアウトによって指定されるフィールド定義と一致している必要があります。

データ構造は、データファイルに含まれるデータ要素(フィールド)、フィールド数やフィールドの並び順、各フィールドのデータ型や長さを示します。テーブルレイアウトとデータファイルが一致していない場合は、雑然としたデータになったり、データが欠落する結果となります。

方法の説明

1. **ナビゲーターの 総覧]**タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**プロジェクト項目のインポート > テーブル]**を選択します。
Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。
2. **プロジェクト]**ダイアログボックスで、テーブルレイアウト ファイル(.layout) を探して選択し、**開く]**をクリックします。

3. 確認ダイアログボックスで、**OK**]をクリックします。

プロジェクトにテーブルレイアウトがインポートされます。プロジェクト内に同じ名前のテーブルレイアウトが既に存在する場合は、インポートしたテーブルレイアウト名の末尾に増分数字が付けられません。

4. インポートしたテーブルレイアウトを新しいデータソースにリンクする必要がある場合は、"Analytics テーブルのデータソースの変更" ページ 784を参照してください。

テーブルレイアウトをエクスポートする

テーブルレイアウトは、Analytics プロジェクトの外部に保存される個別の .layout ファイルとしてエクスポートすることができます。個別のファイルとしてエクスポートされたテーブルレイアウトは、後で任意の Analytics プロジェクトにインポートすることができます。これにより、テーブルレイアウトを一から作成するのではなく、テーブルレイアウトと、それに含まれるフィールド定義を再利用することが可能になります。一度にエクスポートできるテーブルレイアウトは1つだけです。

方法の説明

1. ナビゲーターの **総覧**] タブでテーブルレイアウトを右クリックし、**プロジェクト項目のエクスポート**] を選択します。
2. **名前を付けて保存**] ダイアログボックスで、テーブルレイアウトを保存する場所を選択し、必要であればテーブルレイアウトの名前を変更して、**保存**] をクリックします。確認ダイアログボックスで **OK**] をクリックします。

指定した場所にテーブルレイアウトがエクスポートされます。

メモ

テーブルレイアウト名を64文字の英数字(.layout 拡張子を含まない)に制限し、テーブルレイアウトが Analytics にインポートされるときに名前が切り捨てられないようにします。

名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。先頭に数字を使用することもできません。特殊文字、スペース、先頭の数字はすべて、テーブルレイアウトのインポート時にアンダースコア文字に置換されます。

テーブルレイアウトを削除する

Analytics データファイルまたはデータソースに関連付けられたテーブルレイアウトは、必要であればいつでも削除することができます。テーブルレイアウトを削除すると、関連するビューや、関連するインデックス、テーブルの関連付けも同様に削除されます。

注意

テーブルレイアウトを削除する場合は注意してください。[オプション]ダイアログボックスで **テーブルと一緒にデータファイルを削除する** をオンにした場合は、テーブルレイアウトを削除すると、同時にデータファイルも削除されます。つまり、データを使用できなくなるということです。

テーブルレイアウトと一緒に関連するデータファイルが削除される場合は、**削除の確認** ダイアログボックスが警告を發します。

データファイルはすぐに削除されます。Windows のごみ箱には送られません。

方法の説明

1. **総覧** タブで削除するテーブルレイアウトを右クリックし、**削除** を選択します。
2. 確認ダイアログボックスで、**削除** をクリックします。

テーブルレイアウトのプロパティの設定

テーブルレイアウト ダイアログ ボックスの [テーブルレイアウト オプション] では、テーブルレイアウトに関する基本的なプロパティの多くを変更することができます。選択したテーブルレイアウトに適用されないプロパティやオプションは無効になっています。これは、そのプロパティやオプションが、テーブルレイアウトに関連付けられているデータソースに対応しないものだからです。

[テーブルレイアウト] ダイアログ ボックスのデータ プレビュー領域は、行う変更によって動的に更新されません。

テーブルレイアウトのプロパティを設定するには

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **テーブルレイアウト オプション**]タブをクリックします。

メディアの種類]ドロップダウンリストで、データの物理的なアクセス方法を指定します。通常、この値には "ディスク" が設定され、変更することはできません。

3. **ファイルの種類**]ドロップダウン リストには、レコードをデータソースに保存する方法を指定します。以下のオプションから選択することができます。
 - **固定レコード長** - 各レコードの長さが同じで、各フィールドの位置も全レコードで同一であるデータファイルの場合には、このオプションを選択します。
 - **IBM 可変レコード長** - レコードの長さが可変長のデータファイルの場合には、このオプションを選択します。
 - **テキスト ファイル (CR または CRLF)** - 各レコードの終わりを、キャリッジ リターン (CR) またはキャリッジ リターン/ライン フィード シーケンス (CRLF) で指定されたテキスト ファイルの場合には、このオプションを選択します。
4. データ ファイルの表示に使用する文字セットを変更する場合は、**文字型**]ドロップダウン リストから適切なオプションを選択できます。

ASCII、EBCDIC または Unicode (Unicode 版 Analytics を使用している場合) を選択することができます。選択されている文字セットが、データ ファイルのエンコードと一致しない場合は、データ プレビュー領域のデータが文字化けします。

文字セットは、データ プレビュー領域の左上隅の文字セット トグルスイッチをクリックして変更することもできます。

データ プレビュー領域の左下隅の **16 進**]チェック ボックスをオンにすると、16 進数の形式でデータを表示できます。このオプションは、印刷できない文字や圧縮データ (IBM メインフレーム コンピューターから作成されるパック化された数値データなど) を処理する場合に便利です。ただし、**レコード長**]や **スキップする長さ**]の値を変更する必要があります。

5. レコード長を変更する場合は、**レコード長**]テキスト ボックスの数値を大きくしたり小さくしたりして調整します。

レコード長は、固定長ファイルの各レコードの長さ、または可変長レコードの最長レコードの長さを指します。フィールドの値が右にずれている場合は、おそらくレコード長の値を増やす必要があります。フィールドの値が左にずれている場合は、おそらくレコード長の値を減らす必要があります。

6. ファイルの先頭からデータを除外する場合は、**[スキップする長さ]**テキスト ボックスに値を指定します。

[スキップする長さ]の値は、ファイルの先頭部分でスキップするバイト数を指定します。デフォルト値のゼロは、ファイル全体が処理されることを意味します。データをスキップするには、ゼロより大きい値を入力します。たとえば、先頭から32バイトにヘッダー情報のみが含まれている場合、この情報を除外するためには、値として32を指定します。

7. テーブルレイアウトに関するノートを追加する場合は、**テーブルレイアウト ノートの編集**] をクリックしてノート テキストを入力し、**閉じる**] をクリックします。
8. データプレビュー領域にデータが不正に表示される場合は、**ファイルの分析**] をクリックして**データファイルのプロパティを自動的に調べさせることができます。**
9. データソースが区切り文字付きテキスト ファイルで、そのデータファイルがデータプレビュー領域で不正に表示される(またはデータがなにも表示されない)場合は、フィールド区切りや区切り文字を手作業で指定する必要があります。
 - a. **区切り文字付きファイルの変換**] をクリックします。
 - b. **カスタム区切り文字**ダイアログボックスの**フィールドの区切り文字**および**文字列の区切り文字**に必要な値を入力し、**OK**] をクリックします。
 - c. 新しいデータファイル(.fil)の名前を入力して**保存**] をクリックします。
10. **閉じる**] をクリックして、**テーブルレイアウト** ダイアログボックスを閉じ、変更を保存します。

テーブルレイアウトのプロパティの表示

以下の手順を使用して、特定のテーブルレイアウトのプロパティを表示または管理することができます。データの表示に関連するテーブルレイアウトのプロパティを表示または管理するには、「テーブルレイアウトのプロパティの設定」ページ 779を参照してください。

テーブルレイアウトのプロパティを表示するには

1. ナビゲーターの **総覧**] タブで、Analytics テーブルを右クリックします。
2. **プロパティ**] を選択します。
3. **テーブルプロパティ**] ダイアログ ボックスで、次のタブをクリックして、テーブルプロパティを表示したり変更したりします。
 - **全般** - テーブルレイアウトに関連付けられている Analytics データ ファイル (.fil) またはデータソースの基本プロパティを表示します。ファイルの名前、場所、最終更新日時、および物理サイズが表示されます。

ファイルの場所を開く] をクリックすると、Analytics データ ファイルまたはデータソースが入っているフォルダーに直接移動することができます。
 - **ノート** - テーブルレイアウトに関連付けられているすべての注釈を表示します。既存のノートを変更したり、新しいノートを追加したりすることができます。詳細については、「テーブルレイアウトのノートの追加と編集」ページ 183を参照してください。
 - **ビュー** - テーブルレイアウト内のすべてのビューを表示します。このタブからテーブルレイアウト内のビューを管理できます。詳細については、「ビューでの作業」ページ 841を参照してください。
 - **インデックス** - テーブルレイアウトに関連付けられているすべてのインデックスを表示します。既存のインデックスを変更したり、新しいインデックスを追加したりすることができます。詳細については、「インデックスの管理」ページ 1203を参照してください。

メモ

ビューやインデックスに変更を加えるには、テーブルを開いておく必要があります。

4. 行った変更を保存してダイアログ ボックスを閉じるには、**OK**] をクリックします。

Analytics テーブルのデータの更新

データソースによっては、Analytics テーブルは、テーブルレイアウトを再定義しなくても、テーブルのデータソースの現在の内容で更新することができます。データソースのデータの構造に変更がない限り、Analytics テーブルは必要に応じていつでも更新できます。

更新がサポートされるデータソース

以下のデータソースを持つ Analytics テーブルは、[ソースファイルから更新] オプションを使用して更新することができます。

- ファイルベースのデータソース:
 - 区切り文字付きテキスト
 - Microsoft Access
 - Microsoft Excel
 - Adobe Acrobat (PDF)
 - 印刷イメージ(レポート)
 - SAP プライベート ファイル形式/DART
 - XML
 - XBRL
- ODBC 準拠のデータソース
 - データアクセス ウィンドウまたは `IMPORT ODBC` コマンドを使用して接続する任意のファイルまたはデータベース

メモ

ファイルのアクセスにパスワードが必要な場合は、ファイルベースのデータソースから Analytics テーブルを更新することはできません。1 つの例外は、PDF ファイルからの更新です。

データベースからの更新はパスワードの使用をサポートしません。

古い Excel ファイル(*.xls) や Microsoft Access ファイル(*.mdb) を扱うためには [ソースファイルから更新] オプションを使用するには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、"Microsoft Access データベースエンジンを除外するオプション" ページ 2664を参照してください。

更新の仕組み

ソースから更新を使用して、テーブルを更新すると、テーブルを最初に定義するために使用されるコマンドが再実行され、データソース内のすべての新しいレコードまたは変更が Analytics テーブルに追加されま

す。データソースのデータの構造に変更がなければ、Analytics テーブルレイアウトのフィールド定義が適用されます。

データソースの新しいフィールド値のいずれかが Analytics テーブルレイアウトで指定されたフィールド長よりも長い場合、その値は Analytics テーブルで切り捨てられます。完全な値を取得するには、ソースから更新を使用するのではなく、もう一度 Analytics テーブルを定義する必要があります。

ナビゲーターの [総覧] タブからテーブルを更新する

1. ソースデータを含んでいるファイルが閉じていることを確認します。
2. 更新する Analytics テーブルを右クリックし、[ソースファイルから更新] を選択します。
3. 確認ダイアログボックスで、[はい] をクリックします。

テーブルレイアウト ダイアログボックスを使用してテーブルを更新する

1. ソースデータを含んでいるファイルが閉じていることを確認します。
2. 更新する Analytics テーブルを開きます。
3. [編集 > テーブルレイアウト] の順にクリックします。
4. [テーブルレイアウト オプション] タブをクリックします。
5. [ソースファイルから更新 ] をクリックします。
6. 確認ダイアログボックスで、[はい] をクリックします。

Analytics テーブルのデータソースの変更

複数のソースデータファイルの構造が同一である場合は、すべてのファイルで同じ Analytics テーブルレイアウトを使用することで、手間を省くことができます。

機能の仕組み

1月の請求書データを含むソースデータファイルから Analytics テーブルを定義するとします。翌月の2月のデータを含むソースファイルの構造は同一です。その後の月でも同じことが当てはまります。毎月テーブルレイアウトを再作成するのではなく、同じテーブルレイアウトを再利用し続けることができます。

2月の請求書データを入手したら、1月のテーブルレイアウトをその新しいデータファイルに関連付け直すか、1月のテーブルをコピーしてリンクを変更することで、両方の月のテーブルを保持することができます。

サポートされているデータソース

ソースが区切り文字付きテキストファイル、行区切りのテキストファイル、または Analytics ソースデータファイル(.fil)の場合、テーブルレイアウトを新しいデータソースへリンクできます。

同じデータ構造の要件

テーブルレイアウトと、それに関連付けるデータファイルは、対応している必要があります。つまり、データファイル内のデータの構造は、テーブルレイアウトによって規定されるフィールド定義に対応している必要があります。

データ構造は、データファイルに含まれるデータ要素(フィールド)、フィールド数やフィールドの並び順、各フィールドのデータ型や長さを示します。テーブルレイアウトとデータファイルが一致していない場合は、雑然としたデータになったり、データが欠落する結果となります。

ナビゲーターの 概要] タブからテーブルデータソースを更新する

1. ナビゲーターで、更新するテーブルを右クリックし、**新しいソースデータへのリンク**を選択します。
2. **ファイルの場所の選択**ダイアログボックスが表示された場合、データソースが存在する場所を選択して **OK** をクリックします。

ローカルまたはネットワークの場所の場合は **クライアント**]を選択し、Analytics サーバー上の場所の場合は **サーバー**]とサーバープロファイルを選択します。

3. **ファイルの選択**]ダイアログボックスで、新たにリンクするデータソースを探して選択し、**開く**]をクリックします。

テーブルレイアウト]ダイアログボックスを使用して、テーブルデータソースを変更する

1. ナビゲーターで、更新するテーブルを選択し、**編集 > テーブルレイアウト**]を選択します。
2. **テーブルレイアウト オプション**]タブをクリックします。
3. **新しいソース データへのリンク** ]をクリックします。
4. **ファイルの場所の選択**]ダイアログボックスが表示された場合、データソースが存在する場所を選択して **OK**]をクリックします。

ローカルまたはネットワークの場所の場合は **クライアント**]を選択し、Analytics サーバー上の場所の場合は **サーバー**]とサーバープロファイルを選択します。

5. **ファイルの選択**]ダイアログボックスで、新たにリンクするデータソースを探して選択し、**開く**]をクリックします。

テーブルレイアウトのフィールドの定義

Analytics テーブルレイアウトで、フィールドは、他のデータ単位とともにレコードを形成する、従業員 ID などのデータの単一の単位です。

Analytics テーブルレイアウトでは、次の2種類のフィールドタイプを定義することができます。

- 物理フィールド
- 演算フィールド

Analytics の各フィールドには、物理または演算フィールドの値がどのように処理され表示されるかを決定するデータ型(文字、数値、日付時刻、論理)を割り当てる必要があります。

物理フィールド

物理フィールドは、ファイルまたはデータベースなどのデータソースに物理的に存在する実際のデータに対応します。たとえば、物理フィールド **Amount** には、\$88.50、\$123.00 などの売上金額を含めることができます。

テーブルレイアウトで、レコードはそのデータがデータソースのどこに位置するかを定義し、物理フィールドはレコード内におけるフィールドデータの位置を指定します。

Analytics テーブルを開くには、そのテーブルレイアウトに少なくとも1つの物理フィールドが定義されていなければなりません。通常、データ定義ウィザードまたはデータアクセスウィンドウを使用してテーブルを定義およびインポートした場合には、テーブルレイアウトの物理フィールドは Analytics によって自動的に定義されます。必要であれば、**テーブルレイアウト** ダイアログボックスで物理フィールドを手動で定義することができます。

詳細については、「物理フィールドの定義」ページ 788を参照してください。

演算フィールド

演算フィールドは、Analytics 式を使用される「仮想フィールド」です。これにより、データソースに物理的に存在しない値を演算または計算できます。たとえば、演算フィールド **Total Amount** を作成し、式 `Amount * 1.05` を使用して、5%の売上税を含む合計金額を計算させることができます。

Amount フィールド(物理)	Analytics 式	Total Amount フィールド(演算)
\$88.50	<code>金額 * 1.05</code>	\$92.93
\$123.00	<code>金額 * 1.05</code>	\$129.15

演算フィールドは直接物理フィールドに対応しませんが、多くの場合、上記の例の **Amount** フィールドなどの1つ以上の物理フィールドを参照します。演算フィールド式は他の演算フィールドを参照したり、入力パラメーターとしてフィールドが必要ではない関数を含む場合があります。

演算フィールドは **テーブルレイアウト** ダイアログボックスで定義できます。

詳細については、"演算フィールドの定義" ページ 795を参照してください。

物理フィールドの定義

Analytics テーブルレイアウトでは、データソースの実際の物理データに対応するフィールドは、**物理フィールド**と呼ばれます。物理フィールドを作成するプロセスを**定義**といいます。

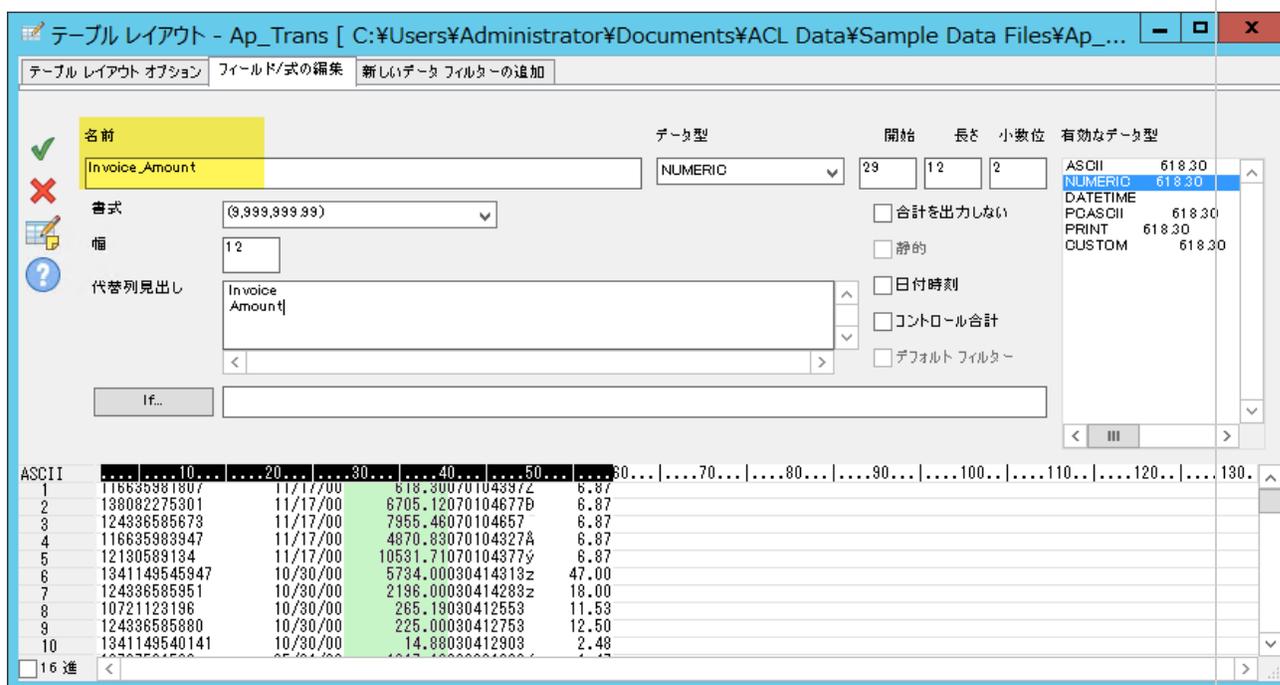
場合によっては**フィールド定義**と呼ばれる物理フィールドは、次のようなメタデータ情報を指定して、未加工フィールド データを構成します。

- フィールドの名前
- レコードのフィールドの開始位置
- フィールドの長さ
- フィールドのデータ型。Analytics がフィールドに保存されているデータを読み取って処理する方法を決定します。

さらに、データ型によって、またデフォルト値を無効にするために提供する設定のために、追加情報を指定する必要もあります。たとえば、数値フィールドに適用する書式設定、またはビューやレポートで使用する列見出しは、空白にしてデフォルト値を割り当てたり、あるいは使用する値を指定することもできます。

物理フィールド定義の例

以下の例は、**テーブルレイアウト**]ダイアログボックスの **Invoice_Amount** フィールドの定義を示します。データプレビュー領域で、フィールドに含まれる実際の物理データは、緑色でハイライト表示されます。



メタデータ要素	説明	値
名前	物理フィールド名	Invoice_Amount
型	データ型	数値
開始位置	フィールドの開始位置	バイト位置 29
長さ	フィールド長	12 バイト
小数点以下の桁数	小数点以下桁数	2
有効なデータ型	データ型候補のクリック可能なリスト	数値 フィールドの最初の値のプレビューを含みます
書式	数値の書式	(9,999,999.99) <ul style="list-style-type: none"> サポートされる数値 0 ~ 9 千の位の桁区切りはカンマです 小数点区切りはピリオドです 負数は括弧で表記されます
幅	ビューとレポートのフィールド表示幅	12 文字
代替列見出し	ビューとレポートのフィールド表示名	請求金額 (2 行)

物理フィールドを定義する

データソースのフィールドの中から、Analytics テーブルレイアウトに追加するフィールドを物理フィールドとして個別に定義する必要があります。

多くの場合、データ定義ウィザードまたはデータアクセス ウィンドウを使用して、データを定義およびインポートするときに、必須の物理フィールドが定義されます。ただし、追加のフィールドを手動で定義するか、テーブルレイアウトで手動ですべてのフィールドを定義できます。

方法の説明

各フィールドの開始位置と長さの指定

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブで、**新しいデータフィールドの追加**  をクリックします。
3. 次のいずれかを実行し、フィールド開始位置と長さを指定します。
 - **クリックしてドラッグ** -データプレビュー領域で、グリッド内の任意のデータ行でフィールド対象部分を強調表示(クリックしてドラッグ)させることができます。
 - **手動で指定 - 開始**]および **長さ**]テキストボックスで、フィールド開始位置と長さを**バイト単位**で手動により指定できます。
開始]および **長さ**]の値を手動で指定する場合は、次のガイドラインに従います。

Analytics のエディション	バイト数と文字数
非 Unicode Analytics	1 バイト = 1 文字
Unicode Analytics、拡張 ASCII(ANSI) データ	1 バイト = 1 文字
Unicode Analytics、Unicode データ	2 バイト = 1 文字

メモ

Unicode データの場合：

- **開始** -]では、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。
- **長さ** -]では、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。

フィールド メタデータの指定

1. **名前**]テキスト ボックスにフィールド名を入力します。

メモ

フィールド名は256文字までの小文字の英字に制限されます。名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

Analytics には多くの予約キーワードがあり、フィールド名にこのキーワードを使用することはできません。詳細については、"予約キーワード" ページ 1421を参照してください。

2. **データ型**]ドロップダウンリストで適切なデータ型を選択または確認します。

指定するデータ型は、ソースデータのデータ型と同じ、あるいはそのデータの使用方法に適したデータ型である必要があります。たとえば、あるフィールドがデータソースでは数値データであっても、Analytics ではそのフィールドを文字データとして定義したいこともあります。

有効なデータ型]の下で、クリック可能なリストには、指定した物理データ型と一致するデータ型が表示されます。データ型は適用可能性が最も高い順に表示されます。一般的なデータ型の後にはシステムやアプリケーション固有のデータ型も表示されます。

3. (オプション) **幅**]テキスト ボックスにフィールドの幅を文字で指定します。

Analytics ビューおよびレポートにフィールドを表示する場合、幅値は列サイズとして使用されます。

4. (オプション) [**代替列見出し**]テキスト ボックスには、表示名を指定します。

Analytics ビューおよびレポートにフィールドを表示する場合、表示名はフィールド名の代わりに列見出しとして使用されます。この値が指定されていない場合は、フィールド名が使用されます。

5. (省略可能) 演算フィールドによって評価されるレコードを制限したい場合は、**[f]**テキスト ボックスで条件を指定します。直接入力するか、または **[f]** ボタンをクリックして、**式ビルダー**を利用してIF ステートメントを作成します。

IF 文によって除外されるレコードは、演算フィールドによって評価されません。たとえば、IF 文 `Invoice_Amount >= 1000` では、\$1000 未満の請求金額のレコードが評価されません。

除外されたレコードの場合、演算フィールド値は、空白、0.00、または False (F) です。これは、演算フィールドのデータ型によって異なります。

6. 選択したデータ型に応じて、以下の設定の値を指定する必要があります。

設定	記述
Dec	この設定は、小数点桁数を指定します。このオプションは、数値 (Numeric) フィールドの場合のみ有効になります。
書式	ビューやレポートにおける数値フィールドの表示書式を制御します。ソースデータの日付時刻フィールドの入力書式も指定します。 このドロップダウンリストは、数値または日付時刻以外のデータ型が選択された場合には無効になります。書式は、ドロップダウンリストから選択する、または直接入力するか、あるいはドロップダウンリストから書式を選択後、その書式を編集することもできます。

設定	記述
	書式]ドロップダウンリストが空白の場合、このフィールドのデータには オプション]ダイアログボックスで指定されたデフォルトの表示形式が適用されます。ここで指定した書式は、デフォルトの書式に優先します。
合計を出力しない	このフィールドの値の合計は算出されなくなります。 Analyticsでは、レポート内の数値フィールドは自動的に集計されます。数値フィールドによっては、単価や口座番号など、合計する必要がない情報が格納されている場合もあります。このオプションは、数値 (Numeric) データ型の場合のみ有効になります。
静的	Analytics が、IF ステートメントを評価するときに使用するデフォルトの動作が変わります。(省略可能な IF ステートメントの詳細については、"フィールド定義を確定する" 下を参照してください。) 静的が選択解除 (デフォルト) -IF ステートメントが False と評価される場合、フィールドのデータカテゴリに応じて、フィールドには空の値 (空白、ゼロ (0)、または False (F)) が割り当てられます。 静的を選択 -IF ステートメントが False と判定された場合、Analytics では空の値ではなく、そのフィールドの最後の有効な値が繰り返し使用されるようになります。IF ステートメントが True と評価され、新しい値が使用されるまで、最後の有効な値が各行で繰り返されます。
日付時刻	数値フィールドが日付時刻フィールドとして解釈されるようになります。 日付時刻]チェックボックスをオンにした場合は、使用する日付時刻書式も 書式]ドロップダウンリストで指定する必要があります。
コントロール合計	フィールドをコントロール合計フィールドとして指定します。 コントロール合計は数値フィールドの値の合計です。これを使用して、データの整合性を確認することができます。新規のテーブルにデータを抽出した場合、または新規のテーブルでデータを並べ替えた場合、コントロール合計フィールドの入力値合計と出力値合計がテーブル履歴に格納されます。入力は元テーブルを指しています。出力は新規テーブルを指しています。2つの合計が一致する場合は、抽出または並べ替え処理で失われたデータはありません。 データ提供者が提供するコントロール合計を Analytics が算出するコントロール合計と比較して、すべてのデータを受け取ったかどうかを確認することができます。 複数のフィールドに対してコントロール合計を指定すると、テーブル履歴は其中で開始位置が最も左にある数値フィールドについてのみ集計を報告します。
デフォルト フィルター	Analytics テーブルを開くたびに、デフォルトのビューのレコードをこのフィールドの値に基づいてフィルターします。 True と判定されたレコードのみが表示され、フィルターは自動的に適用されます。このオプションは、論理データ型の場合のみ有効です。また、各テーブルレイアウトに指定できるデフォルト フィルターは1つのみです。

フィールド定義を確定する

- (省略可能) 含まれる値を制限したい場合は、**[F]**テキスト ボックスで条件を指定します。直接入力するか、または **[F]** ボタンをクリックして、**式ビルダー**を利用して IF ステートメントを作成します。
 - フィールドに含まれる - IF ステートメントを満たす値**

- **フィールドから除外** - IF ステートメントを満たさない値

たとえば、IF ステートメント `Invoice_Amount >= 1000` に \$1000 以上の請求金額が含まれ、\$1000 未満の請求金額が除外されます。

除外された値はフィールドに表示されないか、コマンド処理に含まれません。フィールドのデータカテゴリに応じて、除外された値は、空白、ゼロ(0)、または False(F) として表示されます。IF ステートメントを削除すると、いつでも除外を元に戻すことができます。

2. (オプション) 開いたテーブルビューに新しく定義されたフィールドを自動的に追加しない場合は、**[作成されたフィールドを現在のビューに追加]** の選択を解除します。

オプションを選択したままにした場合、新しいフィールドがテーブルビューに追加されます。フィールドはビューの最後の列として配置されるか、ビューの選択した列の左に移動します。

いつでも手作業でフィールドをビューに追加することができます。詳細については、「ビューに列を追加する」 ページ 849 を参照してください。

3. (省略可能) フィールドに関するノートを追加する場合は、**フィールド ノートの編集**  をクリックしてノート テキストを入力し、**閉じる**  をクリックします。
4. **入力を受け入れる**  をクリックします。
。Analyticsにより、フィールド定義がテーブルレイアウトに追加されます。
5. **閉じる**  をクリックしてテーブルレイアウト ダイアログ ボックスを終了します。

[作成されたフィールドを現在のビューに追加] を選択したままにした場合、関連する列がテーブルビューに追加されます。

日付時刻フィールドの定義

作業対象のデータソースによって、日付時刻情報(日付、日付時刻または時刻)は文字データまたは数値データとして格納されています。日付時刻情報を含むフィールドを手動で定義する場合、Analyticsはこれをデフォルトでは文字データとして扱います。Analyticsで日付時刻情報を正しく読み込むようにするには、データ型として日付時刻を選択し、**書式** ドロップダウン リストからは日付時刻の書式を指定する必要があります。

日付時刻ソース書式

日付時刻のソース書式では、**ソースデータ**内で年、月、日、および時、分、秒を表す文字や数字、また、日付時刻データの各要素を区切るのに使用する文字を識別します。

日付時刻がソースデータに保存される方法を一致させるには、次の操作を行うことができます。

- 既存の日付時刻の選択
- 独自の日付時刻形式の指定
- 既存の形式の選択と修正

たとえば、December 31, 2014 と表される日付が、データソースでは 14-31-12 として格納されている場合、日付時刻書式として `YY-DD-MM` と入力して Analytics が日付の値を正しく解釈できるようにします。

詳細については、「ソース データに含まれる日付および時刻データの書式」 ページ 366を参照してください。

日付時刻表示書式

選択または指定する日付時刻書式が Analytics での日付時刻の表示方法やレポートでの書式設定に影響することはありません。日付時刻の表示形式は **オプション** ダイアログ ボックスの **日付と時刻** タブで指定される **日付の表示書式** 設定や **時刻の表示書式** 設定で決まります。

詳細については、「日付と時刻オプション」 ページ 135を参照してください。

重複フィールドの定義

ほとんどの場合、レコードの物理フィールドを定義するときに、レコードの各バイト位置は、1つのフィールドにのみ割り当てられます。基本的に、テーブルの定義は、レコードの各フィールドの開始位置と長さの定義の問題であり、1つのフィールドは前のフィールドが終了した後に開始します。

しかし、場合によっては、各フィールドが重なるように定義する必要があるかもしれません。その場合、同じバイト位置が複数のフィールドで使用されます。ソースデータの構造が非標準であるか、特定の方法で Analytics でデータを操作する場合に、この状況が発生することがあります。

たとえば、データソースのレコードの先頭 6 バイトを DDMMYY という書式の日付時刻フィールドとして定義した後、月を示す 3 バイト目と 4 バイト目の 2 バイトを数値フィールドとして別定義することができます。この方法によって、日付全体を表すフィールドを経過日数に使用することができ、さらに月を別のフィールドの単独の値として月別合計の作成に使用することができます。

演算フィールドの定義

Analytics テーブルで、実際の物理データではなく式の結果を表示するフィールドは**演算フィールド**といいます。一般的に演算フィールドは、何らかの演算、処理、またはデータ変換を実行します。計算フィールドを作成するプロセスを**定義**といいます。

式の詳細については、「式の使用」ページ 866を参照してください。

演算フィールドを利用する方法

操作する物理データは分析の基盤を提供しますが、多くの場合、物理データから情報を補間するか、計算を実行して、分析を進める必要があります。

物理ソースデータがないと、演算フィールドで、補間し、計算を実行できます。作業している物理データソースに直接存在しない有益なデータを作成できる「仮想フィールド」です。

演算フィールドの使用方法

使用するフィルター	説明
演算結果の表示	在庫ファイルで、演算フィールド Value (値) を作成します。これは、 quantity (数量) フィールドを unit_cost (単価) フィールドで乗算し、在庫品目ごとの合計値を計算します。
物理フィールドのデータ型の変換	数値フィールドを文字データとして扱うために、 STRING() 関数を使用して、数値を文字値に変換する演算フィールドを作成できます。
条件を使用して数値コードをテキスト値に置き換える	物理フィールドの数値の国コードにマッピングして、実際の国名を表示する条件付き演算フィールドを作成します。例：01 ではなく“Canada”、02 ではなく“USA”。
1 つまたは複数の条件を評価し、その結果に基づいてフィールドの値を決定する	販売地域に基づいて商品の消費税を計算する条件付き演算フィールドを作成します。商品が1つの地域で販売される場合、税金は7%で計算されます。商品が別の地域で販売される場合、税金は6%で計算されます。

演算フィールドのデータ カテゴリ

物理フィールドのように、演算フィールドは、次のデータ カテゴリのいずれかに属します。

- 文字
- 数値
- 日付時刻
- 論理

物理フィールドとは異なり、演算フィールドを定義するときには、明示的にデータ型、そしてデータカテゴリを選択しません。代わりに、演算フィールドで指定する**デフォルト値**が、演算フィールドのデータカテゴリを示します。

例

以下の表は、演算フィールドのデフォルト値と関連付けられたデータカテゴリの例です。

演算フィールドのデフォルト値	演算フィールドのデータカテゴリ
"不明なロケーション"	文字
STRING(Employee_number, 10)	
0.00	数値
数量 * 単価	
VALUE(Salary, 0)	
`20180331`	日付時刻
CTOD(Invoice_date, "DD/MM/YYYY")	
T	論理
値 > 1000	

数値演算フィールドの小数点精度の制御

数値の演算フィールドでは、すべての数値演算値の小数精度は、**Default Value** フィールドで指定された式またはリテラル値の精度で統制されます。

- **式** - デフォルト式 `Invoice_Amount * 0.375` を指定し、Invoice Amount フィールドの値の小数点桁数が2桁の場合は、すべての演算値が小数点3桁まで計算され、必要に応じて、端数処理されます。

式の小数点精度は"数式の端数処理と小数点精度の制御" ページ 874 で概説されたルールによって統制されます。

- **リテラル値** - たとえば、既定値として `0.00` を指定すると、演算値はすべて小数点以下2桁まで計算され、必要に応じて四捨五入されます。

小数点精度を上げる

数値演算値の小数点精度を上げるには、**Default Value** フィールドの小数点桁数を増やします。

式

式を1で乗算し、任意の小数点精度が続きます。式の開始に1があることを確認します。以下の例は、小数点4桁まで精度を上げます。

```
1.0000 * Invoice_Amount * 0.375
```

リテラル値

リテラル値の小数部に末尾のゼロを追加します。以下の例は、小数点3桁まで精度を上げます。

```
0.000
```

演算フィールドの種類

次の2種類の演算フィールドを作成できます。これらのフィールドについては次のセクションで説明します。

- 基本演算フィールド
- 条件付き演算フィールド:
 - リテラル値
 - 演算値

基本演算フィールド

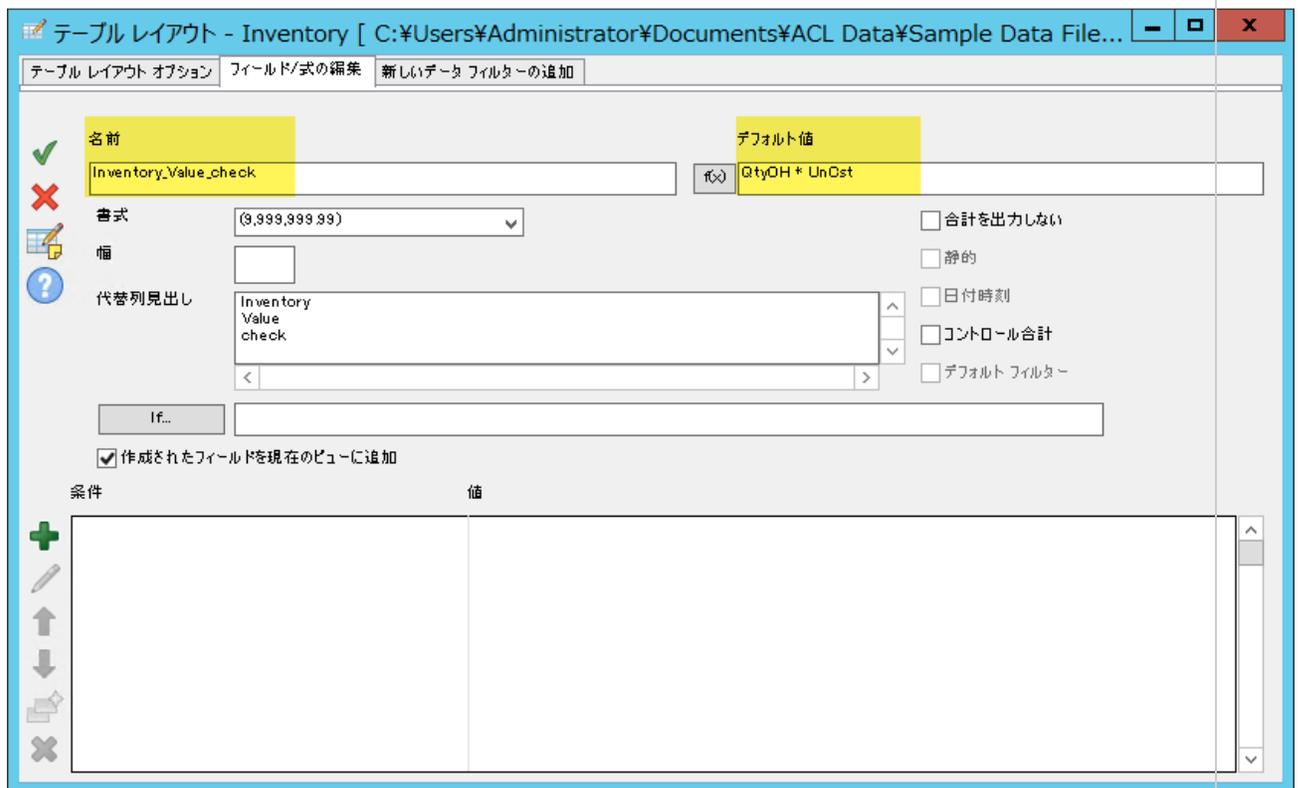
基本演算フィールドは、単一の式またはリテラル値を使用し、レコードの値に関係なく、テーブルのすべてのレコードに適用します。

基本演算フィールドの例

在庫レポート各製品のコストで合計在庫価格を確認します。

演算フィールド **Inventory Value check** を作成します。**Quantity on Hand** フィールドを **Unit Cost** フィールドで乗算します。演算フィールドで計算された値を報告された値と比較し、一致するかどうかを確認します。

以下の例は、**テーブルレイアウト**]ダイアログボックスの **Inventory_Value_check** 演算フィールドの定義を示します。演算式 (`QtyOH * UnCst`) は **Default Value** フィールドに表示されます。



テーブルビューで、物理的なソースデータフィールド (Inventory Value at Cost) の横に演算フィールド (Inventory Value check) を配置し、値を比較できます。

Quantity On Hand	Unit Cost	Inventory Value at Cost	Inventory Value check
870	6.87	5,976.90	5,976.90
460	6.87	3,160.20	3,160.20
1,480	(6.87)	(10,167.60)	(10,167.60)
1,290	6.87	8,862.30	8,862.30
1,500	6.87	10,305.00	10,305.00
2,420	6.87	16,625.40	16,625.40
1,870	6.87	12,846.90	12,846.90
130	47.00	6,110.00	6,110.00
612	18.00	11,016.00	11,016.00
700	11.53	8,071.00	8,071.00
248	12.50	3,100.00	3,100.00
248	2.48	615.04	615.04
0	8.40	0.00	0.00
(12)	49.60	(595.20)	(595.20)

一致しない値を返すフィルターを作成することもできます。

Inventory_Value_check <> Inventory_Value_at_Cost

基本演算フィールドを定義する

単一の式またはリテラル値を使用する基本演算フィールドを定義し、レコードの値に関係なく、テーブルのすべてのレコードに適用します。

方法の説明

演算フィールドの名前とデフォルト値を指定する

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブで、 **新しい式の追加**  をクリックします。
3. **名前**]テキスト ボックスにフィールド名を入力します。

メモ

フィールド名は256文字までの小文字の英字に制限されます。名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

Analytics には多くの予約キーワードがあり、フィールド名にこのキーワードを使用することはできません。詳細については、"予約キーワード" ページ 1421を参照してください。

4. 次のいずれかを実行します。
 - **デフォルト値**]テキスト ボックスに式またはリテラル値を入力します。
この方法は、シンプルな式にのみ適しています。
 - **[f(x)]** をクリックし、**式ビルダー** を使用して式を作成します。
詳細については、"式ビルダーによる式の作成" ページ 872を参照してください。

メモ

数値の演算フィールドでは、すべての数値演算値の小数精度は、 **デフォルト値**]で指定された式またはリテラル値の精度で統制されます。

詳細については、"数値演算フィールドの小数点精度の制御" ページ 796を参照してください。

フィールド メタデータの指定

1. (オプション) **幅**]テキスト ボックスにフィールドの幅を文字で指定します。
Analytics ビューおよびレポートにフィールドを表示する場合、幅値は列サイズとして使用されます。

2. (オプション) [代替列見出し]テキスト ボックスには、表示名を指定します。

Analytics ビューおよびレポートにフィールドを表示する場合、表示名はフィールド名の代わりに列見出しとして使用されます。この値が指定されていない場合は、フィールド名が使用されます。

3. 必要に応じて、以下の一覧の設定から1つ以上の値を選択します。

デフォルト値]テキスト ボックスで指定した式またはリテラル値のデータ カテゴリは、有効な設定を示します。

設定	データ カテゴリ	説明
書式	数値のみ	ビューやレポートにおける数値フィールドの表示書式を制御します。 書式は、ドロップダウンリストから選択する、または直接入力するか、あるいはドロップダウンリストから書式を選択後、その書式を編集することもできます。 書式]ドロップダウンリストが空白の場合、このフィールドのデータには オプション]ダイアログ ボックスで指定されたデフォルトの表示形式が適用されます。ここで指定した書式は、デフォルトの書式に優先します。
合計を出力しない	数値のみ	このフィールドの値の合計は算出されなくなります。 Analyticsでは、レポート内の数値フィールドは自動的に集計されます。数値フィールドによっては、単価や口座番号など、合計する必要がない情報が格納されている場合もあります。
静的	Analytics	が、IF ステートメントを評価するときに使用するデフォルトの動作が変わります。(省略可能な IF ステートメントの詳細については、「フィールド定義を確定する」次のページを参照してください。) 静的が選択解除(デフォルト) -IF ステートメントが False と評価される場合、フィールドのデータ カテゴリに応じて、フィールドには空の値(空白、ゼロ(0)、または False(F)) が割り当てられます。 静的を選択 -IF ステートメントが False と判定された場合、Analytics では空の値ではなく、そのフィールドの最後の有効な値が繰り返し使用されるようになります。IF ステートメントが True と評価され、新しい値が使用されるまで、最後の有効な値が各行で繰り返されます。
日付時刻		このオプションは演算フィールドには使用できません。
コントロール合計	数値のみ	フィールドをコントロール合計フィールドとして指定します。 コントロール合計は数値フィールドの値の合計です。これを使用して、データの整合性を確認することができます。新規のテーブルにデータを抽出した場合、または新規のテーブルでデータを並べ替えた場合、コントロール合計フィールドの入力値合計と出力値合計がテーブル履歴に格納されます。入力元は元テーブルを指しています。出力は新規テーブルを指しています。2つの合計が一致する場合は、抽出または並べ替え処理で失われたデータはありません。 データ提供者が提供するコントロール合計を Analytics が算出するコントロール合計と比較して、すべてのデータを受け取ったかどうかを確認することができます。 複数のフィールドに対してコントロール合計を指定すると、テーブル履歴はその中で開始位置が最も左にある数値フィールドについてのみ集計を報告します。
デフォルト フィルター	論理のみ	フィールドの値 (True または False) に基づき、デフォルト ビューのレコードをフィルターします。True と判断されるレコードのみが表示されます。

設定	データ カテゴリ	説明
		フィールドを含む Analytics テーブルが開くたびに、フィルターが自動的に適用されます。

フィールド定義を確定する

1. (省略可能) 演算フィールドによって評価されるレコードを制限したい場合は、**[f]** テキスト ボックスで条件を指定します。直接入力するか、または **[f]** ボタンをクリックして、**式ビルダー**を利用して IF ステートメントを作成します。

- 演算フィールドによって評価 - IF ステートメントを満たすレコード
- 演算フィールドによって評価されない - IF ステートメントを満たさないレコード

たとえば、IF 文 `Invoice_Amount >= 1000` では、\$1000 未満の請求金額のレコードが評価されません。

除外されたレコードの場合、演算フィールド値は、空白、ゼロ(0)、または False (F) です。これは、演算フィールドのデータ型によって異なります。IF ステートメントを削除すると、いつでも除外を元に戻すことができます。

2. (オプション) 開いたテーブルビューに新しい計算フィールドを自動的に追加しない場合は、**[作成されたフィールドを現在のビューに追加]**の選択を解除します。

オプションを選択したままにした場合、新しいフィールドがテーブルビューに追加されます。フィールドはビューの最後の列として配置されるか、ビューの選択した列の左に移動します。

いつでも手作業でフィールドをビューに追加することができます。詳細については、「ビューに列を追加する」ページ 849を参照してください。

3. (省略可能) フィールドに関するノートを追加する場合は、**フィールド ノートの編集**  をクリックしてノート テキストを入力し、**閉じる**  をクリックします。
4. **入力を受け入れる**  をクリックします。

Analyticsは演算フィールドをテーブルレイアウトに追加します。そのフィールドをコマンドまたはレポート使用することができます。

5. **閉じる**  をクリックしてテーブルレイアウト ダイアログ ボックスを終了します。

[作成されたフィールドを現在のビューに追加]を選択したままにした場合、関連する列がテーブルビューに追加されます。

条件付き演算フィールド

条件付き演算フィールドには、複数の式またはリテラル値が含まれ、条件に基づいてテーブルのレコードに適用します。各レコードに適用される特定の式またはリテラル値は、レコードの値によって異なります。

条件付き演算フィールドとリテラル値の例

請求金額のサイズに応じて、"Small"、"Medium"、または "Large" のリテラル値を各レコードに割り当てます。

演算フィールド **Invoice size** を作成します。このフィールドは、各レコードの請求金額のサイズを特定し、正しいリテラル値を割り当てます。

- "Small" - \$5,000.00 未満の金額
- "Medium" - \$5,000.00 ~ \$9,999.99 の金額
- "Large" - \$10,000.00 以上の金額

以下の例は、**テーブルレイアウト**]ダイアログボックスの **Invoice size** 演算フィールドの定義を示します。リテラル値 "Small" は **Default Value** フィールドに表示されます。リテラル値 "Medium" および "Large" はそれぞれ別の条件で表示されます。

各条件には、関連付けられた値が演算フィールドで使用されるために、True と評価される必要がある演算式が含まれます。レコードが条件のいずれも満たさない場合、デフォルト値が使用されます。

メモ

条件の一覧の順序は重要です。詳細については、"制限が大きい方から制限が少ない方への条件の一覧" ページ 805を参照してください。

The screenshot shows the 'Table Layout' dialog box for 'Ap_Trans'. The 'Name' field is 'Invoice_Size' and the 'Default Value' is 'Small'. The 'Conditions' table is as follows:

条件	値
Discount_Amount >= 10000	"Large"
Discount_Amount >= 5000	"Medium"

テーブルビューでは、各レコードの請求サイズが表示されます。

Invoice Amount	Invoice Size
618.30	Small
6,705.12	Medium
7,955.46	Medium
4,870.83	Small
10,531.71	Large
5,734.00	Medium
2,196.00	Small
265.19	Small
225.00	Small
14.88	Small
1,217.16	Small
158.60	Small
2,230.41	Small
4,324.00	Small

1つのサイズのレコードのみを表示するフィルターを作成できます。

Invoice_size = "Large"

Invoice Amount	Invoice Size
10,531.71	Large

計算値を持つ条件付き演算フィールドの例

請求サイズによって異なる値引割合に基づいて、各レコードの値引金額を計算します。

演算フィールド **Discount amount** を作成します。このフィールドは、各レコードの請求金額のサイズを特定し、正しい割合を使用して値引金額を計算します。

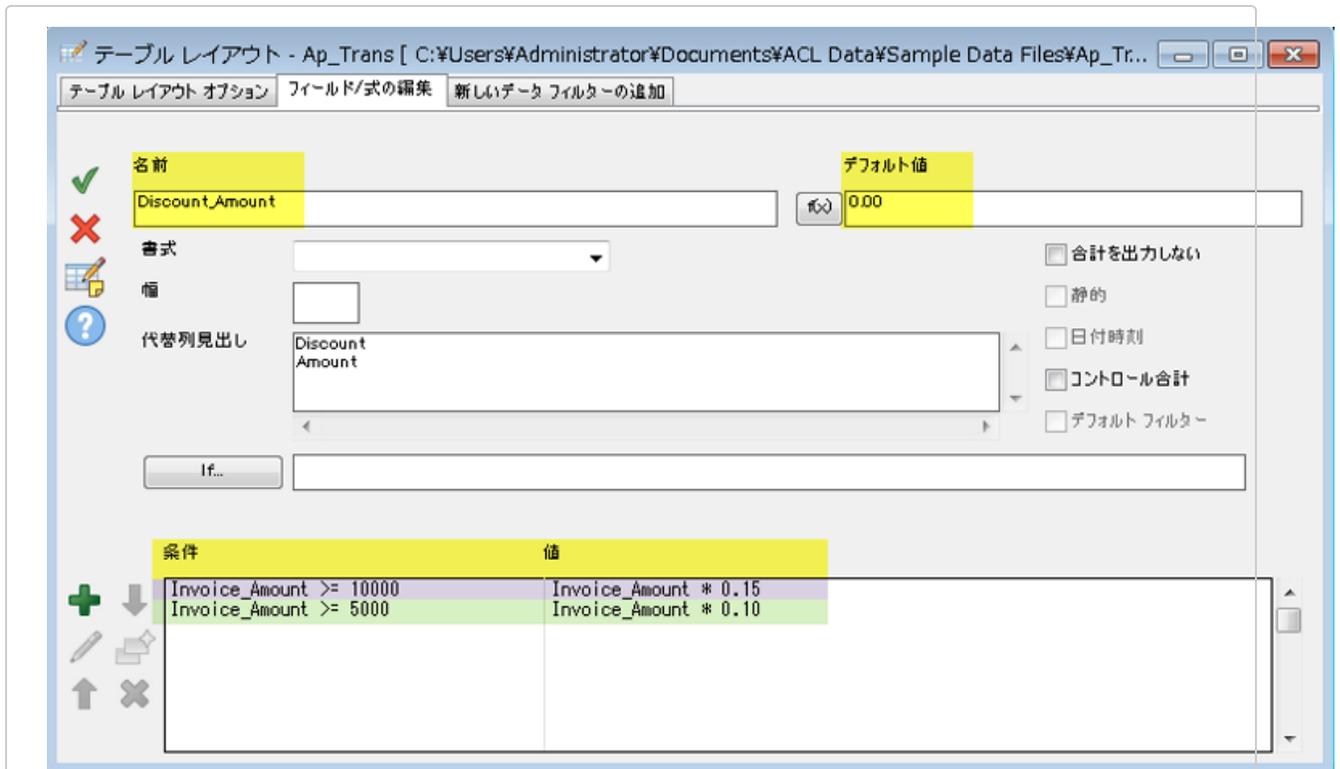
- 0% 値引き - \$5,000.00 未満の金額
- 10% 値引き - \$5,000.00 ~ \$9,999.99 の金額
- 15% 値引き - \$10,000.00 以上の金額

以下の例は、**テーブルレイアウト**]ダイアログボックスの **Discount amount** 演算フィールドの定義を示します。リテラル値 `0.00` は **Default Value** フィールドに表示されます。演算値 `Invoice_Amount * 0.10` および `Invoice_Amount * 0.15` はそれぞれ別の条件で表示されます。

各条件には、関連付けられた演算値が使用されるために、True と評価される必要がある演算式が含まれます。レコードが条件のいずれも満たさない場合、デフォルト値が使用されます。

メモ

条件の一覧の順序は重要です。詳細については、「制限が大きい方から制限が少ない方への条件の一覧」ページ 805を参照してください。



テーブルビューでは、各レコードの値引金額が表示されます。

Invoice Amount	Discount Amount
618.30	0.00
6,705.12	670.51
7,955.46	795.55
4,870.83	0.00
10,531.71	1,579.76
5,734.00	573.40
2,196.00	0.00
265.19	0.00
225.00	0.00
14.88	0.00
1,217.16	0.00
158.60	0.00
2,230.41	0.00
4,324.00	0.00

特定の金額よりも大きい値引きを表示するフィルターを作成できます。

Discount_amount >= 750

Invoice Amount	Discount Amount
7,955.46	795.55
10,531.71	1,579.76

制限が大きい方から制限が少ない方への条件の一覧

複数の条件を定義した場合、Analytics は、**テーブルレイアウト** ダイアログボックスで、それらの条件を条件リストの表示順に上から評価します。

上記の例の **請求書サイズ** では、請求金額が次の順序で条件に対してテストされます。

注文	条件	値
1	Invoice_Amount >= 10000	"大"
2	Invoice_Amount >= 5000	"中"

複数の条件を満たすレコードが意図したとおりに処理されることを保証するには、最も制限が大きい方から制限が少ない方に条件を一覧にし、最も制限が大きい条件が上位になるようにします。

詳しく表示

「制限」の意味

「制限」とは、条件を満たす資格がある値のセットの比率のことです。条件の制限が大きくなるほど、条件を満たす対象の比率が小さくなります。

次の値のセットを考えてください。

- \$12,000
- \$8,000
- \$7,000

最も制限が少ない条件 - すべての3つの値が条件を満たす `Invoice_Amount >= 5000` 値の全体セットが対象となるため、条件の制限が最も少なくなります。

最も制限が大きい条件 - \$12,000 のみが条件を満たす `Invoice_Amount >= 10000` セットの値のいずれかのみが対象となるため、条件の制限が最も大きくなります。

Analytics が条件付き演算フィールド値を割り当てる方法

各レコードに対して、True と判定される最初の条件に関連付けられた演算フィールド値が割り当てられます。割り当てられると、レコードが後続の条件を満たす場合でも、演算フィールド値は変更されません。

請求金額 \$12,000 を考えてください。

- **割り当てられた値 = 大** - 上記の条件の順序では、金額に大の値が割り当てられます。金額は \$10,000 よりも大きいため、これは正しいです。
- **割り当てられた値 = 中** - 上記の条件の順序では、金額に中の値が割り当てられます。金額は \$5,000 よりも大きいため、これも正しいです。ただし、値の割り当ては意図したように動作しません。`Invoice_Amount >= 5000` は制限が少ない条件であり、取り込みたくない金額も取り込んでいるためです。

サブセットの点から制限を考える

別の方法はサブセットの点から制限を考えることです。最初の一覧の条件を満たす資格がある値は、値のセットの最も小さいサブセットを構成します。各追加の条件があると、資格のあるサブセットのサイズが大きくなり、すべての前のサブセットが含まれます。

Analytics が演算フィールド値をレコードに割り当てると、値は変更されません。以下の例では、"大" が請求金額 \$12,000 の請求金額を含むレコードに割り当てられます。レコードが条件 2 を満たす場合でも、値は "中" に更新されません。



条件付き演算フィールドを定義する

複数の式またはリテラル値を含む演算フィールドを定義し、条件に基づいてテーブルのレコードに適用します。

方法の説明

演算フィールドの名前とデフォルト値を指定する

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブで、**新しい式の追加**  をクリックします。
3. **名前**]テキスト ボックスにフィールド名を入力します。

メモ

フィールド名は256文字までの小文字の英字に制限されます。名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

Analyticsには多くの予約キーワードがあり、フィールド名にこのキーワードを使用することはできません。詳細については、「予約キーワード」ページ 1421を参照してください。

4. 次のいずれかを実行します。
 - **デフォルト値**]テキスト ボックスに式またはリテラル値を入力します。
この方法は、シンプルな式にのみ適しています。
 - **[f(x)]** をクリックし、**式ビルダー** を使用して式を作成します。
詳細については、「式ビルダーによる式の作成」ページ 872を参照してください。

メモ

数値の演算フィールドでは、すべての数値演算値の小数精度は、**デフォルト値**]で指定された式またはリテラル値の精度で統制されます。

詳細については、「数値演算フィールドの小数点精度の制御」ページ 796を参照してください。

リテラルテキスト値は、引用符 (" ") で囲む必要があります。リテラル日付値は、バッククォート (` `) で囲む必要があります。

フィールド メタデータの指定

1. (オプション) **幅**]テキスト ボックスにフィールドの幅を文字で指定します。
Analyticsビューおよびレポートにフィールドを表示する場合、幅値は列サイズとして使用されます。
2. (オプション) **[代替列見出し]** テキスト ボックスには、表示名を指定します。
Analyticsビューおよびレポートにフィールドを表示する場合、表示名はフィールド名の代わりに列見出しとして使用されます。この値が指定されていない場合は、フィールド名が使用されます。
3. 必要に応じて、以下の一覧の設定から1つ以上の値を選択します。
デフォルト値]テキスト ボックスで指定した式またはリテラル値のデータカテゴリは、有効な設定を示します。

設定	データ カテゴリ	説明
書式	数値のみ	<p>ビューやレポートにおける数値フィールドの表示書式を制御します。</p> <p>書式は、ドロップダウンリストから選択する、または直接入力するか、あるいはドロップダウンリストから書式を選択後、その書式を編集することもできます。</p> <p>書式]ドロップダウンリストが空白の場合、このフィールドのデータには オプション]ダイアログボックスで指定されたデフォルトの表示形式が適用されます。ここで指定した書式は、デフォルトの書式に優先します。</p>
合計を出力しない	数値のみ	<p>このフィールドの値の合計は算出されなくなります。</p> <p>Analyticsでは、レポート内の数値フィールドは自動的に集計されます。数値フィールドによっては、単価や口座番号など、合計する必要がない情報が格納されている場合もあります。</p>
静的	Analytics	<p>が、IF ステートメントを評価するとき使用するデフォルトの動作が変わります。(省略可能な IF ステートメントの詳細については、「フィールド定義を確定する」ページ 810を参照してください。)</p> <p>静的が選択解除(デフォルト) -IF ステートメントが False と評価される場合、フィールドのデータカテゴリに応じて、フィールドには空の値(空白、ゼロ(0)、または False(F)) が割り当てられます。</p> <p>静的を選択 -IF ステートメントが False と判定された場合、Analytics では空の値ではなく、そのフィールドの最後の有効な値が繰り返し使用されるようになります。IF ステートメントが True と評価され、新しい値が使用されるまで、最後の有効な値が各行で繰り返されます。</p>
日付時刻		このオプションは演算フィールドには使用できません。
コントロール合計	数値のみ	<p>フィールドをコントロール合計フィールドとして指定します。</p> <p>コントロール合計は数値フィールドの値の合計です。これを使用して、データの整合性を確認することができます。新規のテーブルにデータを抽出した場合、または新規のテーブルでデータを並べ替えた場合、コントロール合計フィールドの入力値合計と出力値合計がテーブル履歴に格納されます。入力元は元テーブルを指しています。出力は新規テーブルを指しています。2つの合計が一致する場合は、抽出または並べ替え処理で失われたデータはありません。</p> <p>データ提供者が提供するコントロール合計を Analytics が算出するコントロール合計と比較して、すべてのデータを受け取ったかどうかを確認することができます。</p> <p>複数のフィールドに対してコントロール合計を指定すると、テーブル履歴は其中で開始位置が最も左にある数値フィールドについてのみ集計を報告します。</p>
デフォルト フィルター	論理のみ	<p>フィールドの値 (True または False) に基づき、デフォルト ビューのレコードをフィルターします。True と判断されるレコードのみが表示されます。</p> <p>フィールドを含む Analytics テーブルが開くたびに、フィルターが自動的に適用されます。</p>

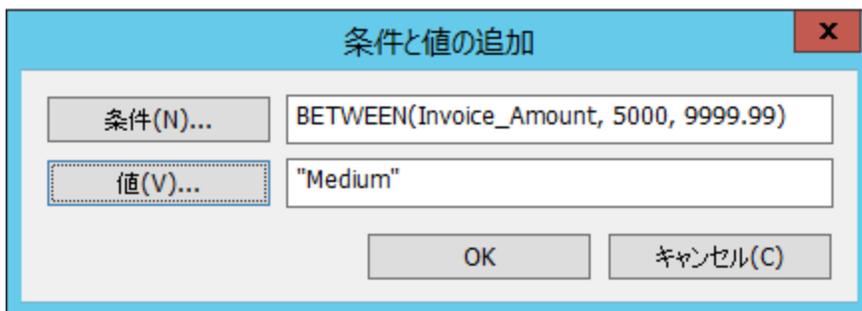
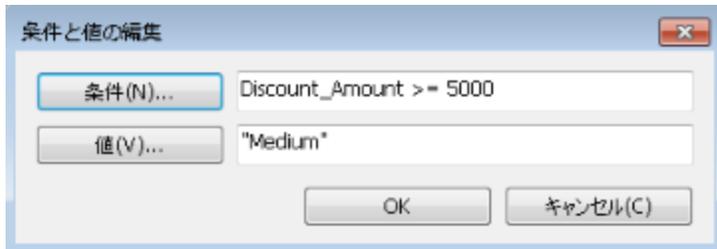
演算フィールドの条件値の指定

条件値は条件と値の組み合わせとして設定されます。レコードが**条件**を満たす場合、演算フィールドは、指定された**値**を使用します。

メモ

指定する値と、**デフォルト値**は、すべて同じデータ型である必要があります。

1. **条件の挿入**  をクリックします。
 2. **条件と値の追加** ダイアログボックスで、次を実行し、**OK** をクリックします。
 - a. **条件** テキスト ボックスに式を入力するか、**条件** をクリックして**式ビルダー**で式を作成します。
 - b. **値** テキスト ボックスに値を入力するか、**値** をクリックして**式ビルダー**で式を作成します。
- 上記の2つの例の条件値は次のとおりです。



3. 別の条件を作成する場合は、次のいずれかを実行します。
 - **条件の挿入**  をクリックし、上記の手順を繰り返します。
 - 既存の条件と類似する条件を作成する場合は、その既存の式を選択して **条件のコピー**  をクリックし、**条件と値の編集**  をクリックして、新しい条件に対する設定を編集します。
4. (省略可能) 条件を選択し、**条件を上に移動**  または **条件を下に移動**  をクリックして、評価される順序を変更します。

メモ

条件の一覧の順序は重要です。詳細については、「制限が大きい方から制限が少ない方への条件の一覧」ページ 805を参照してください。

フィールド定義を確定する

1. (省略可能) 演算フィールドによって評価されるレコードを制限したい場合は、**[f]**テキストボックスで条件を指定します。直接入力するか、または **[f]** ボタンをクリックして、**式ビルダー**を利用してIFステートメントを作成します。

- 演算フィールドによって評価 - IF ステートメントを満たすレコード
- 演算フィールドによって評価されない - IF ステートメントを満たさないレコード

たとえば、IF 文 `Invoice_Amount >= 1000` では、\$1000 未満の請求金額のレコードが評価されません。

除外されたレコードの場合、演算フィールド値は、空白、ゼロ(0)、または False (F) です。これは、演算フィールドのデータ型によって異なります。IF ステートメントを削除すると、いつでも除外を元に戻すことができます。

2. (オプション) 開いたテーブルビューに新しい計算フィールドを自動的に追加しない場合は、**[作成されたフィールドを現在のビューに追加]**の選択を解除します。

オプションを選択したままにした場合、新しいフィールドがテーブルビューに追加されます。フィールドはビューの最後の列として配置されるか、ビューの選択した列の左に移動します。

いつでも手作業でフィールドをビューに追加することができます。詳細については、「ビューに列を追加する」ページ 849を参照してください。

3. (省略可能) フィールドに関するノートを追加する場合は、**フィールド ノートの編集**  をクリックしてノート テキストを入力し、**閉じる**  をクリックします。

4. **[入力を受け入れる]** をクリックします 。

Analyticsは演算フィールドをテーブルレイアウトに追加します。そのフィールドをコマンドまたはレポート使用することができます。

5. **閉じる**  をクリックしてテーブルレイアウト ダイアログ ボックスを終了します。

[作成されたフィールドを現在のビューに追加] を選択したままにした場合、関連する列がテーブルビューに追加されます。

Analytics のデータ型

以下の一覧は、Analytics でサポートされるデータ型とデータ型を使用できるデータソースです。

同じデータ型でも、**データ定義ウィザード**と **テーブルレイアウト** ダイアログ ボックスでは名前が異なる場合があります。両方の名前が下に表示されます。

Analytics データ型は次の 4 つのカテゴリに分けられます。

- 文字
- 数値
- 日付時刻
- 論理

特定のデータ型のフィールドで実行できる Analytics 処理およびフィールドの表示方法は、データカテゴリによって決定されます。例：

- 数値カテゴリのデータ型のフィールドのみを階層化できます。
- 文字カテゴリのデータ型のフィールドは左寄せになります。

操作で誤ったデータ型のフィールドを使用した場合は、Analytics がエラーを表示します。

Analytics データ型 (データ定義ウィザード)	Analytics データ型 (テーブルレイアウト]ダイアログボックス)	Analytics データカテゴリ	外部データソース	追加情報
AccPac Accounting Number	ACCPAC	数値	ACCPAC	ACCPAC 会計アプリケーションで使用されます。このデータフィールドの長さは、常に 6 バイトです。Analytics は、ほかに指定された長さを無効にします。
ACL	ACL	数値	なし。これは、Analytics のシステム データ型です。	Analytics での演算の結果を格納するために、Analytics で生成された 12 バイトのフィールドです。大きな数値を格納するように設計されており、このフィールドは印刷できません。Analytics は、この型がフィールドに対して適切である場合は、自動的に割り当てます。
ASCII テキスト	ASCII	文字	Windows ベースのアプリケーション	ASCII (American Standard Code for Information Interchange) 文字エンコードで格納されているデータに使用されます。 Analytics は拡張 ASCII を使用し、256 文字の出力可能

Analytics データ型 (データ定義ウィザード)	Analytics データ型 ([テーブルレイアウト]ダイアログボックス)	Analytics データ カテゴリ	外部データソース	追加情報
				<p>よび出力不可能文字を定義します。Analytics で使用可能な実際の文字は、オペレーティングシステムのデフォルトの8ビットコード ページで指定されます。</p> <p>ASCII フィールドの最大長は 32767 バイトです。</p>
Basic Floating Point	BASIC	数値	Windows ベースの BASIC アプリケーション	BASIC プログラミング言語用にフォーマットされた浮動小数点データ型に使用されます。このデータ型のフィールド長は、4 バイトまたは 8 バイトです。
Binary Numeric	BINARY	数値	<ul style="list-style-type: none"> ○ PL/1 ○ COBOL COMPUTATIONAL-1 ○ 固定のバイナリデータ型 	<p>最大長は 8 バイトです。これには小数点以下の桁数も含まれており、データ長で指定された桁数を超えることはできません。</p> <p>偶数長のバイナリフィールドは、符号付きバイナリフィールド(2 の補数)として処理され、奇数長の場合は(上位のバイトにゼロを暗黙に追加して)符号なしフィールドとして処理されます。</p>
Custom Text Format	CUSTOM	文字	なし。これは、必要に応じてユーザーが割り当てることができる Analytics データ型です。	<p>データソースからデータが読み取られるときに、ユーザー定義の文字の置換を有効にするために使用されます。このデータ型は、custom.dat という名前のファイルに置換文字が定義されていない限り、データを ASCII テキストとして読み取ります。</p> <p>詳細については、"Custom データ型" ページ 819 を参照してください。</p>
日付時刻	DATETIME	日付時刻	これは、日付、日付時刻および時刻を格納するフィールドに自動または手動で割り当てられる Analytics データ型です。	YYMMDD または YYMMDD hh:mm:ss など、さまざまな書式を使用して格納されている日付、日付時刻および時刻に使用されます。フィールド定義の書式 (Format) 設定は、

Analytics データ型 (データ定義ウィザード)	Analytics データ型 ([テーブルレイアウト]ダイアログボックス)	Analytics データ カテゴリ	外部データソース	追加情報
				データソースから日付時刻データを読み取る方法を指定します。
EBCDIC text	EBCDIC	文字	IBM z/OS および OS/400 のアプリケーション	IBM サーバー オペレーティングシステム上で、8ビット 文字エンコードである、EBCDIC (Extended Binary Coded Decimal Interchange Code: 拡張 2 進法 10 進コード) データに使用されます。このデータ型の長さは、最大 32767 バイトです。
Floating Point	FLOAT	数値	Windows ベースのアプリケーション	倍精度浮動小数点数に使用されます。このデータ型のフィールド長は、4 バイトまたは 8 バイトです。
該当なし	HALFBYTE	数値	Unisys(Burroughs) 社のアプリケーション	Unisys(Burroughs) システムで使われる、半バイトずつがパック形式のデータに使用されます。符号付き数値は、Unisys(Burroughs) の規則に従う必要があります。このデータ型の開始位置と長さは、半バイト単位で指定する必要があります。開始位置は、次のように求めます。 (バイト位置*2)-1 このデータ型は、 テーブルレイアウト ダイアログボックスでのみ選択できます。
IBM Floating Point	IBMFLOAT	数値	IBM z/OS および OS/400 のアプリケーション	IBM 浮動小数点データに使用され、主にメインフレームの科学計算用アプリケーションで使用されています。このデータ型のフィールド長は、4 バイトまたは 8 バイトです。
論理	LOGICAL	論理	これは、論理値を格納するフィールドに自動または手動で割り当てられる Analytics データ型です。	ブール型データ(通常、true または false)を表す単一文字フィールドに使用されます。Analytics では、次の値のセットを解釈することができます。最初の値は true に評価され、

Analytics データ型 (データ定義ウィザード)	Analytics データ型 ([テーブルレイアウト]ダイアログボックス)	Analytics データ カテゴリ	外部データソース	追加情報
				<p>2番目の値は false に評価されます。</p> <p>1/0、T/F、t/f、Y/N、y/n、空白以外/ASCII の空白文字 (16進数 20)</p>
PC Binary	MICRO	数値	Windows ベースのアプリケーション	<p>Integer または Long データ型を表す符号なしのバイナリ数値データに使用されます。最大長は 8 バイトです。小数点以下桁数は小数点以下の桁数として定められた値で、データ長で定められた桁数を越えることはできません。偶数長の Micro フィールドは符号付きバイナリフィールドとして処理され、奇数長の場合は符号なしフィールドとして処理されます。</p>
該当なし	NOTE	文字	なし。これは、Analytics のシステム データ型です。	<p>Analytics テーブルに関連付けられているレコード ノートに関する情報を格納するために、Analytics によって使用されます。Note データ型を使用してフィールドを定義することはできません。</p>
Numeric (Unformatted)	NUMERIC	数値	Windows ASCII または Unicode の印刷可能な数値データ、または COBOL の Display データ型を使用する z/OS または OS/400 の EBCDIC データ	<p>COBOL の Display 型に対応する印刷可能な数値データに使用されます。このフィールド型には任意の分離記号を含めることができますが、最も一般的なのは、先頭または末尾のスペース、オプションで追加する先頭または末尾の記号、桁区切りのカンマ、および明示的な小数点です。</p> <p>このデータ型には最大で、22 桁の数字に 18 文字の分離記号を加えた合計 40 バイトを格納できます。先頭のゼロは空白として扱われます。</p> <p>フィールドに指定された小数点以下の桁数は、それが適切であろうと適用されるため、このデータ型を使用する場合には注意が必要です。たと</p>

Analytics データ型 (データ定義ウィザード)	Analytics データ型 (「テーブルレイアウト」ダイアログボックス)	Analytics データカテゴリ	外部データソース	追加情報
				<p>例えば、小数位 2 桁を指定して値 \$500.50 と \$399 を読み取った場合、最初の値は 500.50 と正しく解釈されますが、2 番目の値は、399.00 ではなく 3.99 と解釈されます。</p> <p>フィールドのデータに明示されている小数点以下の桁数が、指定した小数位と異なる場合、フィールドのデータが適切な桁数に丸められます。</p> <p>Analytics は、丸かっこ"CR"を正しく負の値として解釈しますが、カンマやドル記号 (\$) などの分離記号は無視します。記号は、先頭に指定することも、末尾に指定することもできます。また、固定小数用と浮動小数用とがあります。</p>
Packed Numeric	PACKED	数値	PL/1 の fixed decimal データ型または COBOL の computational-3 データ型	<p>1 バイトにつき 2 桁の数値を格納する、メインフレームのオペレーティングシステムのパックされた数値データに使用されます。右端バイトの下位 4 ビットには符号を示す値が設定されています。通常この値は、正の場合 16 進数の C、負の場合 D です(16 進数の B によって負の数値を表すことはサポートされていません)。右端バイトの上位 4 ビットおよび、その他のバイトの上位と下位の 4 ビットには、対象の数値における対応する位置の数字を表す 16 進数字がそれぞれ 1 つ設定されています。</p> <p>このデータ型の最大長は 12 バイト(23 桁)です。ただし、数値が 22 桁より大きい場合、Analytics はエラーメッセージを出力します。したがって、「テーブルレイアウト」ダイアログボックスでパック数値フィールドを定義する場合には、小数位テキストボックス内に指定する桁数の合計が 22 桁より長くな</p>

Analytics データ型 (データ定義ウィザード)	Analytics データ型 (「テーブルレイアウト」ダイアログボックス)	Analytics データカテゴリ	外部データソース	追加情報
				<p>ないようにする必要があります。たとえば、データの数字が7桁の場合、小数点以下の桁数は、22桁から7桁を引いた15桁までとなり、それより大きい数を指定することはできません。</p> <p>パック数値フィールドは、日付情報を数値書式で格納する場合にも使用できます。</p>
PC DOS Text	PCASCII	文字	Windows	<p>ASCII データ型と同様です。ファイルのデータがDOS アプリケーションで作成されているときに使用できます。</p> <p>Analytics で使用可能な PCASCII 文字はコード ページ 437 で指定されます。</p> <p>PCASCII フィールドの最大長は 32767 バイトです。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>ASCII データ型が必要なときには、PCASCII データ型を使用しないでください。2つのデータ型の拡張文字セットは異なります。</p> </div>
Numeric (Formatted)	PRINT	数値	Windows ASCII または Unicode の印刷可能な数値データ、または COBOL の Display データ型を使用する z/OS または OS/400 の EBCDIC データ	<p>COBOL の Display 型に対応する印刷可能な数値データに使用されます。このフィールド型には任意の分離記号を含めることができますが、最も一般的なのは、先頭または末尾のスペース、オプションで追加する先頭または末尾の記号、桁区切りのカンマ、および明示的な小数点です。</p> <p>このデータ型には最大で、22桁の数字に18文字の分離記号を加えた合計 40 バイトを格納できます。先頭のゼロは空白として扱われます。</p>

Analytics データ型 (データ定義ウィザード)	Analytics データ型 ([テーブルレイアウト]ダイアログボックス)	Analytics データカテゴリ	外部データソース	追加情報
				<p>どの数値にも小数位の数字が含まれているわけでない場合は、Numeric (Unformatted) /NUMERIC 型ではなく、このデータ型を使用する必要があります。たとえば、小数位 2 桁を指定して値 \$500.50 と \$399 を読み取った場合、このデータ型はどちらの値も正しく解釈します(500.50 および 399.00)。</p> <p>フィールドのデータに明示されている小数点以下の桁数が、指定した小数位と異なる場合、フィールドのデータが適切な桁数に丸められます。</p> <p>Analytics は、丸かっこ"CR"を正しく負の値として解釈しますが、カンマやドル記号 (\$) などの分離記号は無視します。記号は、先頭に指定することも、末尾に指定することもできます。また、固定小数用と浮動小数用とがあります。</p>
Unicode	UNICODE	文字	Unicode データ	<p>Unicode 文字データに使用されます。</p> <p>Unicode データの場合、Analytics は UTF-16LE 文字エンコーディングを使用します。</p> <p>このデータ型は Analytics Unicode 版でのみ利用可能です。</p>
UNISYS Packed	UNISYS	数値	Unisys(Burroughs) 社のアプリケーション	<p>Unisys(Burroughs) のパックされたバイト配列のデータを読み取るために使用されます。符号付き数値は、Unisys (Burroughs) の規則に従う必要があります。パックされた符号なしの Unisys データは、Unsigned Packed/UNSIGNED データ型を使用する必要があります。このフィールド型の最大長は 12 バイト、つまり 22 桁です。</p>

Analytics データ型 (データ定義ウィザード)	Analytics データ型 (「テーブルレイアウト」ダイアログボックス)	Analytics データ カテゴリ	外部データソース	追加情報
Unsigned Packed	UNSIGNED	数値	IBM z/OS および OS/400 のアプリケーション	パックされた符号なしデータに使用されます。このデータ型は、1 バイトにつき 2 桁の 10 進数字を格納します。このデータ型の最大長は 11 バイト、つまり 22 桁です。小数点以下の桁数が、このフィールドで可能な最大桁数を超えることはできません。
VAX Floating Point	VAXFLOAT	数値	DEC VAX アプリケーション	DEC(Digital Equipment Corporation) の VAX システムの D 浮動小数点データ型で使用されます。このデータ型の長さは、4 バイトまたは 8 バイトです。
Zoned Numeric	ZONED	数値	IBM、DEC、または Honeywell のメインフレームアプリケーション	<p>1 バイトにつき 1 桁が格納され、ASCII、EBCDIC または Unicode(Unicode 版 Analytics を使用している場合) でエンコードすることができる、ZONED 数値データで使用されます。</p> <p>先頭のゼロは保持され、フィールドの右端バイトの上位 4 ビットには負符号が入ります。ZONED フィールドの最大長は 22 バイトです。</p> <p>Analytics は、IBM、Honeywell、および DEC 形式に準拠する ZONED フィールドを自動的に検出して、形式を調整します。</p>

Custom データ型

Analytics Custom データ型は、非標準の文字データを含んでいるデータソースのフィールドを処理できません。たとえば、特定の文字を標準外またはサポートされていない方法で実装している英語以外の言語のアプリケーションからデータを読み取る場合に、Custom データ型を使用できます。

Custom データ型には ASCII 値が格納されます。ただし、ファイル `custom.dat` を作成できます。このファイルは、非標準またはサポートされていない文字値を標準 ASCII 文字値にマッピングします。

Custom.dat ファイル

`custom.dat` は、各行に 2 つの値が記述されている標準テキスト ファイルです。1 つ目の値は、非標準またはサポートされていない置き換えられる文字で、2 つ目の値は 1 つ目の値を置き換える文字です。値は、次の方法のいずれか、または組み合わせた方法を使用して指定できます。

- 文字コードは、数値を使用して指定します。たとえば、文字 'A' は 65 となります。
- 16 進数値は、2 文字の 16 進数値の前に X を付けて指定します。たとえば、文字 'A' は X41 となります。
- リテラル文字値は、文字の前に C を付けて指定します。たとえば、文字 'A' は CA となります。

`custom.dat` ファイルは Analytics の起動時に読み取られるため、Analytics の実行中にファイルを編集することはできません。また、`custom.dat` に指定する値はどれも、1 バイトに格納できる最大値の 255 を超えてはいけません。`custom.dat` ファイルは、任意のテキスト エディターで作成または編集できます。

例

データソースフィールドは、ドル記号に 16 進数の A4、カンマに A5、小数点には文字コード 5 を使用します。`custom.dat` ファイルを作成して、必要な値を置換します。ファイルには次の行があります。

```
XA4 C$
XA5 C,
5 C.
```

- 1 行目 -は、16 進数の値 A4 が検出されると、それをドル記号 (\$) に置き換えます。
- 2 行目 -は、16 進数の値 A5 が検出されると、それをカンマに置き換えます。
- 3 行目 -は、文字コード 5 が検出されると、それを小数点に置き換えます。

Custom データ型の置換規則を構成する

Custom データ型の置換規則は、`custom.dat` という名前のファイルを作成することによって構成します。このファイルには、置換する必要がある文字と、置換文字の一覧が含まれています。

Custom データ型がフィールド定義で選択されるたびに、`custom.dat` の非標準またはサポートされていない文字がマッピングされた ASCII 文字で自動的に置換されます。単一の `custom.dat` ファイルは、Custom データ型を使用して定義されている Analytics プロジェクトのすべてのフィールドにグローバルに適用されます。

1. テキスト エディターを開いて、新しいファイルを作成します。
2. 次の構文を使用して、置換規則をそれぞれ別の行に入力します。

```
<type> character_to_replace <type> replacement_character
```

- `type` - 文字値の場合は `C`、16 進数値の場合は `X` を指定します。数値 ASCII 文字コードの型を指定しないでください。
 - `character_to_replace` - 置き換える文字、数字、または 16 進数の値を指定します。
 - `replacement_character` - `character_to_replace` 値の代わりに用いる文字、数字、または 16 進数の値を指定します。
3. Analytics 実行ファイル (ACLWin.exe) がインストールされているフォルダーに、`custom.dat` という名前でファイルを保存します。

デフォルトの場所は、`C:\Program Files (x86)\ACL Software\ACL for Windows <バージョン>` です。

Custom データ型は、Analytics を次回起動したときに使用できるようになります。Custom データ型を使用するたびに、定義されている置換規則が自動的に適用されます。

テーブルレイアウトのフィールドの編集

フィールド定義の接続先のデータソースに変更があった場合、あるいはフィールドの書式設定方法を変更する場合は、テーブルレイアウトのフィールド定義を変更することができます。

フィールド定義が演算フィールドによって参照されている場合は、そのフィールド定義への変更が制限されます。そのような場合、フィールド名やデータ型は変更できませんが、それ以外の開始位置や長さなどのプロパティは変更できます。

テーブルレイアウトのフィールドを編集するには

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集** タブで変更するフィールドをダブルクリックします。
3. フィールド定義に対して必要な変更を行い、**入力を受け入れる** をクリックします。

フィールド定義の詳細については、"物理フィールドを定義する" ページ 790を参照してください。

テーブルレイアウトのフィールドの名前を変更する

テーブルレイアウトで定義されているフィールドの名前は必要に応じて変更することができます。フィールドの名前を変更した場合、Analytics オペレーションによって処理されるデータ要素の名前が変更されます。それはビュー内の列の名前を変更することと同じではなく、表示名のみを変更しています。

ビューに含まれているフィールドの名前を変更したときは、ビューに戻るとき、あるいはそのフィールドが含まれるビューを開いたときに、"<変更前のフィールド名> は未定義です" というエラーメッセージが表示されません。

1. **編集 > テーブルレイアウト**] の順にクリックします。
2. **フィールド/式の編集**] タブで名前を変更するフィールドをダブルクリックします。
3. **名前**] フィールドでフィールド名を変更します。
4. (オプション) 同時にビュー内で表示される列名も変更したい場合は、**代替列見出し**] フィールドの列名も変更します。
5. **入力を受け入れる**]  をクリックします。

フィールドは、テーブルレイアウトで名前が変更されます。

6. **閉じる**]  をクリックして**テーブルレイアウト** ダイアログボックスを終了します。
7. "<変更前のフィールド名> は未定義です" エラーメッセージが表示された場合は、**OK**] をクリックします。

代替列見出しを更新した場合は、列名が更新されます。

メモ

名前を変更したフィールドが他のビューにある場合はそれらのビューから削除され、手作業で再追加する必要があります。

テーブルレイアウトからのフィールドの削除

1つまたは複数のフィールドを削除するには

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブで削除対象のフィールドを1つ選択するか、あるいは**Ctrl** キーを押しながら複数のフィールドを選択します。
3. **フィールドの削除** ]をクリックします。
4. 確認プロンプトで、**削除**]をクリックします。

テーブルレイアウトにおけるフィールドのシフト

フィールドのシフトを使用すれば、固定のバイト数で表示されているフィールド定義を簡単に修正することができます。これが必要となるのは、テーブルレイアウトが不正に定義された場合、あるいはデータソースが変更されたが Analytics テーブルを再定義したくない場合です。

データソースの1つのフィールドの長さが1増えたなど、わずかな変更があり、そのフィールドがテーブルレイアウトで定義されていた場合は、後続フィールドの位置をずらすことで、そのテーブルレイアウトを使用し続けることができます。テーブルレイアウトで1つのフィールドの開始位置をシフトすると、そのフィールドの右側に定義されているすべてのフィールドの開始位置が自動的にシフトします。フィールドのシフトが作用するのは物理データフィールドのみです。

レコード長の調整

テーブルレイアウトのフィールドをシフトする場合、以下の状況が当てはまるときは **テーブルレイアウト オプション** タブの **レコード長** の値も調整する必要があります。

- **変更されたフィールド長** - データソースにあるフィールドの長さが変更された場合は、フィールドをシフトする前に、まずテーブルレイアウトでそれに該当するフィールドの長さを手動で調整する必要があります。また、テーブルレイアウトでレコード長の調整が必要になる場合もあります。
- **追加されたフィールド** - データソースに新しいフィールドが追加された場合は、まず、その新しいフィールドが収まるように、**テーブルレイアウト** ダイアログボックスでレコード長を増やす必要があります。シフトされるフィールドの1つが現在のレコード長を超える場合には、フィールドのシフトは実行できません。フィールドがシフトされたら、新しいフィールド定義を追加できます。
- **削除されたフィールド** - データソースからフィールドが削除された場合は、テーブルレイアウト内の対応するフィールドを削除します。フィールドをシフトした後、レコード長を手動で減らす必要が生じるかもしれません。

シフトされたフィールド定義は、レコード長内にある必要があります。

1つ以上のフィールド定義を左右にシフトするときには、フィールドがいずれの方向でもレコード長を超えることができません。

フィールドのシフトは、指定されたフィールド定義と、任意のフィールド定義の両方を、指定された定義の右に移動します。定義のシフトされたブロックがいずれかの方向でレコード長を超える場合、エラーメッセージが表示され、シフト処理は実行されません。

ヒント

レコードの最後を超過しているためにエラーメッセージが表示される場合は、最終フィールド定義を削除し、シフト対象のフィールド定義の場所を作ってください。

テーブルレイアウトでフィールドをシフトする

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブをクリックします。
3. 省略可能。テーブルレイアウトにデータフィルターが定義されている場合は、特定のデータフィルターのみを対象にフィールドをシフトすることができます。リストの上にあるドロップダウンリストから、該当するデータフィルターを選択します。選択したデータフィルターに関連付けられるフィールドのみが表示され、シフトされます。
4. 省略可能。特定のフィールドからフィールドのシフトを開始する場合は、フィールドのリストでフィールドを選択します。

選択するフィールドの開始バイト位置は、**フィールドのシフト**]ダイアログボックスにあらかじめ入力されます。

5. **フィールドのシフト**]をクリックします。
6. **位置からフィールドの位置を開始**]テキストボックスで、あらかじめ入力された値を保持するか、シフトする最初のフィールド定義の開始バイト位置を指定します。

すべてのフィールド定義が、指定されたフィールド定義の右にシフトされます。

非開始バイト位置を指定する場合は、次の開始バイト位置が使用されます。

メモ

非 Unicode Analytics	1 バイト = 1 文字
Unicode Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字
Unicode Analytics、Unicode データ	2 バイト = 1 文字

Unicode データの場合、一般的に、奇数の開始バイト位置を指定します。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。

7. **次のバイト数でフィールドをシフト**]テキストボックスで、フィールド定義をシフトするバイト数を指定します。

フィールド定義を右にシフトする正の数値を入力します。フィールド定義を左にシフトする負の数値を入力します。

メモ

非 Unicode Analytics	1 バイト = 1 文字
Unicode Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字
Unicode Analytics、Unicode データ	2 バイト = 1 文字

Unicode データの場合、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。

8. **[OK]** をクリックしてから、確認ダイアログ ボックスで、**[はい]** をクリックします。

データのダンプ

ダンプ機能では、次のエンコーディングの1つ以上で、レコードまたはファイルの出力可能または出力不可能な文字をすべて表示できます。

- Hex
- ASCII
- EBCDIC
- Unicode (Unicode Analytics のみ)

ダンプ機能を使用して、データの表示または処理に関する問題をトラブルシューティングしたり、ファイルのデータフィールドを特定したりできます。

1. Analytics テーブルで特定のレコードのデータをダンプする場合は、ビューを開いてレコードを選択しておきます。
2. **ツール > 16 進ダンプ** を選択します。
3. ファイルのデータをダンプする場合は、**ファイル** を選択し、**開く** ダイアログ ボックスで対象ファイルを選択して **開く** をクリックします。

Analytics ソース データ ファイル (.fil) または別のファイルタイプを選択できます。

4. 省略可能。 **スキップバイト数**]には、1 以上の値を入力します。データのダンプ前に、ここで指定したバイト数をファイルの開始位置からスキップします。
5. 省略可能。 **列**]で、出力列の幅をバイトで指定します。

メモ

指定された値は、Analytics レコードまたはテーブルに含まれるバイト数を示します。

出力におけるエンコードされた文字数は、レコードまたはテーブルにおける文字数とは1対1で対応しない場合もあります。たとえば、数字 1 の16進エンコードは **31** です。

縦方向に表示した場合の各列のデフォルト数は16バイトであるのに対し、横方向に表示した場合の1列のデフォルト数は64バイトです。指定できる最大バイト数は255です。

6. 省略可能。 **水平** を選択すると、文字エンコードが、縦方向である隣接するブロック(デフォルト)でなく、横方向である行に表示されます。
7. 表示させない文字エンコード(**HEX**、**ASCII**、**EBCDIC**、または**Unicode**(Unicode Analytics のみ) のオプション選択を解除します。
8. 特定の値を検索する場合は、次を実行します。
 - a. **検索**]をクリックします。
 - b. **検索**]テキスト ボックスに、検索文字列を入力します。
 - c. 検索する文字エンコード(**ASCII**、**EBCDIC**、**HEX**) を選択します。
 - d. 検索で大文字と小文字を区別する場合は、 **大文字と小文字を無視**]を選択します。

- e. **検索の開始位置**]パネルでは、ファイルの先頭から検索する場合は **先頭**]を、現在のカーソル位置から検索を開始する場合は **カーソル位置**]を選択します。現在選択されている位置は、**ダンプ**]ダイアログボックスの **位置**]フィールドに表示されています。
 - f. **検索**]をクリックします。
値が見つかった場合、その文字エンコードの場所が強調表示されます。
9. **閉じる**]  をクリックして **ダンプ**]ダイアログボックスを終了します。

テーブルの履歴の表示

Analytics では、Analytics コマンドの結果として作成される Analytics テーブルに関する情報を記録しています。テーブルの履歴には、次のような情報が含まれます。

- テーブルが作成された日時
- 元テーブルおよび新しい出力テーブルの名前
- テーブルの作成に使用したコマンド
- コントロール合計

テーブルの履歴情報を表示するには

1. 履歴を表示させたいテーブルを開きます。
2. [ツールテーブル履歴]の順にクリックします。
3. この履歴情報を印刷するには、履歴表示領域で右クリックし [印刷]を選択します。

ワークスペースを使用したフィールド定義の共有

Analytics ワークスペースは、ほかのテーブルで再利用するために保存されたフィールド定義を1つ以上含んでいる、Analytics プロジェクト項目です。ワークスペースをアクティブにすると、ワークスペース内のフィールドが現在のテーブルで使用できるようになります。ワークスペースには物理フィールド、演算フィールド、またはフィルター(**式ビルダー**で**フィルター**の一覧から選択できます) の定義を保存して再利用することができます。フィールド定義やフィルターを再利用または共有すれば整合性が保証され、複数のテーブルで同じフィールドやフィルターを使用する際は、繰り返し定義する作業が軽減されます。

ワークスペースは、同じ名前と同じデータ型のフィールドが存在するテーブル間で共有することができます。ワークスペースは、たとえば、期間別や1つの会社の部門別の売掛金テーブルなど、特定の種類の一連のテーブルと関連付ける場合などに使用します。

レコードタイプが複数あるファイルの場合、各レコードタイプの定義を別々のワークスペースに保存することができます。この場合、特定のタイプのレコードを処理するために、適切なワークスペースを選択することができます。

ワークスペースを作成する

1. 特定の Analytics テーブルからワークスペースへフィールド定義を追加する場合は、ワークスペースを新規作成する前にそのテーブルを開いておく必要があります。
2. **ファイル > 新規作成 > ワークスペース** の順に選択します。
3. **ワークスペースへのフィールドの追加** ダイアログボックスで、次のいずれかの作業を実行します。
 - **すべて追加** をクリックすると、すべてのフィールドがワークスペースに追加されます。
 - **使用可能なフィールド** リストで個々のフィールドをクリックし、右矢印ボタンをクリックしてワークスペースに追加する。 **使用可能なフィールド** リストで複数のフィールドを
 - **Ctrl** キーを押しながらクリックし、右矢印ボタンをクリックしてワークスペースに追加する。
 - **式** をクリックして**式ビルダー**を開き、ワークスペースに追加する式を作成する。

メモ

ほかの演算フィールドを参照する演算フィールドを追加する場合は、そのような依存関係を持つ演算フィールドを追加する前に、依存関係を持たない(演算フィールドを参照しない)演算フィールドを追加しておく必要があります。

4. **OK** をクリックします。
5. **概要** タブで、ワークスペースファイルを右クリックして **名前の変更** を選択します。

6. テーブルレイアウトの新しい名前を入力し、Enter キーを押します。

メモ

ワークスペースの名前は、英数字 64 文字が上限です。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

ワークスペースを編集する

ワークスペースは編集が可能です。フィールド定義を追加したり、既存のフィールド定義を編集または削除したりすることができます。

1. 特定の Analytics テーブルからワークスペースへフィールド定義を追加する場合は、ワークスペースの編集を開始する前にそのテーブルを開いておく必要があります。
2. ナビゲーターの **総覧**] タブで、編集対象のワークスペース ファイルを右クリックし、**編集**] を選択します。
3. **ワークスペース エディター** でエントリを編集します。フィールド定義のテキストを編集することで、エントリを編集または削除できます。
4. 以下の手順を実行して、ワークスペースにフィールドを追加します。
 - a. ワークスペース ツールバーにある **ワークスペースへのフィールドの追加** ] をクリックします。
 - b. **ワークスペースへのフィールドの追加**] ダイアログ ボックスで、次のいずれかの作業を実行します。
 - **すべて追加**] をクリックして、すべてのフィールドをワークスペースに追加する。
 - **使用可能なフィールド**] リストで個々のフィールドをクリックし、右矢印ボタンをクリックしてワークスペースに追加する。
 - **使用可能なフィールド**] リストで複数のフィールドを
 - **Ctrl** キーを押しながらクリックし、右矢印ボタンをクリックしてワークスペースに追加する。
 - c. **OK**] をクリックします。
5. ナビゲーターの **総覧**] タブで、そのワークスペース ファイルを右クリックし、**閉じる**] を選択します。
6. 確認ダイアログ ボックスで、**はい**] をクリックします。

ワークスペースのアクティブ化

ワークスペースは Analytics テーブルで使用するためにアクティブ化できますが、その場合、ワークスペースのフィールド定義で参照されるフィールドは必ず Analytics テーブルに存在し、使用可能である必要があります。たとえば、ワークスペースに `value` という名前の演算フィールドがあり、このフィールドに `sale_price * quantity` という式が定義されている場合、そのワークスペースを使用するテーブルには `sale_price` と `quantity` の両方のフィールドが必ず含まれていなければなりません。

テーブルのフィールドと同じ名前のフィールドが含まれるワークスペースをアクティブ化すると、そのテーブルのフィールドを置き換えるかどうかを確認するメッセージが表示されます。[はい]をクリックすると、そのテーブルが閉じられるまでは、テーブルのフィールドが一時的にワークスペースのフィールドに置き換えられます。

ワークスペースをアクティブ化した後にテーブルレイアウトを編集した場合、またはアプリケーションで自動的にテーブルレイアウトを保存するような変更を行った場合には、Analytics によって、ワークスペースのフィールドがテーブルレイアウトに永続的に追加されます。ワークスペースのフィールドをテーブルレイアウトに保存すると、そのフィールドをビューに追加できます。

1. ワークスペースを使用するテーブルを開きます。
2. ナビゲーターの **総覧** タブで、そのワークスペースを右クリックし、**アクティブ化** を選択します。

ワークスペースのフィールド定義をテーブルレイアウトに追加する

デフォルトで、ワークスペースのフィールドが Analytics テーブルで使用できるは、その Analytics テーブルが開かれ、ワークスペースがアクティブ化されたときだけです。テーブルを閉じると、ワークスペースは自動的に非アクティブ化され、再びアクティブ化されるまでそのフィールドは利用できなくなります。テーブルが開かれたときに、ワークスペースのフィールドが必ず使用されるようにするには、それらのフィールドをテーブルレイアウトに追加しておく必要があります。ワークスペースのフィールドをテーブルレイアウトに追加する場合は、ワークスペースから定義をコピーし、テーブルレイアウトでフィールドを作成します。このため、今後これらのフィールドに対する変更はワークスペースではなくテーブルレイアウトで実行する必要があります。

1. ワークスペースのフィールド定義を追加するテーブルを開きます。
2. ナビゲーターの **総覧** タブで、そのワークスペースを右クリックし、**アクティブ化** を選択します。
3. **編集 > テーブルレイアウト** の順にクリックします。
4. フィールド リストから 1 つのフィールドをダブルクリックして、フィールドのプロパティを編集します。
5. **幅** テキスト ボックスの数値を 1 増やして変更します。
6. **入力を受け入れる** をクリックします。

変更を保存することで、Analytics はそのフィールド定義への変更を保存し、また、アクティブ化されたワークスペースのフィールド定義もテーブルレイアウトに保存します。

7. 手順の 4 と 5 で編集したフィールドをダブルクリックして、**幅** テキスト ボックスの値を元の値に戻します。
8. **入力を受け入れる** をクリックします。
9. **閉じる** をクリックしてテーブルレイアウト ダイアログ ボックスを終了します。

別の Analytics プロジェクトからワークスペースをコピーする

ある Analytics プロジェクトから別のプロジェクトへワークスペースをコピーすることができます。これにより、そのワークスペースに含まれる物理フィールド、演算フィールド、またはフィルターを一から作成するのではなく、再利用することが可能になります。作業の手間を省いたり、これらの項目を再利用したり、また他の

Analytics ユーザーと共有したりできることに加え、整合性も保証されます。同時に単一のワークスペース、または複数のワークスペースをコピーできます。

Analytics プロジェクトの外部に個別のファイルとして存在するワークスペースをインポートしたい場合は、"ワークスペースをインポートする" 下を参照してください。

1. コピーされたワークスペースを格納するプロジェクトを開きます。
2. ナビゲーターの **総覧**] タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**別のプロジェクトからコピー > ワークスペース**] を選択します。

Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。

3. **プロジェクト ファイルの場所**] ダイアログ ボックスで、ワークスペースのコピー元となる Analytics プロジェクトを探して選択し、**開く**] をクリックします。
4. **インポート**] ダイアログ ボックスで、次のいずれかの作業を行って、1 つ以上のワークスペースを **インポート先 < プロジェクト名 >**] リストに追加します。
 - ワークスペースをダブルクリックします。複数のワークスペースを
 - **Ctrl キー**を押しながらクリックし、右矢印ボタンをクリックする。
 - **すべて追加**] をクリックして、すべてのワークスペースを追加する。

インポート先 プロジェクト名] リストからワークスペースを削除するには、個々のワークスペースをダブルクリックするか、複数のワークスペースを **Ctrl キー**を押しながらクリックした後に左矢印ボタンをクリックするか、または **すべてクリア**] をクリックします。

5. **OK**] をクリックして、コピー先のプロジェクトにワークスペースをコピーします。

プロジェクト内に同じ名前のワークスペースが既に存在する場合は、コピーしたワークスペース名の末尾に増分数字が付けられます。

ワークスペースをインポートする

個別の .wsp ファイルとして Analytics プロジェクトの外部に存在するワークスペースをインポートすることができます。これにより、そのワークスペースに含まれる物理フィールド、演算フィールド、またはフィルターを再作成するのではなく、再利用することが可能になります。作業の手間を省いたり、これらの項目を再利用したり、また他の Analytics ユーザーと共有したりできることに加え、整合性も保証されます。一度にインポートできるワークスペースは 1 つだけです。

別の Analytics プロジェクトからワークスペースをインポートしたい場合は、"別の Analytics プロジェクトからワークスペースをコピーする" 前のページを参照してください。

1. ナビゲーターの **総覧**] タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**プロジェクト項目のインポート > ワークスペース**] を選択します。

Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。

2. **プロジェクト**] ダイアログ ボックスで、ワークスペース ファイル(.wsp) を探して選択し、**開く**] をクリックします。
3. **確認** ダイアログ ボックスで、**OK**] をクリックします。

プロジェクトにワークスペースがインポートされます。プロジェクト内に同じ名前のワークスペースが既に存在する場合は、インポートしたワークスペース名の末尾に増分数字が付けられます。

ワークスペースをエクスポートする

ワークスペースは、Analytics プロジェクトの外部に保存される個別の .wsp ファイルとしてエクスポートすることができます。個別のファイルとしてエクスポートされたワークスペースは、後で任意の Analytics プロジェクトにインポートすることができます。これにより、そのワークスペースに含まれる物理フィールド、演算フィールド、またはフィルターを一から作成するのではなく、再利用することが可能になります。一度にエクスポートできるワークスペースは 1 つだけです。

1. ナビゲーターの **総覧**] タブでワークスペースを右クリックし、**プロジェクト項目のエクスポート**] を選択します。
2. **名前を付けて保存**] ダイアログボックスで、ワークスペースを保存する場所を選択し、必要であればワークスペースの名前を変更して、**保存**] をクリックします。確認ダイアログボックスで **OK**] をクリックします。

指定した場所にワークスペースがエクスポートされます。

メモ

ワークスペース名を 64 文字の英数字 (拡張子 .wsp を含まない) に制限し、ワークスペースが Analytics にインポートされる時に名前が切り捨てられないようにします。

名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。先頭に数字を使用することもできません。特殊文字、スペース、先頭の数字はすべて、ワークスペースのインポート時にアンダースコア文字に置換されます。

ワークスペース ノートを追加または編集する

あるワークスペースに関して、今後の参考のために保持しておきたい、あるいは他のユーザーのために文書化しておきたい情報がある場合は、ノート (注釈) をそのワークスペースに追加して記録することができます。ワークスペース ノートの内容はいつでも編集できます。

ノートを追加、編集、削除または読むために、ワークスペースをアクティブ化する必要はありません。

1. ナビゲーターの **総覧**] タブで、作業対象のワークスペースを右クリックします。
2. **プロパティ**] を選択します。
3. **ワークスペース プロパティ**] ダイアログボックスで **ノート**] タブをクリックします。
4. 新しいノートを入力するか、または既存のノートを編集します。

ノートを削除する場合は、テキストをすべて削除します。

5. **OK**] をクリックしてダイアログボックスを閉じ、変更を保存します。

データ フィルターについて

印刷イメージ(レポート) ファイルや複数レコード タイプのファイルなど、レコード タイプが複数存在するデータソースを定義する場合には、使用するレコード タイプを随時識別できるようにする必要があります。Analytics では、データ フィルターを使用してファイル内の異なるレコード タイプを識別し、必要に応じて不要なレコードを除外します。

データ フィルターを使用すれば、データ ファイルのある特定の部分を識別し、それを Analytics テーブルのレコードおよびフィールドとして定義することができます。データ フィルターの定義には、データ ファイルのどの部分をレコードに含めるか、あるいはレコードから除外するかを示す条件を使用します。

新しいデータ フィルターの追加] タブで新しいデータ フィルターを作成しようとすると、デフォルトで "すべて除外" 条件が作成されています。この条件により、ソース データ ファイルの文字データはすべて除外され、すべての文字が黒の背景色で強調表示されています。

レコードを定義するには、1 文字または一連の文字を選択して、最低 1 つのフィルター条件を指定する必要があります。このフィルター条件を使用して、ACL テーブルに含めるレコードの各インスタンスを一意に識別することができます。データ要素をクリックすると、その要素はフィルター条件と見なされ、その内容がフィルター条件テキスト ボックスに追加されます。あるレコードを含めるフィルター条件を作成すると、そのフィルター条件を満たすすべての行で黒の背景色の強調表示が解除されます。選択したレコードの中で除外したい部分がある場合は、レコードの一部を除外するためのフィルター条件を別に指定できます。

1 つのデータ ファイルに対して複数のデータ フィルターを定義することができます。一般的なレポートでは、ある種目(製品クラスなど)に関する情報が含まれる 1 件のヘッダーレコードが出力されます。ヘッダーレコードに続いてその種目に関連付けられている多くの詳細レコードが出力されます。

この例で、ヘッダーレコードには、製品クラス番号や製品クラス名などの詳細レコードに関連する補足情報が含まれています。この情報を詳細レコードと結合するには、ヘッダーレコードのフィールドを静的フィールドとして定義する必要があります。これを行うには、**フィールド/式の編集**] タブで各フィールドに対し **静的**] チェックボックスをオンにします。新しいデータ フィルターを作成するときには、そのレコード タイプの情報を後続レコードに関係付けるかどうかを尋ねられます。**[はい]** を選択すると、そのレコードに定義したフィールドでは、**静的**] チェックボックスがデフォルトでオンになっています。フィールドごとに手動で **静的**] チェックボックスをオンにすることもできます。

データ フィルターの作成

データ フィルターは、データ ファイルで単独レコードまたは複数レコードを特定するために使用します。データ フィルターの作成では、Analytics テーブルに含めるデータ ファイルの領域を定義する規則、および Analytics テーブルに含めないデータを除外する規則を定める必要があります。プレビュー テーブルで、黒の背景色で強調表示されるデータは除外され、白の背景色で表されるデータは含まれることを示します。

アクティブなデータ フィルターがある場合、そのフィルターは新しいフィルターを作成したときに、自動的に非アクティブ化されます。

既存のデータ フィルターを編集する場合、**テーブルレイアウト** ダイアログ ボックスでそのフィルターを演算フィールドとして編集することができます。**テーブルレイアウト**] ダイアログ ボックスでデータ フィルターを表

示させるには、**オプション**ダイアログボックス(**ツール > オプション**)の **インターフェイス** タブにある **フィールド一覧にフィルターを含める** オプションを選択しておく必要があります。

手順

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **新しいデータフィルターの追加**]タブをクリックします。

データファイルの内容は、画面の下半分のデータプレビュー領域に表示されます。デフォルトで、すべてのデータが除外として表示されています。これは、データフィルターリストの "すべて除外" が適用されていることを示します。このフィルターは変更も削除もできません。

3. レコード内で1文字または一連の文字を選択して、フィルター条件を定義します。単独の文字をクリックして強調表示させる、あるいは複数の文字をクリックとドラッグで選択して **包含**]をクリックすると、指定した条件と一致するすべてのレコードが選択されます。

たとえば、Analytics テーブルに含めるデータファイルの各行の71バイト目に小数点があるとなります。この小数点をフィルターに含める必要があります。除外したい行でも小数点と同じ位置にあった場合は、不正に選択される行を除外する規則を作成する必要があります。

4. 選択されているレコードの一部を除外する場合は、そのレコード内で除外する1文字または一連の文字を選択してフィルター条件を定義し、 **除外**]をクリックします。
5. **入力を受け入れる**] をクリックすると、指定した条件を含むデータフィルターが作成されます。
6. **フィルターを保存する名前**]ダイアログボックスで、フィルターの名前を入力して **OK**]をクリックします。
7. **このフィルターをアクティブにしたままにする**]ダイアログボックスで **はい**]をクリックして、レコードのフィールド定義を開始します。そのフィルターを後で適用する場合は **いいえ**]をクリックします。後で行う場合は、 **フィールド/式の編集**]タブのフィルタードロップダウンリストからそのフィルターを選択します。

そのフィルターがアクティブになると、 **フィールド/式の編集**]タブのデータプレビューで、除外されたデータファイル領域が黒の背景色で強調表示されます。このプレビューテーブルで背景色が白色の領域から個別にフィールドを選択し、レコードに追加することができます。

データフィルターをアクティブ化する

テーブルレイアウト ダイアログボックスの **新しいデータフィルターの追加**]タブでデータフィルターを作成する場合、そのデータフィルターをアクティブ化するかどうかを尋ねられます。また、**テーブルレイアウト** ダイアログボックスの **フィールド/式の編集**]タブにあるドロップダウンリストから適切なオプションを選択することにより、既存のフィルターをアクティブ化したり、アクティブなフィルターを変更したり、選択したフィルターを非アクティブにしたりすることもできます。

データフィルターがアクティブ化されると、そのデータフィルター条件を満たすレコードおよびフィールドのみがビューに抽出され処理することができます。

データフィルターがアクティブ化されると、 **フィールド/式の編集**]タブでは以下のような動作が見られます。

- 定義したフィールドがアクティブなデータフィルターと関連付けられます。
- テーブルのフィールドはデータフィルターの条件に対して評価されます。条件を満たさないフィールドは、**フィールド** リストに表示されません。

- フィルター条件を満たさない行は、データプレビューテーブルで黒色で強調表示されます。
- ドロップダウンリストから "すべてのフィールド" オプションを選択すると、アクティブになっていたフィルターが非アクティブになり、テーブルに定義されたすべてのフィールドが表示されます。

手順

1. **編集 > テーブルレイアウト**]の順にクリックします。
2. **フィールド/式の編集**]タブで、フィールド リストの上にあるドロップダウンリストから、アクティブ化するデータフィルターを選択します。

フィールド リストには、選択したデータ フィルターに属するフィールドのみが表示されます。データ フィルターを非アクティブ化するには、ドロップダウン リストから "すべてのフィールド" を選択します。

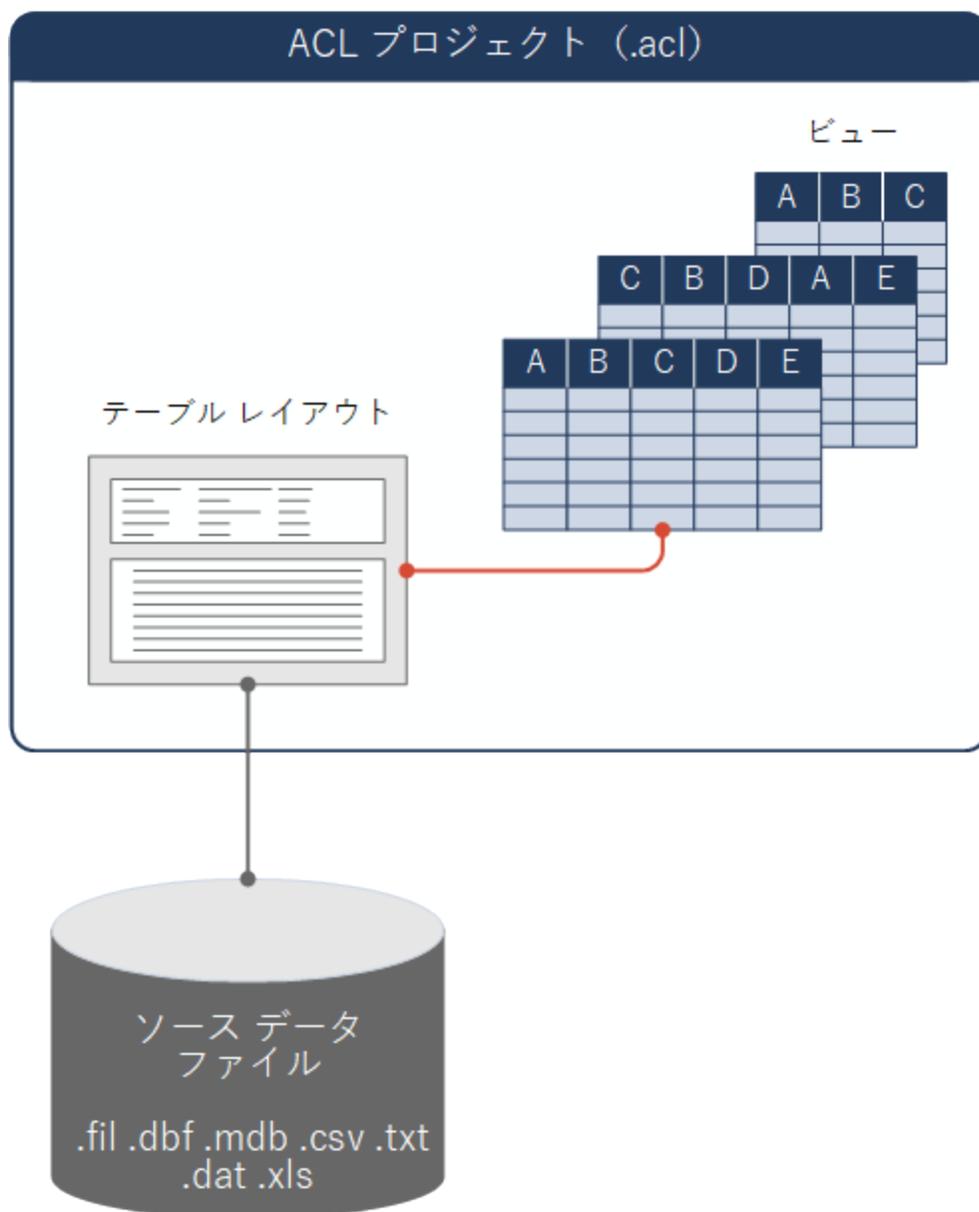
テーブルビューでデータを表示する

Analytics のテーブルビューでは、Analytics テーブルデータが表示される方法を決めることができます。ビューは Analytics 表示領域に表示され、名前付きの列と番号付きのレコードを含んでいます。ビュー内の列は関連するテーブルレイアウト内のフィールド (物理データフィールドまたは演算フィールドのどちらか) を表しています。

テーブルビューと Analytics テーブルに関連付ける方法

以下の図は Analytics テーブルの他のコンポーネントに関連したビューを示しています。

ビューが一般にどのように Analytics テーブルに関連しているかについての情報は、"Analytics テーブルの構造" ページ 118を参照してください。



ビューは柔軟性を提供します

ビューは、関連付けられている Analytics テーブルの中で見たいデータの列だけを表示したり、分析のための最も効果的な方法で列を整理したりすることができる柔軟なツールです。

同じテーブルに対して、フィールドの選択や配置が異なる複数のビューを作成することができます。たとえば、テーブルに含まれる 100 フィールドのうち 5 フィールドにのみ関心がある場合は、該当する 5 フィールドだけを含むビューを作成し、最も便利と考える順番でフィールドを配列することができます。

また、ビューを使用して、Analytics レポートのデータを書式設定することもできます。

Analytics デフォルト_ビュー

Analytics で初めて新しいテーブルを開くときは、"デフォルト_ビュー" というビューが自動的に作成されます。"デフォルト_ビュー" には、関連付けられたテーブルレイアウトに定義されているすべてのフィールドが、レイアウト内に表示される順番で含まれます。

変更なしでデフォルト ビューを使用したり、必要に応じて変更したり、変更したものを新しいビューとして保存したり、デフォルト ビューに基づかない新しいビューを作成したりすることができます。

ビューとテーブルレイアウトの関係

ビューは、元となるテーブルレイアウトに関連付けられますが、**ビューを変更してもテーブルレイアウトは変わりません**。たとえば、ビューから列を削除しても、テーブルレイアウトから対応するフィールドは削除されません。

テーブルレイアウトに加えた変更は、既存のビューに自動的に表示されません。たとえば、テーブルレイアウトに新しい演算フィールドを作成したり、新しい物理データフィールドを定義したりした場合、そのフィールドを必要とするビューには、手動で追加する必要があります。

テーブルレイアウトからフィールドを削除すると、自動的に関連するすべてのビューから列を削除します。

ビューでの作業

以下のセクションでは、ビューで実行できるさまざまな処理について説明します。ほとんどの場合、Analytics の複数の場所で処理を実行できます。

ビュー処理を実行できる場所：

- **ビュー]タブ**
- **ナビゲーターの 総覧]タブ**
- **テーブルプロパティ]ダイアログ ボックス**
- **プロジェクト プロパティ]ダイアログ ボックス**

ビューの作成

初めて Analytics テーブルを開くときには、**デフォルト_ビュー**という名前のビューが自動的に作成されます。このビューには、テーブルレイアウトに定義されているすべてのフィールドが、デフォルトのプロパティで含まれます。

このデフォルト ビューを変更して、別の名前で保存することによって新しいビューを作成できます。あるいは、新しい空のビューを作成し、列を追加して設定を定義することもできます。

以下の手順は、新しい空のビューを作成し、列を追加する方法を示しています。

方法の説明

1. ビューを作成するテーブルを開きます。
2. ビュー タブの下部にあるビューのボタンを右クリックし、**新規作成]**を選択します。
3. **ビューの追加]**ダイアログ ボックスで、ビューの名前を入力して **OK]**をクリックします。

メモ

ビューの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

4. **列の追加]**ダイアログ ボックスで、次のいずれかの作業を実行します。
 - **すべて追加]**をクリックして、テーブルレイアウトのすべてのフィールドをビューに追加する。
 - **使用可能なフィールド]**リストで個々のフィールドをクリックし、右矢印ボタンをクリックしてビューに追加する。
使用可能なフィールド]リストで複数のフィールドを
• **Ctrl** キーを押しながらクリックし、右矢印ボタンをクリックしてビューに追加する。
 - **式]**をクリックして**式ビルダー**を開き、ビューに追加する式または演算フィールドを作成する。
 - **選択済みのフィールド]**リストにフィールドを追加した後、式を使ってその列を変更したい場合は、フィールドを選択し、**編集]**をクリックして**式ビルダー**を開く。

5. (省略可能) **選択したフィールド**]リストで、1つ以上のフィールドを選択し、上矢印  または下矢印  を使用して、フィールドを再配置します。
6. **対象テーブル**]ドロップダウンリストでテーブルを選択し、入れたいフィールドを **選択済みのフィールド**]リストに追加することによって、任意の関連テーブルからフィールドを追加することができます。
7. **OK**]をクリックします。

ビューを開く

初めて Analytics テーブルを開くときには、**デフォルト_ビュー**という名前のビューが自動的に作成され、開きます。このビューにはテーブルレイアウトに定義されているすべてのフィールドが含まれています。

複数のビューが関連付けられているテーブルを開くときは、前回アクティブであったビューが自動的に開きます。

また、テーブルに関連付けられたビューを手動で開くこともできます。

方法の説明

1. **ナビゲーター**で、開きたいビューがあるテーブルをダブルクリックします。
Analyticsは、前回アクティブであったビューを表示します。
2. 別のビューを開くには、**ビュー**]タブの下部にある該当する **ビュー**]ボタンをクリックします。
ビューが開き、アクティブなビューのボタンが強調表示されます。

ビューへの変更の保存

ビューを変更した場合、その変更を既存のビューに保存するか、または変更したビューを新しいビューとして保存し、元のビューは変更されていない状態のままにしておくことができます。

既存のビューでの変更の保存

方法の説明

1. **ファイル > プロジェクトを上書き保存**]の順にクリックします。
2. 確認ダイアログボックスで、**はい**]をクリックします。
以下の操作を行った場合は、既存のビューに変更を保存するように求められます。

- 別のビューに切り替える
- テーブルを閉じる
- 別のテーブルを開く

新しいビューとして変更を保存する

方法の説明

1. ビュー タブの下部にある強調表示されているビューのボタンを右クリックし、**名前を付けて保存**]を選択します。
2. **ビューを別名で保存**]ダイアログ ボックスで、新しいビューの名前を入力して **OK**]をクリックします。

新しいビューが作成されて、自動的に開きます。元のビューは変更されません。

ビュー名を変更する

ビューの目的や、ビューに表示される情報をわかりやすくするために、ビューの名前を変更することができます。

ビューの名前の変更は、ビュータブ、またはナビゲーターの **総覧**]タブを使用して行えます。多数のビュー名を変更する必要がある場合、**概要**]タブがより便利です。

メモ

ビューの名前は、64文字までの英数字に制限されます。名前にはアンダースコア文字 (_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

ビュー]タブを使用したビュー名の変更

方法の説明

1. 名前を変更したいビューのあるテーブルを開きます。
2. ビュー タブの下部にあるビューのボタンを右クリックし、**名前の変更**]を選択します。
3. **ビュー名の変更**]ダイアログ ボックスで新しい名前を入力して、**OK**]をクリックします。

概要]タブを使用したビュー名の変更

方法の説明

1. 名前を変更したいビューのあるテーブルを開きます。
2. ナビゲーターの **総覧**]タブでテーブルを右クリックし、**プロパティ**]を選択します。
3. **ビュー**]タブをクリックします。
4. 名前を変更するビューを選択し、**名前変更**]をクリックします。
5. **ビュー名の変更**]ダイアログ ボックスで新しい名前を入力して、**OK**]をクリックします。
6. (省略可能) 名前を変更する他のビューで手順を繰り返します。
7. **OK**]をクリックして **テーブルプロパティ**]ダイアログ ボックスを閉じます。

ビューをコピーする

ビューをコピーして、同一のビューを同じ Analytics テーブルと関連付け、その後、コピーしたビューを変更することができます。Analytics プロジェクト内のテーブル間でビューをコピーすることもできます。

ヒント

ビューをコピーして変更する方が、新しいビューを一から作成するよりも簡単な場合があります。

ビューのコピーは、ビュー タブ、またはナビゲーターの **総覧** タブを使用して行えます。複数のコピーを作成する必要がある場合は、**概要** タブがより便利です。

ビュー]タブを使用したビューのコピー

方法の説明

1. コピーしたいビューのあるテーブルを開きます。
2. ビュー タブの下部にあるビューのボタンを右クリックし、**名前を付けて保存**]を選択します。
3. **ビューを別名で保存**]ダイアログ ボックスで、コピーされたビューの名前を入力して **OK**]をクリックします。

概要]タブを使用したビューのコピー

方法の説明

1. コピーしたいビューのあるテーブルを開きます。
2. ナビゲーターの **総覧**]タブでテーブルを右クリックし、**プロパティ**]を選択します。
3. **ビュー**]タブをクリックします。
4. コピーするビューを選択し、**コピー**]をクリックします。
増分する数値のサフィックスが付いたビューのコピーが作成されます。
5. (省略可能) 手順を繰り返し、同じビューの複数のコピーを作成するか、別のビューをコピーします。
6. **OK**]をクリックして **テーブルプロパティ**]ダイアログ ボックスを閉じます。

別のプロジェクトのテーブルからビューをコピーする

メモ

テーブル間でビューをコピーする場合、コピー先のテーブルに、ビューに指定されているすべてのフィールドが含まれていない場合は、「未定義」として1つ以上のフィールドが列挙されたエラー メッセージを受け取ります。そのビューを使用することはできませんが、未定義フィールドに関連付けられているビュー列は表示されません。

方法の説明

1. コピーされたビューを格納するテーブルを開きます。
2. ナビゲーターの **総覧**]タブで Analytics プロジェクトを右クリックし、**プロパティ**]を選択します。
Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。
3. **ビュー**]タブをクリックします。
4. コピーするビューを選択し、**適用**]、**OK**]の順にクリックします。

別の Analytics プロジェクトからビューをコピーする

ある Analytics プロジェクトから別のプロジェクトへビューをコピーすることができます。これは、両方のプロジェクトに類似したテーブルがあり、新しいビューを一から作成するのではなく、ビューを再利用したい場合に役立ちます。同時に単一のビュー、または複数のビューをコピーできます。

コピー先のプロジェクトでテーブルが開いている場合、コピーしたビューは、その開いているテーブルに自動的に関連付けられます。テーブルが開いていない場合は、コピーしたビューはプロジェクトに追加され、後でテーブルに関連付けることができます。

コピーしたビューが、プロジェクト内の既存のビューと同じ名前である場合は、コピーしたビュー名の末尾に増分数字が付けられます。

メモ

テーブル間でビューをコピーする場合、コピー先のテーブルに、ビューに指定されているすべてのフィールドが含まれていない場合は、「未定義」として1つ以上のフィールドが列挙されたエラーメッセージを受け取ります。そのビューを使用することはできるかもしれませんが、未定義フィールドに関連付けられているビュー列は表示されません。

方法の説明

ビューをコピーする

1. コピーされたビューを格納するプロジェクトを開きます。
2. 次のいずれかを実行します。
 - コピーしたビューとテーブルを直ちに関連付けたい場合は、そのテーブルを開きます。
 - コピーしたビューとテーブルを後で関連付けたい場合は、コピー先のプロジェクト内のテーブルがすべて閉じていることを確認します。
2番目の方法は、複数のビューをコピーして、さまざまなテーブルと関連付けるつもりである場合に使用します。
3. ナビゲーターの **総覧** タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**別のプロジェクトからコピー > ビュー** を選択します。
Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。
4. **プロジェクト ファイルの場所** ダイアログ ボックスで、ビューのコピー元とする Analytics プロジェクトを探して選択し、**開く** をクリックします。
5. **インポート** ダイアログ ボックスで、次のいずれかの作業を行って、1つ以上のビューを **インポート先 < プロジェクト名 >** リストに追加します。
 - ビューをダブルクリックする。
複数のビューを
 - **Ctrl** キーを押しながらクリックし、右矢印ボタンをクリックする。

- **すべて追加]**をクリックして、すべてのビューを追加する。
[インポート先 < プロジェクト名 >]リストからビューを削除するには、個々のビューをダブルクリックするか、複数のビューを **Ctrl** キーを押しながらクリックした後に左矢印ボタンをクリックするか、または **すべてクリア]**をクリックします。
- 6. **OK]**をクリックして、コピー先のプロジェクトにビューをコピーします。
テーブルが開いている場合、ビューはそのテーブルに関連付けられます。

ビューの関連付け

テーブルとの関連付けをしないで1つ以上のビューをコピーした場合は、次のようにしてビューとテーブルを関連付けます。

1. 適切なテーブルを開きます。
2. ナビゲーターの **総覧]** タブで Analytics プロジェクトを右クリックし、 **プロパティ]** を選択します。
3. **ビュー]** タブをクリックします。
4. テーブルと関連付けるビューを選択し、 **適用]**、 **OK]** の順にクリックします。

ビューをインポートする

個別の **.rpt** ファイルとして Analytics プロジェクトの外部に存在するビューをインポートすることができます。これにより、新しいビューを一から作成するのではなく、ビューを再利用することが可能になります。一度にインポートできるビューは1つだけです。

ビューをインポートするときにプロジェクトでテーブルが開いている場合、インポートしたビューは、その開いているテーブルに自動的に関連付けられます。テーブルが開いていない場合は、インポートしたビューはプロジェクトに追加され、後でテーブルに関連付けることができます。

インポートしたビューが、プロジェクト内の既存のビューと同じ名前である場合は、インポートしたビュー名の末尾に増分数字が付けられます。

メモ

インポートしたビューと関連付けるテーブルに、ビューに指定されているすべてのフィールドが含まれていない場合は、「未定義」として1つ以上のフィールドが列挙されたエラーメッセージを受け取ります。そのビューを使用することはできるかもしれませんが、未定義フィールドに関連付けられているビュー列は表示されません。

方法の説明

ビューをインポートする

1. ナビゲーターの **総覧]** タブで、Analytics プロジェクトのエントリを右クリックし、 **プロジェクト項目のインポート > ビュー]** を選択します。
Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。
2. **プロジェクト]** ダイアログボックスで、ビューファイル(**.rpt**) を探して選択し、 **開く]** をクリックします。

3. 確認ダイアログ ボックスで、**OK**]をクリックします。

プロジェクトにビューがインポートされます。テーブルが開いている場合、ビューはそのテーブルに関連付けられます。

ビューの関連付け

テーブルとの関連付けをしないでビューをインポートした場合は、次のようにしてビューとテーブルを関連付けます。

1. 適切なテーブルを開きます。
2. ナビゲーターの **総覧**]タブで Analytics プロジェクトを右クリックし、**プロパティ**]を選択します。
3. **ビュー**]タブをクリックします。
4. インポートしたビューを選択し、**適用**]、**OK**]の順にクリックします。

ビューをエクスポートする

ビューは、Analytics プロジェクトの外部に保存される個別の **.rpt** ファイルとしてエクスポートすることができます。個別ファイルとしてエクスポートされたビューは、Analytics プロジェクトに後からインポートすることができます。これにより、新しいビューを一から作成するのではなく、ビューを再利用することが可能になります。一度にエクスポートできるビューは1つだけです。

方法の説明

1. ビューに関連付けられているテーブルを開きます。
2. ナビゲーターの **総覧**]タブでテーブルを右クリックし、**プロパティ > ビュー**]を選択します。
3. エクスポートするビューを選択して **エクスポート**]をクリックします。
4. **名前を付けて保存**]ダイアログ ボックスで、ビューを保存する場所を選択し、必要であればビューの名前を変更して、**保存**]をクリックします。確認ダイアログ ボックスで **OK**]をクリックします。
指定した場所にビューがエクスポートされます。

メモ

ビュー名を 64 文字の英数字 (**.rpt** 拡張子を含まない) に制限し、ビューが Analytics にインポートされるときに名前が切り捨てられないようにします。

名前にはアンダースコア文字 (**_**) を使用できますが、他の特殊文字やスペースは使用できません。先頭に数字を使用することもできません。特殊文字、スペース、先頭の数字はすべて、ビューのインポート時にアンダースコア文字に置換されます。

5. **OK**]をクリックして **テーブルプロパティ**]ダイアログ ボックスを閉じます。

ビューを削除する

いつでも必要に応じて、テーブルに関連付けられている個々のビューを削除することができます。ただし、テーブルには関連付けられているビューが必ず1つはなければなりません。Analytics は、テーブルに関連付けられている最後のビューが削除されないようにしています。

ビューを削除しても、テーブルレイアウトや関連するデータファイルへの影響はありません。ビューの削除は、ビュータブ、またはナビゲーターの **総覧**]タブを使用して行えます。

ビュー]タブを使用したビューの削除

方法の説明

1. 削除したいビューのあるテーブルを開きます。
2. ビュータブの下部にあるビューのボタンを右クリックし、**削除**]を選択します。
3. 確認ダイアログボックスで、**削除**]をクリックします。

概要]タブを使用したビューの削除

方法の説明

1. 削除したいビューのあるテーブルを開きます。
2. ナビゲーターの **総覧**]タブでテーブルを右クリックし、**プロパティ**]を選択します。
3. **ビュー**]タブをクリックします。
4. 削除するビューを選択し、**削除**]をクリックします。

ビューは、現在アクティブである場合には削除できません。

5. 確認ダイアログボックスで、**削除**]をクリックします。
6. **OK**]をクリックして **テーブルプロパティ**]ダイアログボックスを閉じます。

ビュー内の列のカスタマイズ

Analyticsビューの個々の列を、画面上で表示する方法やAnalyticsレポートで表示したり処理したりする方法をカスタマイズすることができます。

以下を指定できます。

- ビューに含まれるテーブルレイアウトの列
- 列の順序
- 数値が個別の列に表示される方法
- 列の表示名
- 印刷レポートの列プロパティ

ビューに列を追加する

ビュー内の列として、テーブルレイアウトに定義されている物理データフィールドまたは演算フィールドのいずれかを追加することができます。

また、**式ビルダー**を使用して作成された、一時的に作成した式に基づく列をビューに追加することもできます。

方法の説明

1. 次のいずれかを実行します。
 - 選択した列の左側に1つまたは複数の列を挿入する場合は、ビュー内で1つの列見出しを右クリックする。
 - 最後の列の後ろに1つまたは複数の列を挿入する場合は、ビュー内で(列見出しを選択しないで)任意の表示領域を右クリックする。
2. **列の追加**]をクリックします。
3. **列の追加**]ダイアログボックスで、次のいずれかの手順を実行します。
 - **すべて追加**]をクリックして、すべての列をビューに追加する。
 - **使用可能なフィールド**]リストで個々のフィールドをクリックし、右矢印ボタンをクリックしてビューに追加する。
 - **使用可能なフィールド**]リストで複数の列をCtrlキーを押しながらクリックし、右矢印ボタンをクリックしてビューに追加する。
 - **式**]をクリックして**式ビルダー**を開き、ビューに追加する式を作成する。
 - **選択済みのフィールド**]リストに列を追加した後、式を使ってその列を変更したい場合は、列を選択し、**編集**]をクリックして**式ビルダー**を開く。
4. そのテーブルに関連付けられているテーブルが1つ以上ある場合は、**対象テーブル**]ドロップダウンリストで関連テーブルを選択し、入れたいフィールドを**選択済みのフィールド**]リストに追加することによって、任意の関連テーブルからビューへフィールドを追加することができます。
5. **OK**]をクリックします。

ビューからの列の削除

ビューから不要な列を削除できます。列の基になっている物理データフィールドや演算フィールドは、テーブルレイアウトに存在し続けるので、いつでも必要に応じて列を再度追加することができ、他のビューでも変わらず使用できます。

方法の説明

1. [ビュー] タブで、削除する列の列見出しをクリックします。
隣接する複数の列見出しを選択するには、**Shift** キー + クリックを、隣接していない複数の列見出しを選択するには、**Ctrl** キー + クリックを利用できます。
2. ビューのデータ領域を右クリックし、**選択列の削除**] をクリックします。

ヒント

複数の列を選択している場合は、列見出し行を右クリックしないでください。そうしないと、右クリックした列だけが選択されます。

3. **列の削除**] ダイアログボックスで、**削除**] をクリックします。
4. ビューに加えた変更を保存するには、**ファイル > プロジェクトを上書き保存**] の順にクリックし、確認ダイアログボックスで **はい**] をクリックします。

ビュー内の列のサイズを変更する

ビュー内のセルは、単独、複数、およびすべての単位でサイズ変更できます。

方法の説明

次のいずれかを実行します。

単一の列を手動でサイズ変更するには	その列見出しの右端の区切り線上にカーソルを置き、希望のサイズになるまでドラッグします。
列の内容の幅に合わせて1つ以上の列のサイズを自動的に変更するには	列見出しを選択し、右側の列区切りのいずれかをダブルクリックします。 隣接する複数の列見出しを選択するには、 Shift キー + クリックを、隣接していない複数の列見出しを選択するには、 Ctrl キー + クリックを利用できます。
列の内容の幅に合わせてビューのすべての列のサイズを自動的に変更するには	ビューのデータ領域を右クリックし、 すべての列のサイズの変更] をクリックします。

ビュー内の列の順番変更

ビューで列が表示される順番を変更することができます。複数の列の順番を変更する場合は、各列を個々に選択し、それを適切な場所まで移動させる必要があります。

方法の説明

1. 移動する列の見出しをクリックして、目的の場所までドラッグします。
選択した列を移動させたい場所にあたる、2つの列の間の区切り線の近くまで列をドラッグする必要があります。
2. 移動したいその他すべての列を再配置したら、**ファイル > プロジェクトを上書き保存**の順にクリックして、ビューへの変更を保存します。

ビュー内の列名の変更

ビュー内で表示される列の名前は必要に応じて変更することができます。この列名の変更では表示上の名前が変更されるだけで、テーブルレイアウトにおけるそのフィールドの名前には影響がありません。

テーブルに関連付けられているすべてのビューで使用されるデフォルトの列名を変更したり、あるいは特定のビューで使用される列名を変更したりすることができます。

メモ

個々のビュー内で指定される列名はデフォルトの列名よりも優先されます。

デフォルト列名の変更

方法の説明

1. **編集 > テーブルレイアウト**の順にクリックします。
2. **フィールド/式の編集**タブで名前を変更する列名をダブルクリックします。
3. **代替列見出し**フィールドの列名を変更します。
4. **入力を受け入れる**  をクリックし、**閉じる**  ボタンをクリックして **テーブルレイアウト** ダイアログボックスを終了します。

テーブルに関連付けられているすべてのビューの中で、そのデフォルトの名前を使用している列は自動的に更新されます。ビュー内で個別に列名を指定(変更)していた場合は、その列名が優先されるので更新されません。

特定のビューで使用される列名を変更する

方法の説明

1. 対象の列見出しを右クリックし、**プロパティ**を選択します。
2. **代替列見出し**フィールドで列名を変更し、**OK**をクリックします。

列名をビューレベルの名前からデフォルト名に変更する

方法の説明

1. 対象の列見出しを右クリックし、**プロパティ**を選択します。
2. **代替列見出し**フィールドで名前を削除し、**OK**をクリックします。
3. ビュー間で表示を切り替える、またはテーブルを一旦閉じて開き直すと、列名がリフレッシュされます。

4. 変更保存の確認メッセージで [はい] をクリックすると、その変更がテーブルに保存されます。

ビューにおける数値の書式

ビューおよびレポートで表示される数値に適用される書式は、Analytics の 3 つの異なるレベルで設定することができます。

- アプリケーションレベル(グローバル)
- フィールドレベル
- 列レベル

メモ

これらの書式設定オプションは、数値データ型を使用するフィールドの数値に適用されます。文字データ型を使用するフィールドの数値には適用されません。

レベル	優先	書式を設定する場所	説明
アプリケーションレベル	該当なし	<ol style="list-style-type: none"> [ツール > オプション] の順にクリックします。 デフォルトの数値書式で形式を選択するか指定します。 	フィールドレベルまたは列レベルの書式が指定されていない、Analytics 内のすべての数値フィールドおよび列の書式を指定します。
フィールドレベル	アプリケーションレベル	<ol style="list-style-type: none"> テーブルを開きます。 編集 > テーブルレイアウト] の順にクリックします。 フィールド名をダブルクリックします。 フィールド/式の編集] タブで、書式] で形式を選択または指定します。 	<p>テーブルレイアウト内の個々の数値フィールドと、そのテーブルレイアウトを使用するすべてのビュー内の関連する列の書式を指定します。ただし、列レベルの書式が指定されている列を除きます。</p> <p>フィールドレベルの書式の変更をビューで有効にするには、関連する列を削除してからビューに追加し直すか、またはその列を含んでいる新しいビューを作成する必要があります。</p>
列レベル	アプリケーションレベル フィールドレベル	<ol style="list-style-type: none"> テーブルを開きます。 ビューで、列見出しを右クリックし、プロパティ] を選択します。 列の修正] ダイアログボックスで、書式] で書式を選択または指定します。 	<p>個々のビューの、個々の数値列の書式を指定します。</p> <p>テーブルのビューが複数ある場合は、さまざまなビューで同一の列を異なる書式設定にすることができます。たとえば、印刷レポート用のビューにはドル記号付きの列を表示し、分析に使用するビューではドル記号を除くことができます。</p>

数値書式の構文

Analytics で数値に適用される書式は、それぞれの数値に必要なレイアウトを定義する書式マスクを使用して指定されています。たとえば、書式マスク \$99 は、100ドル未満の任意の値を表示するために使用できます。各 9 は、0 から 9 までの任意の数字を表示できることを示し、値の横にドル記号が表示されます。

デフォルトの書式マスク

以下の表は、Analytics で使用可能なデフォルトの書式マスクの一覧です。また、各マスクが数値 -100234.56 を表示する方法を示します。

定義されているままのデフォルト書式マスクを使用したり、それらを要件に合うように変更したり、独自の書式マスクを定義したりすることができます。

デフォルトの書式マスク	表示 -100234.56:
-999999.99	-100234.56
-9,999,999.99	-100,234.56
(9,999,999.99)	(100,234.56)
-\$9,999,999.99	-\$100,234.56
(\$9,999,999.99)	(\$100,234.56)
9,999,999.99-	100,234.56-

書式マスク要素

書式マスクは、次の構成要素を使用して定義されます。

コンポーネント	説明
桁数プレースホルダー	<p>数字 9 は、0 から 9 までの 1 桁を表示できる各場所を指定するために使用されます。</p> <p>必要以上の桁数のプレースホルダーを指定した場合、余分な桁と、その間にあるカンマなどの記号はすべて表示されません。たとえば、書式マスクを \$9,999 と指定した場合、値 310 は \$310 と表示されます。</p> <p>専用の書式を使用する場合は、列に表示される可能性のある最大桁数の書式を指定してください。なぜなら、指定した桁数を越えた分の桁は、左端の 9 の左にそのまま続けて、分離記号を付加しないで追加されます。たとえば、書式マスクが <code>-9,999.00</code> である場合、<code>1000000.00</code> の値は誤って <code>1000,000.00</code> と書式設定されず(1)の後に桁区切り記号がありません)。</p>
負数のインジケータ	<p>負数のインジケータは次のようになります。</p> <ul style="list-style-type: none"> 数値の前後のマイナス記号: <code>-100.00</code> または <code>100.00-</code> 数値の前後の CR: <code>CR100.00</code> または <code>100.00CR</code> 括弧: <code>(100)</code> <p>負の値のインジケータが指定されていない場合は、デフォルトで数値の前のマイナス記号が使用されます。</p>
千の位の桁区切り(該当する場合)	<p>大きな数には、数を読みやすくするために、桁の集まりの間に書式が適用されていることがよくあります。最も一般的な区切り記号は、カンマ(100,000.00) またはスペース(100 000.00) です。</p>

コンポーネント	説明
	一部の地域の設定では、区切り記号としてピリオドを使用し、小数を表すためにカンマを使用しています。Analytics で使用されるデフォルト値は、 [オプション] ダイアログボックス(ツール > オプション)の 数値 タブにある、 桁区切り記号 テキストボックスに指定されている値を変えることによって変更できます。
小数点インジケータ	ほとんどの場合、小数点を表すためにピリオドが使用されますが、一部の地域の設定ではカンマが使用されます。 Analytics で使用されるデフォルト値は、 [オプション] ダイアログボックス(ツール > オプション)の 数値 タブにある、 小数点記号 テキストボックスに指定されている値を変えることによって変更できます。
値インジケータの記号 (該当する場合)	-表示される値の種類を識別するために、ドル記号やパーセント記号などを書式に追加することができます。

列プロパティの変更

Analytics ビューの各列には多数のプロパティがあり、これらを設定して、データが画面上やビューから生成されたレポートで表示される方法を変更できます。列用に設定されたプロパティによって、テーブルレイアウト内の設定や、そのテーブルに関連付けられている他のビューの設定は変更されません。

名前を付けて保存 コマンドを使ってビューのコピーを作成した場合、ビュー内の列のプロパティはコピーされますが、その後どちらか一方のビューの列プロパティに加えられた変更は、変更が行われたビューに対してのみ適用されます。

1. ビュータブで、列見出しを右クリックして **プロパティ** を選択するか、または列見出しをダブルクリックします。
2. ガイドとして以下のテーブルを使用し、**列の修正** ダイアログボックスで 1 つ以上の変更を行って、**OK** をクリックします。

列の型	オプション	説明
任意	列の内容	列に表示される値を修正できます。 列の内容 をクリックし、 式ビルダー を使用して式を作成または編集します。値の表示のみが修正され、物理データは修正されません。 列の内容 の式は、列の正しいデータ型を返す必要があります。たとえば、数値列の式は数値を返す必要があります。 式で参照されているすべてのフィールドは、テーブルレイアウトに存在する必要があります。
	代替列見出し	ビューの列の表示名のテキスト。 メモ この変更は現在のビューでのみ適用されるもので、 テーブルレイアウト ダイアログボックスで指定されたデフォルトの列見出しを無効にします。詳細については、" ビュー内の列名の変更 " ページ 851 を参照してください。

列の型	オプション	説明
	幅	画面またはレポートの列の表示幅委。幅を文字で入力します。 数値列の場合は、最大桁数と比較して、値を表示するのに十分な幅であることを確認する必要があります。全数値を画面に表示できない場合は、番号記号の文字列(#####)が、エラーを示すために表示されます。
	キー列を並べ替える	レポート出力のみ 列は、レポート出力でデータを並べ替えるために使用されます。 昇順で並べ替え] または 降順で並べ替え] を選択し、並べ順を選択します。 <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> メモ [レポート]ダイアログボックスで、事前並べ替え]を使用する必要があります。 </div>
数値	書式	列に数値を表示するために使用される形式。 書式] ドロップダウンリストから適切な書式を選択するか、またはカスタム書式を入力します。 書式が指定されていない場合は、 [デフォルトの数値書式] の値が使用されます。この値は、 [オプション] ダイアログボックス([ツール > オプション])の 数値] タブで指定されています。
	合計を出力しない	レポート出力のみ 列の値が合計されないようにします。 デフォルトでは、Analytics は自動的にレポート内の数値フィールドの合計を計算します。フィールドに単価などの合計を出しても意味のないデータが含まれている場合は、この動作を止めさせることができます。
	ゼロの場合は空欄にする	レポート出力のみ ゼロ値の代わりに空白のエントリを用います。 列に多数のゼロ値を含むレポートは、ゼロ以外の値のみが表示されると読みやすくなります。
文字 (ビューの一番左列である必要があります)	ブレーク列	レポート出力のみ 列の値が変わるたびにサブセクションを作成して、すべての数値フィールドの小計を計算します。 複数のブレーク列を指定して、ネストされたサブセクションを作成できます。すべてのブレーク列は、ビューの左側で、最初の非ブレーク列の前にまとめて配置する必要があります。 このオプションは、 並べ替えキー列] オプションがオンになっている場合のみ有効になります。 <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> メモ [レポート]ダイアログボックスで、事前並べ替え]を使用する必要があります。 </div> ビューのブレーク列の右側には、太い点線の区切りが表示されます。

列の型	オプション	説明																																																
		<table border="1"> <thead> <tr> <th></th> <th>Vendor State</th> <th>Vendor City</th> <th>Vendor Name</th> <th>Vendor Number</th> <th>Invoice Amount</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>CA</td> <td>Los Angeles</td> <td>More Power Industries</td> <td>11663</td> <td>618.30</td> </tr> <tr> <td>2</td> <td>IA</td> <td>Des Moines</td> <td>NOVATECH Wholesale</td> <td>13808</td> <td>6,705.12</td> </tr> <tr> <td>3</td> <td>LA</td> <td>Sheveport</td> <td>Koro International</td> <td>12433</td> <td>7,955.46</td> </tr> <tr> <td>4</td> <td>CA</td> <td>Los Angeles</td> <td>More Power Industries</td> <td>11663</td> <td>4,870.83</td> </tr> <tr> <td>5</td> <td>MN</td> <td>Minneapolis</td> <td>Stroud & Sons</td> <td>12130</td> <td>10,531.71</td> </tr> <tr> <td>6</td> <td>UT</td> <td>Salt Lake City</td> <td>United Equipment</td> <td>13411</td> <td>5,734.00</td> </tr> <tr> <td>7</td> <td>LA</td> <td>Sheveport</td> <td>Koro International</td> <td>12433</td> <td>7,196.00</td> </tr> </tbody> </table>		Vendor State	Vendor City	Vendor Name	Vendor Number	Invoice Amount	1	CA	Los Angeles	More Power Industries	11663	618.30	2	IA	Des Moines	NOVATECH Wholesale	13808	6,705.12	3	LA	Sheveport	Koro International	12433	7,955.46	4	CA	Los Angeles	More Power Industries	11663	4,870.83	5	MN	Minneapolis	Stroud & Sons	12130	10,531.71	6	UT	Salt Lake City	United Equipment	13411	5,734.00	7	LA	Sheveport	Koro International	12433	7,196.00
	Vendor State	Vendor City	Vendor Name	Vendor Number	Invoice Amount																																													
1	CA	Los Angeles	More Power Industries	11663	618.30																																													
2	IA	Des Moines	NOVATECH Wholesale	13808	6,705.12																																													
3	LA	Sheveport	Koro International	12433	7,955.46																																													
4	CA	Los Angeles	More Power Industries	11663	4,870.83																																													
5	MN	Minneapolis	Stroud & Sons	12130	10,531.71																																													
6	UT	Salt Lake City	United Equipment	13411	5,734.00																																													
7	LA	Sheveport	Koro International	12433	7,196.00																																													
	改ページ	<p>レポート出力のみ</p> <p>ブレイク列値が変化するたびに改ページを挿入します。</p> <p>このオプションは、ブレイク列]オプションがオンになっている場合のみ有効になります。</p>																																																
	重複表示の排除	<p>レポート出力のみ</p> <p>ブレイク列に繰り返し現れる値の代わりに空白のエントリを用います。</p> <p>たとえば、請求書レコードごとに顧客名が記載されている場合、各顧客名の最初のインスタンスだけを記載すれば、レポートが読みやすくなるかもしれません。</p> <p>このオプションは、ブレイク列]オプションがオンになっている場合のみ有効になります。</p>																																																

ビューからのデータのコピー

Analytics ビューからデータをコピーし、別のアプリケーションにそれを貼り付けることができます。ビューからのデータのコピーは、Analytics からデータをエクスポートするもう1つの手段です。場合によっては、データをコピーする方が、より速くて簡単なオプションかもしれません。たとえば、ビューが **クイックソート 昇順** または **クイックソート 降順** メニューコマンドを使用して並べ替えられているときに、データのサブセットを選択します。

ビューからデータをコピーするには

1. 次のいずれかを実行します。
 - ビュー タブでクリックしてドラッグし、コピーしたい値のグループを強調表示します。
 - コピーする列の列見出しをクリックし、複数の列をコピーしたい場合は、**Ctrl** キーを押しながらその他の列見出しをクリックして、列を選択します。
2. ビューのデータ領域を右クリックし、**コピー** をクリックします。複数の列を選択している場合、または値のグループを強調表示している場合は、列見出しを右クリックしないでください。そうしなければ、右クリックした列だけが選択されます。

データがクリップボードにコピーされます。スプレッドシートやテキスト ファイルなど、データのコピー先となるアプリケーションを開き、そのアプリケーションの貼り付けコマンドを使用して、データを挿入することができます。

ビューからのグラフの生成

Analytics のビューでデータを選択し、グラフを生成することができます。数値の列を複数選択するか、あるいは文字または日付時刻の列を1つと1つ以上の数値の列を選択することができます。

ビューのデータをグラフ化するには

1. グラフ化したいデータが含まれるビューを開きます。
2. 次のいずれかを実行します。
 - [ビュー] タブで、グラフ化の対象データ範囲をクリックおよびドラッグして選択します。
 - 隣接している複数の列をまとめて選択するには、最初の列の列見出しをクリックし、次に **Shift** キーを押したまま、最後の列の列見出しをクリックします。選択した最初の列と最後の列の間にある列もすべて選択されます。
 - 隣接していない複数の列を選択するには、**Ctrl** キーを押したまま、それら選択対象の列の列見出しをクリックします。
3. ビュー内で右クリックし、**選択データのグラフ化**] をクリックします。

メモ

複数の列または値のグループを選択している場合は、列見出しを右クリックしないでください。列見出しを右クリックすると、直前の選択が解除され、その列だけが選択されてしまいます。

選択データのグラフ化]メニュー項目が無効な場合は、選択したデータ領域または列がグラフ化できないことを意味します。

グラフウィンドウが開き、グラフが表示されます。グラフウィンドウではグラフプロパティを編集することができます。ただし、グラフの作成に使用したデータのサブセットは既にビューで選択(強調表示)されたものであるため、ドリルダウンコマンド使用できません。

ビューでの無効なデータの表示方法

Analytics は、ビュー内の無効なデータを強調表示するために、次のようなはっきりと識別できる文字シーケンスを使用します。

- ##### - この一連の番号記号 (#) は、列の幅が小さいため、全桁数を表示しきれない数値を識別するために使用されます。

このような値は無効として強調表示されるため、画面上に不正確な値が表示される(たとえば、10100 の代わりに 100 が表示される)ことはありません。しかし、この値が原因で、その値が含まれているレポートでずれが生じます。レポートでは完全な値が印刷されるため、隣接する列がずれる結果となります。

この問題は、すべての桁数を表示できるよう、列の **幅** プロパティの値を増やすことによって解決できます。

- ### ERR ### - この文字シーケンスは、列内の数値が無効であるか未定義である場合に表示されます。

数値のオーバーフローは、ゼロによる除算や、非常に小さな値による除算などの演算によって発生します。値を修正するか、もしくは、ビューを基にレポートを生成する前に、ビューからその列を削除する必要があります。

数値が列の幅よりも大きく、値が無効か未定義である場合は、列幅を広げない限り、番号記号の文字シーケンス(#####)が表示されます。列幅を広げると、「### ERR ###」文字シーケンスが表示されるようになります。

複数のテーブルを開く

Analytics では複数のテーブルを同時に開くことができ、このことは、2 つ以上のデータのセットを視覚的に比較するための便利な方法を提供します。比較を容易にするために、開いているテーブルにフィルターリングやクイックソートを適用することができます。これらの操作の結果は、テーブルを切り替えているときも保持されます。複数のテーブルを開くことはインターフェイスのみの機能です。Analytics スクリプトには適用されません。

Analytics で複数のテーブルを開くには、開いたままにしたいテーブルのビュータブをピン留めして、その後1 つ以上の追加テーブルを開きます。追加テーブルもピン留めすると、開いたままになります。ピン留めされていないテーブルは、別のテーブルを開くたびに自動的に閉じます。

追加のテーブルを開くたびに、追加のビュータブが開きます。個々のビュータブをクリックするか、ナビゲーターでテーブル名をクリックすることにより、テーブルを切り替えることができます。現在表示用に選択されているテーブルはアクティブなテーブルで、アクティブなテーブル名はビュータブとナビゲーターで太字で表示されます。

手順

Analytics では一度に複数のテーブルを開くことができます。各テーブルは別々のビュータブで表示され、テーブル間で切り替えることができます。テーブルはすべて、同じ Analytics プロジェクト内に含まれている必要があります。

1. ナビゲーターで、1 番目に開くテーブルをダブルクリックします。
2. ピンアイコン  をクリックすると、そのビュータブがピン留めされます。
3. さらにテーブルを開いていく場合は、テーブルを開くたびにピンアイコン  をクリックしてから次のテーブルを開きます。

開いたテーブル間を移動するには、それぞれのビュータブをクリックするか、ナビゲーターで表示されるテーブル名をクリックしてください。開いているテーブルのアイコン上には緑色のドット(点)  が付けられています。ナビゲーターでは、アクティブなテーブル(現在表示されているテーブル)の名前は太字で表示されます。

4. 固定されたテーブルを閉じるには、該当する [ビュー] タブで  をクリックします。
5. すべての開いているテーブルを閉じるには、[ビュー] タブの右側にある **すべてのタブを閉じる**  をクリックします。

複数のテーブルを開くことの利点

同時に複数のテーブルを開くことができる一般的な利点は、状況によっては、最初に結合や関連付けのような Analytics 操作を使用してテーブルを処理しなくても、有用な分析を実行できるということです。

たとえば、禁止業者(つまり怪しい業者カテゴリコード)の T&E 取引の再調査を検討してください。2 つのテーブルがあり、1 つには T&E 取引、もう 1 つには禁止業者が含まれているとします。2 つのテーブルは業

者名で結合することができますが、結合を実行する前に、2つのキーフィールドを手動で一致させるために多くの準備作業が必要となります。あるいは、両方のテーブルを開き、禁止業者リストを参照として使用して、取引テーブルから禁止業者の出現を識別するフィルターを作成することができます。適切な場合は、あいまい検索のコンポーネントをT&E取引のフィルターリングに組み込むことができます。次にフィルターの2つの例を示します。

- `ISFUZZYDUP(vendor_name, "Diamond Casino", 5, 99)` は、Diamond Casinoの正確な出現とあいまい重複を分離します。
- `MATCH(mc_code, "7298", "7995", "9222")` は、業者カテゴリコード7298(Health and Beauty Spas)、7995(Betting)および9222(Fines)の正確な出現を分離します。

詳細については、「ISFUZZYDUP()関数」ページ2319と「MATCH()関数」ページ2352を参照してください。

複数のテーブルを開くことに関する追加情報

機能分野	詳細
一般的なテーブルの動作	1つのテーブルだけが開いていようと、複数のテーブルが開いていようと、テーブルは同じように動作します。
テーブル状態の保持	<p>テーブルを切り替えるとき、それぞれのテーブルの状態は保持されます。テーブル状態には次の要素が含まれます。</p> <ul style="list-style-type: none"> ○ フィルターリング ○ クイックソート ○ レコード選択 ○ 列、レコードまたはフィールドの強調 ○ スクロールバーの位置 ○ アクティブなビューの選択 ○ アクティブなインデックスの選択
Analytics 操作	Analytics の操作はアクティブなテーブルのみで実行されます。
ビュータブを閉じてテーブルを閉じる	<ul style="list-style-type: none"> ○ テーブルを含むビュータブを閉じるとテーブルも閉じます。閉じたテーブル上ではどのような操作も実行できません。 <p>バージョン 10.0 より前の Analytics では、ビュータブを閉じると、ビューからテーブルデータを削除しても、テーブルは開いた状態のままであり、データに対して操作を実行し続けることができました。</p>
主テーブルと副テーブル (コマンドラインを含む Analytics インターフェイスのみに適用されます。Analytics スクリプトには適用されません。)	<ul style="list-style-type: none"> ○ 1つ以上の開いたテーブルには、関連付けられた副テーブルと主テーブルを指定できます。それぞれの主と副の関連付けは一意であり、他の主と副の関連付けには影響を与えません。 ○ 異なった主テーブルが同じ副テーブルを持つことができます。 ○ ビュータブで開いているテーブルを副テーブルとして開くことはできません。副テーブルとして開く前に、まずそのテーブルを閉じる必要があります。 ○ 副テーブルとして開いているテーブルをビュータブで開くことはできません。ビュータブで開く前に、まずそのテーブルを閉じる必要があります。 <p>メモ 副テーブルを開くことは、それを主テーブルに関連付け、処理のために利用できるようにすることを意味します。副テーブルはビュータブで開かれません。</p>

機能分野	詳細
	<ul style="list-style-type: none"> ○ 関連付けされていない、または '変動する' 副テーブルは許可されていません。 ● 主テーブルを閉じた場合は、関連付けられた副テーブルが自動的に閉じられます。 ● テーブルを副テーブルとして開くには、別のテーブルが既に開いている必要があります。 ● この制限は Analytics スクリプトには適用されません。スクリプトは、関連付けされていない副テーブルを今も許可しています。
Analytics プロジェクト	<p>開いている複数のテーブルはすべて同じ Analytics プロジェクトに含まれている必要があります。</p> <p>Analytics プロジェクトを閉じるときに複数のテーブルが開いていた場合、そのプロジェクトを再度開いたときには、アクティブなテーブルのみが自動的に開かれます。ほかのテーブルの開いた状態や、主と副の関連付けはすべて保持されません。</p>
サーバー テーブル	<p>複数テーブルモードはローカルテーブルでのみサポートされています。一度に複数のサーバーテーブルを開くことはできません。サーバーテーブルを開くときに1つ以上のローカルテーブルが開いている場合は、すべてのローカルテーブルが自動的に閉じられます。</p>
Analytics コマンド ログ	<p>複数テーブルモードでは、別のテーブルに切り替えたり、テーブルに対してアクションや操作を実行したりするたびに、Analytics コマンド ログにエントリが書き込まれます。また、このログ エントリの前に OPEN<table name> ログ エントリが置かれます。テーブルをピン留めしたり、後続の操作を何も行わずにテーブルを切り替えたりした場合には、ログ エントリは作成されません。複数のテーブルが開いているとき、この先行するログ エントリが、操作の対象となるテーブルを識別します。</p> <p>テーブルをピン留めしたり、後続の操作を何も行わずにテーブルを切り替えたりした場合には、ログ エントリは作成されません。</p>
データ定義 ウィザード	<p>データ定義ウィザードを使用してデータのインポートとテーブルの定義を行うときは、開いているテーブルはすべて、ピン留めされているかどうかにかかわらず自動的に閉じられます。開いているテーブルをすべて閉じることにより、必要に応じてテーブルを上書きできるようにします。</p> <p>コマンド ラインから IMPORT コマンドを実行すると、開いているテーブルを閉じません。この方法でテーブルを上書きしようとしたときに、そのテーブルが開いていてアクティブであると、コマンドは失敗してエラーメッセージが表示されます。</p>
スクリプト	<ul style="list-style-type: none"> ○ Analytics スクリプトを使用して複数のテーブルを開くことはできません。複数テーブルモードは、インターフェイスのみの機能で、ACL スクリプトには相当するものではありません。 ○ Analytics インターフェイスでテーブルをピン留めして開いたままにしても、CLOSE コマンドによってテーブルが閉じられないようにすることはできません。 ○ Analytics スクリプトを実行するときは、開いているテーブルはすべて、ピン留めされているかどうかにかかわらず自動的に閉じられます。アクティブなテーブルと関連付けられた副テーブル(存在する場合)は、スクリプトが別のテーブルを開くことを指定しない限り、自動的に再度開かれます。 <p>アクティブなテーブルとそれに関連する副テーブルを自動的に再度開くことにより、スクリプトを実行する前にテーブルを手動で開いておくことを前提とするスクリプトの使用をサポートします。</p>

複数行にわたるレコードの書式設定

各レコードが複数行に表示されるように、ビューのレイアウトをカスタマイズすることができます。この機能を使用すると、知りたいすべての情報を画面に表示できるため、追加情報を見るためにスクロールさせなければならない状況を回避できます。たとえば、知りたい列を多数含んでいるテーブルがあり、1つの列に詳細な情報を提供するノートが付いている場合は、1つの行にデータ列を表示し、別の行にノートを表示することができます。また、ビューから生成されたレポートのレイアウトの最適化にこの機能を利用できます。たとえば、テーブルに2つの住所フィールド(address_1 と address_2)がある場合、レポートの address_1 フィールドの下の方に address_2 フィールドを1列に並べることによって、フィールドが横に並んでいる場合よりも読みやすくすることができます。

ビューのレコードを複数行にわたるように書式設定するには

1. レコード番号列で、任意のレコードの下部の線の上にカーソルを置きます。レコード番号列とは、ビューの最初の列の左側にある灰色の領域のことです。
2. クリックして、カーソルを下方へドラッグするとレコードに行が追加され、上方へドラッグすると空白行が削除されます。1つのレコードの行数を変更すると、その変更はテーブル内のすべてのレコードに適用され、列見出しも変更を反映させるために同等の行数によって調整されます。
3. 列のデータを別の行へ移動するには、列見出しで対象の列をクリックし、それを追加したい列見出し内の行へドラッグします。列は行の一番左の位置、ただし、その行に既にいくつか列がある場合はその後追加されます。
4. 行内の各列の位置は、列の始めと終わりを示す縦線(境界線)の上にマウスカーソルを置き、その線をクリックして新しい位置へドラッグすることによって変更します。
5. ビューには列見出しのすべての行を表示したくないが、ビューを印刷するときは列見出しの一部を出力しないようにしたい場合は、どの列見出しを可視にするかを変更できます。たとえば、1行目には"Address" ヘッダーのエントリがあり、2行目には"Address_2" ヘッダーのエントリがある場合、通常は2行目を出力しないようにしたいでしょう。可視行を変更するには、ビューの右側にあるスクロールバーの上の黒い線の上にマウスカーソルを置き、カーソルをクリックして上方へドラッグし、含めない列見出しを隠します。列見出しには少なくとも1行はなければならないため、ヘッダーの先頭行の列を非表示にすることはできません。
6. 列および列見出しの位置を設定し終わったら、既存のビューまたは新しいビューに変更を保存します。

分析用のデータの準備

分析する目的のデータに対して、初期の準備作業を実施しなければならない場合があります。場合によっては、インポートした後すぐに、データの分析を開始できます。ただし、多くの場合、次の目的で、1つ以上の準備タスクを実行する必要があります。

- 最終的に分析するデータセットを形成する
- 結果が信頼できることを保証する

準備は、分析の元になる基盤として考えます。適切な基礎は、効果的で信頼できる結果を得るために欠かせません。

一般的な準備タスク

一般的な準備タスクの一部は次のとおりです。

- **変換** - 特定の Analytics コマンドの入力として使用できるようにフィールドのデータ型を変換するか、他のフィールドと合わせる
- **クリーンアップと正規化** - 出力結果が信頼できるように、入力データをクリーンアップして正規化する
- **結合** - 複数のテーブルから結合する
- **サンプル** - レコードのサンプルを抽出する。大きいデータセットのすべてのレコードを検査する時間または予算がない場合があるためです。

これらのタスクはすべて、目的の分析を実行する前に、絶対に必要な最初のステップになることがあります。

データの検証

データの準備の他に、分析するデータの完全性と有効性も検証してください。無効なデータが少しあるだけでも、すべての後続の分析が無効になり、貴重な時間とリソースが無駄になる可能性があります。

式の使用

式は値と演算子を組み合わせたもので、これにより計算を実行したり結果を返したりすることができます。

式は柔軟で役立つツールです。次のことができます。

- さまざまな計算を実行する
- フィルターを作成する
- 分析のデータを準備する
- 演算フィールドを作成する

式の内容

式には、データフィールドや、関数、リテラル、定数、および変数を含めることができ、算術演算子または論理演算子を使ってこれらを組み合わせることができます。

式は手動で直接入力することも、また**式ビルダー**で作成することもできます。この式ビルダーは Analytics のあらゆる場所で式を作成するために提供される標準ユーティリティです。

式の複雑さ

式は必要に応じて、単純な式にも複雑な式にもなり得ます。

単純な式では基本的な算術演算の結果を返します。

例：

```
数量 * 単価
```

複雑な式では、多数のフィールドや関数を参照し、演算子を使って式のパーツを結合させます。

例：

```
PROPER(first_name) + " " + PROPER(last_name)
```

これは、`first_name`(名)フィールドと`last_name`(姓)フィールドの各値をそれぞれ適切な大小文字表記(最初の文字を大文字に、残りの文字を小文字)に変換し、次にそれら2つの値の間にスペースを1つ挿入して、結合しています。

式の型

Analytics は 4 つのタイプの式をサポートします。これは、4 つのサポートされているデータ カテゴリまたはデータ型に対応します。

- 文字
- 数値
- 日付時刻
- 論理

例：

- `金額 + 1` は数値式です。数値に対して演算を実行し、数値の結果を返すためです。
- `金額 > 1` は論理式です。比較を行い、真または偽の論理結果を返します。

作成する式の内容は式タイプに対応する必要があります。

式タイプ	必要な内容	例
文字	次のいずれかを含みます。 <ul style="list-style-type: none"> ◦ 文字フィールド ◦ 文字データを含む変数 ◦ 文字値を返す関数 ◦ 引用された文字列 (文字リテラル) 	製品コードから数字を抽出し、3文字のプレフィックスを破棄します。 <ul style="list-style-type: none"> ◦ <code>SUBSTR(製品コード, 4, 10)</code>
数値	次のいずれかを含みます。 <ul style="list-style-type: none"> ◦ 数値フィールド ◦ 数値データを含む変数 ◦ 数値を返す関数 ◦ 引用符を含まないリテラル数値 - 数字とマイナス記号、必要な場合小数点 	税込み売上価格を計算します。 <ul style="list-style-type: none"> ◦ <code>売上価格 * 1.07</code> 3つのフィールドの中で最大値を見つけます。 <ul style="list-style-type: none"> ◦ <code>MAXIMUM(最小数, 最小手持ち在庫数, 注文数)</code>
日付時刻	次のいずれかを含みます。 <ul style="list-style-type: none"> ◦ 日付時刻フィールド ◦ 日付時刻データを含む変数 ◦ 日付時刻値を返す関数 ◦ 引用された日付時刻値 (日付時刻リテラル) 日付時刻データ型には、日付、日付時刻および時刻という3種類のサブタイプがあります。引用符で囲む日付時刻値の場合はバッククォートが必要です。たとえば、 <code>`20141231`</code> や <code>`20141231.235959`</code> と指定します。バッククォート(`) は、キーボードの Shift + @ キーで入力できます。	2つの日付間の経過日数を計算します。 <ul style="list-style-type: none"> ◦ <code>`20141231` - `20141130`</code> 2つの時刻フィールド間の経過時間を計算します。 <ul style="list-style-type: none"> ◦ <code>終了_時刻 - 開始_時刻</code>
論理	次のいずれかを含みます。 <ul style="list-style-type: none"> ◦ 真または偽の論理結果を生成する処理 (T または F) ◦ 論理値を返す関数 T または F が式の一部となる場合は、引用符を付けしないで	支払日が期限を過ぎたすべてのレコードを検索します。 <ul style="list-style-type: none"> ◦ <code>支払_日付 > 支払_期日</code> 3つの都市のテーブルでレコードをフィルタリングします。

式タイプ	必要な内容	例
	入力する必要があります。 メモ 論理式は、フィールド、変数、またはあらゆるデータ型のリテラルを参照できます。	<ul style="list-style-type: none"> MATCH(Vendor_City, "Phoenix", "Austin", "Los Angeles")

Analytics による式の評価方法

Analytics は次の規則に従って、式を評価します。

演算子の優先順位	算術的および論理的な優先順位は、演算子が評価される順序をsします。"Analytics 式の演算子" 下を参照してください。 演算子の評価順序を変更するには、丸かっこ () を使います。
演算子データ型	各演算子は、そのオペランドが互換性があるデータ型である場合にのみ動作します。
関数のかっこ	すべての Analytics 関数には丸かっこが必要です。評価は、最初に関数の丸かっこ内のすべてのもの、その後その関数の丸かっこの外にある式の他の部分の順に行われます。
文字列の比較	異なる長さの文字列を比較した場合、デフォルトでは、より短い文字列が採用されます。 オプション] ダイアログボックスの テーブル] タブにある 正確な文字比較を行う] オプションが有効になっている場合は、より長い文字列が採用されます。 詳細については、"テーブルオプション" ページ 125を参照してください。
小数点精度	数式で異なる精度の数値が混在した場合、計算結果には、小数点以下の桁数が最も多いオペランドの小数点以下桁数が保持されます。(このデフォルト動作は SET MATH コマンドで変更できます。) 例： <ul style="list-style-type: none"> 4 + 5.0 = 9.0 6 * 2.000000 = 12.000000 1.1 * 1.1 = 1.2 1.1 * 1.10 = 1.21 詳細については、"数式の端数処理と小数点精度の制御" ページ 874を参照してください。

Analytics 式の演算子

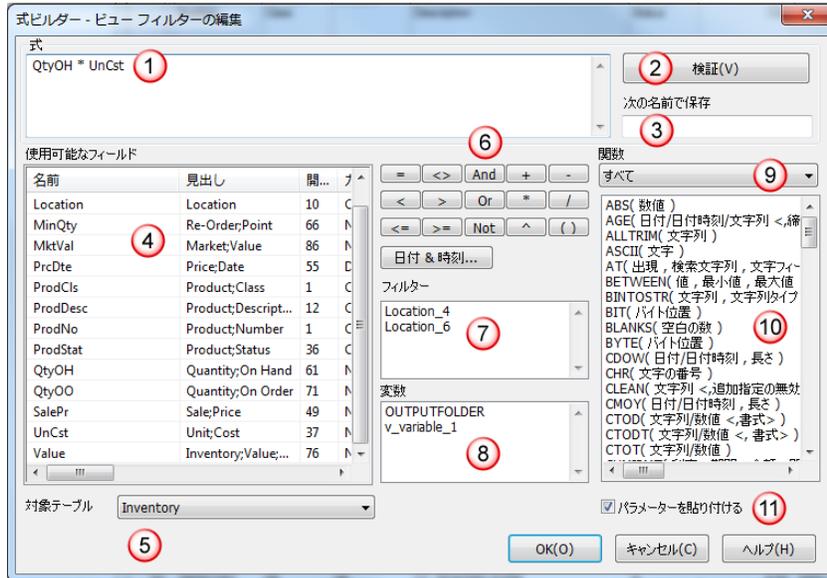
以下の表は、式を作成するときを使用できる演算子の一覧を示します。

演算子は優先順位の降順で記載されています。式に同じ優先順位の演算子がある場合、かっこを使用して特定の評価順序を指定しないかぎり、演算子は左から右に評価されます。

演算子(優先順位順に一覧表示)	説明
()	丸かっこ - 演算子の優先順位を修正するか、または関数のパラメーターを囲みます。
NOT -	論理 NOT 単項マイナス - マイナス記号。負数を示します。
^	累乗 - 数値のべき乗を求めます。
* / 同じ優先順位の演算子は左から右に評価されます。	乗算 除算
+ - 同じ優先順位の演算子は左から右に評価されます。	加算 減算
+	文字列の連結
> < = >= <= <> 同じ優先順位の演算子は左から右に評価されます。 メモ 2つの記号の演算子にはスペースを含めることができません。たとえば、>=と入力します。>=は不可です。	次の値より大きい より小さい 等しい 以上 以下 等しくない
AND(または &)	論理 AND
OR(または)	論理 OR

式ビルダーの概要

式ビルダーは、式を作成できる Analytics ユーティリティです。ユーザー定義式を作成できる場合は、式ビルダーを使用できます。



式とは何か

式とは、データフィールド、演算子、関数、フィルター、および変数などの要素を組み合わせたステートメントで、Analytics が評価して値を返すものです。

コンテキストに基づく式の確認

式ビルダーが表示される状況に応じて、Analytics は式が必要な型の値に評価されるかどうかをチェックします。たとえば、ビューのレコードのフィルターリングに使用される式を作成している場合、式は True または False の論理値として評価される必要があります。

式ビルダー ユーザー インターフェイス 要素

1	[式] テキストボックス	<p>新しい式を作成するか、既存の式を編集するためのテキスト</p> <p>式に必要な構文を手入力したり、式ビルダーに含まれるリストやボタンを使用して、必要な情報を入力することができます。</p> <p>リストとボタンを使用した場合、その情報は [式] テキスト ボックス内の現在のカーソル位置に追加されます。</p>
---	--------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------

2	\ 検証] ボタン	<p>式]テキストボックスで、式の構文を自動的に検証する</p> <p>検証]をクリックすると、式ビルダー内で式の妥当性を迅速にチェックできるので時間を節約することができます。</p>
3	テキストボックスとして保存	完全に保存された式の名前を入力するフィールド
4	使用可能なフィールド リスト	選択されているテーブル内のすべてのデータ フィールド および演算 フィールドを一覧表示します フィールド エントリをダブルクリックすると、 式] テキスト ボックスにそのエントリが追加されます。
5	テーブル]ド ロップダウン リスト	<p>関連するテーブルを選択するためのドロップダウン リスト</p> <p>式には、1 つまたは複数の関連テーブルのフィールドを含めることができます。そのフィールドが現在のビューに表示されていなくてもかまいません。また、関連テーブルからビューにフィールドを追加しようとなかろうと、関連テーブルのフィールドを含んでいるフィルターを作成し、その結果を現在のビューで見ることができます。</p>
6	演算子 ボタン、日時セレクト	演算子、日付、日付時刻、または時刻を 式] テキストボックスに追加するためのボタン
7	フィルター リスト	<p>選択テーブルに関連付けられている名前付きフィルターを一覧表示します。</p> <p>フィルター エントリをダブルクリックすると、式]テキスト ボックスにそのエントリが追加されます。</p>
8	変数リスト	<p>選択テーブルに関連付けられている変数を一覧表示します。</p> <p>変数 エントリをダブルクリックすると、式]テキスト ボックスにそのエントリが追加されます。</p>
9	関数ドロップダウンリスト	関数カテゴリを一覧表示します。これを使用して、 関数] リストに表示される関数を選別できます。
10	関数のリスト	<p>Analytics で使用可能な関数と、その必要な構文を一覧表示します。オプションのパラメーターは、山かっこで(<>) 囲まれています。関数をダブルクリックすると、式]テキスト ボックスにその関数が追加されます。関数の種類に応じて、パラメーターは定数、リテラル、フィールド名、あるいは式になり得ます。この式には、別の関数を含めることができます。</p> <p>ヒント</p> <p>関数]リストに挙げられる個々の関数の使用方法については、一覧の下の ヘルプ] ボタンをクリックしてください。</p>
11	パラメーターを貼り付ける チェックボックス	<ul style="list-style-type: none"> オン - 関数およびパラメーター プレースホルダーは、式]テキスト ボックスにコピーされます オフ - 開かっこと閉じるかっこを含む関数だけが 式]テキスト ボックスにコピーされます。

式ビルダーによる式の作成

式ビルダーは、式を作成および編集するための標準インターフェイスを提供します。式ビルダーダイアログボックスは、Analyticsでの作業時、必要に応じて表示されます。次のいずれかの操作を実行すると、式ビルダーが開きます。

- 任意のダイアログボックスにある **[f]** または **式** をクリックする。
- 表示領域の **[ビュー]** タブにある **ビューフィルター** の **編集**  ボタンをクリックする。
- **編集 > テーブルレイアウト** メニューを選択し、**フィールド/式の編集** タブにある **新しい式の追加**  をクリックしてから、**式**  をクリックする。
- **編集 > フィルター** または **編集 > 変数** をクリックし、**新規作成** をクリックする。
- **スクリプト エディター** ツールバーにある **式の挿入**  をクリックする。

[f] をクリックして、またはフィルターの編集によって式ビルダーを開いた場合は、論理式の作成に制限されます。

式を作成するには

1. **式ビルダー** ダイアログボックスで、次の1つまたは複数の手順を用いて式を作成します。

式の要素をクリックまたはダブルクリックして挿入すると、その要素は **式** テキストボックス内のカーソルのある位置に挿入されます。

- **式** テキストボックスで、式の一部または全部を手動で入力します。
- **使用可能なフィールド** リストでフィールド名をダブルクリックします。作業中のテーブルに関連付けられているテーブルがある場合は、**対象テーブル** ドロップダウンリストから、フィールドを追加するテーブルを選択できます。
- 演算子を挿入するには、数学記号または論理記号のボタンをクリックします。
- 有効な日付、日付時刻、または時刻を挿入するには、**日付 & 時刻** をクリックして **日付と時刻セクター** ダイアログボックスを使用します。
- **フィルター** リストのフィルター名、または **変数** リストの変数名をダブルクリックします。
- **関数** リストで関数名をダブルクリックします。

関数 ドロップダウンリストを使用すると、関数にフィルターをかけて種類を絞ることができます。**[パラメーターを貼り付ける]** チェックボックスがオンになっている場合は、パラメーターのプレースホルダーがコピーされるので、それを適切なフィールドまたは値に置き換える必要があります。また、オプション(省略可能)パラメーターについては、指定する場合はそれを囲んでいる山かっこ (<>) を削除し、使用しない場合はオプションパラメーター全体を削除します。

ヒント

関数 リストに挙げられる個々の関数の使用方法については、一覧の下の **ヘルプ** ボタンをクリックしてください。

2. **検証]**をクリックして、構築した式の妥当性を検査します。式の構文が正しいこと、また、式の戻り値に特定の型が要求される場合は式が正しい型の値を返すことを確実にするため、式がチェックされます。
3. 次のいずれかの手順を行って、式を作成します。
 - a. 式を永続的に保存する場合は、**次の名前**で保存]テキスト ボックスに名前を入力して **OK]**をクリックします。**式ビルダー**へのアクセス元となった場所またはテキスト ボックスに、式の名前がコピーされます。変数を作成している場合は、名前を指定する必要があります。
 - b. 式を永続的に保存しないで作成する場合は、**OK]**をクリックします。**式ビルダー**へのアクセス元となった場所またはテキスト ボックスに、式がコピーされます。

数式の端数処理と小数点精度の制御

乗算または除算が関係する計算では、Analytics は 2 つのオペランドのうち、小数点以下の桁数がより大きい側の桁数に四捨五入します。この端数処理方法は、ほとんどの数式を評価するために Analytics が使用する固定小数点演算に関連付けられています。

複数のオペランドの式 ($a*b/c$) の場合、端数処理は式の各段階で適用されます。最初の評価された段階で開始し、式が完全に評価されるまで繰り返されます。

メモ

Analytics での端数処理方法に注意しないことは、演算エラーの最も一般的な原因の 1 つです。

固定小数点演算

Analytics の端数処理動作は、Analytics が数値演算で使用する固定小数点演算 (金融関数を除く) に関連しています。Analytics は次の 2 つの理由で固定小数点演算を使用します。

- 処理速度が向上する
- ユーザー側で小数の桁数や丸めを制御することができる

乗算の丸め

式 $1.1 * 1.1$ を考えます。正しい答えは 1.21 です。ただし、Analytics は結果を小数点 1 桁まで端数処理します。これは、オペランドの小数点桁数の大きい数であるためです。

$$1.1 * 1.1 = 1.2$$

オペランドのいずれかが小数点 2 桁の場合、Analytics は結果の小数部をオペランドの大きい方の小数点桁数まで端数処理します。この特定の式では、端数処理は不要です。

$$1.10 * 1.1 = 1.21$$

除算の丸め

式 $7/3$ を考えます。正しい答えは 2.333... ですが、どちらのオペランドにも小数がないため、Analytics はこれを小数なしに丸めます。

$$7/3 = 2$$

オペランドのいずれかまたは両方に小数点がある場合、Analytics は結果の小数部をオペランドの大きい方の小数点桁数まで端数処理します。

$$7/3.00 = 2.33$$
$$7.000/3.00 = 2.333$$

小数を追加して丸めを制御する

端数処理を制御し、目的の小数点精度を得るための最も簡単な方法は、1 で式を乗算し、結果に必要な小数点精度数が続きます。たとえば、1.0000 で乗算し、結果が小数点 4 桁まで正確であることを確認します。

例

問題

次の式では、Analytics は結果を小数点 2 桁に端数処理します。これは、要件に合う精度ではありません。

$$7.21 * 2.33 = 16.80$$

$$7.21 / 2.33 = 3.09$$

解決策

結果の精度を高めるには、1 と任意の小数点精度で乗算します。

$$1.0000 * 7.21 * 2.33 = 16.7993$$

$$1.000000 * 7.21 / 2.33 = 3.094421$$

注意

式の最初に1を置きます。別の場所に1を置くと、精度調整が動作しない場合があります。最初の2つのオペランドの精度が既に端数処理を発生させているためです。

$$7.21 * 2.33 * 1.0000 = 16.8000$$

$$7.21 / 2.33 * 1.000000 = 3.090000$$

括弧を使用するときには注意してください

丸かっこを使用して数学演算の順序を指定する場合は注意してください。括弧を使用する場合、精度調整が動作しない場合があります。括弧のオペランドの精度が既に端数処理を発生させているためです。

$$1.0000 * (7.21 * 2.33) = 16.8000$$

括弧内の1を統合すると、問題を解決できます。

$$(1.0000 * 7.21 * 2.33) = 16.7993$$

複数オペランド式の端数処理動作

端数処理動作と小数点精度は、式のオペランド数に関係なく、同じように動作します。Analytics は、組み合わせで式を評価するときに、2つのオペランドの小数点桁数の大きい方に結果を端数処理します。

ただし、累積端数処理が複数オペランドの式で動作する方法のため、評価される最初の2つのオペランドによって確立された小数点制度は、式結果に適用される精度です。

累積端数処理のもう一つの特性は、小数点精度の損失が複数オペランド式の各段階で増えるということです。

小数点 2 桁の精度の例

以下の表は、Analytics が次の複数オペランド式を計算するときに、端数処理を適用する方法を示します。

$$1.1 * 1.12 * 1.123 * 1.1234 = 1.5514$$

式の最初に評価される段階の小数点桁数の大きい方は2です(1.1 * 1.12)。この小数点2桁の精度は、複数オペランド式の残りでも保持されます(赤色の桁数で表示)。

結果の違い列は、精度の累積損失が評価の連続段階でどのように増えるのかを示します。

Analytics 計算 (評価順)	Analytics 結果 (端数処理)	端数処理されてい ない計算 (評価順)	端数処理されてい ない結果	結果の違い
1.1 * 1.12	1.23	1.1 * 1.12	1.232	0.002
1.23 * 1.123	1.381	1.232 * 1.123	1.383536	0.002536
1.381 * 1.1234	1.5514	1.383536 * 1.1234	1.5542643424	0.0028643424

精度の詳細

ACL 結果(端数処理)列では、最初の2つの後のすべての小数点桁数は、対応する端数処理されていない結果と比べると、不正確です。データ分析の一部として数値計算を実行するとき、不正確さの度合いは、主な問題になることが多くあります。

端数処理された結果は、結果を生成する特定の組み合わせの式のコンテキストでは、不正確ではありません。たとえば、小数点3桁に端数処理された $1.23 * 1.123 = 1.38129$ は、ルールに従うと、1.381です。ただし、1.23は以前に1.232から端数処理されました。これは、特定の組み合わせの計算に既にある程度の不正確さが含まれていることを意味します。

小数点5桁の精度の例

以下の表は、Analytics が1.00000を加算した後に、どのように端数処理を適用し、5桁の小数点精度を指定するのかを示します。

$$1.00000 * 1.1 * 1.12 * 1.123 * 1.1234 = 1.55427$$

式の最初に評価される段階の小数点桁数の大きい方は5です(1.00000 * 1.1)。この小数点5桁の精度は、複数オペランド式の残りでも保持されます(赤色の桁数で表示)。

Analytics 計算 (評価順)	Analytics 結果 (端数処理)	端数処理されていない計算 (評価順)	端数処理されていない結果	結果の違い
1.00000 * 1.1	1.10000	1.00000 * 1.1	1.10000	0.00000
1.10000 * 1.12	1.23200	1.10000 * 1.12	1.23200	0.00000
1.23200 * 1.123	1.38354	1.23200 * 1.123	1.383536	0.000004
1.38354 * 1.1234	1.55427	1.383536 * 1.1234	1.5542643424	0.0000056576

演算の順序を指定する

丸かっこは、計算で求められる精度レベルを決める重要な役割を果たします。丸かっこを使用して数学演算の正規順序を変更する場合は注意してください。

日歩の計算

シナリオ

年利 12% で \$100,000 の日歩を計算する必要があります。

1つのアプローチ

最初に日歩を計算してから、日率を 100,000 で乗算できます。ただし、このアプローチでは端数処理の問題が発生します。

$$100000 * (0.12/365) = 0.00$$

Analytics は最初に 0.12 を 365 で除算します。それに基づくと、Analytics の端数処理は 0.00 になります。真の結果は 0.00032876712... ですが、結果は小数点 2 桁で端数処理されるため、すべての後続の桁数が失われます。

正しい答えが 32.876712... であっても、端数処理された結果は 100000 で乗算され、0.00 になります。

代替アプローチ

最初に年利の合計金額を計算してから、金額を 365 で除算できます。このアプローチにより、端数処理の問題を解決できます。

$$100000 * 0.12/365 = 32.88$$

括弧が削除されると、計算の各段階の結果は 1 より大きく、小数点の端数処理の問題を回避します。答えはペンスまで正確になります(2 桁)。

小数点精度の動作の変更

SET MATH コマンドを使用して、数式で小数点精度の動作を変更できます。デフォルトでは、Analytics は、2 つのオペランドを評価するときに、小数点桁数の大きい方または最大値を使用します。最大数を使用すると、式の各段階で最大精度が維持されます。

コマンドラインまたは Analytics スクリプトで SET MATH を使用すると、Analytics セッション中のデフォルト動作が変更されます。以下のオプションの概要では、式 `1.275 * 1.3` の異なる結果が示されます。実際の端数処理されていない結果は `1.6575` です。

コマンド	説明	<code>1.275 * 1.3</code> の結果
<code>SET MATH FIRST</code>	オペランドのペアの最初のオペランドの小数点桁数を使用します	1.658
<code>SET MATH LAST</code>	オペランドのペアの最後のオペランドの小数点桁数を使用します	1.7
<code>SET MATH MIN</code>	オペランドのペアで最も少ない小数点桁数を使用します	1.7
<code>SET MATH MAX</code>	オペランドのペアで最も多い小数点桁数を使用します	1.658
デフォルト		
SET MATH の詳細については、"SET コマンド" ページ 2081を参照してください。		

財務関数での丸めの制御

Analytics の他の数値演算と違い、財務関数は浮動小数点演算で評価されます。

浮動小数点演算には2つの計算方法があります。

- 総計を返す関数は、1250.00 のように小数点以下 2 桁まで計算します。
- 利率を返す関数は、0.01676584、つまり 1.676584% のように小数点以下 8 桁まで計算します。

財務関数では、利率を 12 や 365 などの定数で割って、月利や日歩を計算します。たとえば、0.08/365 は年率 8% に基づいた日歩要因になります。Analytics の標準の除算規則では、ほとんどの場合、大幅な丸め誤謬が生じます。たとえば、0.08/365 の結果は、実際の利率 0.000219178 ではなく、0.00 になります。このため、利率パラメーターを評価する場合には、標準の除算規則は一時的に無効になります。

財務関数の利率パラメーターが、0.08/365 のように2つの数量の除算の場合、まず、この2つの数量は別々に評価されます。その後、除算が実行されて、正確な結果を出します。ただし、各数量は Analytics の標準の丸め規則で計算されているので、複雑な計算の場合、この方法で丸め誤謬を完全になくすことはできません。

利率が特定の率または 08/1/365 のように単純な2つの数量の除算ではない場合、利率計算に丸めが生じたという警告メッセージがログに記録されます。スクリプト モードでない場合は、警告]ダイアログボックスに表示されます。この場合、返された結果が過剰に丸められていないかを確認する必要があります。

演算フィールドで利率が計算される場合、Analytics は利率が丸められたかどうかを判断できないので、結果が正しいかを確認する必要があります。

数式でのオーバーフローエラーの防止

結果の桁数が小数点以下の桁数も含めて22桁を超える計算では、**数値のオーバーフローで停止する**]オプションがオンになっている場合は、オーバーフローエラーを作成できます。オーバーフローが発生すると、Analytics は処理を停止し、ビューに「###ERR###」と表示します。

オーバーフローは、財務計算やパーセンテージ、現在価値などの、非常に大きな数値または小数点以下の桁数が大きい数値を扱う計算を行う場合に発生します。また、財務関数で負の値の支払期間などの不正なパラメーターを使用している場合、あるいは計算でゼロによる除算を行っている場合にもオーバーフローは発生します。

数値のオーバーフローで停止する]オプションはオフにすることができます。そうすると、Analytics はコマンドの処理を続行しますが、超過した桁を左端から切り捨てるため、結果は不正確なものとなります。

条件付きフィールドを作成することで、ゼロによる除算を回避することができます。たとえば、「給与総額/労働時間」という計算の場合、ゼロによる除算が発生するのは、対象となる特定の期間中に従業員の働いた時間が0時間である場合です。0以外の値が入っているフィールドのみが Analytics で評価されるようにするには、次のような値と条件を使って条件付き演算フィールドを作成します。

- デフォルト値:0
- 条件: 労働時間 <> 0
- 値: 給与総額/労働時間

数値のオーバーフローで停止する]オプションをオンのままにしておき、大きな数値または小数点以下の桁数が大きい数値に注意する方がより安全です。また、財務関数のパラメーターや、ゼロによる除算が発生する可能性のある計算にも注意してください。

期間に負の値が指定されているなど、明らかに無効なパラメーターが財務関数に指定されている場合、この無効なパラメーターは次のいずれかの方法で処理されます。

- **数値のオーバーフローで停止する**]がオンの場合、Analytics は処理を停止します。
- **数値のオーバーフローで停止する**]がオフの場合、関数は -1 の値を返します。

式の使用時の2つの一般的なエラー

Analytics で式を初めて使用するユーザーは、式の作成時に次のような2つの一般的なエラーに遭遇することが多いでしょう。

- 式でデータ型と操作が適合していないと、"式の型が一致しません" エラーメッセージが表示されません。
- 論理式のみが有効とされる状況または場所で、文字、数値または日付時刻の式が作成されると、"'式'タイプは不正です" または "論理式が必要です" エラーメッセージが表示されます。

"式の型が一致しません"

式を有効にするには、式の値のデータ型を、実行してる計算や操作に適合させる必要があります。たとえば、2つの文字フィールドで掛け算したり、2つの日付フィールドで割り算したりすることはできません。そのようなことを行くと、"式の型が一致しません" エラーメッセージが表示され、処理が実行されません。

以下に示す方法を使用すると、このエラーを回避することができます。ただし、特定の状況によっては、これらが適切な解決法とならないこともあります。

- **値は同じですが、操作を変更する**

たとえば、2つの時刻フィールドを加算することはできないので、`終了時刻 + 開始時刻` は不正です。しかし、一方の時刻からもう一方の時刻を減算することはできるので、`終了時刻 - 開始時刻` は有効です。

- **操作は同じですが、値の1つ以上を変更する**

たとえば、文字値から数値を減算することはできないので、`DATE() - 2` は不正です。これは、パラメーターを使用しない `DATE()` 関数は現在のオペレーティングシステムのシステム日付を文字値として返すからです。ただし、日付値から数値を減算することはできるので、`TODAY() - 2` は有効です。これは、`TODAY()` 関数が現在のオペレーティングシステムのシステム日付を日付値として返すからです。

- **操作は同じですが、値の1つ以上のデータ型を変更する**

たとえば日付と数値を比較することはできないので、`Finish_Date > 20141231` は不正です。しかし、日付と日付は比較できるので、`Finish_Date > `20141231`` は有効です。逆引用符(`)を追加することで、数値リテラルが日付リテラルに変更されます。

場合によっては、関数を使用して式内の値またはフィールドを、式に適したデータ型に変換することもできます。変換関数の詳細については、"フィールドの一致" ページ 921 を参照してください。

"'式'タイプは不正です" または "論理式が必要です"

式の戻り値は、その状況に対して適正なデータ型である必要があります。式が必要とされるアプリケーションの多くの領域で、その戻り値は特定のデータ型でなければなりません。必要とされるデータ型でない場合は "'式'タイプは不正です" または "論理式が必要です" エラーメッセージが表示され、処理が実行されません。式が有効であったとしても、式がその状況に適さない不正なデータ型を返す場合は、Analytics でエラーメッセージが表示されます。

たとえば、`Quantity_on_Hand * Unit_Cost` という式の場合、それを演算フィールドの作成に使用するのであれば、一般的には有効なので、期待どおりに動作し、数値合計が得られます。しかし、同じ式でも、それをビューフィルターの作成時、または IF ステートメントで使用する場合は、True または False (T または F) の論理値を返す必要があるためエラーが返されます。その式を `Quantity_on_Hand * Unit_Cost > 5000` に変更すると、フィルターや IF ステートメントで正しく動作します。

どの場所から**式ビルダー**を開くかが、重要なポイントです。[フィルター] テキスト ボックスの右側にある

ビューフィルターの編集  をクリックする、またはさまざまな場所で存在する **[f]** ボタンをクリックした場合、**式ビルダー**では論理式を作成する必要があります。

式での日付時刻の使用

日付、日付時刻および時刻を計算するために式を使用することができます。

- 経過日数、経過日数と時間、または経過時間の計算

例: ``20141231` - `20141130`` は2つの日付間の日数 `31` を返します。

- 日付、日付時刻、または時刻の正または負の調整

例: ``20141231` - 15` は15日前の日付 `2014年12月16日` を返します。

- 日付、日付時刻または時刻の比較

例: ``20141231 183000` > `20141231 171500`` は最初の日付が2番目の日付より新しいので `T` (True) を返します。

日付および時刻関数

日付時刻データの操作を支援するために、Analytics はさまざまな有用なタスクを実行する日付と時刻の関数を多数提供しています。日付時刻式を構成するときに、これらの関数を利用できます。

次の表に、日付時刻関数をタスク別にグループ分けして示します。**式ビルダー**の **関数** ドロップダウンリストを "日付時刻" でフィルターすると、同様の関数の一覧を見ることができます。

実行されるタスク	関数
日付と締切日の間の経過日数(年齢)、現在の日付、または任意の2つの日付間の経過日数を返します。	AGE()
指定された月数だけ前または後の日付、月末を計算する	GOMONTH() , EOMONTH()
日付の曜日、または月を特定します	CDOW() , CMOY()
日付の曜日を表す数値(1 ~ 7)を返します	DOW()
日付時刻値から日付または時刻を抽出します	DATE() , TIME()
日付時刻値から年、月、日、時、分、または秒を抽出します	DAY() , MONTH() , YEAR() , HOUR() , MINUTE() , SECOND()
シリアル日付時刻値、文字日付時刻値、または数値日付時刻値を Datetime データ型の標準の日付時刻値に変換します	STOD() , STODT() , STOT() , CTOD() , CTODT() , CTOT()
現在のオペレーティングシステム日付、日付時刻、または時刻を返します	TODAY() , DATETIME() , NOW()

時間の量と時点

式の中で日付時刻を操作するとき、時間の量と時点を区別することは重要です。この違いによって、異なる種類の式を構築することが要求されます。

時刻

時刻値 **08:30:00** が指しているのは、時間の量(8時間30分)かもしれないし、時点(午前8時30分)かもしれません。

メモ

時刻の表示書式(ツール>オプション>日付と時刻)は、最初の例では **hh:mm:ss** で、2番目の例では **hh:mm:ss PM** です。どちらの例も、後続のセクションで説明されているシリアル日付時刻の計算を必要とします。

時間の量

ある時刻から別の時刻を減算した場合、結果は、時間の量である経過時間となります。

`01:15:00` を返します

```
STOT(`T083000` - `T071500`)
```

時間の時点

時刻に数値を加算する、または時刻から数値を減算した場合、結果は、最初の時刻より前か後の別の時点を作成する、正または負の調整になります。

`07:00:00 AM` を返します

```
`T083000` - (1.00000000/24*1.5)
```

日付

時間の量

ある日付から別の日付を減算した場合、結果は、時間の量である経過日数となります。

2つの日付間の日数である `31` を返します

```
`20141231` - `20141130`
```

時間の時点

日付に数値を加算する、または日付から数値を減算した場合、結果は、最初の日付より前か後の別の時点になります。

31 日前の日付 `30 Nov 2014` を返します

```
`20141231` - 31
```

有効および無効な日付時刻式

日付時刻式は、日付時刻サブタイプ(日付、日付時刻、および時刻)と演算子の多くの可能な組み合わせを網羅しています。すべての組み合わせが正当な式であるわけではありません。たとえば、経過日数を調べるために、ある日付から別の日付を引くことはできますが、2つの日付を足すことはできません。この演算は非論理的だからです。しかし、ある日付に数値を加算して、その後の日付を生成することはできます。

次の規則が日付時刻式に適用されます。

- **日付時刻を減算または比較** -日付、日付時刻、または時刻の値の任意の組み合わせを減算または比較演算で使用することができます。
- **数値および日付時刻を加算または減算** -整数、帯分数、および分数を日付、日付時刻、または時刻の値に対し加算または減算することができます。
- **日付時刻を加算** -日付、日付時刻、または時刻の値を互いに加算することはできません。一週間の労働時間などのように時間の量を加算する必要がある場合は、時刻の時、分、および秒の部分を数値として抽出するために、Analytics 関数を使用することができます。それらの数値

で計算を実行することができます。詳細については、「関数を使った時刻の加算」ページ 889を参照してください。

- **日付時刻および数値を比較** -日付、日付時刻、または時刻の値を数値と比較することはできません。

次の表は日付時刻式で可能な組み合わせを要約し、各組み合わせが有効か無効であることを示します。つまり Analytics によって処理できるかどうかを示します。

メモ

式が有効であっても、必ずしもその式が分析目的に有効にかなうとは限りません。たとえば、Analytics は「終了_日付 > 開始_時刻」という式を処理しますが、結果は常に True (T)となるため、日付と時刻の比較は論理的な目的を果たしません。

	日付値	日付時刻値	時刻値	数値
日付値	有効: 減算 比較	有効: 減算 比較	有効: 減算 比較	有効: 減算 加算
	無効: 加算	無効: 加算	無効: 加算	無効: 比較
日付時刻値	有効: 減算 比較	有効: 減算 比較	有効: 減算 比較	有効: 減算 加算
	無効: 加算	無効: 加算	無効: 加算	無効: 比較
時刻値	有効: 減算 比較	有効: 減算 比較	有効: 減算 比較	有効: 減算 加算
	無効: 加算	無効: 加算	無効: 加算	無効: 比較

日付時刻式によって返されるデータ型

日付時刻式によって返される結果のデータ型は、式の中の値や演算子に依存します。

日付時刻式	結果のデータ型
減算(日付時刻値のみ) 減算では、日付、日付時刻、または時刻の値の任意の組	数値 シリアル日付、シリアル日付時刻、またはシリアル時刻

日付時刻式	結果のデータ型
み合わせを使用できます	詳細については、「シリアル日付時刻」ページ 896を参照してください。
加算または減算(日付時刻値と数値) 整数、帯分数、または分数を日付、日付時刻、または時刻の値に対し加算または減算できます	日付時刻 日付時刻データ型の日付、日付時刻、または時刻サブタイプ
比較(日付時刻値のみ) 比較演算では、日付、日付時刻、または時刻の値の任意の組み合わせを使用できます	論理 T(True) または F(False)

日付時刻リテラルの書式

- 日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。

日付は時間の前に置く必要があります、日付と時刻の間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、文字 't'、または文字 'T' です。

- 時刻値** - 24 時間形式を使用して時刻を指定する必要があります

UTC(Coordinated Universal Time: 協定世界時)からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
thhmmss	`t235959`
Thhmm	`T2359`

形式の例	リテラル値の例
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

関数を使った時刻の加算

Analytics では、時刻値どうしを直接加算することはできません。ただし、Analytics 関数を使用して、時刻の時、分、秒の部分を数値として抽出し、それらの数値で計算を実行することができます。

次の勤務時間記録表 (Timesheet) データ例では、時間と分を使用してこのアプローチを説明します。

	Date	Start_Time	End_Time	Elapsed	Hours	Minutes	Part Hours	Hours+Part Hours
1	2013/04/08	08:17:00	18:05:00	09:48:00	9	48	0.800	9.800
2	2013/04/09	09:48:00	17:48:00	08:00:00	8	0	0.000	8.000
3	2013/04/10	09:28:00	17:52:00	08:24:00	8	24	0.400	8.400
4	2013/04/11	08:28:00	18:30:00	10:02:00	10	2	0.033	10.033
5	2013/04/12	10:12:00	18:42:00	08:30:00	8	30	0.500	8.500

<< ファイルの終わり >>

計算を生成するために、いくつかの演算フィールドが必要となります。

演算フィールドの名前	式	説明
Elapsed	STOT(End_Time - Start_Time)	一日の労働時間を計算する減算操作。STOT() 関数は、計算結果をシリアル時刻値から標準の時刻値に変換します。
Hours	HOUR(Elapsed)	時の部分を Elapsed 値から数値として抽出します。
Minutes	MINUTE(Elapsed)	分の部分を Elapsed 値から数値として抽出します。 (表示目的です。計算には必要ありません。)
Part Hours	MINUTE(Elapsed)/60.000	分の部分を Elapsed 値から数値として抽出し、60 分の小数部として計算します。
Hours+Part Hours	Hours + Part_Hours	Hours と Part Hours の数値を加算します。

最後のステップとして、一週間の合計時間を計算するために、Hours+Part Hours フィールドを合計します。



日付、日付時刻、または時刻の正または負の調整

日付、日付時刻、または時刻の値に正または負の調整を行うことができます。たとえば、15 日を加減算したり、3 時間を加減算したりすることができます。

日付に正負の調整を行う

日付に正または負の調整を行うことは簡単です。別の日付を基にある日付を算出するために、必要な日数を加算または減算します。

15 Jan 2015 を返します。

```
`20141231` + 15
```

16 Dec 2014 を返します。

```
`20141231` - 15
```

日付時刻または時刻に正負の調整を行う

日付に調整を行うよりも、日付時刻または時刻の値に正または負の調整を行うことはやや複雑です。

時刻値どうしや、時刻値を日付時刻値に直接加算することはできません。たとえば、次の式のいずれかを使用して3時間の正の調整をしようとする場合、エラーが発生します。

`式の型が一致しません`エラーを返します。

```
`t120000` + `t030000`
```

`式の型が一致しません`エラーを返します。

```
`20141231 235959` + `t030000`
```

シリアル時刻相当値を使用して正の調整を行う

3時間に相当するシリアル時刻(0.125)を加算することにより、有効な式を作成することができます。しかし、このような式を手作業で作成するには、相当するシリアル時刻を知っている必要がありますので不便です。

`15:00:00`を返します。

```
`t120000` + 0.125
```

`01 Jan 2015 02:59:59`を返します。

```
`20141231 235959` + 0.125
```

シリアル日付時刻または時刻結果で負の調整を行う

日付時刻または時刻の値に負の調整を行うことの方が簡単ですが、結果はシリアル日付時刻値またはシリアル時刻値になります。この値は、人が読めるように標準の日付時刻値または時刻値に変換する必要があります。

`0.3750000000000000`を返します。

```
`t120000` - `t030000`
```

`09:00:00`を返します。

```
STOT(`t120000` - `t030000`)
```

演算フィールドを作成して、より簡単に正負の調整を行う

次の方法を使用すると、より簡単に日付時刻または時刻に正または負の調整を行うことができます。

1. 加算または減算したい時間に相当するシリアル時刻を計算する演算フィールドを作成します。

0.10416668 を返します。これは 2-1/2 時間に相当するシリアル時刻です。

```
(1.00000000/24*2.5)
```

Analyticsが結果を丸めないように、小数点以下に複数のゼロを持つ '1' を指定する必要があります。

(1.00000000/24*1)、(1.00000000/24*8)、(1.00000000/24*10.25) など、掛ける数字を変えることで、時間の適正な数値を取得することができます。

2. 同じ演算フィールドで、計算したシリアル時刻をソースの時刻または日付時刻の値に加算する、またはこれらの値から減算します。

フィールドの値 + 2-1/2 時間を返します。

```
<時刻または日付時刻フィールド> + (1.00000000/24*2.5)
```

3. 日付時刻値に対して日と時間の両方の加減算をしたい場合は、計算に適切な日数を含めてください。

フィールドの値 + 2 日 2-1/2 時間を返します。

```
<日付時刻フィールド> + 2 + (1.00000000/24*2.5)
```

日付時刻式の例

以下の表では有効な日付時刻式の例を示します。

- "経過日数、日数と時間、または時間の計算" 次のページ
- "日付、日付時刻、または時刻の正または負の調整" ページ 894
- "日付、日付時刻または時刻の比較" ページ 895
- "変換関数を使用する日付時刻式" ページ 895

メモ

多くの例では、結果はシリアル日付時刻として返されます。つまり、整数または24時間の小数部分で表される日付、日付時刻、または時刻の値です。

シリアル日付時刻の戻り値を標準の日付時刻の値に変換するために、STOD()、STODT()、およびSTOT()関数を使用できます。詳細については、「シリアル日付時刻」ページ 896を参照してください。

経過日数、日数と時間、または時間の計算

式	結果
`20141231` - `20141130`	31 2つの日付間の経過日数
終了_日付 - 開始_日付	終了_日付と開始_日付の間の経過日数
`20141231 235959` - `20141130 114530`	31.51005787037037 2つの日付時刻間の経過日数と時間をシリアル時刻として表現
STRING(INT(`20141231 235959` - `20141130 114530`),5) + " " + TIME(STOT(MOD(`20141231 235959` - `20141130 114530`, 1)))	31 12:14:29 上記の例の2つの日付時刻間の経過日数と時間を日、時、分、秒として表現 現在のAnalyticsの時刻表示書式はhh:mm:ssとします
(`20141231 235959` - `20141130 114530`) * 24	756.24138888888888 上記の例の2つの日付時刻の間の経過時間は、時間と時間の小数部分として表現されます。
終了_日付時刻 - 開始_日付時刻	終了_日付時刻と開始_日付時刻の間の経過日数と時間をシリアル時刻として表現
STRING(INT(終了_日付時刻 - 開始_日付時刻), 5) + " " + TIME(STOT(MOD(終了_日付時刻 - 開始_日付時刻, 1)))	終了_日付時刻と開始_日付時刻の間の経過日数と時間を日、時、分、秒として表現
`T235959` - `T114530`	0.51005787037037 2つの時刻間の経過時間をシリアル時刻として表現
STOT(0.51005787037037)	12:14:29 上記の例のシリアル時刻が、現在のAnalyticsの時刻表示書式を使用して時刻値に変換されます。
STOT(`T235959` - `T114530`)	12:14:29 2つの時刻間の経過時間を現在のAnalyticsの時刻の表示書式を使用して時刻として表現

式	結果
終了_時刻 - 開始_時刻	終了_時刻と開始_時刻の間の経過時間をシリアル時刻として表現

日付、日付時刻、または時刻の正または負の調整

式	結果
支払_期日 + 15	支払_期日フィールドの値を 15 日増加
`20141231` - 15	16 Dec 2014 日付を 15 日減少。現在の Analytics の日付表示書式は DD MMM YYYY とします
`20141231 235959` + (1.00000000/24*1.5)	01 Jan 2015 01:29:59 日付時刻 + 1.5 時間
`20141231 235959` - (1.00000000/24*1.5)	31 Dec 2014 22:29:59 日付時刻 - 1.5 時間
STODT(`20141231 235959` - `T013000`)	31 Dec 2014 22:29:59 日付時刻 - 1.5 時間
`20141231 235959` + 2 + (1.00000000/24*1.5)	03 Jan 2015 01:29:59 日付時刻 + 2 日と 1.5 時間
`20141231 235959` - 2 - (1.00000000/24*1.5)	29 Dec 2014 22:29:59 日付時刻 - 2 日と 1.5 時間
`t235959` + (1.00000000/24*1.5)	01:29:59 時刻 + 1.5 時間
`T173000` - (1.00000000/24*1.5)	16:00:00 時刻 - 1.5 時間
STOT(`T173000` - `T013000`)	16:00:00 時刻 - 1.5 時間
STOT(STOT(`T173000` - `T013000`) - `T010000`)	15:00:00 時刻 - 1.5 時間 - さらに 1 時間

日付、日付時刻または時刻の比較

式	結果
`20141231` > `20141230`	T (True)
支払_期日 <= `20141231`	2014年12月31日以前の支払_期日フィールドのすべての値
支払_日付 > 支払_期日	支払_期日を過ぎた支払_日付フィールドのすべての値
CTOD(DATE(支払_日時, "YYYYMMDD"), "YYYYMMDD") > 支払_期日	支払_期日を過ぎた支払_日時フィールドのすべての値 日付時刻と日付の値を比較するには、まず、支払_日時フィールドの日付時刻値から日付を文字値として抽出し、その後、それを支払_期日と比較するために日付値に変換を戻します。 日付の書式が一致していることを保証するために、同一の書式が、DATE() 書式パラメータ(出力書式)とCTOD() 書式パラメータ(入力書式)に指定されています。
ログイン_時刻 > `t100000`	10:00:00より遅いログイン_時刻フィールドのすべての値

変換関数を使用する日付時刻式

式	結果
STOT(CTOT("t120000") - CTOT("t090000"))	03:00:00 2つの文字時間値の間の経過時間 まず、減算操作で使用できるように、文字時刻値が標準の時刻値に変換されます。その後、減算操作の計算結果である数値のシリアル時刻が標準の時刻値に変換されます。
CTOT(TIME(`20141231 125959`)) < `T235959`	T (True) まず、日付時刻値から時刻を文字値として抽出し、その後、それを23:59:59と比較するために時刻値に変換を戻します。

シリアル日付時刻

Analytics では、日付、日付時刻または時刻の保存と日時計算の実行にシリアル日付時刻を使用しています。

日付時刻計算の操作では、シリアル日付時刻値が発生する場合があります。たとえば、時刻の値を対象とする減算操作では、結果がシリアル時刻の形式で返されます。

シリアル日付時刻とは

シリアル日付時刻とは、日付を表す整数と24時間の時刻を表す小数を使用した数値です。小数点の前(左側)の部分は日付、小数点の後(右側)の部分は時刻です。

シリアル日付時刻	相当する標準の日付時刻
42004	2015/01/01
42004.5000000	01 January 2015 12:00:00
0.7500000	18:00:00
42004.74618055555556	01 Jan 2015 17:54:30

日付部分

日付部分は、1900年1月1日からの経過日数を表します。シリアル日付の1は1900/01/02に相当します。シリアル日付0(ゼロ)はカウントされません。シリアル日付42004は2015/01/01に相当します。

時刻部分

シリアル日付時刻の時刻部分は24時間制を使用します。シリアル時刻値は次のように計算されます。

$$1 / 86,400 \text{ (1日の秒数)} * \text{(秒で表される特定の時刻値)}$$

ヒント

シリアル時刻値の考えるもう一つの方法は、24時間の日の割合を表すということです。

標準の時刻	シリアル時刻
01:00:00	0.04166666666667 (1時間、1日24時間のうちの1/24時間)

標準の時刻	シリアル時刻
08:00:00	0.3333333 (1日24時間のうちの1/3)
12:00:00	0.5000000 (1日24時間のうちの半分)
17:54:30	0.74618055555556 (17時間54分30秒)
18:00:00	0.7500000 (1日24時間のうちの3/4)

シリアル日付時刻の標準日付時刻値への変換

3つの変換関数を使用すると、シリアル日付時刻値を日付時刻データ型の標準日付時刻値に変換することができます。

- [STOD\(\)](#) - “Serial to Date”(シリアル日付を日付値へ)
- [STODT\(\)](#) - “Serial to Datetime”(シリアル値を日付時刻値へ)
- [STOT\(\)](#) - “Serial to Time”(シリアル時刻を時刻値へ)

シリアル日付時刻を変換して、日付時刻式の結果を人間が読み取りやすいようにする、あるいは日付時刻データ型を必要とする別の式で使用するために、数値のシリアル日付時刻値を日付時刻データ型に変換させることができます。

次の表では、3つの関数の使用例を示します。

式	結果
STOD(42004)	2015/01/01
STODT(42004.5000000)	2015/01/01 12:00:00
STOT(0.7500000)	18:00:00
STODT(42004.74618055555556)	01 Jan 2015 17:54:30
STOT('T173000' - 'T093000')	08:00:00

標準日付時刻値のシリアル日付時刻への変換

通常、標準日付時刻値のシリアル日付時刻への変換の必要はありません。シリアル日付時刻は、日付時刻保存と計算のために、Analytics によって内部的に使用されます。

標準日付時刻に対応するシリアル日付時刻値を表示する場合は、次の方法を使用します。

標準日付時刻値	変換式	対応するシリアル日付時刻
2015/01/01	`20150101`-`19000101`	42004
17:54:30	$1.0000000000 * ((\text{HOUR}(\text{'t175430'}) * 3600) + (\text{MINUTE}(\text{'t175430'}) * 60) + \text{SECOND}(\text{'t175430'})) / 86400$	0.7461805556

UTC オフセットが時刻日付式に与える影響

UTC とは何か

UTCとは協定世界時のことで、経度ゼロ度の位置の時刻です。これは、時刻および時刻ゾーンを調整するために、世界時刻基準として使用されます。UTCはグリニッジ標準時(GMT)と密接に関連しています。多くの目的において、この2つの時刻は同一であると考えられます。

UTC オフセットとは何か

UTC オフセットとは、特定の時刻ゾーンとUTC(経度ゼロ度の時刻)の間の時間と分の差です。たとえば、ニューヨークはUTC-05:00です。これは、ニューヨークの時刻が、UTC±00:00のロンドンの時刻より5時間遅れていることを意味します。

Analyticsでは、UTC オフセットの時刻データは、`hh:mm:ss±hh:mm`の形式を使用します。例:`23:59:59-05:00`

AnalyticsでUTCオフセットを処理する方法

Analyticsで、UTC オフセット付きのローカル時刻データを含む日付時刻式を処理する場合、そのUTC オフセットは調整され、式はローカル時刻に相当するUTCに対して計算を実行します。日付時刻関数もUTC オフセットを調整します。たとえば、式または関数で`23:59:59-05:00`というローカル時刻が存在した場合、実際には、その時刻に相当するUTC `04:59:59`に対して計算が実行されます。

日付時刻式や関数がこのように動作するのは、内部的にAnalyticsがUTC オフセット付きのローカル時刻を、それに相当するUTCとして保存するからです。

影響を受ける可能性がある日付

深夜0時を境に調整が前後する場合、UTC オフセットの調整が日付時刻データの日付に影響する可能性があります。たとえば、`31 Dec 2014 23:59:59-05:00`に相当するUTCは`01 Jan 2015 04:59:59`です。

AnalyticsでのUTC表示オプション

デフォルトで、Analyticsは調整されたUTC オフセット付きのローカル時刻を、相当するUTCとして表示するので、計算で使用されるときには実際の時刻が見られます。UTC オフセットが表示されたローカル時

刻を表示するようにするオプションもあります。UTC 表示の詳細については、「日付と時刻オプション」ページ 135を参照してください。

UTC に基づくデータの作業

UTC に基づくデータで作業を行う場合、その調整処理を考慮しないと、結果が正しくないものになると思われるかもしれません。

以下のテーブルの例では、UTC オフセットが日付時刻式に与える影響について示しています。補足のため、相当する UTC を使用した式のバージョンも示します。相当する UTC を使用する式の方は、式の処理時に Analytics で見えるものではありません。

日付時刻式	相当する UTC を使った式	結果
<code>`T235959` > `T230000`</code> <code>`T235959-0500` > `T230000`</code>	該当なし <code>`T045959` > `T230000`</code>	T (True) F (False)
<code>`20131231 235959` + 1</code> <code>`20131231 235959-0500` + 1</code>	該当なし <code>`20140101 045959` + 1</code>	01 Jan 2014 23:59:59 02 Jan 2014 04:59:59
<code>CROW(`20141231T235959`, 9)</code> <code>CROW(`20141231T235959-0500`, 9)</code>	該当なし <code>CROW(`20150101T045959`, 9)</code>	水曜日 木曜日
<code>MONTH(`20141231T235959`)</code> <code>MONTH(`20141231T235959-0500`)</code>	該当なし <code>MONTH(`20150101T045959`)</code>	12 1
<code>STOT(`T235959` - `T225959`)</code> <code>STOT(`T235959-0500` - `T225959-0400`)</code>	該当なし <code>STOT(`T045959` - `T025959`)</code>	01:00:00 02:00:00

監査データの検証

分析作業を実行する前に、必ず、作業しようとしているデータの完全性と有効性を検証してください。テーブルに無効なデータが少しあるだけでも、すべての後続のデータ分析が無効になる可能性があります。

操作	説明
検証	文字、数値、日付時刻フィールドの値の有効性をテストします
カウント	テーブルのレコード集計と同時に、次の作業も実行できます。 結果をコントロールカウントと比較できます
合計	1つ以上の数値フィールドを合計します 結果をコントロール合計と比較できます

データの検証

データの検証では、アクティブなテーブル内のデータの妥当性エラーを検証します。検証はテーブル内のデータがテーブルレイアウトに合致しており、指定されたデータ型に一致していることを確認します。

検証で実行されるチェック

検証では、次のチェックを実行します。

- **文字フィールド** -には、文字、数字、および記号などの、有効な印刷可能な文字のみが入っている
- **数値フィールド** -には、数字、小数点、および通貨記号などの、有効な数値文字のみが入っている
- **日付時刻フィールド** -が、有効な日付、日付時刻、または時刻を含んでいる。

エラーが検出された場合は、該当するレコード番号、フィールド名、および無効なデータの16進数コードが出力に含まれ、ログに記録されます。

手順

データが、指定したデータ型を含め、テーブルレイアウトと一致しているかを検証し、妥当性エラーを出力することができます。

方法の説明

1. **データ** > **検証** を選択します。
2. **メイン** タブで、次のいずれかを行います。
 - **検証フィールド** リストから検証するフィールドを選択します。
 - **検証フィールド** をクリックして、フィールドを選択するか、または式を作成します。
フィールドを選択する順序が、結果に表示する列の順序になります。
3. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

4. **出力** タブをクリックします。

5. **出力先**]パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果をテキストファイルに保存または追加するには、このオプションを選択します。ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

6. 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは、使用している Analytics のエディションに応じて、"ASCII テキストファイル" または "Unicode テキストファイル" のいずれかのみです。結果を新しいテキストファイルに保存するか、または既存のテキストファイルに追加します。
 - **名前** - **名前**]テキストボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

- **ローカル** - 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。
7. **詳細**]タブをクリックします。
 8. **適用範囲**]パネルで、適切な出力オプションを選択します。

- **すべて**
- **先頭**
- **次へ**
- **While**

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
------------	-------------------------------------------------------------------

先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>	

9. **[エラー制限]** テキスト ボックスで、列挙する無効なレコードの最大数を指定するか、またはデフォルトの 10 のままにしておきます。

制限に達すると、処理が停止され、その時点で検出されている無効なレコードが出力されます。

メモ

エラー制限のデフォルト値は、**[ツールオプション]**の順にクリックして **[コマンド]** タブを選択し、**[エラー制限]**の値を更新することによって変更できます。

10. 出力の種類として **[ファイル]**を選択した場合、出力結果を既存のテキスト ファイルの末尾に追加するときは、**[既存のファイルに追加する]**を選択します。
11. **[OK]**をクリックします。
12. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

自動検証を有効にする

Analytics で、テーブルを開くたびにデータが自動的に検証されるよう設定することができます。このオプションが有効な場合は、すべての Analytics テーブルに適用されます。

方法の説明

1. **[ツール > オプション]**の順にクリックします。
2. **[数値]**タブをクリックします。
3. **[データを検証する]**をオンにします。

4. 無効なデータが含まれているフィールドを空白で表示したい場合は、**無効なデータを空白にする**]をオンにします。このオプションをオンにしていない場合、無効なデータが含まれているフィールドには「###ERR###」が表示されます。
5. **OK**]をクリックします。

レコードのカウント

テーブル内の総レコード数や、指定した条件に一致するレコードのみの数をカウントすることができます。結果は Analytics の表示領域に表示されます。

条件が指定されていない場合は、テーブル内の総レコード数がステータスバーに表示されます。ビューにグローバルフィルターが適用されている場合は、フィルターが適用された後にテーブルに残っているレコードの数がステータスバーに表示されます。

手順

1. **分析 > カウント**]の順に選択します。
2. **メイン**]タブで、任意で次のいずれかを行うことができます。
 - **[f]**テキスト ボックスに条件を入力します。
 - **[f]**をクリックし、**式ビルダー**を使用して IF ステートメントを作成します。
IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。
3. **詳細**]タブをクリックします。
4. **適用範囲**]パネルで、適切な出力オプションを選択します。
 - **すべて**
 - **先頭**
 - **次へ**
 - **While**

詳しく表示

<p>すべて</p>	<p>デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。</p>
<p>先頭</p>	<p>このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。</p>
<p>次へ</p>	<p>このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p>
<p>While</p>	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

5. **[OK]** をクリックします。

フィールドの合計

アクティブなテーブル内の数値フィールドや数式の合計を計算できます。合計は、1 つまたは複数の数値フィールドの算術的合計を求めます。通常、これはデータの完全性や正確性を証明するため、またコントロール合計を生成するために使用されます。結果は Analytics の表示領域に表示されます。

ヒント

請求書番号などの文字フィールドには、数値が含まれていることがあります。この型のデータを合計するには、VALUE() 関数を使用する演算フィールドを作成し、文字データを数値データに変換した後、演算フィールドの合計を求めます。

手順

1. **分析 合計**] の順に選択します。
2. **メイン**] タブで、次のいずれかを行います。
 - **合計フィールド**] リストから合計するフィールドを選択します。
 - **合計フィールド**] をクリックして、フィールドを選択するか、または式を作成します。
フィールドを選択する順序が、結果に表示する列の順序になります。
3. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

4. **詳細**] タブをクリックします。
5. **適用範囲**] パネルで、適切な出力オプションを選択します。
 - **すべて**
 - **先頭**
 - **次へ**
 - **While**

詳しく表示

<p>すべて</p>	<p>デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。</p>
<p>先頭</p>	<p>このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。</p>
<p>次へ</p>	<p>このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p>
<p>While</p>	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

6. **[OK]** をクリックします。

データの結合

Analytics では、一度に1つのテーブルのデータしか分析できません。このため、分析を実行する前に、2つ以上のテーブルからデータを結合して、1つのテーブルに入れる必要があります。

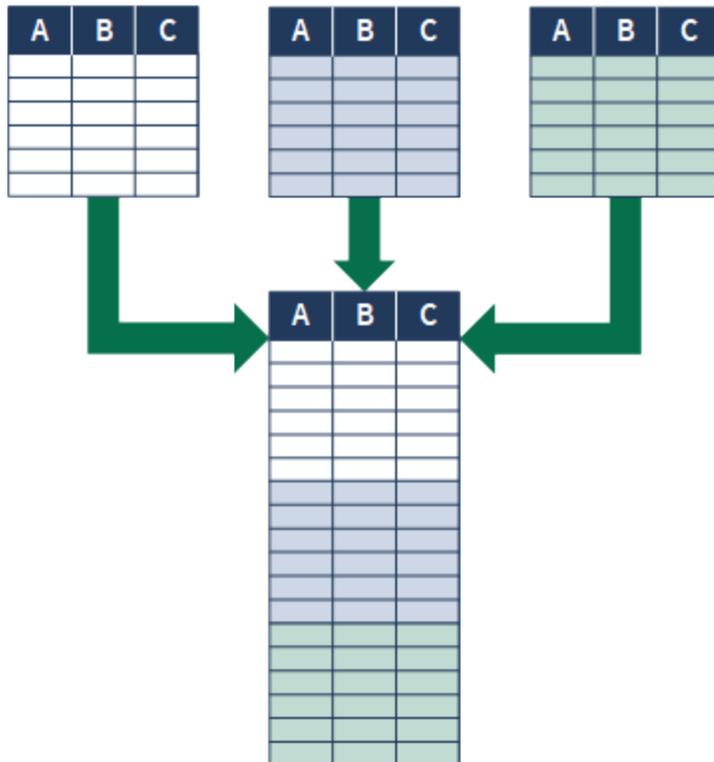
Analytics では、データを結合するために次の方法が提供されています。

レコードの結合	フィールドの結合
<ul style="list-style-type: none"> 最後に追加 抽出/最後に追加 マージ 	<ul style="list-style-type: none"> 結合 関連付け

ソースデータまたは分析目標の本質は、使用するデータ結合方法を規定します。以下に、5つの方法の概要を示します。

最後に追加

2つ以上のテーブルを追加すると、それらのテーブル内のレコードが結合されて新しいテーブルに出力されます。追加されるテーブル内の全フィールドを追加するか、または共通フィールドのみを追加するかを選択できます。



例

シナリオ

1年全体に相当するデータに対して分析を実行したいが、データが12の月間取引テーブルに分散している場合を考えます。

アプローチ

12の月間テーブルのデータを、それらのすべてを含む単一の年間テーブルに追加し、その後分析を実行します。

詳細な情報

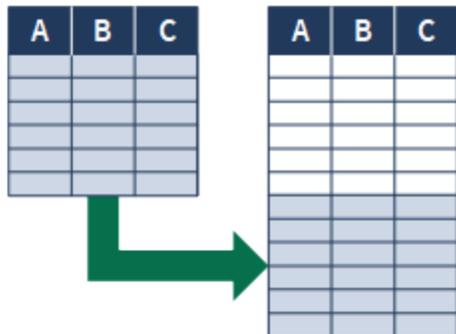
詳細については、「テーブルの追加」ページ928を参照してください。

抽出/最後に追加

データを抽出または追加するときに、1つのテーブルからレコードを抽出し、別のテーブルの末尾にそれらを追加することができます。抽出はコピーと同じです。末尾に追加は追加と同じです。

レコード全体でなく、レコード内のフィールドのサブセットを抽出することもできます。

(対象テーブル)に追加するテーブルのサイズは大きくなります。新しいテーブルは作成されません。



例

シナリオ

従業員レコードのセット全体に対して分析を実行したいが、新しい従業員のレコードがまだ従業員マスターテーブルに含まれていない場合を考えます。

アプローチ

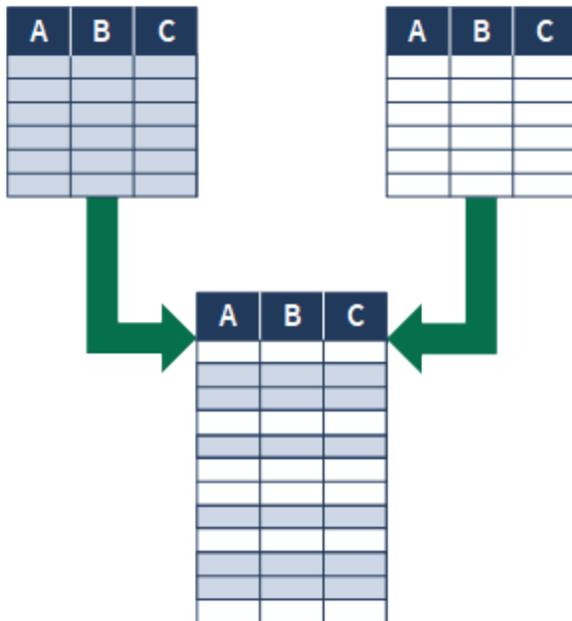
新しい従業員のレコードを抽出し、それらを従業員マスターテーブルの最後に追加して、分析を実行します。

詳細な情報

詳細については、「データの抽出と追加」 ページ 940を参照してください。

マージ

テーブルをマージするときには、2つの並べ替えられたテーブルのレコードを、並べ替えられた新しい3番目のテーブルに挿入します。挿入は、既存の並べ順に従って、レコードを結合することを意味します。



例

シナリオ

従業員レコードのセット全体に対して分析を実行したいが、レコードが2つの課員テーブルに分散している場合を考えます。

どちらのテーブルもラストネームで並べ替えられているため、レコードを結合後に再度並べ替えるオーバーヘッドが生じないようにしたいと思います。

アプローチ

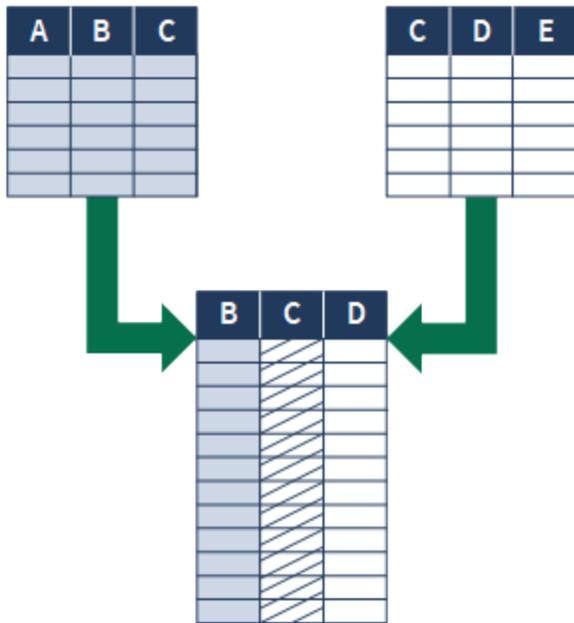
2つのテーブルにあるレコードを、新しい第3のテーブルに結合します。マージを行うことで、ラストネームでの並べ替えが保持されます。

詳細な情報

詳細については、「テーブルのマージ」ページ 951を参照してください。

結合

テーブルを結合するときには、共通キーフィールドを使用して、2つのテーブルのレコードまたは選択したレコードを新しい3番目のテーブルに取り込みます。共通キーフィールドは、結合される両方のテーブルに現れる従業員 ID などの識別フィールドです。



例

シナリオ

考えられる不適切な報酬のデータを分析する方法の1つとして、従業員でもある業者を特定する必要があります。

アプローチ

業者マスターテーブルを従業員テーブルと結合し、共通キーフィールドの住所を使用します。

結合された出力テーブルには、同じ住所の業者と従業員が含まれます。

詳細な情報

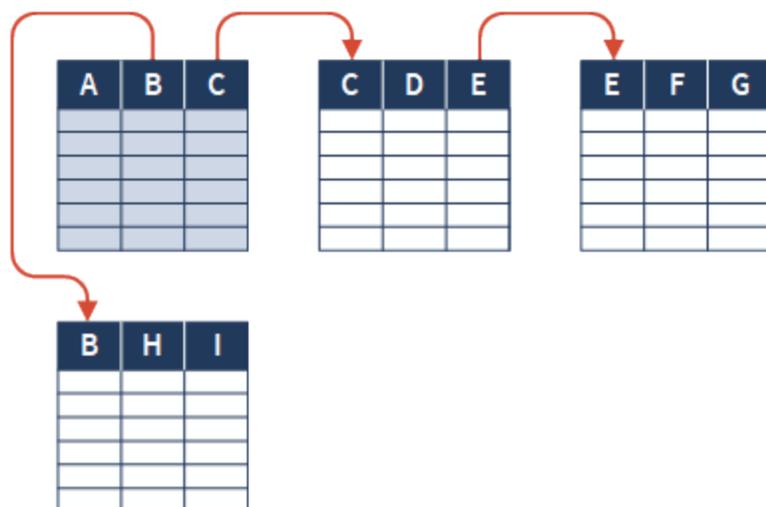
詳細については、「テーブルの結合」ページ 960を参照してください。

関連付け

テーブルを関連付けるときには、最大 18 テーブルを結合します。共通キーフィールドを使用し、各テーブルペアを関連付けます。

関連付け(実際には結合)は、テーブル間に一時的なプログラム関連付けを作成し、1つの物理テーブルに存在するかのようにテーブルのデータにアクセスするようにはできることです。ただし、物理テーブルは作成されません。いつでもソーステーブルの関連付けを解除できます。

共通キーフィールドは、関連付けられる各テーブルペアに現れる従業員 ID などの識別フィールドです。一般的には、各テーブルペアで異なる共通キーフィールドを使用します。



例

シナリオ

3月の顧客と販売された製品に関する詳細を含む売上レポートを作成する必要がある場合に、データが3つのテーブルに分散しているとします。

アプローチ

顧客マスターテーブルを注文テーブルに関連付け、注文テーブルを製品マスターテーブルに関連付けて、レポートに必要なすべての情報を含むテーブルの一時的な関連付けを作成します。

- 顧客マスターテーブル内の顧客名と顧客拠点 -
- 注文テーブル内の注文詳細 -

- 製品マスターテーブル内の製品詳細 -

詳細な情報

詳細については、「テーブルの関連付け」ページ 999を参照してください。

どのデータ結合方法を使用すべきですか。

データの結合方法を選択するときには、複数の考慮事項があります。最初は以下のガイドラインを使用できます。

使用...	条件...
最後に追加	<ul style="list-style-type: none"> ○ 複数のテーブルを最小の労力で結合したいとします。 ○ ソーステーブルのレコードは、構造が似ているか、またはまったく同じです。
抽出/最後に追加	<ul style="list-style-type: none"> ○ 2つのソーステーブルのテーブルのレコードまたはフィールドは、構造がまったく同じです。
マージ	<ul style="list-style-type: none"> ○ 2つのソーステーブルのテーブルのレコードは、構造がまったく同じです。 ○ 両方のソーステーブルは同じ並べ替え順序を使用して並べ替えられます。 <p>ヒント</p> <p>マージを正常に実行するには、特別な方法があります。追加してから並べ替える、あるいは抽出または追加してから並べ替えるのでも、同じ結果を得ることができます。</p> <p>2つのソーステーブルが既に並べ替えられている場合は、マージはより効率的で、より迅速に実行できます。</p>
結合	<ul style="list-style-type: none"> ○ 2つのソーステーブルのレコードには異なるレコード構造があります。 ○ 共通キーフィールド内の一致する、または一致しない値に基づいて、レコードを包含または除外する必要があります。 ○ 物理的な結合テーブルが必要な調査分析を行っています。
関連付け	<ul style="list-style-type: none"> ○ 異なるレコード構造の最大 18 テーブルを関連付け(実際には結合)しようとしています。 ○ 共通キーフィールド内の一致する、または一致しない値に基づいて、レコードを包含または除外する必要があります。 ○ 結合されたデータを、新しいテーブルに出力する必要がありません。 ○ レポートのように、テーブル間の一時的な関連付けのみが必要な情報作業を実施しています。 <p>ヒント</p> <p>必要であれば、テーブルの関連付けの後、別の操作を実行し、関連付けられたテーブル内にあるフィールドの任意の組み合わせを新しい物理テーブルに抽出することができます。</p>

データ構造

データを結合するときどの方法を選択するかは、そのソースデータの構造によって大きく左右されます。データ構造(レコード構造)は、レコードに含まれるデータ要素、データ型、フィールドの長さ、および列数と並べ順を示します。

データ構造の詳細については、「データ構造およびデータ書式の要件」ページ 918を参照してください。

実験が必要な場合があります

状況によっては、データを結合するためにどの方法を用いれば良いかがすぐには明らかにならないことがあります。実行したいタスクに最も適した方法を決定するために、データの小さなサブセットを使って試してみる必要があるかもしれません。

小さなサブセットを使用すると、大きなテーブルの処理に伴う長い処理時間を回避できるほか、パターンが調べやすくなる可能性もあります。

目的を達成するために、2つ以上のデータの結合方法を使用することができます。

1つのデータ結合方法を最初に使用してから、最初の方法の出力結果で2番目の方法を使用し、複数のテーブルを含む、より複雑なデータ結合の状況に対処できる場合があります。

例

1. 最初に、月次取引テーブルを結合し、年次取引テーブルを作成します。
2. 顧客 ID などの共通キーフィールドを使用して、年次取引テーブルを、顧客名などのデータを含むマスターテーブルに結合します。

データの結合の代替方法

場合によっては、Analytics のネイティブ データ結合方法のいずれか以外の方法を使用して、データを結合する方が簡単であったり、実践的であることがあります。

メモ

代替方法の適合性は、データ分析ワークフローとソース データの本質によって異なります。方法は適切な場合とそうでない場合があります。

代替方法	説明
データアクセス ウィンドウを使用したテーブルの結合	データ アクセス ウィンドウを使用してデータを Analytics にインポートするときに、最大 10 個のテーブルを結合できます。 詳細については、「データ アクセス ウィンドウのテーブルの結合」ページ 388 を参照してください。
サードパーティ アプリケーションを使用したデータの結合	Excel や Access などの作成元アプリケーションのネイティブ機能を使用して、データを結合し、結合されたデータを Analytics にインポートします。
データの往復	同じデータ要素を含むフィールド間の不一致のため、Analytics でデータの追加またはマージが難しい場合は、データの往復を試すことができます。 a. Analytics テーブルのいずれかを区切りフラット ファイルにエクスポートします。 b. 追加の Analytics テーブルを最初のテーブルにエクスポートして追加します。 c. 単一のフラット ファイルにすべてのデータを一体化したら、そのフラット ファイルを Analytics に再インポートします。 区切り文字付きフラット ファイルは、Analytics テーブルに比べるとデータ構造の不整合についてそれほど厳格ではありません。 ○ データ型 やフィールド長は同じでなくてもかまいません。 ○ データ要素、および列の数と順序だけは同一である必要があります。 この手法は、Analytics 内部でデータとフィールドを一致させるよりも労力が少なく済むかもしれません。

データ構造およびデータ書式の要件

データ構造の概要

データ構造またはレコード構造は次のことを指します。

- レコードに含まれるフィールド(データ要素)
- フィールドの数と順序
- フィールドのデータ型と長さ

フィールドは、姓、名、住所、業者 ID など、レコードに含まれる個々のデータ単位です。

ヒント

2つのテーブルのデータ構造の比較に、DISPLAY コマンドを使用する方法については、"データ構造の比較" ページ 209を参照してください。

データ書式の概要

データ書式は、次のようなフィールドに含まれる値の特性を示します。

- 位置揃え
- 大文字/小文字
- 日付の書式

同じデータ構造または書式の要件

Analytics では、データの結合方法に応じて、結合されるテーブル内のレコードまたはフィールドは、同じデータ構造を持っている必要があります。場合によっては、フィールド値は同じ書式である必要があります。

ヒント

データ構造またはフィールドのデータ書式が異なる場合は、Analytics 関数を使用してフィールドを一致させることができる可能性があります。詳細については、"フィールドの一致" ページ 921を参照してください。

データ結合の要件	データ構造の要件	データ書式の要件
最後に追加	フィールドが直接互いに追加されるためには、同じ物理名を持ち、同じデータカテゴリに属している必要があります。	要件なし
抽出/最後に追加	抽出されるテーブルおよび追加されるテーブ	日付時刻書式が同じである必要があります

データ結合の要件	データ構造の要件	データ書式の要件
	ル内の、レコード、ビュー内の全フィールド、または選択したすべてのフィールドは構造内で正確に同じである必要があります。	
マージ	マージされるテーブルの全体レコードは、構造がまったく同じである必要があります。	日付時刻書式が同じである必要があります
結合 関連付け	結合または関連付けられるテーブルの共通のキーフィールドは、構造がまったく同じである必要があります。	結合および関連付けでは、値が正しく照合されるように、共通キーフィールドの値はあらゆる点で等しく書式設定されている必要があります。

データ構造の詳細要件

結合されるテーブルの要件	最後に追加 (共通フィールド)	抽出/最後に追加 (レコード全体、 ビュー内のフィールド または選択フィールド)	マージ (レコード全体)	結合 (共通キーフィールド)	関連付け (共通キーフィールド)
フィールド(データ要素)	同じデータ要素である必要があります	同じデータ要素である必要があります	同じデータ要素である必要があります	同じデータ要素である必要があります	同じデータ要素である必要があります
フィールド名	同じである必要があります	異なってもかまいません	異なってもかまいません	異なってもかまいません	異なってもかまいません
フィールドの数と順序	数が同じである必要があります 順序は異なっても構いません	同じである必要があります	同じである必要があります	該当なし	該当なし
対応するフィールドのデータ型	データカテゴリが同じである必要があります 日付時刻サブタイプが同じである必要があります 文字および数値データサブタイプの自動調整	同じである必要があります	同じである必要があります	同じである必要があります 文字 - 数値結合の自動一致	同じである必要があります
対応するフィールドのフィールド長	異なってもかまいません	同じである必要があります	同じである必要があります	データ型によって、同じである必要があるか、または同じであること	同じであることが推奨されます

結合されるテーブルの要件	最後に追加 (共通フィールド)	抽出/最後に追加 (レコード全体、 ビュー内のフィールド または選択 フィールド)	マージ (レコード全体)	結合 (共通キーフィールド)	関連付け (共通キーフィールド)
				が推奨されます 文字キーフィールドの長さの自動一致	

データ書式の詳細要件

対応するフィールドの要件	最後に追加 (共通フィールド)	抽出/最後に追加 (レコード全体、 ビュー内のフィールド または選択 フィールド)	マージ (レコード全体)	結合 (共通キーフィールド)	関連付け (共通キーフィールド)
日付/日付時刻書式	日付、日付時刻、または時刻書式の自動調整	同じである必要があります	同じである必要があります	異なってもかまいません	異なってもかまいません
位置揃え	異なってもかまいません	異なってもかまいません	異なってもかまいません 違いは、キーフィールドの並べ替え順序に影響を与えます	同じである必要があります	同じである必要があります
大文字/小文字	異なってもかまいません	異なってもかまいません	異なってもかまいません 違いは、キーフィールドの並べ替え順序に影響を与えます	同じである必要があります	同じである必要があります
フィールド値の書式 (標準ハイフネーション、ストリート形式の略語、など)	異なってもかまいません	異なってもかまいません	異なってもかまいません 違いは、キーフィールドの並べ替え順序に影響を与えます	同じである必要があります	同じである必要があります

フィールドの一致

Analytics でテーブルを正しく結合するためには、まず、結合される 2 つのテーブル内の 1 つまたは複数のフィールドを一致させる必要があるかもしれません。

一致の概要

一致とは、別のテーブルの対応するフィールドの構造を同一にするプロセスのことです。たとえば、フィールドのデータ型を統一することなどが含まれます。

また、2 つの対応するフィールド内の値の書式を同一にすることも指します。ID 番号のハイフンの使い方を統一することなどが含まれます。

対応するフィールドの構造またはフィールドの値の書式が同一でない場合は、雑然としたデータが生じる結果となったり、結合操作を実行できない可能性があったり、または結合や関連付けで値を正しく照合できない可能性があります。

関数と演算フィールドを使用してフィールドを一致させる

Analytics 関数を使用して演算フィールドを作成することが、フィールドを一致させるための主要な手法です。たとえば、変換関数を使用すると、フィールドのデータ型をある型から別の型へ変換することができます。ほかの関数では、フィールド長、位置揃え、および大文字小文字を変更したり、フィールド内の値の書式を統一化したりすることができます。

2 つのフィールド間の不一致の程度によっては、正常にフィールドを一致させるために一連の関数を使用する必要があるかもしれません。

フィールドを一致させると、次の方法のいずれかを使用してデータを結合できます。

- 共通キーフィールドの**結合**または**関連付け**-は、一致したフィールドと元のキーフィールドまたは 2 つの一致したフィールドを使用します。
- 1 つまたは両方のテーブルの**追加**、**抽出と最後に追加**、または**マージ**-では、1 つ以上の一致したフィールドが作成され、そのフィールドを基準に抽出が行われて、一致した演算フィールドは実際の演算値が入力された物理フィールドに変換されます。この抽出後のテーブルと物理フィールドをデータ結合操作で使用します。

詳細については、「データの抽出」ページ 201と「演算フィールドの抽出と追加」ページ 949を参照してください。

フィールドを一致させるための Analytics 関数

フィールドを一致させるために使用できる Analytics 関数について、以下に概略を説明します。特定の関数の使い方の詳細については、「関数の概要」ページ 2139を参照してください。

Analytics 関数	カテゴリ	目的
STRING()	データ型の変換 (N から C)	数値データを文字データに変換します。
ZONED()		数値データを文字データ(ASCII ZONED データ形式)に変換し、データの先頭にゼロを追加します。
VALUE()	データ型の変換 (C から N)	文字データを数値データに変換します。
CTOD()	データ型の変換 (C または N から D)	文字または数値の日付を日付データに変換します。
CTODT()		文字または数値の日付時刻を日付時刻データに変換します。
CTOT()		文字または数値の時刻を時刻データに変換します。
DATE()	データ型の変換 (D から C)	日付データを文字データに変換します。
DATETIME()		日付時刻データを文字データに変換します。
TIME()		時刻データを文字データに変換します。
STOD()	データ型の変換 (シリアルの N から D)	シリアル日付を日付データに変換します。
STODT()		シリアル日付時刻を日付時刻データに変換します。
STOT()		シリアル時刻を時刻データに変換します。
SUBSTRING()	長さの調整	文字列の指定された部分を抽出します(既存の文字列全体と等しくすることができます)。フィールド長を短くしたり長くしたりするために使用できます。指定した長さが既存の文字列より長い場合は、末尾に空白が追加されます。
BLANKS()		指定された長さの空白文字列を作成します。文字データの先頭または末尾に空白を追加するために使用できます。
LTRIM()	長さ調整/両端揃え	文字データから先頭の空白を除去します。
TRIM()		文字データから末尾の空白を除去します。
ALLTRIM()		文字データから先頭と末尾の空白を除去します。
RJUSTIFY()		末尾の空白をすべて先頭の空白に変換させて、文字データを右寄せにします。

分析用のデータの準備

Analytics 関数	カテゴリ	目的
UPPER()	大文字と小文字の変換	アルファベット文字を大文字に変換します。
LOWER()		アルファベット文字を小文字に変換します。
PROPER()		各単語の先頭文字を大文字にし、残りの文字を小文字に変換します。
INCLUDE()	書式の変更	文字列から指定された文字を抽出します。 たとえば、英数字のデータから数字だけを抜き出すことができます。
REMOVE()		文字列から指定された文字を抽出し、末尾に空白を追加することによって、元の文字列の長さを保持します。
EXCLUDE()		文字列から指定された文字を取り除きます。 たとえば、英数字のデータから数字を取り除くことができます。また、"123-45-4536" からハイフンを除いて、文字列 "123454536" を出力することができます。
OMIT()		文字列から、指定された文字または部分文字列を取り除きます。 たとえば、業者名から "Corporation"、"Inc." または "Ltd." を取り除きます。
INSERT()		文字列に指定された文字を挿入します。 たとえば、"123454536" にハイフンを挿入して、文字列 "123-45-4536" を出力することができます。
SPLIT()		文字データをスペースやカンマなどの区切り文字に基づいてセグメントに分け、指定されたセグメントを抽出します。
CLEAN()		タブやキャリッジリターンなどの無効な文字、および指定された任意の文字を文字列から削除し、それ以降のすべての文字と、削除された文字を空白に置き換えます。
REPLACE()		既存の文字列のすべてのインスタンスを新しい文字列で置き換えます。たとえば、"Rd." を "Road" と置き換えることができます。
DEC()		数値フィールドの小数点以下の桁数を指定します。

データ結合方法の比較

Analytics で異なるデータ結合方法を使用する利点と問題点について、以下で説明します。

メモ

追加、抽出と追加、およびマージは比較されます。これは、これらの方法が同一または類似のレコード構造を持つテーブル同士を結合するからです。

結合は関連付けと比較します。これは、この2つの方法が異なるレコード構造を持つテーブル同士を結合するからです。

詳細については、「データ構造およびデータ書式の要件」ページ 918を参照してください。

追加、抽出と追加、およびマージ

要件/機能	追加	抽出と追加	マージ
結合されるテーブルは同じデータ構造である必要があります。	いいえ	はい	はい
結果として生じる結合されたテーブルは並べ替えられています。	いいえ ソーステーブルから抽出されたレコードは、対象テーブルにグループとして追加されます。	いいえ ソーステーブルから抽出されたレコードは、グループとして、対象テーブルの最後に追加されます。	はい 両方のテーブルが並べ順に基づき、新しい3番目のテーブルに挿入されます。
2つのテーブルを基とするデータにアクセスして分析する	はい	はい	はい
3つ以上のテーブルを基とするデータにアクセスして分析する	はい	単一の抽出および追加操作ではサポートされていません。複数の操作が必要となります。	単一のマージ操作ではサポートされていません。複数の操作が必要となります。
物理的に別個の新規 Analytics テーブルに結果を出力する	はい	いいえ	はい
両方のテーブルのキーフィールドの要件： <ul style="list-style-type: none"> 並べ替え 同じデータ型 同じ長さ 	該当なし 追加はキーフィールドを使用しません。	該当なし 抽出と追加はキーフィールドを使用しません。	はい
キーフィールドの数	該当なし	該当なし	1個以上のキーフィールドを各テーブルから選択できます

要件/機能	追加	抽出と追加	マージ
	追加はキーフィールドを使用しません。	抽出と追加はキーフィールドを使用しません。	す。

結合と関連付け

メモ

結合したいテーブルが同じレコード構造をしている場合は、通常、追加、抽出/追加、またはマージを使用する必要があります。

機能の比較

機能	結合	関連付け
使用例	永久に結合された新しい3番目のテーブルを出力するため、調査作業の準備ステップに適しています。	永続化されるという要件のない仮想テーブルを作成するため、情報作業に適しています。
2つのテーブルを基とするデータに同時にアクセスして分析する	はい	はい
3つ以上のテーブルを基とするデータに同時にアクセスして分析する	いいえ 単一の結合操作ではサポートされていません。複数の結合操作が必要となります。	はい 単一の関連付け操作で、同時に最大18個までのテーブルのアクセス/分析をサポートしています。
物理的に別個の新規 Analytics テーブルに結果を出力する	はい	いいえ 必要であれば、別の操作を実行し、関連付けられたテーブル内にあるフィールドの任意の組み合わせを新しいテーブルに抽出することができます。
キーフィールドの数	1個以上のキーフィールドを各テーブルから選択できます。	テーブルの組ごとに1つのキーフィールドに限定されます。 テーブルの組間の正確な関係を確立するために複数のキーフィールドが必要な場合は、必要なキーフィールドを連結するために、各テーブルに演算フィールドを作成します。
操作の実行速度	遅い 結合操作の期間は、結合の複雑さと、主テーブルが並べ替えられているかどうかによって変化します。	速い 実際のレコードの照合は、関連付けの操作中には行われません。このため、実行期間は結合よりもかなり短くなります。

機能	結合	関連付け
ファイルの後続の処理	速い 結合の結果は、フラット ファイル(.fil ソース データ ファイル)に格納されます。フラット ファイルは非常に速く処理できます。	遅い 関連テーブル間のレコードの照合は後続の処理時に行われるため、処理時間が増えます。
ソース データ ファイルからの更新	いいえ 結合の結果は、新しいソース データ ファイルに含まれる新しい第 3 のテーブルへ出力され、結合にかかわったソース データ ファイルとの関連付けはなくなります。	はい 関連テーブルは関連付けられたままで、関係にかかわるソース データ ファイルから更新することができます。
一致した主レコードおよび副レコード (一致する 1 件目の副レコード)	はい	直接はサポートされていません テーブルを関連付けした後、フィルターを使用して、一致する副レコードがある主レコードを分離します。
一致した主レコードおよび副レコード (重複する副レコードのすべての一致) 多対多一致とも呼ばれます	はい	いいえ
不一致の主レコード	はい	直接はサポートされていません テーブルを関連付けした後、フィルターを使用して、一致する副レコードがない主レコードを分離します。
すべての主レコードおよび一致した副レコード	はい	はい
すべての副レコードおよび一致した主レコード	はい	いいえ
すべての主レコードおよび副レコード (一致および不一致)	はい	いいえ

要件の比較

要件	結合	関連付け
結合されるテーブルは同じデータ構造である必要があります。	いいえ	いいえ
キーフィールドのデータ型は、テーブルの組ごとで同じでなければならない	変わる 文字 - 数値、または数値 - 文字の結合では必要ありません。Analytics に	はい

分析用のデータの準備

要件	結合	関連付け
	よって自動的に調整されます。その他すべての可能性では必須です。	
キーフィールドの長さは、テーブルの組ごとで同じでなければならない	推奨(強制されません) 2つの文字キーフィールドの長さは、Analyticsによって自動的に調整されます。	推奨(強制されません)
処理に必要なディスクの空き領域	多い 結合は、新しい第3のテーブルを作成します。このテーブルは、結合の性質によって、元の2つのテーブルを合わせたよりも大きくなる場合があります。	少ない 子テーブルのインデックスを作成するための最低限のディスク空き容量が必要です。
テーブルは並べ替えられているか、インデックスが作成されている必要がある	並べ替え、[あらかじめ並べ替える]、またはインデックスは、副テーブルでは必須、主テーブルでは任意です。	子テーブルではインデックスが必須(テーブルを関連付けるときに自動的に作成されます)で、親テーブルの並べ替えまたはインデックスは任意です。

テーブルの追加

テーブルの追加により、2 つ以上の Analytics テーブル内のレコードが新しいテーブルに結合されます。分析を実行する前に、複数のテーブルを単一のテーブルの最後に追加しなければならない場合があります。

たとえば、1 年全体に相当するデータに対して分析を実行したいが、データが 12 の月間 Excel ワークシートに分散している場合を考えます。個別のワークシートを Analytics にインポートした後に、それらを最後に追加し、分析用の単一の年次テーブルを作成できます。

追加作業の方法

ここで、追加とはレコードグループを別のレコードグループの末尾に付加することです。各ソーステーブル内のレコードが、それらのテーブルを選択する順序と同じ順序で追加されます。新しいテーブルには、最初に選択したテーブルのレコード、2 番目に選択したテーブルのレコード、... というような順序で格納されます。

各ソーステーブルは、レコード構造が同じでも異なってもどちらでも構わず、また、並べ替えても並べ替えなくても構いません。

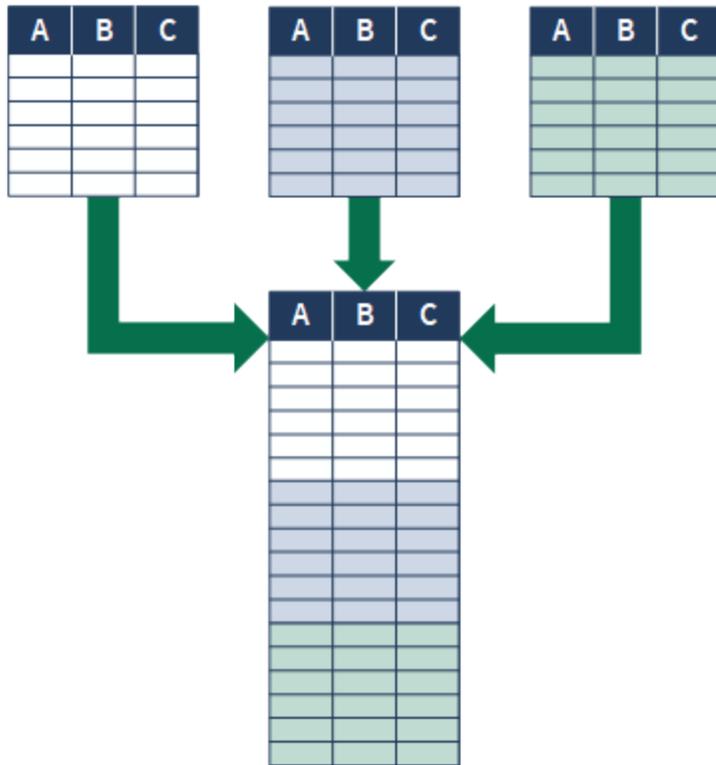
フィールドが追加される仕組み

複数のテーブルのレコードを追加するときには、レコード内の個別のフィールドが追加される方法は、フィールドに同一の名前があるか一意の名前があるかどうかによって異なります。

名前が同じフィールド

同じ物理名と同じデータカテゴリを持つソーステーブルフィールドは、互いのフィールドに直接追加されません。

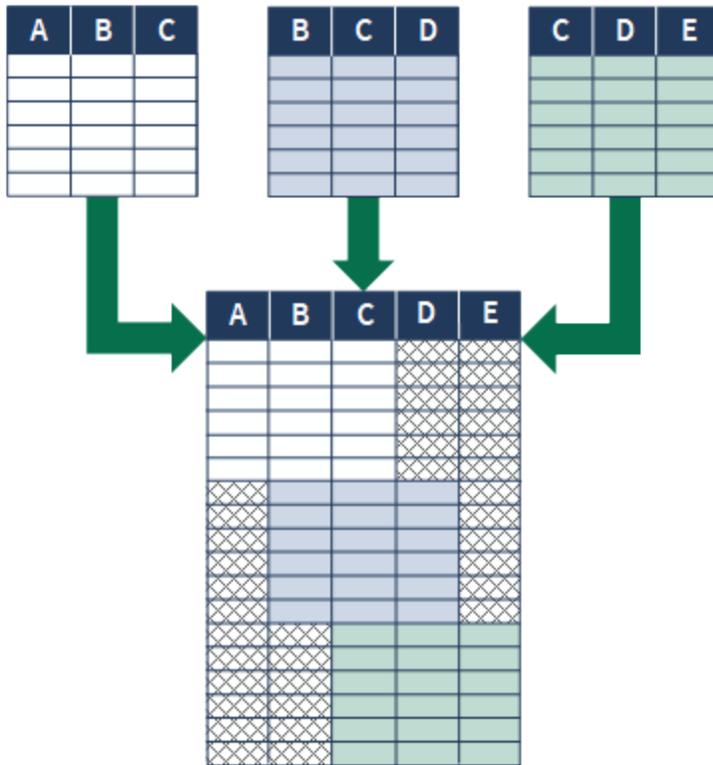
次の図で、フィールド A、B、C はすべて直接追加されます。



名前が一意のフィールド

すべてのソーステーブルにおいて一意である物理名を持つフィールドは、出力テーブルには追加されますが、互いのフィールドには直接追加されません。

次の図で、フィールド A と E は、このフィールド追加法が使用されたフィールドの例です。

**ヒント**

名前が異なるフィールドを直接追加したい場合は、事前に各テーブルレイアウト内のフィールドの物理名を統一しておいてください(フィールドが同じデータカテゴリに属しているか、フィールドのデータカテゴリを一致させていると仮定します。) 詳細については、"物理フィールドを定義する" ページ 790を参照してください。

追加の実行に適する場面

追加は、構造が同じまたは類似である複数のテーブル内のデータを結合したい場合に使用できます。たとえば、月次テーブルまたは四半期テーブルを年次テーブルに結合する場合に使用することをお勧めします。

ヒント

APPEND コマンドを 1 回実行しただけで、APPEND オプションを指定した EXTRACT コマンドの複数回の実行結果が置き換えられます。

例

シナリオ

1年全体に相当するデータに対して分析を実行したいが、データが12の月間取引テーブルに分散している場合を考えます。

アプローチ

12の月間テーブルのデータを、それらのすべてを含む単一の年間テーブルに追加し、その後分析を実行します。

追加に適さない場合とは

追加は結合および関連付けの代替としては使用できません。その理由は、共通キーフィールド内の一一致する、または一致しない値に基づいてレコードを包含または除外することができないためです。追加では、すべてのソーステーブル内の全レコードが出力テーブルに追加されます。

類似性がまったくないテーブルの追加

類似性がまったくないテーブル、つまり共通のフィールドがない2つ以上のテーブルを追加することができます。類似性がないテーブルを追加することが分析目的に資する場合があります(これは追加の意図した主な用途ではありません)。

すべてのフィールドまたは共通フィールドのみの追加

テーブルを追加する場合、オプションは2つあります。

- すべてのソーステーブル内の全フィールドを追加する
- すべてのソーステーブルに**共通**のフィールド、つまりすべてのテーブルに存在するフィールドのみを追加する

フィールドが共通と見なされるためには、同じ物理名を持ち、次のいずれかのデータカテゴリのうち同じデータカテゴリに属している必要があります。

- 文字
- 数値
- 日付時刻
- 論理

例：3つの従業員テーブルの追加

従業員データを分析する前に、3つの従業員テーブルを従業員マスターテーブルに追加したいとします。

これら3つのテーブルには、どのテーブルにも存在する3つの共通フィールドがあります。

- Employee_number
- First_name
- Last_name

さらに、1つまたは2つのテーブルには存在するが、3つのテーブルすべてに存在するわけではない、共通でないフィールドが2つあります。

- Middle_name
- Email

入力

これら追加対象となる3つのテーブルは次のとおりです。

テーブル名	フィールド																														
Employees_central	<table border="1"> <thead> <tr> <th></th> <th>Employee_number</th> <th>First_name</th> <th>Middle_name</th> <th>Last_name</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>190</td> <td>Sybil</td> <td>Denise</td> <td>Johnson</td> </tr> <tr> <td>2</td> <td>170</td> <td>Catherine</td> <td>Eleanor</td> <td>Exelby</td> </tr> <tr> <td>3</td> <td>140</td> <td>Abed</td> <td>Aziz</td> <td>Bhatti</td> </tr> <tr> <td colspan="5"><< ファイルの終わり >></td> </tr> </tbody> </table>		Employee_number	First_name	Middle_name	Last_name	1	190	Sybil	Denise	Johnson	2	170	Catherine	Eleanor	Exelby	3	140	Abed	Aziz	Bhatti	<< ファイルの終わり >>									
		Employee_number	First_name	Middle_name	Last_name																										
	1	190	Sybil	Denise	Johnson																										
	2	170	Catherine	Eleanor	Exelby																										
3	140	Abed	Aziz	Bhatti																											
<< ファイルの終わり >>																															
Employees_east	<table border="1"> <thead> <tr> <th></th> <th>Employee_number</th> <th>First_name</th> <th>Last_name</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>330</td> <td>Vincent</td> <td>Scarpetta</td> </tr> <tr> <td>2</td> <td>160</td> <td>Oliver</td> <td>Woye</td> </tr> <tr> <td>3</td> <td>60</td> <td>Savi</td> <td>Madan</td> </tr> <tr> <td colspan="4"><< ファイルの終わり >></td> </tr> </tbody> </table>		Employee_number	First_name	Last_name	1	330	Vincent	Scarpetta	2	160	Oliver	Woye	3	60	Savi	Madan	<< ファイルの終わり >>													
		Employee_number	First_name	Last_name																											
	1	330	Vincent	Scarpetta																											
	2	160	Oliver	Woye																											
3	60	Savi	Madan																												
<< ファイルの終わり >>																															
Employees_west	<table border="1"> <thead> <tr> <th></th> <th>Employee_number</th> <th>First_name</th> <th>Middle_name</th> <th>Last_name</th> <th>Email</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>110</td> <td>John</td> <td>David</td> <td>Mullen</td> <td>jmullen@example.net</td> </tr> <tr> <td>2</td> <td>280</td> <td>Emma</td> <td>Clare</td> <td>Pickford</td> <td>epickford@example.net</td> </tr> <tr> <td>3</td> <td>120</td> <td>Jorge</td> <td>Alberto</td> <td>Garcia</td> <td>jpgarcia@example.net</td> </tr> <tr> <td colspan="6"><< ファイルの終わり >></td> </tr> </tbody> </table>		Employee_number	First_name	Middle_name	Last_name	Email	1	110	John	David	Mullen	jmullen@example.net	2	280	Emma	Clare	Pickford	epickford@example.net	3	120	Jorge	Alberto	Garcia	jpgarcia@example.net	<< ファイルの終わり >>					
		Employee_number	First_name	Middle_name	Last_name	Email																									
	1	110	John	David	Mullen	jmullen@example.net																									
	2	280	Emma	Clare	Pickford	epickford@example.net																									
3	120	Jorge	Alberto	Garcia	jpgarcia@example.net																										
<< ファイルの終わり >>																															

出力 - すべてのフィールドが追加されます。

すべてのフィールドを追加する場合は、これら追加される3つのテーブル内の全レコードと全フィールドが出力テーブルに追加されます。

ソーステーブルにフィールドがない場所は、出力テーブルでは空白値になります。

テーブル名	フィールド																																																																		
Employees_ master	<table border="1"> <thead> <tr> <th></th> <th>Employee_number</th> <th>First_name</th> <th>Middle_name</th> <th>Last_name</th> <th>Email</th> </tr> </thead> <tbody> <tr><td>1</td><td>190</td><td>Sybil</td><td>Denise</td><td>Johnson</td><td></td></tr> <tr><td>2</td><td>170</td><td>Catherine</td><td>Eleanor</td><td>Exelby</td><td></td></tr> <tr><td>3</td><td>140</td><td>Abed</td><td>Aziz</td><td>Bhatti</td><td></td></tr> <tr><td>4</td><td>330</td><td>Vincent</td><td></td><td>Scarpetta</td><td></td></tr> <tr><td>5</td><td>160</td><td>Oliver</td><td></td><td>Woye</td><td></td></tr> <tr><td>6</td><td>60</td><td>Savi</td><td></td><td>Madan</td><td></td></tr> <tr><td>7</td><td>110</td><td>John</td><td>David</td><td>Mullen</td><td>jmullen@example.net</td></tr> <tr><td>8</td><td>280</td><td>Emma</td><td>Clare</td><td>Pickford</td><td>epickford@example.net</td></tr> <tr><td>9</td><td>120</td><td>Jorge</td><td>Alberto</td><td>Garcia</td><td>kgarcia@example.net</td></tr> <tr><td colspan="6"><< ファイルの終わり >></td></tr> </tbody> </table>		Employee_number	First_name	Middle_name	Last_name	Email	1	190	Sybil	Denise	Johnson		2	170	Catherine	Eleanor	Exelby		3	140	Abed	Aziz	Bhatti		4	330	Vincent		Scarpetta		5	160	Oliver		Woye		6	60	Savi		Madan		7	110	John	David	Mullen	jmullen@example.net	8	280	Emma	Clare	Pickford	epickford@example.net	9	120	Jorge	Alberto	Garcia	kgarcia@example.net	<< ファイルの終わり >>					
		Employee_number	First_name	Middle_name	Last_name	Email																																																													
	1	190	Sybil	Denise	Johnson																																																														
	2	170	Catherine	Eleanor	Exelby																																																														
	3	140	Abed	Aziz	Bhatti																																																														
	4	330	Vincent		Scarpetta																																																														
	5	160	Oliver		Woye																																																														
	6	60	Savi		Madan																																																														
	7	110	John	David	Mullen	jmullen@example.net																																																													
	8	280	Emma	Clare	Pickford	epickford@example.net																																																													
9	120	Jorge	Alberto	Garcia	kgarcia@example.net																																																														
<< ファイルの終わり >>																																																																			

出力 - 共通フィールドのみが追加されます。

共通フィールドのみを追加する場合は、追加される3つのテーブル内の全レコードと共通フィールドのみが出力テーブルに追加されます。

テーブル名	フィールド																																												
Employees_ master	<table border="1"> <thead> <tr> <th></th> <th>Employee_number</th> <th>First_name</th> <th>Last_name</th> </tr> </thead> <tbody> <tr><td>1</td><td>190</td><td>Sybil</td><td>Johnson</td></tr> <tr><td>2</td><td>170</td><td>Catherine</td><td>Exelby</td></tr> <tr><td>3</td><td>140</td><td>Abed</td><td>Bhatti</td></tr> <tr><td>4</td><td>330</td><td>Vincent</td><td>Scarpetta</td></tr> <tr><td>5</td><td>160</td><td>Oliver</td><td>Woye</td></tr> <tr><td>6</td><td>60</td><td>Savi</td><td>Madan</td></tr> <tr><td>7</td><td>110</td><td>John</td><td>Mullen</td></tr> <tr><td>8</td><td>280</td><td>Emma</td><td>Pickford</td></tr> <tr><td>9</td><td>120</td><td>Jorge</td><td>Garcia</td></tr> <tr><td colspan="4"><< ファイルの終わり >></td></tr> </tbody> </table>		Employee_number	First_name	Last_name	1	190	Sybil	Johnson	2	170	Catherine	Exelby	3	140	Abed	Bhatti	4	330	Vincent	Scarpetta	5	160	Oliver	Woye	6	60	Savi	Madan	7	110	John	Mullen	8	280	Emma	Pickford	9	120	Jorge	Garcia	<< ファイルの終わり >>			
		Employee_number	First_name	Last_name																																									
	1	190	Sybil	Johnson																																									
	2	170	Catherine	Exelby																																									
	3	140	Abed	Bhatti																																									
	4	330	Vincent	Scarpetta																																									
	5	160	Oliver	Woye																																									
	6	60	Savi	Madan																																									
	7	110	John	Mullen																																									
	8	280	Emma	Pickford																																									
9	120	Jorge	Garcia																																										
<< ファイルの終わり >>																																													

自動調整

次の場合、Analyticsにより、フィールドを追加するために、フィールドが自動的に調整されます。

フィールドのデータカテゴリ	調整の実行
文字	<ul style="list-style-type: none"> 異なるフィールド長は調整されます。 フィールドを ASCII または UNICODE データ型に変換することで、カスタム、PCASCII、EBCDIC など様々な文字データ型が調整されます。
数値	<ul style="list-style-type: none"> 異なるフィールド長は調整されます。フィールドは ACL データ型に変換されます。 異なる小数点以下桁数が定義されている場合は、調整されます。小数点以下桁数は、必要に応じて数値の末尾にゼロが付加されて、最大の桁数に統一されます。フィールドは ACL データ型に変換されます。 フィールドを ACL データ型に変換することで、Print、Float、EBCDIC、Micro など複数のデータ

フィールドのデータカテゴリ	調整の実行
	型が調整されません。
日付時刻	<ul style="list-style-type: none"> フィールドを Analytics のデフォルト書式に変換することで、ソースデータ内にある複数の日付書式、日付時刻書式、時刻書式が調整されます。 <ul style="list-style-type: none"> YYYYMMDD YYYYMMDD hh:mm:ss hh:mm:ss

自動調整が実行されない場合とは

次の場合、フィールドは Analytics によって自動調整されません。エラーメッセージが表示され、追加操作は実行されません。

- 名前が同じ 2 つのフィールドが異なるデータカテゴリに属する
- 名前が同じ 2 つの日付時刻フィールドが異なる日付時刻サブタイプ(日付、日付時刻、時刻)に属する
- 名前が同じ 2 つの日付時刻フィールドにおいて、タイムゾーンインジケータの使用有無が一致していない

これらの状況のいずれかが発生した場合は、"ユーザーが指定した一致" 下を参照してください。

ユーザーが指定した一致

[最後に追加] ダイアログボックスの 2 つのオプションを使用すると、異なるデータカテゴリに属する同一の名前のフィールドを一致させ、エラーなくフィールドを追加することができます。オプションは、文字型データカテゴリの同じ名前のフィールドを統一することによって動作します。

- 共通フィールドを調整するために文字データ型を使用する - ことで文字型でないフィールドを文字データカテゴリに変換するのは、そうすることが調整上必要な場合に限りです。
- すべてのフィールドを文字型データに変換すると、一致のために必要とされるか否かにかかわらず、文字データカテゴリにすべてのテーブルが最後に追加されます

例

シナリオ

Employee_ID フィールドが 1 つのテーブル内の文字型データであり、もう 1 つのテーブル内の数値データである 2 つのテーブルを最後に追加したいとします。

アプローチ

[最後に追加] ダイアログボックスで、**文字データ型を使用して、共通フィールドを(異なるデータ型と)合わせます**を選択します。数値的 Employee_ID フィールドは文字データに変換され、2 つ

のフィールドはエラーなしで追加されます。Employee_ID フィールドは文字データに変換され、2つのフィールドはエラーなしで最後に追加されます。

ヒント

文字型データのカテゴリを一致させることがニーズを満たしていない場合は、異なる手法を使用してフィールドを手作業で一致させるか、1つ以上のフィールドを再定義することができます。詳細については、「フィールドの一致」ページ 921と「物理フィールドを定義する」ページ 790を参照してください。

演算フィールドがサポートされていない

演算フィールドを追加できません。テーブルを追加する場合、ソーステーブル内のすべての演算フィールドは自動的に出力テーブルから除外されます。

あるソーステーブル内の演算フィールドが、別のソーステーブル内の物理フィールドと同じ名前を持っている場合には、エラーメッセージが表示され、追加処理は実行されません。

ヒント

演算フィールドを追加するには、まずこのようなフィールドを抽出して、物理フィールドに変換します(詳細については、「データを抽出して追加する」ページ 945を参照してください)。次に、演算フィールドの抽出先となったテーブルを追加操作で使用します。別のアプローチとしては、追加された出力テーブルに演算フィールドを再作成します。

レコード ノート フィールドがサポートされていない

レコード ノート フィールドを追加できない場合です。テーブルを追加する場合、ソーステーブル内のすべてのレコード ノート フィールドは自動的に出力テーブルから除外されます。

あるソーステーブル内のレコード ノート フィールドが、別のソーステーブル内の物理フィールドと同じ名前を持っている場合には、エラーメッセージが表示され、追加処理は実行されません。

レコード ノート フィールドは、レコードにノートを追加したときに Analytics によって自動的に生成されるものです。詳細については、「レコード ノートの追加と編集」ページ 185を参照してください。

追加に関する追加情報

以下のテーブルでは追加に関する追加情報を提供します。

機能分野	詳細
レコード長	<p>追加時にすべてのソース テーブル内の全フィールドを含めるには、出力テーブルのレコード長がソース テーブル内の最も長いレコードより長くなる可能性があります。</p> <p>出力テーブルのレコード長が Analytics の上限である 32 KB を超えると、エラー メッセージが表示されます。</p>
日付時刻フィールド	<p>2 つ以上の日付時刻フィールドを追加するには、次の条件を満たす必要があります。</p> <ul style="list-style-type: none"> ◦ 物理名が同じ ◦ データ カテゴリが同じ(日付時刻) ◦ データ サブタイプが同じ(日付、日付時刻、時刻) ◦ タイムゾーン インジケータの使用有無の一致 - 追加対象となるすべてのフィールドで使用するか、使用しないかのいずれか <p>メモ</p> <p>異なる日付時刻フィールドを文字データ カテゴリに変換してすることでこれらのフィールドを一致させ、それから最後に追加することができます。このアプローチでは、データを 1 つのテーブルに結合することができます。ただし、ソース データの性質によっては、結合されたデータを日付時刻データに戻すことができない場合があります。</p>
小数点以下桁数	<p>小数位がある数値フィールドの追加は、特定の動作によって制御されます。</p> <p>小数位設定</p> <p>追加操作では、テーブルレイアウトのフィールド定義にある小数位設定に定義された小数点以下桁数が使用されます。</p> <p>メモ</p> <p>小数位設定は、ソース データ内の実際の小数点以下桁数とは同じでない場合があります。小数位設定を超える小数点以下桁数は未定義のため、計算では丸められます。</p> <p>統一されていない小数位設定</p> <p>最後に追加された数値フィールドの小数位設定が統一されていない場合には、それらのフィールドは ACL データ型に変換され、最も長い小数位設定に自動的に調整されます。</p> <p>最も長い小数位設定を超えるソース データ ファイル内の小数点以下桁数は、追加操作で生成される出力テーブルから除外されます。</p> <p>統一されている小数位設定</p> <p>追加された数値フィールドの小数位設定が統一されている場合には、データ型変換も調整も行われません。</p> <p>小数位設定を超えるソース データ ファイル内の小数点以下桁数は、追加操作で生成される出力テーブルに追加されます。</p>
並べ替え	<p>ソース テーブルに存在するすべての並べ替え順は、出力テーブル内のそれぞれのレコード セットで別々に保持されます。</p> <p>たとえ両方のテーブルのレコードが並べ替えられていても、結果として生じる結合されたテーブルは、並べ替えられていないテーブルと見なされます。これは、抽出されたレコードを出力先テーブルの末尾にグループとして追加するときに、出力先テーブルの既存の並べ替え順が何も考慮されないからです。</p> <p>たとえば、月次または四半期テーブルを追加して年次テーブルを作成する場合は、月次または四</p>

機能分野	詳細												
	<p>半期データの内部並べ替えは保持されます。必要であれば、追加操作を実行した後で、出力テーブルを並べ替えることもできます。</p>												
<p>フィールドの順番</p>	<p>共通フィールド</p> <p>ソーステーブルの共通フィールドは、同じ順序になっていなくても追加できます。</p> <p>たとえば、以下のフィールドは順序が異なっても正常に追加されます。</p> <table border="1" data-bbox="451 495 1411 674"> <thead> <tr> <th>テーブル</th> <th>フィールド</th> </tr> </thead> <tbody> <tr> <td>表 1</td> <td>ラストネーム ファーストネーム ミドルネーム</td> </tr> <tr> <td>表 2</td> <td>ファーストネーム ミドルネーム ラストネーム</td> </tr> </tbody> </table> <p>出力テーブル内でのフィールドの順序は、最後に追加ダイアログボックスで選択した最初のテーブルによって決まります。したがって、上の例では、出力テーブルの順序は次のようになります。</p> <ul style="list-style-type: none"> ラストネーム ファーストネーム ミドルネーム <p>共通でないフィールド</p> <p>ソーステーブル内の共通でないフィールドは、選択したテーブルグループに出現している順に出力テーブルに出力されます。</p> <p>たとえば、以下の2つのテーブルを追加する場合を考えます。</p> <table border="1" data-bbox="451 972 1411 1150"> <thead> <tr> <th>テーブル</th> <th>フィールド</th> </tr> </thead> <tbody> <tr> <td>表 1</td> <td>役職 ラストネーム ファーストネーム ミドルネーム</td> </tr> <tr> <td>表 2</td> <td>ファーストネーム ミドルネーム ラストネーム 生年月日</td> </tr> </tbody> </table> <p>出力テーブルでの順序は次のようになります。</p> <ul style="list-style-type: none"> 役職 ラストネーム ファーストネーム ミドルネーム 生年月日 	テーブル	フィールド	表 1	ラストネーム ファーストネーム ミドルネーム	表 2	ファーストネーム ミドルネーム ラストネーム	テーブル	フィールド	表 1	役職 ラストネーム ファーストネーム ミドルネーム	表 2	ファーストネーム ミドルネーム ラストネーム 生年月日
テーブル	フィールド												
表 1	ラストネーム ファーストネーム ミドルネーム												
表 2	ファーストネーム ミドルネーム ラストネーム												
テーブル	フィールド												
表 1	役職 ラストネーム ファーストネーム ミドルネーム												
表 2	ファーストネーム ミドルネーム ラストネーム 生年月日												
<p>代替列見出し</p>	<p>ソーステーブルの代替列見出しは、出力テーブルにも出力されます。複数のソーステーブルで同じフィールドに代替列見出しがある場合は、最初に選択したテーブルの代替列見出しが優先的に出力されます。</p>												

テーブルを追加する

2つ以上の Analytics テーブルを追加することで、ソーステーブル内のすべてのデータ、またはソーステーブル内の共通フィールドのみを含んだ、新しいテーブルを作成できます。

最後に追加 ダイアログボックスでテーブルを選択した順序に基づいて、それらのテーブル内のレコードが新しいテーブルにおいて縦方向に追加されます。

手順

1. Analytics のメインメニューから、**データ > 最後に追加** を選択します。
2. **最後に追加** ダイアログボックスの **利用可能なテーブル** リストで、新しいテーブルへのそれらのテーブルの追加順序と同じ順序で、それらのテーブルをダブルクリックします。
それらのテーブルが **選択されたテーブル** 領域に追加されます。テーブル名の前にある番号は、**選択されたテーブル** 領域におけるそのテーブルの順序を示します。この順序は、各テーブルが出カテーブルに追加された順序でもあります。
3. (オプション) 選択したテーブルの順序を並べ替えたり、各テーブルが出カテーブルに最後に追加された順序を変更したりするには、任意のテーブルをドラッグします。

メモ

テーブルは、そのヘッダーを使ってドラッグして、別のテーブルにドロップします。

4. (オプション) **テーブルフィールドの非表示**  をクリックすると、テーブルフィールド一覧が折りたたまれます。または、**選択したテーブルの削除**  をクリックすると、**選択されたテーブル** 領域からテーブルが削除されます。
5. (オプション) 選択したすべてのテーブルに存在するフィールドのみを出カテーブルに追加したい場合は、**共通フィールドのみ** をオンにします。

このオプションをオンにする場合、結合対象のテーブルの間で少なくとも1つのフィールドが共通である必要があります。

メモ

フィールドが"共通"と見なされるためには、同じ物理名を持ち、同じデータカテゴリに属しているか、同じデータカテゴリに属するように一致している必要があります。

演算フィールドを追加できません。詳細については、「演算フィールドがサポートされていない」ページ 935を参照してください。

6. (オプション) 出カテーブルに **ソーステーブル** フィールドを含めたい場合は、**テーブル名の追加** をオンにします。

出力テーブルの各レコードの [ソーステーブル] フィールドには、そのレコードのソーステーブルが表示されます。

ヒント

追加するソーステーブルの名前を含めておくと、出力テーブルのデータを分析する際に役立つ可能性のある情報が得られます。

7. (オプション) 同じ名前のフィールドのデータカテゴリを一致させる必要がある場合は、以下のいずれかを選択します:

- **文字データ型を使用した共通フィールドの調整** -により、文字型でないフィールドが文字データカテゴリに変換されるのは、そうすることが調整上必要な場合に限りです。

たとえば、数値データ型を使用しているあるテーブル内の請求日フィールドと、日付データ型を使用している別のテーブル内の請求日フィールドが、どちらも文字データカテゴリに変換されます。

- **すべてのフィールドを文字型データに変換すると、一致のために必要とされるか否かにかかわらず、文字データカテゴリにすべてのテーブルが最後に追加されます**

8. (オプション) 出力テーブルを自動的に開くようにするには、**出力テーブルの使用**] をオンにします。

9. **保存先**] テキストボックスに、新しいテーブルの名前を指定します。

絶対ファイルパスや相対ファイルパスを指定したり、**参照**] をクリックして別のフォルダーを指定したり、あるいはプロジェクトの場所以外の場所に新しいテーブルを保存したりすることもできます。

例: `C:\Data\Annual_table.fil` または `Data\Annual_table.fil`。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

10. **OK**] をクリックして、新しいテーブルへの追加を確定します。

フィールドの調整に関する通知が表示される場合には、フィールドを変換して調整したくない理由があるときを除いて、**はい**] をクリックします。詳細については、「自動調整」ページ 933 を参照してください。

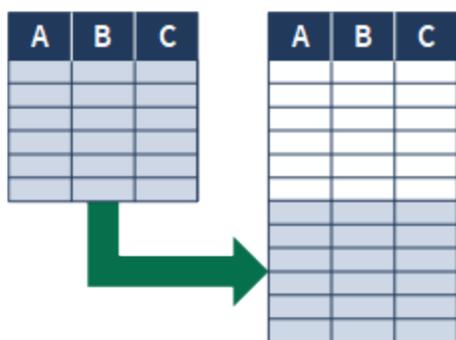
上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

データの抽出と追加

データの抽出と追加では、1つの Analytics テーブルからレコードまたはフィールドを抽出して、別の Analytics テーブルの最後にグループとして追加することができます。抽出はコピーと同じです。末尾に追加は追加と同じです。2つのテーブルを並べ替えるか並べ替えを解除します。

(対象テーブル)に追加するテーブルのサイズは大きくなります。新しいテーブルは作成されません。

抽出/最後に追加の操作を複数回繰り返すことによって、たとえば、月次または四半期のテーブルを組み合わせて1つの年次テーブルにするなどの、有用なタスクを実行できます。



例

シナリオ

従業員レコードのセット全体に対して分析を実行したいが、新しい従業員のレコードがまだ従業員マスターテーブルに含まれていない場合を考えます。

アプローチ

新しい従業員のレコードを抽出し、それらを従業員マスターテーブルの最後に追加して、分析を実行します。

ヒント

APPEND コマンドを1回実行しただけで、APPEND オプションを指定した EXTRACT コマンドの複数回の実行結果が置き換えられます。

詳細については、「テーブルの追加」ページ 928を参照してください。

抽出と追加のベスト プラクティス

データを抽出および追加するときには、元のデータファイルには決してレコードを追加しないことが、最良の方法です。

最初に、元のテーブルから新しいテーブルへレコードを抽出することにより、新しい出力先テーブルを作成する必要があります。その後、1 つまたは複数のソーステーブルからレコードを抽出し、それらを新しいテーブルに追加します。

この方法をとれば、抽出/最後に追加の処理で何らかの問題が発生した場合に元のデータファイルが保持されます。

データを抽出および追加するときのさまざまなオプション

データの抽出および追加時には、3 つの異なるオプションを選択できます。

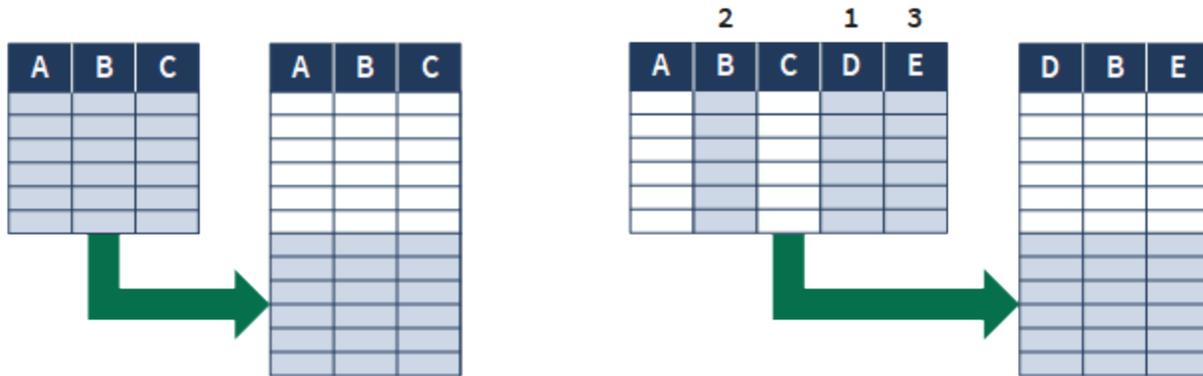
オプション	説明
レコード単位で抽出	レコード全体を抽出します。 <ul style="list-style-type: none"> フィールドは、テーブルレイアウトに現れる順序と同じ並びで抽出されます。 ソーステーブルとターゲット テーブルのデータ構造は完全に同じである必要があります。
ビュー単位で抽出	現在のビューにあるすべてのフィールドを抽出します。 <ul style="list-style-type: none"> フィールドは、ビューに現れる順序と同じ並びで抽出されます。 ソーステーブルとターゲット テーブルで、対応するフィールドのデータ構造が完全に同一である必要があります。
フィールド単位で抽出	フィールドを個別に選択して抽出します。 <ul style="list-style-type: none"> フィールドは、選択した順序で抽出されます。 ソーステーブルとターゲット テーブルで、対応するフィールドのデータ構造が完全に同一である必要があります。

オプション間の主な違い

次の図では、レコード単位、ビュー単位、フィールド単位の抽出および追加の主な違いを示しています。

レコード単位で抽出および追加	図の左側の2つのテーブルとフィールド A、B、Cに示すように、フィールドの数と順序はソースとターゲット テーブルで同じである必要があります。
ビュー単位、フィールド単位の抽出および追加	図の右側の2つのテーブルに示すように、 ソースとターゲット テーブルのフィールドの数と順序同じである必要がありません。 このような状況では、ターゲット テーブルのフィールド数と順序に一致するように、ソーステーブ

ルでビューを調整することか、抽出するときに適切なフィールドを選択することができます。
次の例では、D、B、Eの順序でビューにフィールドを配置するか、フィールドを選択します。
フィールド A および C を省略します。



抽出および追加するときの並べ替えの仕組み

抽出して追加するときには、ソースおよびターゲット テーブルに存在するすべての並べ替え順は、結果として生じる結合されたテーブル内のそれぞれのレコード セットで別々に保持されます。

たとえ両方のテーブルのレコードが並べ替えられていても、結果として生じる結合されたテーブルは、並べ替えられていないテーブルと見なされます。これは、抽出されたレコードを出力先テーブルの末尾にグループとして追加するときに、出力先テーブルの既存の並べ替え順が何も考慮されないからです。

たとえば、月次または四半期テーブルを抽出して、年次テーブルを作成する場合は、月次または四半期データの内部並べ替えは保持されます。必要であれば、抽出/最後に追加操作を1回または複数回実行した後、結果として生じた結合テーブルを並べ替えることができます。

サーバー テーブルおよびローカル テーブルからの抽出と追加

サーバー テーブルとローカル テーブルの両方からデータを抽出/最後に追加することができます。サーバー テーブルから抽出されたデータは、サーバー上またはローカルコンピュータ上のテーブルに追加できます。ローカル テーブルから抽出されたデータは、ローカルコンピュータ上のテーブルにのみ追加できます。

データを抽出および追加するときの必要条件

データを抽出および追加する場合、その操作を成功させるためには、対象とするデータが、ある条件を満たしている必要があります。データが条件を満たさない場合は、規則性のないデータ、データ不足、あるいは不正確なデータとなる可能性があります。

フィールドレベルのデータ構造の違いによって、データの正常な抽出/追加が妨げられている場合は、フィールドを一致させることができるかもしれません。詳細については、「フィールドの一致」ページ 921を参照してください。

レコード構造が異なる(つまり、データ要素が同一でない)テーブルについては、結合または関連付けを使用します。

ヒント

場合によっては、Analytics 外でデータを結合する方が簡単あるいは実用的であることもあります。フィールド間で一貫性がないために、Analytics でデータを追加することが困難な場合は、「データの結合の代替方法」ページ 917を参照してください。

次の表は、異なる抽出および追加オプションの要件をまとめています。

要件	レコード単位で抽出および追加	ビュー単位で抽出および追加	フィールド単位で抽出および追加
フィールド(データ要素) ソーステーブルとターゲットテーブルで、フィールド(データ要素)が同一でなければならない。	はい	いいえ ターゲットテーブルのフィールドが、ソーステーブルのフィールドのサブセットとなることもあります。	いいえ ターゲットテーブルのフィールドが、ソーステーブルのフィールドのサブセットとなることもあります。
フィールドの順番 ソーステーブルとターゲットテーブルの対応するフィールドが、テーブルレイアウトにおいて同じ順序で並んでいる必要がある。	はい	いいえ ソーステーブルビューのフィールドは、ターゲットテーブルのテーブルレイアウトのフィールドと同じ順序で並んでいる必要があります。	いいえ ソーステーブルのフィールドは、ターゲットテーブルのテーブルレイアウトのフィールドと同じ順序で選択する必要があります。
フィールドの数 ソーステーブルとターゲットテーブルで、フィールドの数は同じである必要があります。	はい	いいえ ソーステーブルビューのフィールド数は、ターゲットテーブルのテーブルレイアウトのフィールド数と同じである必要があります。	いいえ ソーステーブルで選択するフィールド数は、ターゲットテーブルのテーブルレイアウトのフィールド数と同じである必要があります。
ビューの構造 ソーステーブルビューのデータ構造は、ターゲットテーブルのテーブルレイアウトと同一である必要がある。	いいえ	はい	いいえ
レコード長 ソーステーブルとターゲットテーブルで全レコード長が同一である必要がある。	はい	いいえ	いいえ
フィールド長	はい	はい	はい

要件	レコード単位で抽出および追加	ビュー単位で抽出および追加	フィールド単位で抽出および追加
ソーステーブルとターゲットテーブルで、対応するフィールドの長さが同一である必要がある。			
フィールド名 ソーステーブルとターゲットテーブルで、対応するフィールドの名前が同一である必要がある。	いいえ	いいえ	いいえ
	ターゲット フィールド名 は結果の結合されたテーブルで使用されます。		
開始位置 ソーステーブルとターゲットテーブルで、対応するフィールドの開始位置が同一である必要がある。	はい	いいえ	いいえ
データ型 ソーステーブルとターゲットテーブルで、対応するフィールドのデータ型が同一である必要がある。	はい	はい	はい
日付時刻書式 ソーステーブルとターゲットテーブルで、対応するフィールドの日付書式および日付時刻書式が同一である必要がある。	はい	はい	はい

データを抽出して追加する

1つのAnalytics テーブルからレコードまたはフィールドを抽出し、別のAnalytics テーブルの末尾にそれらをグループとして追加することができます。フィールドのレコードには、2つのテーブルで同じ構造が必要です。2つのテーブルを並べ替えるか並べ替えを解除します。結果として生じる結合されたテーブルは、並べ替えられていないものと見なされます。

手順

メモ

手順の後に詳細情報が表示されます。"抽出]ダイアログボックスのオプション"見開きページを参照してください。

1. ナビゲーターで、レコードまたはフィールドを抽出するテーブルを開きます。
2. **データ > 抽出]**の順に選択します。
3. **メイン]**タブで、次のいずれかを選択します。
 - **[レコード]** -はレコード全体を抽出します。
 - **ビュー** -は現在のビューにあるすべてのフィールドを抽出します。

メモ

ビュー内のフィールドの数、選択、および順序は、出力先テーブルのテーブルレイアウトで定義されるフィールドの数、選択、および順序と厳密に一致している必要があります。

4. **[フィールド]**を選択すると、**抽出フィールド]**リストから適切なフィールドを選択します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

メモ

選択するフィールドの数、選択、および順序は、出力先テーブルのテーブルレイアウトで定義されるフィールドの数、選択、および順序と厳密に一致している必要があります。

5. **保存先]**テキスト ボックスに、出力先テーブルの名前を指定します。

6. **詳細]タブ**

- a. (省略可能)レコードのサブセットのみが処理されることを指定するには、**範囲]パネル**でオプションのいずれかを選択します。
- b. **OK]をクリック**します。

抽出]ダイアログボックスのオプション

次の表は、**抽出]ダイアログボックス**のオプションの詳細を示します。

メイン]タブ

オプション - 抽出ダイアログボックス	説明
	<p>抽出方法を指定します。</p> <ul style="list-style-type: none"> ○ レコード] -はレコード全体を抽出します。レコード内のフィールドは、テーブルレイアウトに現れる順序と同じ並びで抽出されます。 ○ ビュー] -は現在のビューにあるすべてのフィールドを抽出します。フィールドは、ビューに現れる順序と同じ並びで抽出されます。 ○ フィールド] -はフィールドを個別に選択して抽出します。フィールドは、選択した順序で抽出されず。 <p>1つ以上の演算フィールドを抽出するには</p> <ul style="list-style-type: none"> ○ レコード]を選択すると、抽出されたフィールドは演算式として保持されます。 ○ ビュー]または フィールド]を選択すると、抽出されたフィールドは適切なデータ型の物理フィールドに変換され、実際に計算された値が格納されます。 <p>メモ 演算フィールドと物理フィールドをお互いに追加することはできません。 詳細については、「演算フィールドの抽出と追加」ページ 949を参照してください。</p>
レコード ビュー フィールド	<p>テーブルの関係における子テーブルからデータを抽出したい場合：</p> <ul style="list-style-type: none"> ○ その子テーブルのフィールドがビューに追加済みであれば、フィールド]あるいは ビュー]を選択します。 <p>レコード]オプションを使って子テーブルのデータを抽出することはできません。</p>
抽出フィールド	<p>フィールド]を選択した場合は、抽出するフィールドを指定します。</p> <ul style="list-style-type: none"> ○ 抽出フィールド]リストから適切なフィールドを選択します。 ○ 抽出フィールド]をクリックして、適切なフィールドを選択するか、または式を作成して、OK]をクリックします。 <p>テーブルの関係における子テーブルからフィールドを選択したい場合：</p> <ul style="list-style-type: none"> ○ 抽出フィールド]をクリックします。選択済みのフィールド]ダイアログボックスの 対象テーブル]ドロップダウンリストを使用して、適切な子テーブルを選択できます。
もし次の	(省略可能)レコードを処理から除外する条件を作成できます

オプション - 抽出ダイアログボックス	説明
	[F] テキスト ボックスに条件を入力するか、または [F] ボタンをクリックし、 式ビルダー を利用して WHILE ステートメントを作成することができます。
変換先	<p>対象テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> ◦ 保存先] テキスト ボックスに、出力先テーブルの名前を指定します。 ◦ 保存先] ボタンをクリックして、保存] または ファイルを保存する名前] ダイアログボックスで出力先テーブルの名前を指定するか、または既存のテーブルを出力先テーブルとして選択します。 <p>また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にある出力先テーブルにデータを追加したりすることもできます。たとえば、C:\Results\GL_2011.fil または Results\GL_2011.fil と指定します。</p> <p>データの追加先に関係なく、出力先テーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p>
ローカル	<p>サーバーテーブルに接続している場合は、出力テーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ◦ ローカルを選択 -] では、Analytics プロジェクトと同じ場所、または移動先の場所に出カテーブルを保存します。 ◦ AX サーバーの Prefix フォルダーに出カテーブルを保存する場合は、ローカル-] を選択解除されたままにしておきます。
出力テーブルを開く	操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。

詳細] タブ

オプション - 抽出ダイアログボックス	説明
範囲パネル	<p>処理されるソーステーブルのレコードを指定します。</p> <ul style="list-style-type: none"> ◦ すべて - (デフォルト) ソーステーブルのすべてのレコードが処理されます。 ◦ 先頭 -- このオプションを選択してテキスト ボックスに数を入力すると、ソーステーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 ◦ 次 -- このオプションを選択してテキスト ボックスに数を入力すると、ソーステーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。 ◦ While -- WHILE ステートメントを使用して、条件に基づいてソーステーブル内のレコードの処理を制限するには、このオプションを選択します。 <ul style="list-style-type: none"> • [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 • WHILE ステートメントでは、指定した条件が true と評価される間のみ、レコードを処理することができます。

オプション - 抽出ダイアログボックス	説明
	<ul style="list-style-type: none"> • While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。 <p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
EOF(ファイルの終わり処理)	<p>(省略可能) テーブルの終わりに達したときに、もう一度抽出操作が実行されるようにします。</p> <p>EOF パラメーターは通常、大規模な分析プロセスの一環としてレコードを抽出するとき、Extract コマンドがスクリプトのグループ内で発生する場合に使用されます。シーケンシャルレコード間の比較に基づいてレコードを抽出する場合は、テーブルの最後のレコードが確実に抽出されるように、EOF を使用する必要があるかもしれません。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <ul style="list-style-type: none"> ◦ レコードまたはフィールドと出力先テーブルの構造がまったく同じであることがわかっている場合は、既存のファイルに追加する]を選択します。 ◦ 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、既存のファイルに追加する]を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。 <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加する]を選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <p>上書きを確認するメッセージが表示されたら、最後に追加]を選択します。</p> <p>最後に追加]オプションが表示されることを期待していたのに表示されない場合は、[いいえ]をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>

演算フィールドの抽出と追加

演算フィールドと物理フィールドをお互いに追加することはできません。出力元と出力先のテーブルの一方に演算フィールドが含まれている場合は、対応する演算フィールドと物理フィールドの間に不一致が存在するおそれがあります。強制的に追加操作を実行した場合([抽出]ダイアログボックスで **既存のファイルに追加する**]をオンにする)、不一致によって雑然としたデータが生じる結果となります。

レコード単位、ビュー単位、フィールド単位の抽出

データの抽出時に選択するオプションである [レコード]、[ビュー]または [フィールド]によって、対応する演算フィールドと物理フィールドの間に不一致が生じるか、または不一致が調整されます。

- [レコード]を選択すると、抽出された演算フィールドは演算式として保持されます。
- [ビュー]または [フィールド]を選択すると、抽出された演算フィールドは物理フィールドに変換され、実際に計算された値が格納されます。

演算フィールドを伴う抽出/追加を行うときは、適切なオプションを選択し、場合によっては、以下に概説されている予備手順を実行する必要があります。

演算フィールドと物理フィールドを一致させる

次のテーブルは、抽出と追加を行うときのさまざまなフィールドの組み合わせを取りまとめ、対応する演算フィールドと物理フィールドを一致させるために必要な手順を概説しています。

対応する2つの演算フィールドが、異なる式を使用していることもあります。このような状況では、出力先の式が出力元の式よりも優先されます。また、結合されたデータが有効であることを確認するために、いくつかの予備手順を実行することが必要になるかもしれません。

抽出の [レコード] オプションを使用する場合に調整する

	出力先: 演算フィールド	出力先: 物理フィールド
出力元: 演算フィールド	<p>追加しても不一致は生じません。</p> <p>出力元と出力先の演算フィールドが異なる式を使用している場合は、出力先フィールドの式が優先されます。これにより、出力元フィールドの計算値は無効になる可能性があります。</p> <p>両方の式を見直して、相違が正当なものであることを確認してください。そうであれば、抽出の [ビュー] または [フィールド] オプションを使用して、出力元と出力先の両方の演算フィールドを新しいテーブルの物理</p>	<p>追加すると不一致が生じます。</p> <p>改善策: 抽出の [ビュー] または [フィールド] オプションを使用して、出力元テーブルからデータを抽出します。</p>

	出力先:演算フィールド	出力先:物理フィールド
	フィールドに変換してから、新しいテーブルに追加することができます。	
出力元:物理フィールド	追加すると不一致が生じます。 改善策:抽出の [ビュー] または [フィールド] オプションを使用して、既存の出力先テーブルから新しい出力先テーブルを作成します。出力元テーブルから必要なレコードまたはフィールドを抽出し、それらを新しい出力先テーブルに追加します。	追加しても不一致は生じません。

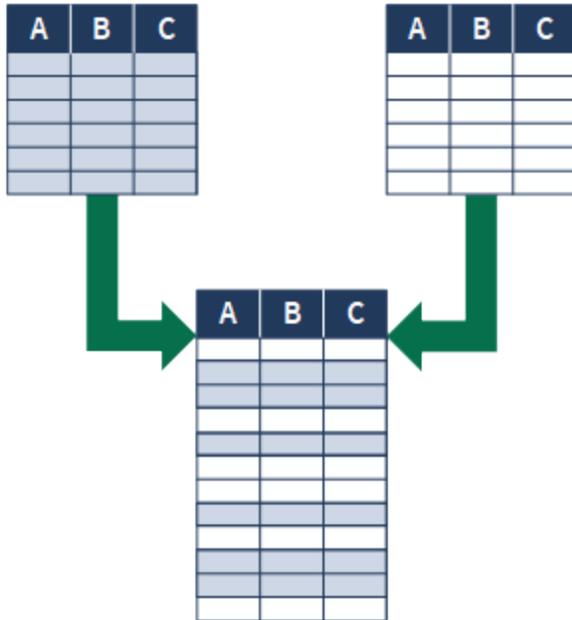
抽出の [ビュー] または [フィールド] オプションを使用する場合に調整する

	出力先:演算フィールド	出力先:物理フィールド
出力元:演算フィールド	追加すると不一致が生じます。 改善策:抽出の [レコード] オプションを使用して、出力元テーブルからデータを抽出します。	追加しても不一致は生じません。
出力元:物理フィールド	追加すると不一致が生じます。 改善策:抽出の [ビュー] または [フィールド] オプションを使用して、既存の出力先テーブルから新しい出力先テーブルを作成します。出力元テーブルから必要なレコードまたはフィールドを抽出し、それらを新しい出力先テーブルに追加します。	追加しても不一致は生じません。

テーブルのマージ

テーブルのマージを使用すると、レコード構造が同一である2つの並べ替え済みの Analytics テーブルを結合して、元のテーブルと同じ並び順になっている第3の新しいテーブルに出力することができます。マージはレコードの挿入によって動作します。既存の並び順に従って、レコードを結合することを意味します。

並べ替えられた従業員テーブルの並び順を保持する統合テーブルと組み合わせるなど、有用なタスクを実行するために、マージ処理を使用できます。



例

シナリオ

従業員レコードのセット全体に対して分析を実行したいが、レコードが2つの課員テーブルに分散している場合を考えます。

どちらのテーブルもラストネームで並べ替えられているため、レコードを結合後に再度並べ替えるオーバーヘッドが生じないようにしたいとします。

アプローチ

2つのテーブルにあるレコードを、新しい第3のテーブルに結合します。マージを行うことで、ラストネームでの並べ替えが保持されます。

テーブルには同じレコード構造が必要です

2つのテーブルを正常にマージするには、両方のテーブルのレコードは、構造がまったく同じであることが必要です。

- データ要素、およびフィールドの数と順序は同一である必要があります。
- 対応するフィールドのデータ型は同じである必要があります。
- 対応するフィールドの開始位置と長さは、同一である必要があります。
- 日付時刻フィールドについては、日付時刻形式が同じである必要があります。

メモ

[マージ]ダイアログボックスには、文字フィールドと文字の演算フィールドのみが表示されます。表示されないフィールドも、2つのテーブル間で同一のデータ構造である必要があります。

レコード構造の比較

2つのテーブルをマージする前に、テーブルの対応するフィールドを比較し、同じ構造であることを確認できます。詳細については、「データ構造の比較」ページ 209を参照してください。

フィールドレベルのデータ構造の違いによって、テーブルの正常なマージが妨げられている場合は、対応するフィールドを一致させることができる可能性があります。詳細については、「フィールドの一致」ページ 921を参照してください。

ヒント

場合によっては、Analytics 外でデータを結合する方が簡単あるいは実用的であることもあります。フィールド間で一貫性がないために、Analytics でデータをマージすることが困難な場合は、「データの結合の代替方法」ページ 917を参照してください。

共通のキーフィールドを使用したテーブルのマージ

共通のキーフィールド(従業員番号やベンダーID、姓など両方のテーブルに存在するデータ要素)を使用してテーブルをマージします。2つの元のテーブルのレコードは、元のテーブルで使用される並べ順での配置に基づき、マージされたテーブルに配置されます。

マージするテーブルのキーフィールドには複数の要件が適用されます。

キーフィールドの特性	要件
データ要素	同じである必要があります。たとえば、両方のキーフィールドが姓フィールドです。
並べ替え順序	同じで、昇順である必要があります。

キーフィールドの特性	要件
	<p>メモ</p> <p>マージ操作時に主キーフィールドを並べ替えるには、主テーブルをあらかじめ並べ替える]オプションを使用することができます。副キーフィールドが並べ替えられていない場合は、マージを実行する前に、まず別個の並べ替え操作でフィールドの並べ替えを行ってください。</p>
データ型	文字である必要があります。
フィールド型	物理フィールドまたは演算フィールドにできます。
フィールド名	異なることができます。
開始位置	同じである必要があります。
フィールド長	同じである必要があります。

主テーブル、副テーブル、キーフィールド

マージ操作の対象となるテーブルとキーフィールドは、テーブルを開く順序に基づき、主および副として識別されます。

- 主テーブル-最初に開いたテーブル
- 主キーフィールド - 主テーブルから選択したキーフィールド
- 副テーブル-2番目に開いたテーブル
 - 副テーブルを開くことは、それを主テーブルに関連付け、処理のために利用できるようにすることを意味します。副テーブルはビュータブで開かれません。
- 副キーフィールド - 副テーブルから選択したキーフィールド

主および副となるテーブルやキーフィールドは自由に選択できますただし、このマージが成功するのは、その対象テーブルとキーフィールドがマージの必要条件を満たしている場合のみです。

詳細については、「キーフィールドについて」ページ 231を参照してください。

複数のキーフィールドのテーブルのマージ

複数の主キーフィールドおよび副キーフィールド(複数の共通キー)を使用して2つのテーブルをマージする場合は、さらに以下の要件が適用されます。

- キーフィールドはすべて昇順に並んでいる必要があります。これは各テーブルで、入れ子になっている並べ替えパターンがあることを意味します。
- 各テーブルでキーフィールドを選択する順序は、各テーブルにおける入れ子の並べ替えパターンの順序と同じでなければなりません。
- 両方のテーブルで、入れ子の並べ替えパターンが同じである必要があります。

マージに関する追加情報

以下のテーブルではマージに関する追加情報を提供します。

機能分野	詳細
出力テーブルのサイズ	結果として生じる結合テーブル内のレコード数は、マージされる2つのテーブル内のレコード数の合計になります。
レコードとフィールド	マージできるのはレコード全体のみです。
キーフィールドのデータ型	文字フィールドまたは文字の演算フィールドのみ、キーフィールドに指定できます。 ヒント Analytics 関数を使用して、数値または日付時刻フィールドを文字フィールドに変換できます。詳細については、「フィールドの一致」ページ 921を参照してください。
同じキーフィールド値	主テーブルと副テーブルのレコードのキーフィールド値が同一である場合は、主テーブルのレコードが副テーブルのレコードより先に並べ替えられます。
対応するフィールドの名前	主テーブルと副テーブルの対応するフィールドは、同じ名前である必要はありません。結果として生じる結合テーブルでは、主テーブルのフィールド名が優先されます。
対応する演算フィールド	対応する演算フィールドがある場合、主テーブルの演算フィールドの式は副テーブルの式よりも優先されます。
パフォーマンスのヒント	サイズが異なる2つのテーブルをマージするときに、大きい方のテーブルを主テーブルとして使用すると、処理が少なくて済みます。
並べ替えの代わりのインデックス	主キーフィールドと副キーフィールドは、並べ替える代わりに昇順でインデックスを付けることができます。インデックスは並べ替えよりもパフォーマンス上の利点を提供します。 副テーブルへのインデックスの適用は、コマンドラインまたはスクリプト内のみで行うことができます。
範囲パラメーター	処理されるレコードを絞り込むための If、While、First、Next パラメーターは、主テーブルにのみ適用されます。
テーブルの位置	マージするためには、その対象となるテーブルは同じ Analytics プロジェクト内に存在している必要があります。サーバーにあるテーブルであれば同じサーバー上にあり、同じサーバープロファイルを使用してアクセスする必要があります。ローカルのテーブルとサーバーのテーブルをマージすることはできません。

テーブルのマージ

各テーブルの共通キーフィールドを使用して、マージを使用すると、レコード構造が同一である2つの並べ替え済みのAnalytics テーブルを結合して、元のテーブルと同じ並び順になっている第3の新しいテーブルに出力することができます。

メモ

テーブルを正常にマージするには、両方のテーブルのデータは、構造がまったく同じである必要があります。詳細については、「テーブルのマージ」ページ 951を参照してください。

手順

メモ

手順の後に詳細情報が表示されます。「マージ ダイアログ ボックスのオプション」下を参照してください。

1. ナビゲーターで、主テーブルを開き、(ナビゲーターで)副テーブルを右クリックして **副テーブルとして開く** を選択します。

主および副テーブルアイコンが数字 1 ~ 2 で更新され、相互への関連を示します  

2. **データ > マージ** を選択します。
3. **メイン** タブ
 - a. **主キー** リストから主キーフィールドを選択します。
 - b. **副キー** リストから副キーフィールドを選択します。
4. **保存先** テキスト ボックスに、新しいマージされたACL テーブルの名前を指定します。
5. **詳細** タブ
 - a. (省略可能)レコードのサブセットのみが処理されることを指定するには、**範囲** パネルでオプションのいずれかを選択します。
 - b. **OK** をクリックします。

マージ ダイアログ ボックスのオプション

次の表は、**マージ** ダイアログ ボックスのオプションの詳細を示します。

[メイン] タブ

オプション - マージ ダイアログ ボックス	説明
副テーブル	副テーブルを選択する代替方法。
主キー 副キー	<p>2つのテーブルをマージするために使用する共通キーフィールドを指定します。</p> <ul style="list-style-type: none"> 主キーおよび副キーリストで直接共通キーフィールドを選択できます。 [主キー] または [副キー] をクリックすると、選択したフィールド ダイアログボックスを開き、共通キーフィールドを選択するか、主キーで式を作成できます。 <p>キーフィールドのガイドライン:</p> <ul style="list-style-type: none"> データ型 - 両方のキーフィールドは文字フィールドである必要があります。 データ構造 - 次の要素は両方のキーフィールドで正確に同じである必要があります。 <ul style="list-style-type: none"> 開始位置 フィールド長 並べ替え - 主テーブルのキーフィールドは、昇順で並べ替えられている必要があります。 複数のキーフィールド - 必要に応じて、1つのテーブルに2つ以上の共通キーフィールドを含めることができます。詳細については、「複数のキーフィールドのテーブルのマージ」ページ 953を参照してください。
主テーブルをあらかじめ並べ替える	<p>キーフィールドまたはフィールドで主テーブルを並べ替えます。</p> <ul style="list-style-type: none"> キーフィールドまたはフィールドが既に適切に並べ替えられているか、インデックスが付いている場合は、事前並べ替え をオフにできます。 事前並べ替えを行うと、テーブルのマージに時間がかかるため、必要な場合にのみこの機能を使用してください。 副キーフィールドには「あらかじめ並べ替える」オプションがないので、事前に昇順で並べ替えておくかインデックスを付ける必要があります。
ローカル	<p>サーバーテーブルに接続している場合は、マージされたテーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ローカルを選択 -] では、Analytics プロジェクトと同じ場所、または移動先の場所に出カテーブルを保存します。 AX サーバーの Prefix フォルダーに出カテーブルを保存する場合は、ローカル -] を選択解除されたままにしておきます。
出カテーブルを開く	操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。
もし次の	<p>(省略可能) レコードを処理から除外する条件を作成できます</p> <ul style="list-style-type: none"> [F] テキスト ボックスに条件を入力するか、または [F] ボタンをクリックし、式ビルダー を利用して WHILE ステートメントを作成することができます。 条件が主テーブルにのみ適用されます。

オプション - マージダイアログボックス	説明
変換先	<p>出力テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> ○ 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 ○ プロジェクト フォルダー以外の場所に出力テーブルを保存するには、-絶対ファイルパスや相対ファイルパスを指定したり、保存先]をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

詳細]タブ

オプション - マージダイアログボックス	説明
範囲パネル	<p>処理される主テーブルのレコードを指定します。</p> <ul style="list-style-type: none"> ○ すべて - (デフォルト) 主テーブルのすべてのレコードが処理されます。 ○ 先頭 -- このオプションを選択してテキスト ボックスに数を入力すると、主テーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 ○ 次 -- このオプションを選択してテキスト ボックスに数を入力すると、主テーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 <p>行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p> <ul style="list-style-type: none"> ○ While -- WHILE ステートメントを使用して、条件に基づいて主テーブル内のレコードの処理を制限するには、このオプションを選択します。 <ul style="list-style-type: none"> ● [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダー を利用して WHILE ステートメントを作成することができます。 ● WHILE ステートメントでは、指定した条件が true と評価される間のみ、レコードを処理することができます。 ● While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。

オプション - マージダイアログボックス	説明
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加するを選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <p>上書きを確認するメッセージが表示されたら、適切なオプションを選択します。</p> <p>最後に追加オプションが表示されることを期待していたのに表示されない場合は、いいえをクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>

結合または関連付けの一般的な用途

結合や関連付けの一般的な用途は、取引テーブル内のレコードをマスターテーブル内のレコードに照合することです。

例

- 従業員のT&E(クレジットカード)請求額を従業員のT&Eの限度額と比較して、払い戻し限度額を超えているすべての従業員を識別する。
- 取引の承認を職務の分担リストと比較して、内部統制を回避しているすべての従業員を識別する。
- 顧客の勘定残高をクレジット限度額と比較して、クレジットの限度額を超えているすべての顧客を識別する。

結合や関連付けのもう1つの一般的な用途は、2つのファイルの内容を比較することです。

例

- 従業員レコードを業者リストと比較して、従業員であると同時に業者としても登録されている従業員をチェックする。
- 医療請求レコードを保険金請求と比較して、その請求金額が正しいことを確認する。

主または親テーブルにするテーブル

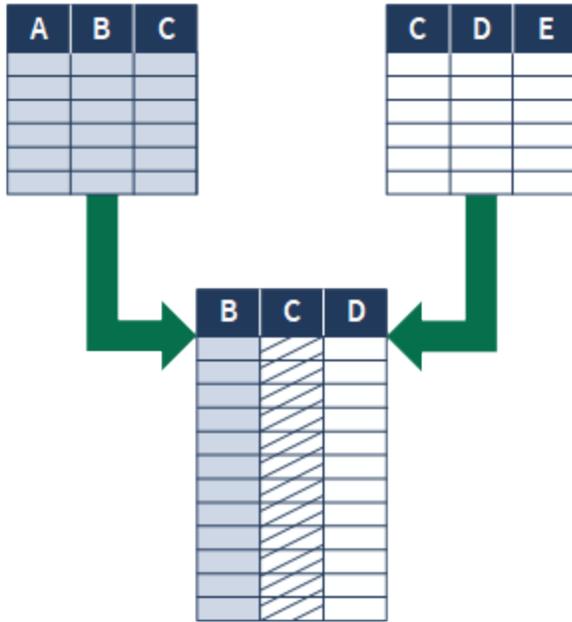
Analyticsにおける結合のほとんどは多対一結合で、関連付けは常に多対一関係であるため、通常、取引テーブルが主テーブルまたは親テーブル、そしてマスターテーブルが副テーブルまたは子テーブルになります。複数の取引(多)、たとえばいくつかのT&E請求額などを、単一のマスターレコード(一)、たとえばT&E限度額などに結合する、あるいは関連付けることができます。

主/親テーブルと副/子テーブルの順序を逆にすると異なる結果が生じるため、主/親と副/子の特定は注意して行ってください。原則として、テーブル内のレコードをすべて分析したい場合は、そのテーブルを主/親テーブルにすべきです。

テーブルの結合

テーブルを結合すると、レコード構造が異なる2つの Analytics テーブルを結合して、第3の新しいテーブルとして出力できます。新しいテーブルに含める2つの元のテーブルからフィールドの組み合わせを選択できます。

異なる複数のフィールド(データ要素)があるとレコード構造は異なります。結合は、分析を始めるにあたり、永続的に結合されたデータのセットを必要とする調査作業に適しています。



例

シナリオ

考えられる不適切な報酬のデータを分析する方法の1つとして、従業員でもある業者を特定する必要があります。

アプローチ

業者マスターテーブルを従業員テーブルと結合し、共通キーフィールドの住所を使用します。結合された出力テーブルには、同じ住所の業者と従業員が含まれます。

メモ

このトピックでは、データインポート処理の一部として、データアクセスウィンドウでテーブルを結合する方法の詳細については、「データアクセスウィンドウのテーブルの結合」ページ 388を参照してください。

このトピックでは、Analytics で Analytics テーブルを結合することについて説明します。

共通のキーフィールドを使用したテーブルの結合

共通のキーフィールド(従業員番号やベンダーID、または住所など両方のテーブルに存在するデータ要素)を使用してテーブルを結合します。2つのキーフィールドに同一の値が存在する場合は1つの一致のみが生じます。この一致は別のテーブルのレコードを個別に結合します。

以下の例では、ベンダー マスター テーブルと従業員 マスター テーブルが、共通キーとして各テーブルの住所フィールド(**Vendor_Street**、および **Emp_Address**)を使用して結合されます。出力テーブルには2つの結合されたレコードがあります。例では、各テーブルの結合されていないレコードは、出力テーブルにも含まれます。これは、選択できるオプションです。

Vendor_Num	Vendor_Name	Vendor_Street	Emp_Address	Emp_Name	Emp_Num
13373	Lilydale Hardware	111 South Main Street			
11663	More Power Industries	150 North Michigan Ave.	150 North Michigan Ave.	Catherine Exelby	10
13411	United Equipment	250 Williams Street			
11182	Industrial Equipment Co-Op	400 High Street S.E.	400 High Street S.E.	John Mullen	50
11247	Meridian Industries	444 Derby Lane			
			800 North Lindbergh Blvd.	Savi Madan	60

類似またはほぼ同一のキーフィールド値を使用したテーブルの結合

Analytics 曖昧結合は、キーフィールドと値の曖昧一致を使用して、2つの Analytics テーブルを新しい3番目のテーブルに結合します。ほとんどの点において、曖昧結合は標準の Analytics 結合に似ています。主な違いは、曖昧結合は、キーフィールド値の完全一致に基づくレコードの結合のほか、近似一致に基づくレコードの結合ができることです。

詳細については、「曖昧結合」ページ 985を参照してください。

キーフィールドの要件

結合するテーブルのキーフィールドには複数の要件が適用されます。

詳しく表示

キーフィールドの特性	要件
データ要素	同じである必要があります。たとえば、両方のキーフィールドは従業員番号フィールドです。
データ型	データ型でも、またキーフィールドでもかまいませんが、関連付けするテーブル間で統一されている(データ型が同じになっている)必要があります。たとえば、2つの文字フィールドです。 例外は文字数値および数値文字のデータ型結合であり、Analyticsが自動的に一致させます。詳細については、「テーブル結合時の自動一致」ページ 997を参照してください。 日付時刻のサブタイプ(日付、日付時刻および時刻)の結合対象は、同じサブタイプのみです。
フィールド型	物理フィールドまたは演算フィールドにできます。
フィールド名	異なることができます。
開始位置	異なることができます。
フィールド長	<ul style="list-style-type: none"> 文字キーフィールド - は同じである必要があります。 文字キーフィールドの長さも自動的に揃えます。詳細については、「テーブル結合時の自動一致」ページ 997を参照してください。 数値キーフィールド - は同じにしておくことをお勧めします。 日付時刻キーフィールド - は異なることができます。
文字フィールドの両端揃えおよび大文字と小文字	同じである必要があります。

主テーブル、副テーブル、キーフィールド

結合操作の対象となるテーブルとキーフィールドは、テーブルを開く順序に基づき、主および副として識別されます。

- 主テーブル - 最初に開いたテーブル
- 主キーフィールド - 主テーブルから選択したキーフィールド
- 副テーブル - 開く2番目のテーブル

副テーブルを開くことは、それを主テーブルに関連付け、処理のために利用できるようにすることを意味します。副テーブルはビュータブで開かれません。

- 副キーフィールド - 副テーブルから選択したキーフィールド

主および副となるテーブルやキーフィールドは自由に選択できますが、この結合が成功するのは、その対象キーフィールドが「キーフィールドの要件」上を満たしている場合のみです。

詳細については、「キーフィールドについて」ページ 231を参照してください。

一致するレコードと不一致のレコード

結合を操作するときには、一致するレコードと一致しないレコードの両方を検討する必要があります。

- 一致したレコード - した主レコードおよび副レコードの値が同一である場合は、その主レコードと副レコードが一致します。

メモ

選択する結合タイプによっては、一致する副キー値の重複する発生は結合されない可能性があります。詳細については以下を参照 "なぜ結合された出力テーブルには一部の副テーブルレコードがないのですか。" 見開きページ

- 一致しないレコード - 主レコードおよび副レコードの値が同一である場合は、その主レコードと副レコードが一致しません。

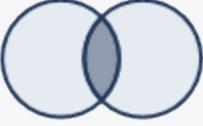
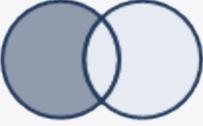
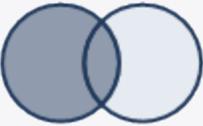
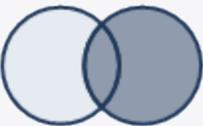
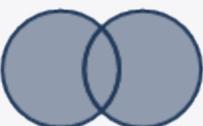
結合後のテーブルに含まれるレコード

一致および不一致のキーフィールド値、および実行する結合のタイプによって、元の2つのテーブルから新しい結合されたテーブルに抽出されるレコードが決まります。

結合のタイプ

Analytics では6種類の結合タイプをサポートします。概要は以下のとおりです。具体的な例については、"結合タイプの例" ページ 976を参照してください。

結合タイプ	結合後のテーブルに含まれるレコード			
	主テーブルの一致レコードを含める	不一致の主レコード	副テーブルの一致レコードを含める	副にテーブルの不一致レコードを含める
一致した主レコードおよび副レコード (一致する1件目の副レコード) 	✓		✓ 含まれない: 副テーブルで一致する副キーの重複も含めます	
一致した主レコードおよび副レコード (重複する副レコードのすべての一致)	✓		✓ 含まれて結合される: 副テーブルで一致する副キーの重複も含めます	

結合タイプ	結合後のテーブルに含まれるレコード			
	主テーブルの一致レコードを含める	不一致の主レコード	副テーブルの一致レコードを含める	副にテーブルの不一致レコードを含める
				
不一致の主レコード 		✓		
すべての主レコードと、キーに一致する副レコード 	✓	✓	✓ 含まれない: 副テーブルで一致する副キーの重複も含めます	
すべての副レコードと、キーに一致する主レコード 	✓		✓ 含まれて結合されない: 副テーブルで一致する副キーの重複も含めます	✓ 含まれる: 副テーブルで一致しない副キーの重複も含めます
すべての主レコードおよび副レコード 	✓	✓	✓ 含まれて結合されない: 副テーブルで一致する副キーの重複も含めます	✓ 含まれる: 副テーブルで一致しない副キーの重複も含めます

なぜ結合された出力テーブルには一部の副テーブルレコードがないのですか。

6つのAnalytics結合タイプのうちの5つは、一致する副キー値の重複を結合しません。一致する主キー値の重複は結合されますが、すべて一致する副キー値の1つめに結合されます。これらの結合タイプは、多対一結合として、広く知られています。

すべての一致する副キー値を結合するには、次のいずれかを実行します。

- **テーブルを逆にする** - 結合で、主および副テーブルを逆にします。元の主テーブルのキーフィールドの値が一意の場合に、この方法が適しています。両方のテーブルで主キー値の重複がある場合、この方法では、必要な結果が得られない可能性があります。
- **多対多結合を使用** - 一致する主および副(すべての副一致)結合タイプを使用します。

多対一結合と多対多結合

多対一結合と呼ばれる Analytics 結合、および多対多結合と呼ばれる 6 つの結合タイプの 1 つを確認できます。これらの用語は、Analytics 結合の動作を広く説明するための役立ちます。ただし、これらの用語は一般化されているため、結合動作を完全に表すわけではありません。

Analytics 多対一結合

1つの式では、Analytics で使用できる結合タイプは多対一結合です。主キーフィールドのすべての値が一意の場合、一対多結合としても機能します。

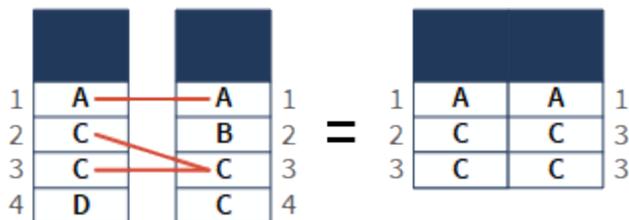
詳しく表示

Analytics 多対一結合

- **結合** - 一致する主キー値の重複する発生はすべて、一致する副キー値の最初の発生に結合されます
重複するキー一致と、最初の副キー一致は結合されたテーブルに含まれます。
- **結合しない** - 一致する副キー値の重複するレコードは結合されません
重複した副キー一致は結合テーブルから除外されます。ただし、すべての副レコードを含む結合タイプを選択した場合を除きます。すべての副レコードを含める場合、重複する副キー一致が、結合されていないレコードとして、結合された出力テーブルに表示されます。

多対一結合

以下の例では、主キー値 'C' の両方の出現が出力テーブルで結合されますが、副キー値 'C' の最初の出現のみが結合されます。



ヒント

キーフィールド値が結合するテーブルのいずれかで一意の場合は、そのテーブルが副テーブルになります。たとえば、トランザクションテーブルをマスターテーブルに結合している場合、マスターテーブルが副テーブルになります。

この方法で結合を構造化すると、確実にすべての一致するレコードが結合され、出力テーブルに含まれます。

Analytics 多対多結合

1つの Analytics 結合タイプ - 一致する主および副 (すべての副一致) - すべての一致する主および副レコードが含まれます。この結合タイプは、多対多結合とも呼ばれます。

主キーフィールドのすべての値が一意の場合、多対多結合は一对多結合としても機能します。

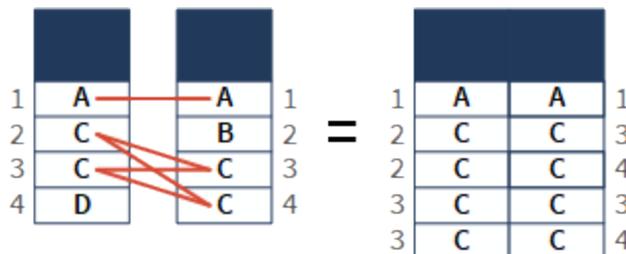
詳しく表示

Analytics 多対多結合

- **結合** - 一致する主キー値のすべての発生は、一致する副キー値のすべての発生に結合されます。重複するキー一致と、重複する副キー一致はすべて、結合され、出力テーブルに含まれます。
- **結合しない** - 一致する副キー値の重複する発生は結合されません

多対多結合

以下の例では、主キー値 'C' の両方の出現は出力テーブルで結合され、副キー値 'C' の両方の出現も結合されます。

**ヒント**

重複する一致が副キーに存在するかどうか不確かである場合は、多対多結合タイプを選択します。結合されるレコードを除外しないことが保証されます。

重複する副キー一致を意図的に除外する場合は、多対多結合タイプを選択しないでください。

Analytics 結合とSQL 結合フィルターの違い

結合]ダイアログボックスを使用して実行する Analytics 結合と、**データアクセス**]ウィンドウを使用してデータを ACL にインポートするときには、主な違いがあります。

- **Analytics 結合** - 重複する副キー一致値は結合されません(多対多結合を除く)。
- **SQL 結合** - 重複する副キー一致値は、選択する結合タイプに関係なく、すべて結合されます。

データアクセス]ウィンドウの SQL 結合の詳細については、「データアクセス ウィンドウのテーブルの結合」ページ 388 を参照してください。

メモ

Analytics は ACL 固有の方法で、「多対多結合」という用語を使用しています。SQL 多対多結合と同義ではありません。

結合されるテーブルの並び替え

結合されたテーブルは主キーフィールドを基準に昇順に並び替えられています。これは、結合の実行時に主キーフィールドを並び替えている(**あらかじめ並び替える**)、あるいは主テーブルが既にこの並び順になっていることを前提とします。

主キーフィールドを基準に並び替えられてない場合は、結合後のテーブルは主テーブルの既存のソート順を使用します。

並び替えと結合に関する追加情報

- 主テーブルが、必ずソート済みである必要はありませんが、主テーブルがまったくソートされていない、または並び順が降順の場合は、処理時間が大幅にかかります。
- 部分的にソートされている主テーブルのキーフィールドを使用して結合を実行する場合(たとえば、主テーブルがまず月順にソートされ、次に勘定コード順にソートされているときに、勘定コードをキーとして結合する場合など)、処理時間はそれほどかかりません。
- 結合の実行時、主テーブルと副テーブルには **あらかじめ並び替える**] オプションがあります。
- 主キーフィールドと副キーフィールドは、並び替える代わりにインデックスを付けることができます。副テーブルのキーフィールドは、昇順でインデックスが付けられている必要があります。副テーブルへのインデックスの適用は、コマンドラインまたはスクリプト内のみで行うことができます。

結合に関する追加情報

以下のテーブルでは結合に関する追加情報を提供します。

機能分野	詳細
一致しないレコードお	結合に不一致の主レコードまたは不一致の副レコードが含まれる場合、欠落するフィールド値の対

機能分野	詳細
よびフィールド値の欠落	処として、文字や日付時刻フィールドには空白を、数値フィールドにはゼロを、そして論理フィールドには "F" を表示します。
副テーブルのキーフィールドの重複または空白	副テーブルのキーフィールドの値が重複していたり欠落していたりすることで、その後の分析が無効になってしまうような場合、事前に副テーブルから重複や空白を除去することで解決する可能性があります。
部分一致	<p>キーフィールド値の部分一致はサポートされていません。一致するには、値が100% 同じである必要があります。</p> <p>例：</p> <ul style="list-style-type: none"> ○ 一致 - AB-123, AB-123 ○ 不一致 - AB-123, 123 <p>メモ Analytics では、部分一致がサポートされます。"曖昧結合" ページ 985</p>
キーフィールド長の同一化は不要	<p>Analytics では、テーブルの結合時に、(文字キーフィールド以外で) 主キーフィールドと副キーフィールドの長さが同一であることを強制しません。</p> <p>数値キーフィールドには同一の長さを使用することをお勧めします。必要であれば、結合を行う前にフィールド長を手作業で揃えてください。フィールド長が異なる数値キーフィールドを使用して結合を行った結果は信頼性がありません。</p> <p>日付、日付時刻、または時刻を用いた操作を実行する場合、Analytics は Analytics 内部の日付時刻形式を使用しているため、日付時刻キーフィールドの長さが異なることがあります。</p>
行揃えと大文字小文字の同一化	<p>文字キーフィールドを使用してテーブルの結合を行う場合、両フィールドの行揃えおよび大文字小文字の表記が同一でなければなりません。</p> <ul style="list-style-type: none"> ○ 両方のキーフィールドは同じ行揃えであることが必要です。LTRIM() 関数を使用すると、キーフィールドから先頭のスペースを削除します。 ○ 両方のキーフィールドはそれぞれ、フィールド内の単語(アルファベット)の大文字、小文字または大文字/小文字の混在の表記が同じあることが必要です。大文字小文字の表記を同一にするには、UPPER()、LOWER() または PROPER() 関数を使用します。
結合に含まれないレコードのカウント	<p>実行する結合タイプによって、主テーブルおよび/または副テーブルのレコードが結合後のテーブルに含まれないこともあります。コマンド ログでは、含まれなかった主レコードの数 (<n> 件のレコードがバイパス(除外) されました) が表示されますが、除外された副レコードの数は表示されません。</p>
結合操作で使用される条件式と適用範囲オプション	<p>多対一結合の場合、処理されるレコードを絞り込むための If、While、First、Next パラメーターは、主テーブルにのみ適用されます。多対多結合の場合は、If 式と While 式によって副テーブルを参照することもできます。</p>
結合されるテーブルでの同一フィールド名	<p>主キーフィールドの名前と副キーフィールドの名前が同じ、またその他の出力対象フィールドの名前がテーブル間で同一の場合は、出力テーブルのレイアウトで副テーブルの対象フィールド名の末尾に2が付けられます。たとえば、元のフィールド名が vendor_ID であれば vendor_ID2(以降、vendor_ID3 ...など) となり、出力テーブルでフィールド名が重ならないようにします。</p> <p>ただし、出力テーブルのビューの代替列見出し列には、変更されていない同じ名前が表示されます。</p>
副テーブルとして使	<p>現在、子テーブルとして主/親テーブルに関連付けられているテーブルは、結合で副テーブルとして選</p>

分析用のデータの準備

機能分野	詳細
用できないテーブル	<p>択することはできません。この制限を回避するには、その主/親テーブルのレイアウト、または子テーブルのレイアウトのコピーを作成し、そのコピーされたレイアウトを使って結合します。また、関連付けの関係を削除するという選択肢もあります。</p>
結合されるテーブルの場所に関する制限事項	<p>テーブルを結合する場合、その対象となるテーブルは同じ Analytics プロジェクト内に存在している必要があります。サーバーにあるテーブルであれば同じサーバー上にあり、同じサーバー プロファイルを使用してアクセスする必要があります。ローカルのテーブルとサーバーのテーブルを結合することはできません。</p>
結合されるテーブルのサイズ	<p>実行する結合タイプによって、結合後のテーブルに含まれるレコードの数は、結合前の元の2つのテーブルに含まれるレコードの総数に比べ、多くなったり、少なくなったり、あるいは等しくなることもあります。</p>
UTC ベースのデータと非 UTC データの結合	<p>UTC ベースおよび非 UTC の日付時刻キー フィールドは、2つのテーブルの結合に使用できます (UTC は協定世界時で、経度 0 度地点における時刻)。日付時刻または時刻を用いた操作を実行する場合、Analytics では Analytics 自身の内部日付時刻形式を使用するので、以下の2つの日付時刻は同一であると解釈され、一致します。</p> <ul style="list-style-type: none"> ◦ UTC ベース - 31/12/2014 10:30:15-05:00 ◦ 非 UTC - 31/12/2014 15:30:15 <p>Analytics 操作の中で、UTC ベースの時刻データと非 UTC の時刻データを混在させる場合は注意が必要です。Analytics は上の2つの時刻値を一致させます。しかし、一方の値はタイムゾーンを参照し、もう一方の値は参照しないため、論理的には意味がない可能性があります。UTC の詳細については、"日付と時刻オプション" ページ 135を参照してください。</p>

テーブルの結合

各テーブルの共通キーフィールドを使用して、異なるレコード構造を持つ2つのソート済み Analytics テーブルを結合することができます。第3のテーブルに含まれるフィールドは、元の2つテーブルにあるフィールドを組み合わせたものです。

メモ

主と副の関係を逆にすると異なる結果が生じるため、結合では主テーブルと副テーブルの特定に注意してください。詳細については、「結合または関連付けの一般的な用途」ページ 959を参照してください。

手順

メモ

手順の後に詳細情報が表示されます。「[結合]ダイアログボックスのオプション」次のページを参照してください。

1. ナビゲーターで、主テーブルを開き、(ナビゲーターで)副テーブルを右クリックして **副テーブルとして開く**を選択します。

主および副テーブルアイコンが数字 1 ~ 2 で更新され、相互への関連を示します  

2. **データ > 結合** の順にクリックします。
3. **メイン** タブ
 - a. **結合タイプ**を選択します。

結合タイプについては、以下で説明します。
 - b. **主キー** リストから主キーフィールドを選択します。
 - c. **副キー** リストから副キーフィールドを選択します。
 - d. **主フィールド** および **副フィールド** リストから、結合テーブルに含めたいフィールドを選択します。

メモ

結合テーブルに含める場合は、主キーおよび副キーフィールドを明示的に選択する必要があります。

ヒント

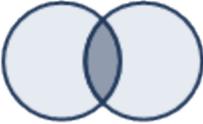
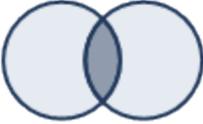
隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

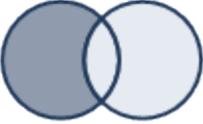
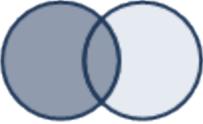
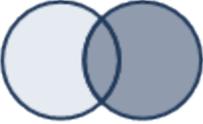
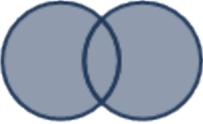
4. **保存先**]テキスト ボックスに、新しい結合された ACL テーブルの名前を指定します。
5. (省略可能)[**詳細**]タブ:
 - a. レコードのサブセットのみが処理されることを指定する場合は、**範囲**]パネルでオプションのいずれかを選択します。
 - b. 出力結果を既存の Analytics テーブルの末尾に追加する場合は、**既存のファイルに追加する**]を選択します。
6. **OK**]をクリックします。
新しい結合されたテーブルが出力です。

結合]ダイアログ ボックスのオプション

次の表は、**結合**]ダイアログ ボックスのオプションの詳細を示します。

[メイン]タブ

オプション- 結合]ダイアログ ボックス	説明
結合タイプ	使用する Analytics 結合タイプを指定します。 詳細については、参照してください。"結合後のテーブルに含まれるレコード" ページ 963
一致した主レコードおよび副レコード (一致する 1 件目の副レコード) 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ◦ すべての一致した主レコードおよび最初に一致した副レコード
一致した主レコードおよび副レコード (重複する副レコードのすべての一致) 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ◦ すべての一致した主レコードおよびすべての一致した副レコード ◦ 主および副テーブル間の各一致に 1 つのレコード

オプション- 結合]ダイアログボックス	説明
不一致の主レコード 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ○ 不一致の主レコード
すべての主レコードと、キーに一致する副レコード 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ○ すべての主レコード(一致と不一致の両方)、最初に一致した副レコード
すべての副レコードと、キーに一致する主レコード 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ○ すべての副レコード(一致と不一致) およびすべての一致する主レコード 重複する副一致の最初のインスタンスのみが主レコードに結合されます。
すべての主レコードおよび副レコード 	結合された出力テーブルの内容: <ul style="list-style-type: none"> ○ すべての主および副レコード(一致と不一致) 重複する副一致の最初のインスタンスのみが主レコードに結合されます。
副テーブル	副テーブルを選択する代替方法。
主キー 副キー	2つのテーブルを結合するために使用する共通キーフィールドを指定します。 <ul style="list-style-type: none"> ○ 主キーおよび副キーリストで直接共通キーフィールドを選択できます。 ○ [主キー]または[副キー]をクリックすると、[選択したフィールド]ダイアログボックスを開き、共通キーフィールドを選択するか、主キーで式を作成できます。 キーフィールドのガイドライン: <ul style="list-style-type: none"> ○ データ型 - キーフィールドはどのようなデータ型でもかまいませんが、関連付けするテーブル間で統一されている(データ型が同じになっている)必要があります。 1つの例外は、文字および数値キーフィールドを相互に結合できることです。 ○ 日付時刻のサブタイプ - 日付時刻のサブタイプ(日付、日付時刻および時刻)の結合対象は、同じサブタイプのみです ○ 長さ

オプション- 結合]ダイアログボックス	説明
	<ul style="list-style-type: none"> • 数値キーフィールドの長さは同じにしておくことをお勧めします。 • 文字キーフィールドの長さが異なる場合、これらのフィールドは自動的に調整されます。 • 日付時刻キーフィールドの長さは同じでなくてもかまいません。 ○ 名前と開始位置 -キーフィールドの名前と開始位置はテーブル間で異なってもかまいませんが、同じデータ要素になっている必要があります。 ○ 複数のキーフィールド -必要に応じて、1つのテーブルに2つ以上の共通キーフィールドを含めることができます。詳細については、"複数のキーフィールドの使用" ページ 1013を参照してください。
<p>主フィールド 副フィールド</p>	<p>結合テーブルに含めるフィールドを指定します。</p> <ul style="list-style-type: none"> ○ 主キーおよび副キーリストで直接フィールドを選択できます。 ○ 主フィールド]または 副フィールド]をクリックすると、選択したフィールド]ダイアログボックスを開き、フィールドを選択するか、1つ以上の主フィールドで式を作成できます。 ○ 主フィールドと副フィールドの選択順序は、結合されるテーブルにおけるフィールドの表示順序に対応します。 <p>グループとして、結合されるテーブルで、主フィールドは副フィールドの前に現れます。</p>
<p>主テーブルをあらかじめ並べ替える 副テーブルをあらかじめ並べ替える</p>	<p>キーフィールドで主テーブルまたは副テーブルを並べ替えます。</p> <ul style="list-style-type: none"> ○ 1つまたは両方のキーフィールドが既に適切に並べ替えられているか、インデックスが付いている場合は、事前並べ替え]をオフにできます。 ○ 事前並べ替えを行うと、テーブルの結合に時間がかかるため、必要な場合にのみこの機能を使用してください。 ○ 副テーブルのキーフィールドは、昇順で並べ替えられているか、インデックスが付けられている必要があります。
<p>ローカル</p>	<p>サーバーテーブルに接続している場合は、結合テーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ○ ローカルを選択 -]では、Analytics プロジェクトと同じ場所、または移動先の場所に出カテーブルを保存します。 ○ AX サーバーの Prefix フォルダーに出カテーブルを保存する場合は、ローカル-]を選択解除されたままにしておきます。
<p>出力テーブルを開く</p>	<p>操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。</p>
<p>もし次の</p>	<p>(省略可能)レコードを処理から除外する条件を作成できます</p> <ul style="list-style-type: none"> ○ [f]テキスト ボックスに条件を入力するか、または [f] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 ○ ほとんどの Analytics 結合タイプで、条件は主テーブルのみを参照できます。 ○ 結合タイプの一致する主および副レコード(すべての副一致)では、条件は主テーブルまたは副テーブル、あるいはその両方を参照できます。

オプション- 結合]ダイアログボックス	説明
	<p>メモ</p> <p>[式ビルダー]で副テーブルフィールドにアクセスするには、テーブルから]ドロップダウンリストで副テーブルを選択します。</p> <p>If 条件は、任意の範囲オプション(先頭、次へ、While)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
変換先	<p>出力テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 プロジェクト フォルダー以外の場所に出力テーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先]をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ</p> <p>Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(<code>_</code>)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

詳細]タブ

オプション- 結合]ダイアログボックス	説明
範囲パネル	<p>処理される主テーブルのレコードを指定します。</p> <ul style="list-style-type: none"> すべて-(デフォルト) 主テーブルのすべてのレコードが処理されます。 先頭-- このオプションを選択してテキスト ボックスに数を入力すると、主テーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 次-- このオプションを選択してテキスト ボックスに数を入力すると、主テーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 <p>行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p> <ul style="list-style-type: none"> While-- WHILE ステートメントを使用して、条件に基づいて主テーブル内のレコードの処理を制限するには、このオプションを選択します。

オプション- 結合]ダイアログボックス	説明
	<ul style="list-style-type: none"> • While]テキスト ボックスに条件を入力するか、または While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 • WHILE ステートメントでは、指定した条件が true と評価される間のみ、レコードを処理することができます。 • While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。 <p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
<p>既存のファイルに追加する</p>	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加する]を選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>
<p>OK</p>	<p>処理を実行します。</p> <ul style="list-style-type: none"> ◦ 文字キーフィールドと数値キーフィールドを対応付けて結合を行う場合、または長さが異なる文字キーフィールドで結合を行う場合は、それらのフィールドを一致させるよう試みることを提示するメッセージが表示されます。 ◦ 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。 <p>最後に追加] オプションが表示されることを期待していたのに表示されない場合は、[いいえ]をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>

結合タイプの例

Analytics で 6 種類の結合タイプを示すためのサンプルデータについて説明します。この例は、異なる結合タイプを使用して、結合された出力テーブルに必要な情報を正確に取り込む方法を示します。

6 つの結合タイプの概要ビューについては、参照してください。"結合後のテーブルに含まれるレコード" ページ 963

キーフィールド値の曖昧一致を使用して、2 つの Analytics テーブルを結合するには、"曖昧結合" ページ 985を参照してください。

サンプルデータ

最初の 5 つの例は、以下に示すサンプルデータを使用します。

主テーブル

	Employee number	Cheque amount
1	001	1850.00
2	002	2200.00
3	003	1500.00
4	003	500.00

<< ファイルの終わり >>

副テーブル

	Employee number	Pay per period
1	001	1850.00
2	003	2000.00
3	004	1975.00
4	005	2450.00

<< ファイルの終わり >>

サンプルデータと例の詳細

給与台帳テーブル

ある単独の給与期間(1か月)におけるすべての給与支払いについての情報が含まれています。従

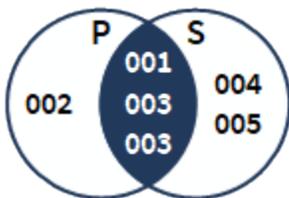
(主)	業員番号 003 の従業員には2度の給与支払いがありました。
従業員レコードテーブル (副)	人事部によって管理されています。従業員レコードテーブルには、有効な従業員の一覧と1か月ごとに支払われる給与金額が含まれています。1つの従業員 002 はテーブルに表示されません。
結合	次の例では、従業員番号を共通キーフィールドとして使用し、給与台帳テーブルと従業員レコードテーブルを結合します。 すべての5つの例は、 多対一結合 です。
目標	各例では、給与の不規則性をテストします。

一致した主レコードおよび副レコード(最初の副一致)

例

テスト -従業員に正しく支払われたことを検証します。

アプローチ - 従業員レコードテーブル(S)との一致がある給与台帳テーブル(P)のすべてのレコードで1つの出力レコードを作成する結合タイプを使用します。



出力テーブル -給与が支払われた従業員の中で、従業員レコードテーブルに存在している従業員がすべて出力されます。

	Cheque amount	Employee number	Employee number	Pay per period
1	1850.00	001	001	1850.00
2	1500.00	003	003	2000.00
3	500.00	003	003	2000.00
<< ファイルの終わり >>				

主テーブルにある従業員番号 003 の2件のレコードが、副テーブルでは同じ従業員番号 003 で同じレコードにまとめられていることに注目してください。

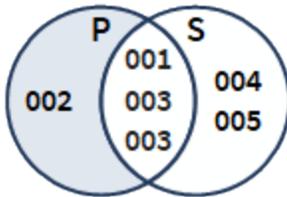
分析 - 出力テーブルで、**期間別支払い小切手金額**を比較し、従業員に正しく支払われたことを検証します。従業員 003 は2つの小切手を受領していますが、合計金額 \$2000 は正しいです。

不一致の主レコード

例

テスト - 従業員のリストにない人に支払われたかどうかを確認します。

アプローチ - 従業員レコードテーブル(S)との一致がない給与台帳テーブル(P)のすべてのレコードで1つの出力レコードを作成する結合タイプを使用します。



出力テーブル - 給与が支払われた従業員の中で、従業員レコードテーブルに存在していない人がすべて出力されます。

	Cheque amount	Employee number
1	2200.00	002
	<< ファイルの終わり >>	

分析 - 出力テーブルのすべてのレコードのフォローアップが必要です。

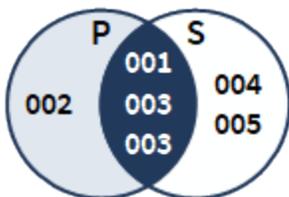
おそらく、従業員 002 は有効な従業員で、エラーにより従業員レコードテーブルから省略されたか、正しくない従業員番号で登録されています。あるいは、従業員 002 は詐欺の目的で作成された幽霊社員の可能性があります。

すべての主レコードおよび一致した副レコード

例

テスト - 発行されたすべての小切手の金額を検証します。

アプローチ - 従業員レコードテーブル(S)との一致があるかどうかに関係なく、給与台帳テーブル(P)のすべてのレコードで1つの出力レコードを作成する結合タイプを使用します。



出カテーブル-支払われた人の全一覧が含まれます。

	Cheque amount	Employee number	Employee number	Pay per period
1	1850.00	001	001	1850.00
2	2200.00	002		0.00
3	1500.00	003	003	2000.00
4	500.00	003	003	2000.00
<< ファイルの終わり >>				

分析 - 出カテーブルで、期間別支払い小切手金額を比較し、従業員に正しく支払われたことを検証します。従業員 002 は \$2200 が支払われましたが、Pay Per Period フィールドによると何も支払われないはずでした。

メモ

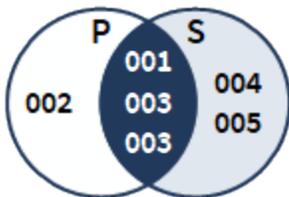
Analytics では、主テーブルの不一致レコードに対する副テーブルのフィールドの欠落には、空白かゼロを挿入します。

すべての副レコードおよび一致した主レコード

例

テスト - 従業員レコード テーブルのすべて従業員が支払われたことを検証します。

アプローチ: - 給与台帳テーブル (P) との一致があるかどうかに関係なく、従業員レコード テーブル (S) のすべてのレコードで 1 つの出カレコードを作成する結合タイプを使用します。



出カテーブル-すべての従業員の全一覧と支払い内容が含まれます。

	Cheque amount	Employee number	Employee number	Pay per period
1	1850.00	001	001	1850.00
2	1500.00	003	003	2000.00
3	500.00	003	003	2000.00
4	0.00		004	1975.00
5	0.00		005	2450.00
<< ファイルの終わり >>				

分析 - 出力テーブルで、期間別支払い小切手金額を比較し、従業員に正しく支払われたことを検証します。従業員 004 および 005 はまったく支払われていないことを確認できます。

メモ

Analytics では、副テーブルの不一致レコードに対する主テーブルのフィールドの欠落には、空白かゼロを挿入します。

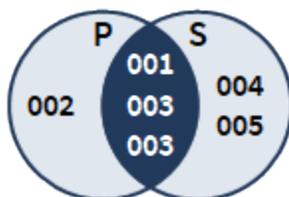
すべての主レコードおよび副レコード(一致および不一致)

例

テスト - すべての給与および従業員データを検査します。

アプローチ - 作成する結合タイプ:

- 従業員レコードテーブル(S) との一致がある給与台帳テーブル(P) のすべてのレコードで 1 つの出力レコード
- テーブルのすべての一致しないレコードの 1 つの出力レコード



出力テーブル - 一致するか不一致かに関係なく、すべての給与および従業員データが含まれます。

	Cheque amount	Employee number	Employee number	Pay per period
1	1850.00	001	001	1850.00
2	2200.00	002		0.00
3	1500.00	003	003	2000.00
4	500.00	003	003	2000.00
5	0.00		004	1975.00
6	0.00		005	2450.00
<< ファイルの終わり >>				

分析 - 出力テーブルで、期間別支払い小切手金額を比較します。

- 従業員が支払われ、正しく支払われたことを確認します
- 支払われ、従業員レコードテーブルにない人を特定します
- 給与がまったく支払われなかった従業員を特定します

メモ

Analytics では、不一致レコードに対するフィールドの欠落には、空白かゼロを挿入します。

一致した主レコードおよび副レコード(すべての副一致)

この例は、以下に示すサンプルデータを使用します。

主テーブル

Payroll_Ledger			
	Employee number	Cheque amount	Pay date
1	004	1975.00	2018/01/15
2	004	1975.00	2018/01/31
3	004	1975.00	2018/02/15
4	004	1975.00	2018/02/28
5	005	2450.00	2018/01/15
6	005	2450.00	2018/01/31
7	005	2450.00	2018/02/15
8	005	2450.00	2018/02/28
9	006	2100.00	2018/01/15
10	006	2100.00	2018/01/31
11	006	2300.00	2018/02/15
12	006	2300.00	2018/02/28
<< ファイルの終わり >>			

副テーブル

サンプルデータと例の詳細

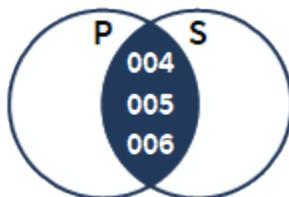
Employee_Records			
	Employee number	Pay per period	Start date
1	004	1975.00	2016/10/19
2	005	2450.00	2017/05/17
3	006	2100.00	2015/09/15
4	006	2300.00	2018/02/01
<< ファイルの終わり >>			

給与台帳テーブル (主)	給与台帳テーブル全体には、2018年の全給与期間におけるすべての給与支払い情報が含まれています。この例ではサンプルとして、従業員に対する1月と2月の給与支払い情報を使用します。
従業員レコードテーブル (副)	<p>人事部によって管理されています。従業員レコードテーブルの内容:</p> <ul style="list-style-type: none"> 有効な従業員の包括的なリスト 期間ごとの各従業員の支払い 各従業員の開始日 新しい職位の従業員の開始日 <p>従業員 006 のレコードは2件あります。</p> <ul style="list-style-type: none"> 開始日 昇格と昇級のデータ
結合	次の例では、従業員番号を共通キーフィールドとして使用し、給与台帳テーブルと従業員レコードテーブルを結合します。 例は 多対多結合 です。
目標	例では、給与の不規則性をテストします。

例

テスト -従業員に正しく支払われたことを検証します。

アプローチ: -給与台帳テーブル(P)と従業員レコードテーブル(S)のレコード間のすべての一致で1つの出力レコードを作成する結合タイプを使用します。



メモ

結合対象の両方のソーステーブルで、一致するキー値が複数存在するため、すべての副一致を含む結合タイプを使用して、関連するすべてのデータを捕捉し、正しい結果を導くようにしてください。

出力テーブル-支払日ごとに、給与が支払われた従業員の中で、従業員レコードテーブルに存在している従業員がすべて出力されます。

	Pay date	Cheque amount	Employee number	Employee number	Pay per period	Start date
1	2018/01/15	1975.00	004	004	1975.00	2016/10/19
2	2018/01/31	1975.00	004	004	1975.00	2016/10/19
3	2018/02/15	1975.00	004	004	1975.00	2016/10/19
4	2018/02/28	1975.00	004	004	1975.00	2016/10/19
5	2018/01/15	2450.00	005	005	2450.00	2017/05/17
6	2018/01/31	2450.00	005	005	2450.00	2017/05/17
7	2018/02/15	2450.00	005	005	2450.00	2017/05/17
8	2018/02/28	2450.00	005	005	2450.00	2017/05/17
9	2018/01/15	2100.00	006	006	2100.00	2015/09/15
10	2018/01/15	2100.00	006	006	2300.00	2018/02/01
11	2018/01/31	2100.00	006	006	2100.00	2015/09/15
12	2018/01/31	2100.00	006	006	2300.00	2018/02/01
13	2018/02/15	2300.00	006	006	2100.00	2015/09/15
14	2018/02/15	2300.00	006	006	2300.00	2018/02/01
15	2018/02/28	2300.00	006	006	2100.00	2015/09/15
16	2018/02/28	2300.00	006	006	2300.00	2018/02/01
	<< ファイルの終わり >>					

分析 - 出力テーブルで、期間別支払い小切手金額を比較し、各支払日に従業員に正しく支払われたことを検証します。

すべての副テーブル一致を含む結合タイプ(Analytics 多対多結合)を実行したため、2月15日以降に従業員006によって受領された小切手金額における\$200分の増加は、2月1日以降の\$200の増加を示す一致する従業員レコードによって説明されます。

冗長な結合されたレコードを削除 - 結合されるデータの本質によっては、多対多結合で、冗長な結合レコードが作成される可能性があります。上記の例では、従業員006の結合レコードの一部には、無効な支払日-開始日の組み合わせが含まれていました。フィルターを使用して、無効な組み合わせを削除し、出力テーブルの読み取りを簡単にします。

```
Emp_Num="004" OR Emp_Num="005" OR (Emp_Num="006" AND Pay_date <=
`20180131` AND Start_date = `20150915`) OR (Emp_Num="006" AND Pay_date
> `20180131` AND Start_date = `20180201`)
```

	Pay date	Cheque amount	Employee number	Employee number	Pay per period	Start date
1	2018/01/15	1975.00	004	004	1975.00	2016/10/19
2	2018/01/31	1975.00	004	004	1975.00	2016/10/19
3	2018/02/15	1975.00	004	004	1975.00	2016/10/19
4	2018/02/28	1975.00	004	004	1975.00	2016/10/19
5	2018/01/15	2450.00	005	005	2450.00	2017/05/17
6	2018/01/31	2450.00	005	005	2450.00	2017/05/17
7	2018/02/15	2450.00	005	005	2450.00	2017/05/17
8	2018/02/28	2450.00	005	005	2450.00	2017/05/17
9	2018/01/15	2100.00	006	006	2100.00	2015/09/15
11	2018/01/31	2100.00	006	006	2100.00	2015/09/15
14	2018/02/15	2300.00	006	006	2300.00	2018/02/01
16	2018/02/28	2300.00	006	006	2300.00	2018/02/01
	<< ファイルの終わり >>					

曖昧結合

Analytics 曖昧結合は、キーフィールドと値の曖昧一致を使用して、2つの Analytics テーブルを新しい3番目のテーブルに結合します。ほとんどの点において、曖昧結合は標準の Analytics 結合に似ています ("テーブルの結合" ページ 960を参照)。主な違いは、曖昧結合は、キーフィールド値の完全一致に基づくレコードの結合のほか、近似一致に基づくレコードの結合ができることです。

曖昧結合は、主キーおよび副キーに同じ種類のデータが少し異なる形式で含まれているときに役立ちます。あるいは、キーのデータに、完全一致を妨げる可能性がある誤字などの少しの不規則性があります。

例

シナリオ

考えられる不適切な報酬のデータを分析する方法の1つとして、従業員でもある業者を特定する必要があります。

アプローチ

ベンダー マスターテーブルと従業員テーブルが、共通キーとして各テーブルの住所フィールド (**Vendor_Street**、および **Emp_Address**) を使用して結合されます。ただし、キーフィールドの住所データの形式は少し異なります。このため、標準の結合ではなく、曖昧結合を使用します。

データの一部の概要

重要なデータクレンジングと調整作業がないと、以下に示す主キーおよび副キー値は、非常に一致の確率が高い住所であっても、標準の Analytics 結合では結合されません。

主キー値	副キー値
605 3rd Avenue	605 Third Avenue
400 High St SE	400 High Street S.E.
2203 Rowan Street	2203 Rowen St

データクレンジングと調整を行ったとしても、"Rowan" や "Rowen" といった綴りにわずかな違いがあるキー値はおそらく一致しません。

キー値は、曖昧結合設定に応じて、曖昧結合で結合できます。

出力結果

以下の結合されたテーブルの例では、正確なキーフィールド一致は紫でハイライトされています。曖昧キーフィールド一致は緑でハイライトされています。

	Vendor_Num	Vendor_Name	Vendor_Street	Emp_Address	Emp_Name	Emp_Num
1	11663	More Power Industries	150 North Michigan Ave.	150 North Michigan Ave.	Catherine Exelby	10
2	11435	Group Services	605 3rd Avenue	605 Third Avenue	Eileen Henderson	22
3	13928	Liberty Trading	300 North Meridian Street	300 North Meridian Street	Jorge Marin	65
4	11182	Industrial Equipment Co-Op	400 High St SE	400 High Street S.E.	John Mullen	50
5	13136	Muller Corp.	2203 Rowan Street	2203 Rowen St	Sybil Johnson	43

<< ファイルの終わり >>

曖昧結合と曖昧重複

曖昧結合は2つのテーブルのキーフィールドの値を分析します。ほぼ同一の値の単一のAnalyticsテーブルの単一のフィールドをテストするには、"あいまい重複分析" ページ 1276を参照してください。

曖昧結合の効果の改善

次の手法を1つ以上取り入れると、曖昧結合の効果を大幅に高めることができます。

- 主キーおよび副キーフィールド値の個別の要素を並べ替える
- 主キーおよび副キーフィールド値から汎用要素を削除する
- 主キーおよび副キーフィールド値を調整する

これらの手法では、より厳密な曖昧設定を使用して、同じ曖昧一致を取得しながら、誤検出一致の数を減らすことができます。これらの手法は別々に使用するか組み合わせて使用できます。

式または演算フィールドを作成する

これらの手法のいずれかを使用するには、適切なAnalytics関数といずれかまたは両方のキーフィールドを使用して、式または演算フィールドを作成する必要があります。

式の詳細については、"式の使用" ページ 866を参照してください。

演算フィールドの詳細については、"演算フィールドの定義" ページ 795を参照してください。

メモ

曖昧結合]ダイアログボックスでは、副キーフィールドに対して式を作成できません。ただし、Analytics コマンドラインまたはスクリプトでは、手動で副キーフィールド式を作成できます。別のオプションは、副キーフィールドとして使用する演算フィールドを作成することです。

キーフィールド値の個別の要素を並べ替える

`SORTWORDS()` 関数では、主キーフィールド値と副キーフィールド値の個別の要素を連続する順序に並べ替えることで、曖昧結合の効果を改善することができます。

アドレスの構成要素などの要素を並べ替えると、同じ情報を含む、形式が異なり、相互により密接に類似したキーフィールド値を作成できます。より密接な類似性により、キーフィールド値が相互の曖昧一致として選択される確率が上がります。

詳細については、「`SORTWORDS()` 関数」ページ 2498を参照してください。

`SORTWORDS()` の概要を説明する動画については、「[Fuzzy Matching Using SORTWORDS\(\)](#)」(英語のみ)をご覧ください。

メモ

キーフィールド値の要素の並べ替えは、レーベンシュタイン距離アルゴリズムを使用した曖昧一致に最適です。

ダイクストラアルゴリズムを使用しているときには、要素の並べ替えが有効な場合とそうでない場合があります。本番設定でダイクストラアルゴリズムと `SORTWORDS()` を使用するかどうかを決定する前に、サンプルデータのセットをテストしてください。

注意

曖昧結合と `SORTWORDS()` を使用する場合は、`SORTWORDS()` を比較対象の両方の文字列または両方のフィールドに適用する必要があります。

キーフィールド値から汎用要素を削除する

`OMIT()` は、主キーフィールド値および副キーフィールド値から "Corporation" や "Inc."、カンマ、ピリオド、アンパサンド (&) 文字などの一般的要素を除去することによって、曖昧結合の効果を高めることができます。

一般的要素と句読点の除去により、曖昧一致は、意味のある違いが発生する可能性のあるキーフィールド値の部分だけに集中されます。

詳細については、「`OMIT()` 関数」ページ 2390を参照してください。

キーフィールド値を一致させる

`REPLACE()` または `REGEXREPLACE()` 関数は、主キーフィールド値と副キーフィールド値の同じ要素の変異形を一致させることで、曖昧結合の効果を改善できます。たとえば、"Street"、"St."、"St" を一

致させ、単一の値 "St" を使用できます。

要素を一致させると、同じ情報を含む、形式が異なり、相互により密接に類似したキーフィールド値を作成できます。より密接な類似性により、キーフィールド値が相互の曖昧一致として選択される確率が上がります。

詳細については、"REPLACE() 関数" ページ 2469(簡単な置換) および"REGEXREPLACE() 関数" ページ 2455(複雑な置換) を参照してください。

出力テーブルサイズとコマンド パフォーマンス

出力テーブルサイズ

曖昧結合は、Analytics 多対多結合と似ています。すべての主キー値は、すべての副キー値と一致する可能性があります。出力テーブルのサイズは、主または副入力テーブルのサイズの数倍の大きさになることがあります。

コマンド パフォーマンス

曖昧一致アルゴリズムは、指定された度合いの曖昧さのキー値または完全に一致する値のみが実際に結合されることを保証します。ただし、すべての考えられる主と副の一致はテストされる必要があります。つまり、曖昧結合処理は時間がかかることがあります。実行する必要がある個別のテスト数は、主テーブルのレコード数を副テーブルのレコード数で乗算した値です。

最初の副一致との一致を制限する

最初の副キー一致のみを結合]を選択すると、処理時間が大幅に短縮され、出力結果のサイズが小さくなります。このオプションを有効にすると、各主キー値が、最初に発生した一致する副キー値にのみ結合されることを指定します。

このオプションは、次のいずれかの状況でのみ有効にしてください。

- **一致するかどうか** - 完全一致またはあいまい一致の一致が2つのテーブル間に存在するかどうかのみを確認し、すべての一致を特定するために必要な処理時間を減らしたい場合
- **1つ以下の一致** - 各主キー値に対する一致が副テーブルで多くても1つだけであることが確実な場合

このオプションは、主キー値と副キー値の間のすべての考えられる結合を出力結果煮含める必要がある場合には使用しないでください。

メモ

最初の副キー一致のみを結合]を選択し、最初の一致が完全一致であった場合は、後続の主キー値のあいまい一致は、結合された出力テーブルに含まれません。

ベスト プラクティス

主および副入カテゴリーを準備し、曖昧さの度合いを指定するときには、出カテゴリーサイズとコマンドパフォーマンスに注意してください。

- **データのカスタマイズ** - 関連するレコードのみが主および副テーブルに含まれることを保証します。一部のレコードに一致の可能性がない場合は、曖昧一致を実行する前に、除外します。
- **テスト実行** - 大きいデータセットの場合は、データの一部のみでテストを実行します。これは、曖昧一致アルゴリズムの適切な設定を得るためのより効率的な方法です。保守的な曖昧設定から開始し、必要に応じて、徐々に緩めて行きます。

曖昧一致アルゴリズム

曖昧結合を実行するときには、2つの異なる曖昧一致アルゴリズムから選択します。

- **ダイス係数**
- **レーベンシュタイン距離**

このアルゴリズムは、相互に完全に独立しているため、異なる結果を生成することができます。1つのアプローチは、曖昧結合を2回(各アルゴリズムで1回ずつ)実行し、結果を比較することです。一般的に、各結果セットの多数の曖昧一致は重複しますが、一部の一致は各結果セットに対して一意であることがあります。

曖昧度

各アルゴリズムの曖昧さの度合いを指定します。これにより、結果セットのサイズと構成を動的に変更することができます。「曖昧度」とは、2つの値がどれほど近く一致しているのかを差します。

選択するアルゴリズムに応じて、次の設定を使用して、曖昧度を制御します。

アルゴリズム	設定
ダイス係数	<ul style="list-style-type: none">◦ N-gram◦ パーセント
レーベンシュタイン距離	<ul style="list-style-type: none">◦ 距離

異なる曖昧さの度合いで実験する保守的に開始し、小さい結果セットを生成します。次に、明らかに一致ではない(誤検出)結合された値が多くなりすぎるまで、徐々に設定を緩めます。

ダイス係数

ダイス係数アルゴリズムは、0.0000 ~ 1.0000 の尺度で、主キー値と副キー値の間の類似性の度合いを測定することで動作します。2つの値のダイス係数が大きいほど、類似性が高くなります。

詳しく表示

ダイス係数	意味
1.0000	各文字列は同じ文字から構成されています。ただし、文字の順序と大文字小文字の表記は異なる場合があります。 2つの値の n -gram は 100% 同一です。 N -gram は以下で説明します。
0.7500	2つの値の n -gram は 75% 同一です。
0.0000	2つの値には同一の n -gram がないか、 N-gram 設定で指定された長さが比較対象の2つの値の短い方よりも長くなっています。

N-gram

まず、ダイス係数は比較対象の2つの値を n -gram 文字ずつに分割して計算されます。 N -gram は文字の重複するブロックです。 n の長さは、**N-gram** 設定で指定する長さです。

ここには、上記の例の2つの値があります。長さが2文字の n -gram に分割されています($n = 2$)。

2203 Rowan Street	22 20 03 3_ _R Ro ow wa an n_ _S St tr re ee et
2203 Rowen St	22 20 03 3_ _R Ro ow we en n_ _S St

ダイス係数は、同一の2つの値の n -gram の割合を表します。この場合、28 n -gram 中 20 が同一であり、71.43% です。0.7143 は小数として表されます。

メモ

N-gram 設定の長さを大きくすると、2つの値の類似度の基準が厳しくなります。

パーセント

パーセント 設定を指定するときには、曖昧一致となるための、2つの値の許容可能な最低ダイス係数を指定しています。たとえば、`0.7500` を指定すると、一致を作成するためには、2つの値の少なくとも75%の n -gram が同一である必要があります。

パーセント設定	意味	2203 Rowan Street / 2203 Rowen St
0.7500	曖昧一致となるためには、2つの値の少なくとも75%の n -gram が同一である必要があります。	一致しません。結合されたテーブルに含まれません (ダイス係数 = 0.7143)
0.7000	曖昧一致となるためには、2つの値の少なくとも70%の n -gram が同一である必要があります。	一致します。結合されたテーブルに含まれます (ダイス係数 = 0.7143)

ダイス係数の仕組みの詳細については、「`DICECOEFFICIENT()` 関数」ページ 2238を参照してください。

レーベンシュタイン距離

レーベンシュタイン距離アルゴリズムは、0 から開始する整数の尺度で、主キー値と副キー値の間の差異を測定して機能します。この尺度は、ある値を他の値と同一にするために必要な、1 文字の編集の回数を表します。2 つの値間のレーベンシュタイン距離がより大きければ、相違も大きくなります。

詳しく表示

レーベンシュタイン距離	意味
0	各値は、同一の順序の、同一の文字セットから構成されます。大文字と小文字は異なる場合があります。
2	2 つの値を同一にするには、2 つの単一の文字の編集が必要です。 例: "Smith" と "Smythe" <ul style="list-style-type: none"> ○ 編集 1 - 'y' を 'i' に代入 ○ 編集 2 - 'e' を挿入
3	2 つの値を同一にするには、3 つの単一の文字の編集が必要です。 例: "Hanssen" と "Jansn" <ul style="list-style-type: none"> ○ 編集 1 - 'J' を 'H' に代入 ○ 編集 2 - 's' を削除 ○ 編集 3 - 'e' を削除

距離

距離設定を指定するときには、曖昧一致となるための、2 つの値間の許容可能な最大レーベンシュタイン距離を指定しています。たとえば、2 を指定する場合、2 つの値を同一にするために、2 回以下の編集が必要です。

距離設定	意味	Hanssen / Jansn
2	曖昧一致となるためには、2 つの値を同一にするために、2 回以下の編集が必要です。	一致しません。結合されたテーブルに含まれません (レーベンシュタイン距離 = 3)
3	曖昧一致となるためには、2 つの値を同一にするために、3 回以下の編集が必要です。	一致します。結合されたテーブルに含まれます (レーベンシュタイン距離 = 3)

レーベンシュタイン距離の仕組みの詳細については、「LEVDIST() 関数」ページ 2335を参照してください。関数とは異なり曖昧結合で使用されるレーベンシュタイン距離アルゴリズムは、自動的に先頭と末尾の空白を取り除きます。大文字と小文字は区別されません。

手順

曖昧結合を使用すると、キーフィールドと値の曖昧一致を使用して、2つの Analytics テーブルを新しい 3 番目のテーブルに結合できます。

方法の説明

メモ

手順の後に詳細情報が表示されます。" [曖昧結合] ダイアログ ボックス オプション" 次のページを参照してください。

1. ナビゲーターで、主テーブルを開き、(ナビゲーターで) 副テーブルを右クリックして **副テーブルとして開く**を選択します。

主および副テーブルアイコンが数字 1 ~ 2 で更新され、相互への関連を示します  

2. **データ > 曖昧結合** の順にクリックします。
3. **メイン** タブで、使用する曖昧一致アルゴリズムを選択します。

- **ダイス係数**
- **レーベンシュタイン**

4. 選択するアルゴリズムに応じて、設定を使用して、曖昧度を制御します。

ダイス係数

- **N-gram**
- **割合**

レーベンシュタイン

- **距離**

設定は以下で説明します。

5. (省略可能) **最初の副キー一致のみを結合**を選択すると、各主キー値が、最初に発生した一致する副キー値にのみ結合されることを指定します。
6. **主キー** リストから主キーフィールドを選択します。
1つの主キーフィールドのみを選択できます。これは、文字フィールドである必要があります。
7. **副キー** リストから副キーフィールドを選択します。
1つの副キーフィールドのみを選択できます。これは、文字フィールドである必要があります。
8. **主フィールド** および **副フィールド** リストから、結合テーブルに含めたいフィールドを選択します。

メモ

結合テーブルに含める場合は、主キーおよび副キーフィールドを明示的に選択する必要があります。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

9. **保存先**]テキスト ボックスに、新しい結合された ACL テーブルの名前を指定します。
10. (省略可能) **[詳細]**タブ:
 - a. レコードのサブセットのみが処理されることを指定する場合は、**範囲**]パネルでオプションのいずれかを選択します。
 - b. 出力結果を既存の Analytics テーブルの末尾に追加する場合は、**既存のファイルに追加する**]を選択します。
11. **OK**]をクリックします。
新しい結合されたテーブルが出力です。

曖昧結合]ダイアログボックス オプション

次の表は、**曖昧結合**]ダイアログボックスのオプションの詳細を示します。

メイン]タブ

オプション- 曖昧結合]ダイアログボックス	説明
ダイス係数	<p>主キー値と副キー値の間の曖昧一致には、ダイス係数を使用します。</p> <ul style="list-style-type: none"> ◦ N-gram - 使用する n-gram の長さ 1 以上の整数を指定します。 n-gram の長さを大きくすると、2 つ値の間の類似度の基準が厳しくなります。 ◦ 割合 - あいまい一致と見なされるための、2 つの値の許容可能な最低ダイス係数 0.0000 ~ 1.0000 の小数を指定します(例:0.7500)。小数点 4 桁の最大値を使用します。 値を小さくすると、あいまい度が高い値(相互の関連性が低い値)が含まれるため、一致の件数が多くなります。
レーベンシュタイン	<p>主キー値と副キー値の間の曖昧一致には、レーベンシュタイン距離を使用します。</p> <ul style="list-style-type: none"> ◦ 距離 - あいまい一致であると見なされるための、2 つの値間の許容可能な最低レーベンシュタイン距離 1 以上の整数で指定します。 値を大きくすると、あいまい度が高い値(相互の関連性が低い値)が含まれるため、一致の件数が多くなります。
最初の副キー一致のみを結合	<p>各主キー値が、最初に発生した副キー一致にのみ結合されることを指定します。 このオプションをオフにする場合、デフォルト動作では、各主キー値をすべての副キー一致に結合します。</p>

オプション- [曖昧結合] ダイアログ ボックス	説明
副テーブル	副テーブルを選択する代替方法。
主キー 副キー	<p>2つのテーブルを結合するために使用する共通キーフィールドを指定します。</p> <ul style="list-style-type: none"> 主キーおよび副キーリストで直接共通キーフィールドを選択できます。 [主キー]または[副キー]をクリックすると、選択したフィールド]ダイアログボックスを開き、共通キーフィールドを選択するか、主キーで式を作成できます。 <p>キーフィールドのガイドライン:</p> <ul style="list-style-type: none"> データ型 - キーフィールドは文字データ型にする必要があります。 長さ - キーフィールドの長さが異なる場合、これらのフィールドは自動的に調整されます。 名前と開始位置 - キーフィールドの名前と開始位置はテーブル間で異なってもかまいませんが、同じデータ要素になっている必要があります。 複数のキーフィールド - テーブルごとに1つのキーフィールドのみがサポートされます。
主フィールド 副フィールド	<p>結合テーブルに含めるフィールドを指定します。</p> <ul style="list-style-type: none"> 主キーおよび副キーリストで直接フィールドを選択できます。 [主フィールド]または[副フィールド]をクリックすると、選択したフィールド]ダイアログボックスを開き、フィールドを選択するか、1つ以上の主フィールドで式を作成できます。 主フィールドと副フィールドの選択順序は、結合されるテーブルにおけるフィールドの表示順序に対応します。 <p>グループとして、結合されるテーブルで、主フィールドは副フィールドの前に現れます。</p>
出力テーブルを開く	操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。
もし次の	<p>(省略可能) レコードを処理から除外する条件を作成できます</p> <ul style="list-style-type: none"> [f] テキスト ボックスに条件を入力するか、または [f] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 条件は、主テーブル、副テーブル、または両方を参照できます。 <p>メモ</p> <p>[式ビルダー]で副テーブルフィールドにアクセスするには、テーブルから]ドロップダウンリストで副テーブルを選択します。</p> <p>If 条件は、任意の範囲オプション(先頭、次へ、While)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
変換先	<p>出力テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。

オプション- 曖昧結合]ダイアログ ボックス	説明
	<p>○ プロジェクト フォルダー以外の場所に出カテーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先]をクリックし、別のフォルダーに移動します。</p> <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出カテーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

詳細]タブ

オプション- 曖昧結合]ダイアログ ボックス	説明
<p>範囲パネル</p>	<p>処理される主テーブルのレコードを指定します。</p> <ul style="list-style-type: none"> ○ すべて-(デフォルト) 主テーブルのすべてのレコードが処理されます。 ○ 先頭--このオプションを選択してテキスト ボックスに数を入力すると、主テーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 ○ 次--このオプションを選択してテキスト ボックスに数を入力すると、主テーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 <p>行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p> <ul style="list-style-type: none"> ○ While-- WHILE ステートメントを使用して、条件に基づいて主テーブル内のレコードの処理を制限するには、このオプションを選択します。 <ul style="list-style-type: none"> ● [While]テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 ● WHILE ステートメントでは、指定した条件が true と評価される間のみ、レコードを処理することができます。 ● While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。

オプション- [曖昧結合]ダイアログボックス	説明
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうかが不確かな場合は、既存のファイルに追加するを選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <ul style="list-style-type: none"> 異なる長さの文字キーフィールドの結合を行う場合、または長さが異なる文字キーフィールドで結合を行う場合は、Analytics がそれらのフィールドを一致させるよう試みることを提示するメッセージが表示されます。 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。 <p>最後に追加 オプションが表示されることを期待していたのに表示されない場合は、いいえをクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>

テーブル結合時の自動一致

Analytics で2つのテーブルを結合する場合、次の2つの状況ではキーフィールドを自動的に一致させません。

- 1つの文字キーフィールドと1つの数値キーフィールドを使用します
- 長さが異なる文字キーフィールドを使用します

自動一致によって、特殊な結合が簡略化され、関連する作業を減らすことができます。

Analytics でキーフィールドを自動的に一致させる場合、その操作と関連する構文はコマンド ログに記録されます。

文字キーフィールドと数値キーフィールドによる結合の自動一致

Analytics では、文字キーフィールドと数値キーフィールドでテーブルを結合する場合、結合の実行時に VALUE() 関数を使用して文字フィールドを数値に変換することでデータ型を自動的に一致させます。

例

共通キーフィールドとして社会保障番号を使用して、2つのテーブルを結合します。

- 1つのキーフィールドには文字データとして書式設定された数値と記号が含まれます: 555-44-3322
- 他のキーフィールドには数値データとして書式設定された数値のみが含まれます: 555443322

Analytics は文字と数値の結合を自動的に調整するため、関数を使用して最初にフィールドを手動で調整せずに、標準結合を実行できます。

追加の詳細

- 文字フィールドに含まれるアルファベット文字、ハイフンやかっこなどの区切り記号は無視され、数値フィールドの値による一致で、数値のみを一致の対象とすることができます。
- アルファベット文字の位置は数値の一致に影響ありません。
- 文字フィールドの元のデータ型およびすべての文字(アルファベット文字と区切り文字を含む)は、結合後のテーブルに保持されます。
- 文字または数値フィールドを主キーフィールドにできます。
- 文字フィールドも、文字フィールドに含まれる数字も、数値フィールドとして同じ長さにする必要はありません。フィールド長に関係なく、同一の数字のみが一致します。

負の数値に関連する自動一致

文字キーフィールドと数値キーフィールドの自動一致では、負の数値の一致を直接サポートしません。VALUE() 関数の意図する動作に対応するため、Analytics は文字キーフィールドに対して ABS() 関数も使用します。この関数は結合の実行中、一時的にすべての数値を正の値に変換します。

負の数値を含むキーフィールドに自動一致を使用したい場合は、通常の方法で結合を実行した後、手動で編集したコマンドログ エントリを使用して結合を再度実行し、VALUE() 関数のみを文字キーフィールドに適用するようにします。この方法は、文字キーフィールドと負の値を含む数値キーフィールドによる結合時には正しい結果が得られますが、どちらかのフィールドに数値以外のデータが存在する場合は正確に動作しません。

文字キーフィールド長の自動一致

長さの異なる文字キーフィールドを選択した場合、Analytics ではテーブル結合の実行時に、短い方のフィールドに空白を追加してフィールドの長さを自動的に一致させます。短い方のフィールドの元の長さは、結合後のテーブルに保持されます。

フィールド長の自動一致は、文字ベースの演算キーフィールド、および複数の文字キーフィールドで構成される共通キーに対しても動作します。

フィールド長を自動的に一致させるのは文字キーフィールドのみです。数値や日付時刻のキーフィールドでは長さの自動一致を行いません。

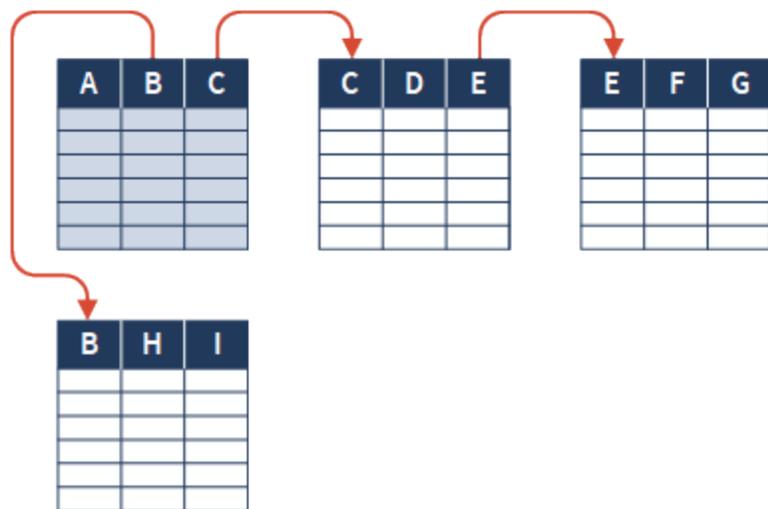
調整に影響されない値の一致

文字キーフィールドの長さを一致させても、値の一致については値自体の厳密な一致が必要です。たとえば、値がショートバージョンとロングバージョン("ABC"と"ABC Corporation"など)の場合、これらの値が、長さを一致させたフィールドに含まれていたとしても、一致は発生しません。

テーブルの関連付け

テーブルの関連付けを使用すると、それぞれ異なるレコード構造を持つ Analytics テーブルを最大 18 個まで関連付けることができます。関連付けられたテーブルにあるフィールドを組み合わせることで、単一テーブルに存在するデータのようにアクセスおよび分析することができます。

異なる複数のフィールド(データ要素)があるとレコード構造は異なります。いくつかの物理テーブル間でデータの関係をすばやく図式化するのに必要な情報作業の場合、あるいはレポートの生成前に、対応する完全な名前によってコードをリンクする場合には関連付けを利用できます。



例

シナリオ

3月の顧客と販売された製品に関する詳細を含む売上レポートを作成する必要がある場合に、データが3つのテーブルに分散しているとします。

アプローチ

顧客マスターテーブルを注文テーブルに関連付け、注文テーブルを製品マスターテーブルに関連付けて、レポートに必要なすべての情報を含むテーブルの一時的な関連付けを作成します。

- 顧客マスターテーブル内の顧客名と顧客拠点 -
- 注文テーブル内の注文詳細 -
- 製品マスターテーブル内の製品詳細 -

関連付けは仮想テーブルを作成します

関連付けられたテーブルの結果は仮想的なものです。それらのテーブルは独立しており、どの時点においても関連付けられません。

関連付けは結合やマージと異なり、新しいテーブルを作成しません。その代わりに、関連付けられたテーブルまたは子テーブルのフィールドは、関係を確立した親テーブルで使用できるようになります。

必要であれば、別の操作を実行し、親テーブルや子テーブル内にあるフィールドの任意の組み合わせを新しいテーブルに抽出することができます。

共通のキーフィールドを使用したテーブルの関連付け

共通のキーフィールド(従業員番号やベンダーID、または住所など両方のテーブルに存在するデータ要素)を使用してテーブルを関連付けます。キーフィールドに同一の値が存在する場合は1つの一致のみが生じます。この一致は別のテーブルのレコードを個別に関連付けます。基本的な関連付け操作では、部分一致はサポートされていません。

関連付けるテーブルのキーフィールドには複数の要件が適用されます。

キーフィールドの特性	要件
データ要素	同じである必要があります。たとえば、両方のキーフィールドは従業員番号フィールドです。
データ型	データ型でも、またキーフィールドでもかまいませんが、関連付けするテーブル間で統一されている(データ型が同じになっている)必要があります。たとえば、2つの文字フィールドです。 日付時刻のサブタイプ(日付、日付時刻および時刻)の関連付け対象は、同じサブタイプのみです。
フィールド型	物理フィールドまたは演算フィールドにできます。
フィールド名	異なることができます。
開始位置	異なることができます。
フィールド長	同じである必要があります。
文字フィールドの両端揃えおよび大文字と小文字	同じである必要があります。

親テーブル、子テーブル、およびキーフィールド

関連付け操作の対象となるテーブルとキーフィールドは、そのテーブルを追加する順序に基づき、親および子として識別されます。

- **親テーブル**-最初に追加したテーブル(1つのテーブルを開き、関連付け操作を開始すると自動的に追加される)
- **親キーフィールド**-親テーブルから選択したキーフィールド
- **子テーブル**-2番目以降に追加したテーブル
- **子キーフィールド**-子テーブルから選択したキーフィールド

親および子となるテーブルやキーフィールドは自由に選択できますただし、この関連付けが成功するのは、その対象キーフィールドが関連付けの必要条件を満たしている場合のみです。

詳細については、「キーフィールドについて」ページ 231を参照してください。

子テーブルのフィールドへのアクセス

関連付けが確立されると、子テーブルのフィールドを親ビューに追加することができます。ただし、これは必ずしも行う必要はありません。親テーブルを通じて子テーブルのフィールドに対しアクセスおよび分析を行うことができます。これには Analytics の各種ダイアログボックスの **対象テーブル**]ドロップダウンリストや**式ビルダー**を使用できます。親ビューにフィールドが追加されているかどうかは関係ありません。

親テーブルからアクセスされる子テーブルのフィールドは、抽出元テーブルを示すため「**テーブル名.フィールド名**」という形式で表示されます。関連付けられたテーブルへのアクセスおよび関係の編集は、子テーブルではなく親テーブルを通じて行うことができます。

関連テーブルの並べ替えとインデックスの作成

関連付けの結果生じる仮想テーブルは、親テーブルの既存のソート順を使用します。テーブルを関連付ける前に、親テーブルのキーフィールドを並べ替えたりインデックスを作成する必要はありません。ただし、関連付けには親テーブル用のオプション **あらかじめ並べ替える**]がないので、並べ替えやインデックス作成を事前に行うこともできます。

関連付け操作の内部的な機能の一部として、子テーブルのキーフィールドには自動的にインデックスが作成され、昇順に並べ替えられます。これら子テーブルのインデックスは、その子テーブルの関連付けが解除された後も保持されますが、必要に応じて手作業で削除できます。

関連付けに関する追加情報

次のテーブルでは関連付けに関する追加情報を提供します。

機能分野	詳細
レコードの一致	<p>一組のテーブルの関連付けは、論理的には"主テーブルの全レコード" オプションを使用した結合に相当します。つまり、一致する主レコードと副レコード(親レコードと子レコード)、および一致しない主レコードを含む多対一結合タイプの使用です。</p> <p>この相当する多対一結合と同様、関連付け操作では、親キーの値と、子キーの値で最初に見つかった一致のみを有効とします。子キーの値で同じ一致がほかに存在していた場合でも、それらは無視されます。テーブルの関連付けを計画する場合は、このような動作を考慮に入れる必要があります。特に、子テーブルで、一致対象のキー値が複数存在することが妥当である場合などは留意してください。親テーブルを子テーブルに、また子テーブルを親テーブルにというように、2つのテーブルの関係を逆転させてみるのも1つの方法です。</p>
一致しないレコードおよびフィールド値の欠落	<p>親キーの値が、関連付けられている子テーブルに存在しない場合、Analytics では、欠落するフィールド値の対処として、文字や日付時刻フィールドには空白を、数値フィールドにはゼロを、そして論理フィールドには"F"を表示します。</p>
子テーブルのキーフィールドの重複または空白	<p>子テーブルのキーフィールドの値が重複していたり欠落していることで、その後の分析が無効になってしまうような場合、事前に子テーブルから重複や空白を除去することで解決する可能性があります。</p>
関連付けられたテーブルにあるデータの抽出	<p>関連付けられたテーブルからデータを抽出する場合は2つのオプションがあります。</p> <ul style="list-style-type: none"> ◦ 抽出]ダイアログボックスの ビュー]または フィールド]オプションのいずれかを使用して、親テーブルと子テーブルからデータの一部またはすべてを抽出し、新しいAnalytics テーブルに出力できます。新しいテーブルはどのテーブルにも関連付けられていません。 ◦ ビュー]オプションを使用する場合、まず子テーブルの適切なデータを親ビューに追加する必要があります。 ◦ 抽出]ダイアログボックスの レコード]オプションを使用して、親テーブルからデータを抽出し、新しいAnalytics テーブルに出力できます。その新しいテーブルは元の親テーブルの関連付けを保持します。レコード]オプションは子テーブルのデータの抽出をサポートしません。
キーフィールド長の同一化は不要	<p>Analytics では、親テーブルと子テーブルで共通キーフィールドの長さが同一であることを強制しません。</p> <p>同一の長さのフィールドを使用することをお勧めします。必要であれば、関連付けを行う前にフィールド長を手作業で揃えてください。フィールド長が異なるキーフィールドを使用して関連付けを行った結果は信頼性がありません。</p> <p>日付、日付時刻、または時刻を用いた操作を実行する場合、Analytics は Analytics 内部の日付時刻形式を使用しているため、日付時刻キーフィールドの長さが異なることがあります。</p>
キーフィールドのデータ型の変更	<p>親テーブルと子テーブルのキーフィールドのデータ型は、それらのフィールドが既にテーブルの関連付けに使用されている場合は、変更することができません。いずれかのフィールドのデータ型を変更する必要がある場合は、まず関係を削除しておく必要があります。いずれかのフィールドのデータ型を変更することによって、対応付けていたもう一方のフィールドと異なるデータ型になった場合は、それら2つのフィールドをテーブルの関連付けに使用することはできなくなります。</p>
条件付きインデックスの回避	<p>子テーブルのキーフィールドに対しては条件付きインデックスを使用しないでください。その代わりに、親テーブルとその関連テーブルに対して操作を実行するときに条件を適用します。</p> <p>関連付けの構築時に条件付きインデックスを使用すると、関連付けの階層のさまざまな段階で意図しないデータの欠落が生じる可能性があります。より安全に行うには、関連付けを保証するのに十分なデータセットを表す関連付けを構築し、その後、必要であれば条件を適用します。</p>

機能分野	詳細
関連付けられるテーブルの場所に関する制限事項	<p>テーブルを関連付ける場合、その対象となるテーブルは同じ Analytics プロジェクト内に存在している必要があります。サーバーにあるテーブルであれば同じサーバー上にあり、同じサーバー プロファイルを使用してアクセスする必要があります。ローカルのテーブルとサーバーのテーブルを関連付けることはできません。</p>
行揃えと大文字小文字の同一化	<p>文字キーフィールドを使用してテーブル間の関連付けを行う場合、両フィールドの行揃えおよび大文字小文字の表記が同一でなければなりません。</p> <ul style="list-style-type: none"> ◦ 両方のキーフィールドは同じ行揃えであることが必要です。LTRIM() 関数を使用すると、キーフィールドから先頭のスペースを削除します。 ◦ 両方のキーフィールドはそれぞれ、フィールド内の単語(アルファベット)の大文字、小文字または大文字/小文字の混在の表記が同じあることが必要です。大文字小文字の表記を同一にするには、UPPER()、LOWER() または PROPER() 関数を使用します。
UTC ベースのデータと非 UTC データの関連付け	<p>UTC ベースおよび非 UTC の日付時刻キーフィールドは、2 つのテーブルの関連付けに使用できません(UTC は協定世界時で、経度 0 度地点における時刻)。日付時刻または時刻を用いた操作を実行する場合、Analytics では Analytics 自身の内部日付時刻形式を使用するので、以下の2 つの日付時刻は同一であると解釈され、一致します。</p> <ul style="list-style-type: none"> ◦ UTC ベース - 31/12/2014 10:30:15-05:00 ◦ 非 UTC - 31/12/2014 15:30:15 <p>Analytics 操作の中で、UTC ベースの時刻データと非 UTC の時刻データを混在させる場合は注意が必要です。Analytics は上の2 つの時刻値を一致させます。しかし、一方の値はタイムゾーンを参照し、もう一方の値は参照しないため、論理的には意味がない可能性があります。UTC の詳細については、"日付と時刻オプション" ページ 135を参照してください。</p>

テーブルの関連付け

一組のテーブルごとに、共通キーフィールドを使用して異なるレコード構造を持つ複数の Analytics テーブルを関連付けることができます。テーブルが関連付けられると、親テーブルを使用して、関連付けたテーブルにあるフィールドを組み合わせたデータに対しアクセスおよび分析が行えるようになります。

メモ

親と子の関係を逆にすると異なる結果が生じるため、関係の親テーブルと子テーブルの特定に注意してください。詳細については、"結合または関連付けの一般的な用途" ページ 959を参照してください。

手順

メモ

手順の後に詳細情報が表示されます。" [関係] ダイアログ ボックス オプション" 次のページを参照してください。

1. ナビゲーターで親テーブルを開きます。
2. **データ | テーブルの関連付け** の順にクリックします。
3. **関係** ダイアログ ボックスで、**テーブルの追加** をクリックして 1 つまたは複数の子テーブルを選択します。

親テーブルを含め、最大 18 個のテーブルを関連付けることができます。

ヒント

隣接する複数のテーブルを選択するには、**Shift キー + クリック**を、隣接していない複数のテーブルを選択するには、**Ctrl キー + クリック**を利用できます。

子テーブルを個々に追加するには、**テーブルの追加** リストでその子テーブルをダブルクリックします。

4. **追加** をクリックした後、**閉じる** をクリックします。

ヒント

関係 ダイアログ ボックスまたはダイアログ ボックスのテーブルのサイズを変更し、テーブルを移動して、スペースを広げることができます。これにより、作業したり、より多くの情報が表示されます。

5. 親テーブルのキーフィールドを、子テーブルの対応するキーフィールドへドラッグします。
2 つのキーフィールド間で、テーブル間の関連付けを示す矢印が表示されます。

親および子テーブルは、子テーブルキーフィールドのインデックスを使用して関連付けられます。詳細については、「子テーブルインデックス」見開きページを参照してください。

- さらにテーブルの関連付けを追加する場合は、最初の一組のテーブルと同様、キーフィールドからキーフィールドヘドラッグすることでテーブルを関連付けていきます。

追加する関連付けは、それぞれ親テーブルへの直接リンクまたは間接リンクを作成する必要があります。

メモ

2つのテーブルの各インスタンスに関連付けを行えるのは1回のみです。既に関連付けた一組のテーブルに対して、さらに関連付けを行おうとすると、その操作は拒否されエラーメッセージが表示されます。**テーブルの追加** ボタンをクリックして、対象とする適切なテーブルを選択すると、そのテーブルのインスタンスを新たに追加できます。

詳細については、「複数のキーフィールドを個別に使用」 ページ 1015を参照してください。

- (省略可能) 個別の関係またはテーブルを **関係** ダイアログボックスから削除するには、次の手順を実行します。

- 関係を削除するには - キーフィールド矢印を右クリックし、**削除** を選択します
- テーブルを削除するには、テーブルの本体を右クリックして、**テーブルの削除**

] をクリックします。メモ

テーブルに既存の関係がある場合は、最初に関係を削除する必要があります。

- 完了**] をクリックして **関係** ダイアログ ボックスを閉じます。

関連付けたテーブルにあるフィールドを組み合わせたデータに対し、アクセスおよび分析を行うことができるようになります。すべてのフィールドが単独テーブルに存在するように扱えます。

関係] ダイアログ ボックス オプション

次の表は、**関係** ダイアログ ボックスのオプションの詳細を示します。

オプション - 関係ダイアログボックス	説明
テーブルの追加	テーブルの追加 ダイアログ ボックスを開きます。
テーブルの追加 ダイアログボックス	関係に含めるテーブルを指定します。
キーフィールド矢印	共通キーフィールドを指定し、各テーブルペアを関連付けます。 <ul style="list-style-type: none"> 共通キーフィールドを選択するには、キーフィールドをキーフィールドにドラッグします。

オプション - 関係ダイアログボックス	説明
	<ul style="list-style-type: none"> ○ キーフィールド矢印が配置されたら、それをクリックし、関係の編集]を選択して、共通キーフィールドを変更できます。 <p>キーフィールドのガイドライン:</p> <ul style="list-style-type: none"> ○ データ型 - キーフィールドはどのようなデータ型でもかまいません。各テーブルペアでは、キーフィールドに同じデータ型が必要です。 ○ 日付時刻のサブタイプ - 日付時刻のサブタイプ(日付、日付時刻および時刻)の関連付け対象は、同じサブタイプのみです ○ 長さ - キーフィールドの長さはテーブル間で同一にしておくことをお勧めします。 ○ 名前と開始位置 - キーフィールドの名前と開始位置はテーブル間で異なってもかまいませんが、同じデータ要素になっている必要があります。 ○ 複数のキーフィールド - 必要に応じて、1つのテーブルに2つ以上の共通キーフィールドを含めることができます。詳細については、「複数のキーフィールドの使用」ページ 1013を参照してください。
テーブルを整列する	(省略可能) 関係]ダイアログボックスの作業領域を右クリックして テーブルの整列]を選択し、テーブルの配置やキーフィールドの矢印を整理してください。
完了	<p>処理を実行します。</p> <p>関連付けたテーブルにあるフィールドを組み合わせさせたデータに対し、アクセスおよび分析を行うことができるようになります。すべてのフィールドが単独テーブルに存在するかのよう扱えます。</p> <p>親テーブルからアクセスする場合、Analyticsの各種ダイアログボックスで見られる対象テーブル]ドロップダウンリストおよびビルダー]で、関連付けられたテーブルを選択し、フィールドを選択して、分析または処理できます。</p>

子テーブル インデックス

親および子テーブルは、子テーブルキーフィールドのインデックスを使用して関連付けられます。

インデックスが存在しない場合	子テーブルキーフィールドにインデックスが存在しない場合、親テーブルと子テーブルを関連付けるときに、Analyticsが自動的に作成します。
特にインデックス名を付ける場合	<p>特にAnalyticsによって自動作成された子テーブルインデックスに名前を付ける場合:</p> <ol style="list-style-type: none"> a. 右クリックして、親テーブルのキーフィールドを子テーブルの[]にドラッグします。 b. 名前付きインデックスの関連付け]を選択します。 <p>名前付きインデックスを使用した関連付けは、インデックスが存在する場合は無効です。</p> <ol style="list-style-type: none"> c. インデックスの名前を指定し、必要であれば、デフォルト (Analytics プロジェクトを含むフォルダー) 以外の作成先も指定します。 d. OK]をクリックします。
複数のインデックスが存在する場合	子テーブルでキーフィールドに2つ以上のインデックスが存在する場合、それらの適切なインデックスのリストが表示されます。適切なインデックスを選択して OK]をクリックします。

(省略可能) 子テーブルフィールドを親ビューに追加します

子テーブルのフィールドを親テーブルのビューに追加する必要はありませんが、追加しておけば関連データがわかりやすくなります。

1. 親テーブルのビュー領域内で右クリックしてから **列の追加**]を選択します。
2. **対象テーブル**]ドロップダウンリストから子テーブルを選択します。
使用可能なフィールド]リストの名前フィールドには、子テーブルのフィールドが「子テーブル名.フィールド名」という書式で表示されます。
3. 子テーブルのフィールドリストから、親ビューに追加するフィールドを1つまたは複数選択します。
親ビューに子テーブルのフィールドが表示される順序は、フィールドを選択した順序どおりです。
4. さらに別の子テーブルのフィールドを表示する場合は、追加する子テーブルを **対象テーブル**]ドロップダウンリストから選択し、その子テーブルのフィールドを選択して親ビューに追加します。
5. **OK**]をクリックします。

子テーブルのフィールドが親ビューに追加されます。Analytics では、親テーブルの不一致レコードに対する子テーブルのフィールドの欠落した値には、空白かゼロを挿入します。

関係の編集

テーブルの関係は、設定後も編集することができます。関連付けられるテーブルの追加または削除、キーフィールドの変更および式の作成が可能です。関係全体を削除することもできます。

1. 親テーブルを開いて、**データ > 関連付け**の順に選択します。
2. **関係**]ダイアログボックスで、以下のいずれかを行います。

<p>関係全体を削除する</p>	<p>2つのテーブルを接続する矢印を右クリックし 削除]を選択します。</p> <p>関係の削除対象である子テーブルのフィールドが、現在の親ビューまたはアクティブな演算フィールドで参照されている場合は、その関係を削除することはできません。関係を削除する前に、ビューからそのフィールドを削除する、または演算フィールドを削除するか非アクティブにする必要があります。演算フィールドは、別のビューに切り替える、あるいは親テーブルを一度閉じて開き直すことで非アクティブにすることができます。</p> <p>関係に、親テーブルへ間接的にリンクする別の子テーブルがある場合も、その関係を削除することはできません。まず、ほかの子テーブルへの関係を削除しておく必要があります。</p>
<p>テーブルから関係を削除する</p>	<p>テーブルの本体を右クリックして、テーブルの削除]をクリックします。</p> <p>先にテーブルの関係(テーブルを接続する矢印)を削除しておかないとテーブルは削除できません。親テーブルを削除することはできません。関係を持たないテーブルは、関係]ダイアログボックスを閉じたときに、自動的に削除されます。</p>
<p>テーブルを関係に追加する</p>	<p>テーブルの追加]をクリックし、1つ以上のテーブルを選択して 追加]をクリックしたら 閉じる]をクリックします。通常の方法で新しいテーブルへの関係を作成します。</p>
<p>2つのテーブル間の関係を編集する</p>	<p>2つのテーブルを接続する矢印を右クリックし 関係の編集]を選択するか、矢印をダブルクリックします。</p> <p>親のキーフィールドや子のキーフィールドを変更したり、子テーブルに対するインデックが複数存在する場合には別のインデックスを選択することができます。 OK]をクリックして、変更を保存します。</p> <p>メモ 親テーブルを閉じて開き直すと、関係がリフレッシュされ変更が反映されます。</p>
<p>式の作成</p>	<p>親テーブルを右クリックして 式の新規作成]を選択し、式ビルダー]を開きます。</p> <p>親テーブルのフィールドおよび関連付けられている子テーブルのフィールドを使用して、式を作成することができます。式を作成できるのは、親テーブルからのみです。</p>
<p>テーブルと関係の矢印を自動的に配置する</p>	<p>作業領域を右クリックし、テーブルの配置]を選択します。</p>

3. **完了**]をクリックして **関係**]ダイアログボックスを閉じます。

テーブルの関連付けの構築方法

テーブル間の関連付けは階層型になっています。1つの親テーブルが階層の最上位にあり、複数の子テーブルと関連付けることができます。

テーブルが関連付けられると、それらのテーブルのフィールドは単独のテーブルに存在しているかのように扱うことができ、任意に組み合わせてデータにアクセスし分析することが可能です。

テーブルを関連付けるための一般ガイドライン

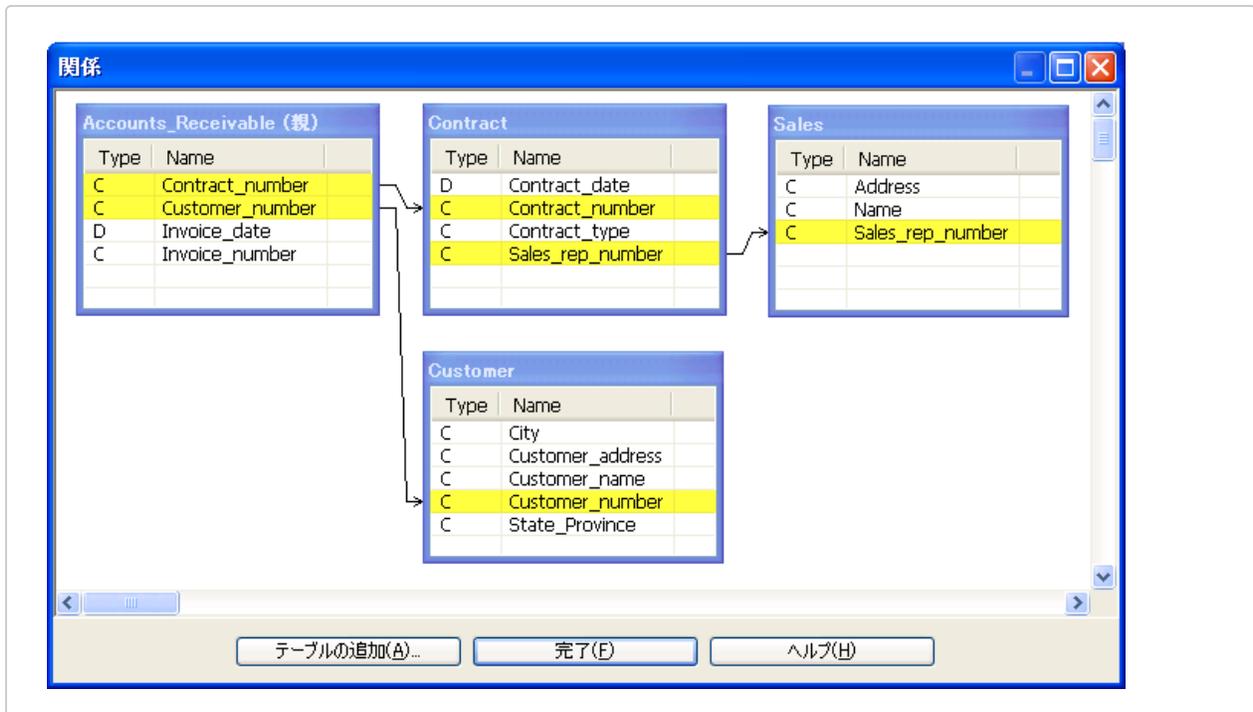
- 使用できる親テーブルは1つのみです。
- 子テーブルは、自身の子テーブルに関連付けることができます。これは親テーブルから見て孫テーブル、さらにひ孫テーブルといった関係になります。
- 親テーブルを含め、最大 18 個のテーブルを互いに関連付けることができます。

関係] ダイアログボックスはグラフィカルな作業領域を提供しているので、複数の関係を簡単に管理することができます。

例

下図では、テーブルとキーフィールドが次のように関連付けられています。

親テーブル	共通キーフィールド	子テーブル	共通キーフィールド	孫テーブル
売掛金	Contract_number	Contract	Sales_rep_number	営業部
	Customer_number	Customer		



テーブルを関連付けるための詳細ガイドライン

特に多くのテーブルを伴う関連付けの場合、データを希望どおりに表示させる関係を構築するには、ある程度の計画が必要です。以下のガイドラインを参考にして関係を構築してください。

キーフィールド間の一致

各テーブルに共通するキーフィールド、そのキーフィールドに含まれる値のデータ型、および生じる可能性が高い一致について慎重に検討してください。キーフィールド間の一致の完全性と正確度は、その後の情報の検査や分析の品質に影響を与えます。

親子関係を元に戻す

1つのテーブルともう1つのテーブルで親子の関係を作成したときの影響、およびその2つのテーブルの関係を逆転させた場合に結果がどのように異なるかについて考慮してください。

Analytics における関連付けのタイプは多対一であることを覚えておいてください。単独または同一の親キーの値は、子キーの値で最初に見つかった一致と関連付けられます。子キーの値の2番目以降の一致、およびそれらを含むレコードは無視されます。

子テーブルに複数の妥当なキー値が存在する場合は、子テーブルを親にする方が、完全な結果が得られる可能性があります。これは現在の親テーブルに妥当な重複が含まれていないことも前提とします。

両方のテーブルに妥当な重複が含まれる場合は、多対多結合を使用して使用してテーブルを結合するのがよい方法でしょう。

関係階層でのテーブルの位置の効果

関連付けの階層の各レベルで、子テーブルは階層上の1つ低いレベルにあるテーブルに対して親となることができます。

親と子の間では、キーの値の一致が多対一であるため、テーブルが階層上の下位へ行くにつれて、そのデータの多くが最後の関連付けから省かれる可能性が高くなります。このような多対一による一致で起こり得る累積的影響は、共通キーフィールドのすべての値において一対一の一致が存在している場合は問題になりません。

テーブル内の全レコードを必ず関連付けに含めるようにするには、そのテーブルを親テーブルにします。

中間テーブルの使用

共通キーフィールドがない2つのテーブルを関連付けたい場合は、1つ以上の中間テーブルを使用して関連付けを構築できる場合があります。

上の図では、**Contract** テーブルは中間テーブルとして機能し、**Accounts Receivable** および **Sales** テーブルを間接的に関連付けます。中間テーブルにあるデータ自体は必要な場合もあれば不要な場合もあります。

基本関係の関連付けのバリエーション

基本的な親子関係以外に、状況によって以下の関係のタイプが可能です。

<p>2つのテーブルの各インスタンスに関連付けを行えるのは1回のみです。</p>	<p>既に関連付けた一組のテーブルに対して、さらに関連付けを行おうとすると、その操作は拒否され次のメッセージが表示されます。</p> <p>"これらのファイルのいずれかは既に関係の一部です。別の関連付けを作成するには、そのファイルのインスタンスをもう1つ追加します。"</p> <p>関係] ダイアログボックスの テーブルの追加] ボタンをクリックし、適切なテーブルを選択してそのテーブルのインスタンスを追加できます。テーブルのインスタンスは、元のテーブル名の末尾に増分数字を付けた名前か、あるいは指定した名前前で追加されます。</p> <p>この代わりに、ナビゲーターで適切なテーブルレイアウトのコピーを作成し、そのコピーを 関係] ダイアログボックスにコピーすることもできます。</p>
<p>複数のキーフィールドのテーブルの関連付け</p>	<ul style="list-style-type: none">○ キーフィールドが連結される場合、複数のキーフィールドの組み合わせを使用してテーブルを関連付けることができます。 <p>詳細については、"複数のキーフィールドを組み合わせで使用" ページ 1013を参照してください。</p> <ul style="list-style-type: none">○ 親テーブルは、同じ子テーブルの別々(2つ以上)のインスタンスに関連付けることができます。次の間の関係があります。 <ul style="list-style-type: none">• 同じ親キーフィールドと2つの異なる子キーフィールド• 2つの異なる親キーフィールドと同じ子キーフィールド

	詳細については、"複数のキーフィールドを個別に使用" ページ 1015を参照してください。
テーブルの自己関連付け	テーブルは自身の別のインスタンスと関連付けることができます。

複数のキーフィールドの使用

2つの状況では、複数の共通キーフィールドを使用して、テーブルを正確に結合するか関連付ける必要があります。

使用...	処理するとき...	例
複数のキーフィールドの組み合わせ	1つの共通キーフィールドのみの値では一意性が保証されず、2つのテーブルを正確に結合または関連付けることができません。	業者 ID フィールドとロケーションフィールドの両方を使用して、2つのテーブルを正確に結合または関連付ける必要があります。
個別の複数のキーフィールド	2つのテーブルを結合または関連付けるために必要な値が、結合または関連付け対象のテーブルのどちらか一方で2つ(以上)のキーフィールドに分かれています。	Name でテーブルを結合するか関連付けます。主または親テーブルには単一の Name フィールドがあります。ただし、名前は副または子テーブルの2つのフィールドにいずれかで発生します。

複数のキーフィールドを組み合わせて使用

1つの共通キーフィールドのみの値では一意性が保証されず、2つのテーブルを正確に結合または関連付けることができない場合は、複数の共通キーフィールドを組み合わせて使用します。

例

共通キーフィールドとして `Vendor_ID` を使用して、2つのテーブルを結合するか、関連付けます。ただし、一部の業者には同じ業者 ID の複数のロケーションがあります。

この例では、業者 A4538 にはバンクーバー、リッチモンド、コキットラムにロケーションがあります。

1つのキーフィールド

`Vendor_ID` のみを使用してテーブルを結合または関連付けようとすると、副テーブルまたは子テーブルのレコードのうち、最初に一致した業者の場所以外のレコードは、結合されるテーブル(多対一結合)または関連付けられるテーブルには含まれないので、テーブル間による場所の一致に誤りが生じます。

主/親 テーブル

Location	Vendor_ID
Vancouver	A4538
Richmond	A4538
Coquitlam	A4538

副/子 テーブル

Vendor_ID	Location
A4538	Vancouver
A4538	Richmond
A4538	Coquitlam



複数のキーフィールドの組み合わせ

製造元の場所をすべて出力し、かつ場所を正確に一致させるには、両方のテーブルで、製造元 ID (Vendor_ID) と場所 (Location) の両フィールドをキーフィールドとして使用する必要があります。複数のキーフィールドを組み合わせれば、各フィールドの値によって1つの一意な値が形成され、これを使用して2つのテーブル間でレコードを確実に一致させることができます。

主/親 テーブル

Location	Vendor_ID
Vancouver	A4538
Richmond	A4538
Coquitlam	A4538

副/子 テーブル

Vendor_ID	Location
A4538	Vancouver
A4538	Richmond
A4538	Coquitlam

テーブルの結合時、複数のキーフィールドを組み合わせ指定する

テーブルを結合するときには、次の方法を使用して、複数のキーフィールドを組み合わせ指定できます。

- **結合** ダイアログボックスで複数のキーフィールドを選択します。
- 各テーブルで、2つ以上のキーフィールドを連結(加算)する演算フィールドを作成し、その演算フィールドを使用してテーブルを結合します。詳細については、「キーフィールドの連結」ページ 1020を参照してください。

結合 ダイアログボックスで複数のキーフィールドを選択します。

結合 ダイアログボックスで各テーブルの複数のキーを選択すると、次の条件が適用されます。

データ構造	1つのキーフィールドの使用時に適用される、データ構造およびデータ形式の要件は、複数のキーフィールドの使用時に各テーブルで対応させるキーフィールドにも適用します。詳細については、「データ構造およびデータ書式の要件」ページ 918を参照してください。
データ型	1つのテーブル内で、複数キーフィールドのデータが異なってもかまいません(氏名の苗字、名前、および誕生日など)。
並べ替え順序	複数のキーフィールドを選択すると、出力テーブルでは並べ替え順序が入れ子になります。ただし、これは結合の実行で主テーブルが 事前に並べ替えられている (主テーブルをあらかじめ並べ替える]オプションを選択)ことを前提とします。キーフィールドを選択する順序が、入れ子の並べ替え順序の優先度を示します。

テーブルの関連付け時、複数のキーフィールドを組み合わせ指定する

テーブルを関連付けるときには、次の方法を使用して、複数のキーフィールドを組み合わせ指定できます。

- 各テーブルで、2つ以上のキーフィールドを連結(加算)する演算フィールドを作成し、その演算フィールドを使用してテーブルを関連付けます。詳細については、"キーフィールドの連結" ページ 1020を参照してください。
- 各テーブルで、複数のキーフィールドのデータを含めることができる長さのフィールドを新たに定義し、その新しいフィールドを使用してテーブルを関連付けます。詳細については、"物理フィールドを定義する" ページ 790を参照してください。

メモ

結合とは異なり、テーブルの関連付けを行う場合、一組みのテーブルに対して選択できるキーフィールドは1つのみです。このため、複数のキーフィールドを組み合わせて使用するには、上のいずれかの方法を用いる必要があります。

複数のキーフィールドにデータを含む新しいフィールドの定義

複数のキーフィールドにデータを含む新しいフィールドを定義するときには、次の条件が適用されます。

データ構造	1つのキーフィールドの使用時に適用される、データ構造およびデータ形式の要件は、複数のキーフィールドを含む新しく作成されたフィールドにも適用します。詳細については、"データ構造およびデータ書式の要件" ページ 918を参照してください。
フィールドの隣接	この方法は、その複数のキーフィールドが各テーブルで隣接している場合にのみ機能します。フィールドが隣接されるようにするには、対象のフィールドを必要な順序で選択し、新しいテーブルに抽出することで実現できます。
データ型	複数のキーフィールド全体を含めることができる新しいフィールドのデータ型は、ソースデータと対応しています。 複数のキーフィールドのデータ型がそれぞれ異なる場合は、テーブルの関連付けに使用する目的のため、それらを文字フィールドとする新しいフィールドを作成できます。

複数のキーフィールドを個別に使用

2つのテーブルを正確に結合または関連付けるために必要な値が、結合または関連付け対象のテーブルのどちらか一方で2つ(以上)のキーフィールドに含まれている場合、それらの複数のキーフィールドを個別に使用する必要があります。

複数のキーフィールドのフィールドごとに、結合作業を1回ずつ実行するか、関連付けを1つずつ設定し、完全なデータセットがある統合された最終的な結果を生成します。

テーブルの結合または関連付けで複数のキーフィールドのうち1つしか使用しなかった場合は、その結果のデータは不完全なものになります。

例 1: 副または子テーブルの2つのキーフィールド

名前を用いて2つのテーブルを結合または関連付けます。主または親テーブルには**Name**フィールドがあります。ただし、副テーブルまたは子テーブルには**Name_1** および **Name_2** という別々の名前フィールドがあります。副テーブルまたは子テーブルで、一致の対象となる名前は2つの名前フィールドのうちどちらかに存在します。

テーブルの結合

名前の間ですべての可能な一致を取り込むには、2つの連続結合を実行し、副テーブルのキーフィールドのいずれかのみを使用して各結合を行う必要があります。1番目に行った結合の出力テーブルを、2番目に行う結合の主テーブルとして使用します。

各結合を実行する場合、一致および不一致のどちらの主レコード(すべての主レコード)も含める結合タイプを選択する必要があります。このため、処理のどの時点でも不一致の主レコードを失いません。

メモ

以下の図は、結合されるテーブルのキーフィールドのみを示します。通常、テーブルにはキーフィールド以外のデータも含まれています。

最初の結合(主キー Name 副キー Name_1)

主テーブル	副テーブル
Name	Name_1 Name_2
Ann Wilson	Ann Wilson
John Smith	John Smith
Robert Brown	Robert Brown

2番目の結合(主キー Name 副キー Name_2)

主テーブル(最初の結合の結果)	副テーブル
Name_1 Name	Name_1 Name_2
Ann Wilson	Ann Wilson
John Smith	John Smith
Robert Brown	Robert Brown

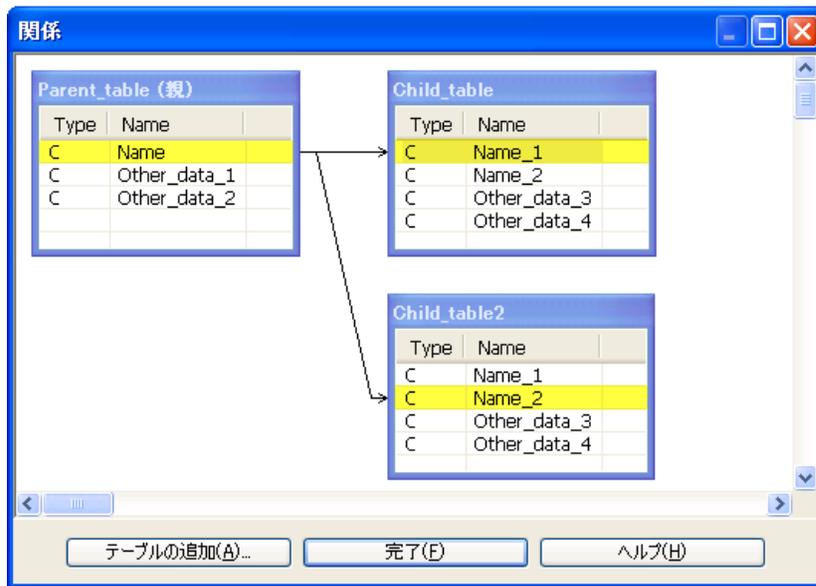
最終結果

Name_1	Name	Name_2
Ann Wilson	Ann Wilson	
John Smith	John Smith	
	Robert Brown	Robert Brown

テーブルの関連付け

名前の間ですべての可能な結合を取り込むには、親キーフィールドと2番目の子キーフィールドの間で追加の関係の子テーブルの追加インスタンスを追加する必要があります。

関係]ダイアログボックスの **テーブルの追加]** ボタンをクリックし、適切なテーブルを選択して子テーブルのインスタンスを追加します。



最終結果

Name_1	Name	Name_2
Ann Wilson	Ann Wilson	
John Smith	John Smith	
	Robert Brown	Robert Brown

例 2: 主または親テーブルの 2 つのキー フィールド

税フィルター ID 番号を用いて 2 つのテーブルを結合または関連付けます。主または親テーブルには税還付金上方が含まれ、副または子テーブルは税フィルター ID 番号を社会保障番号にマッピングします。

主テーブルまたは親テーブルには、主申告者(本人)の ID 番号を含む ID フィールドがあり、副申告者(配偶者)が存在する場合は、その ID 番号を含む **Secondary_ID** フィールドがあります。副テーブルまたは子テーブルには、すべての人の ID 番号を含む ID フィールドがあります。社会保障番号が主申告者と副申告者の両方に結び付けられた、結合データまたは関連付けデータを生成します。

テーブルの結合

社会保障番号を主および副税フィルターの両方に関連付けるには、2 つの連続結合を実行し、主テーブルのキーフィールドのいずれかのみを使用して各結合を行う必要があります。1 番目に行った結合の出力テーブルを、2 番目に行う結合の主テーブルとして使用します。

各結合を実行する場合、一致および不一致のどちらの主レコード(すべての主レコード)も含める結合タイプを選択する必要があります。このため、処理のどの時点でも不一致の主レコードを失いません。

メモ

以下の図は、結合されるテーブルのキーフィールドのみを示します。通常、テーブルにはキーフィールド以外のデータも含まれています。

最初の結合(主キー ID 副キー ID)

主テーブル

Secondary_ID	ID
M-9022385	A-2984745
	G-8142438
B-9817633	T-1129374

副テーブル

ID	SSN
A-2984745	555-44-3322
B-9817633	999-33-7744
G-8142438	777-55-1199
M-9022385	111-66-5588
T-1129374	333-22-0066

2 番目の結合(主キー Secondary_ID 副キー ID)

主テーブル(最初の結合の結果)

SSN	ID	Secondary_ID
555-44-3322	A-2984745	M-9022385
777-55-1199	G-8142438	
333-22-0066	T-1129374	B-9817633

副テーブル

ID	SSN
A-2984745	555-44-3322
B-9817633	999-33-7744
G-8142438	777-55-1199
M-9022385	111-66-5588
T-1129374	333-22-0066

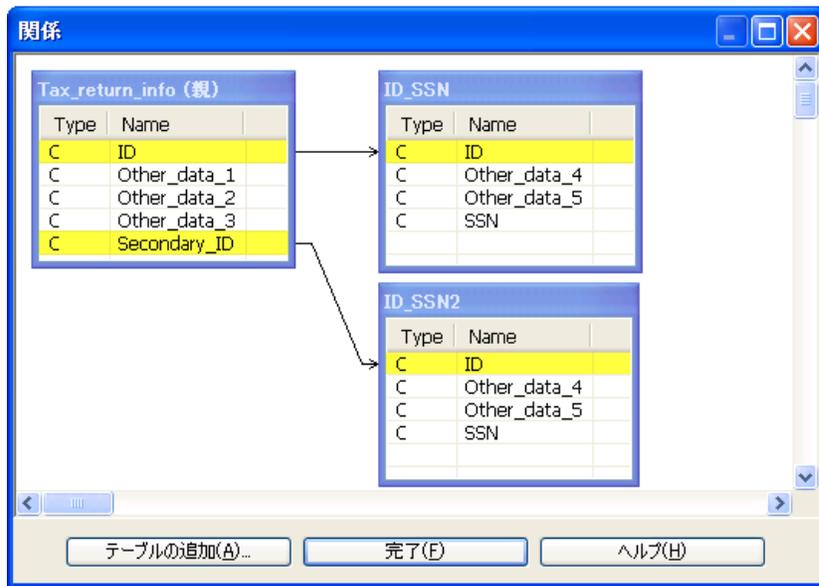
最終結果

SSN	ID	Secondary_ID	SSN_2
555-44-3322	A-2984745	M-9022385	111-66-5588
777-55-1199	G-8142438		
333-22-0066	T-1129374	B-9817633	999-33-7744

テーブルの関連付け

社会保障番号を主および副税フィルターの両方に関連付けるには、2 番目の親キーフィールドと子キーフィールドの間の子テーブルの追加インスタンスを追加する必要があります。

関係]ダイアログボックスの **テーブルの追加]** ボタンをクリックし、適切なテーブルを選択して子テーブルのインスタンスを追加します。



最終結果

SSN	ID	Secondary_ID	SSN_2
555-44-3322	A-2984745	M-9022385	111-66-5588
777-55-1199	G-8142438		
333-22-0066	T-1129374	B-9817633	999-33-7744

キーフィールドの連結

単独のキーフィールドでは一意性が保証されず、2つのテーブルを正確に関連付けることができない場合は、2つ以上のキーフィールドを連結する演算フィールドを各テーブルで作成し、その演算フィールドを使用してテーブルを関連付けることができます。

この方法は、テーブルを結合する場合にも使用できますが、これが必ずしも必要というわけではありません。テーブルの結合では1つのテーブルで複数のキーフィールドが選択できるからです。

単独のキーフィールドと同様に、連結されたキーフィールドも、関連付けられる2つのテーブルで同一のデータ構造とデータ形式を持っている必要があります。

メモ

連結できるのは文字キーのみであるため、連結前に Analytics 関数を使用して文字以外のデータを文字データとして変換することができます。詳細については、「フィールドの一致」ページ 921 を参照してください。

1. 親テーブルを開いて、**編集 > テーブルレイアウト**]の順に選択します。

2. **新しい式の追加** ]をクリックします。
3. 連結するキーフィールド用の名前を **名前**]に入力します。
4. **f(x)**  をクリックして**式ビルダー**を開きます。
5. 2つ以上のキーフィールドと加算演算子 (+) を使用して式を作成します。

例: `vendor_ID + location_code`

6. **OK**]をクリックします。

"式の型の不一致" エラーになった場合、キーフィールドの1つ以上が文字のキーフィールドでない可能性があります。

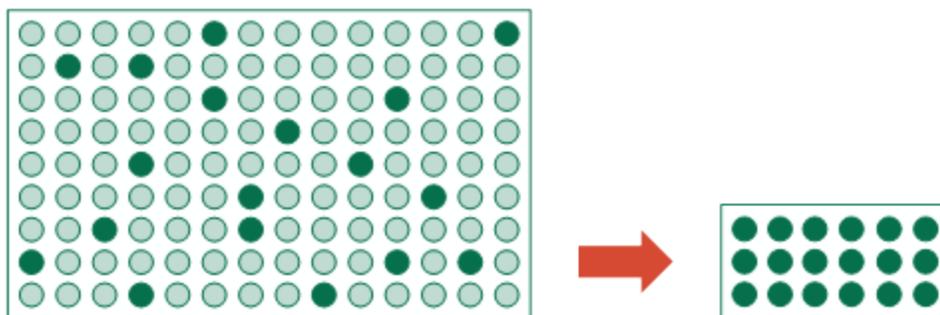
7. **入力を受け入れる** ] をクリックし、**閉じる** ] ボタンをクリックして **テーブルレイアウト**] ダイアログボックスを終了します。
8. 子テーブルを開き、同じ手順を繰り返して同一の連結キーフィールドを子テーブルに追加します。
9. 連結されたキーフィールドを使用して2つのテーブルを関連付けます。

データのサンプリング

取引の勘定またはクラスにおける、規定された統制からの逸脱率、または金額虚偽表示の合計金額を発見する必要があります。ところが、データセット内のすべてのレコードを検査する時間または予算がない場合があります。

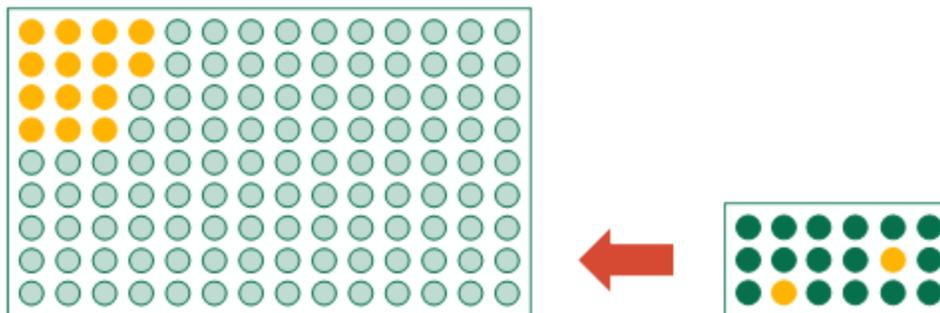
Analytics を使用して、**サンプル**という統計的に有効なデータのサブセットを抽出し、代わりに大幅に小さいデータのセットを分析できます。

サンプルの抽出



より小さいデータセットの分析結果をデータの母集団全体に対して**適用/推定**することができます。これにより、全体の逸脱率や全体の虚偽表示金額について、推定値が作成されます。

結果の推定



サンプル選択と推定で使用される統計式により、「推定された結果が、実際にすべてのレコードを検査した場合に得られる結果に近い」という妥当かつ測定可能な信頼度が保証されます。

メモ

本ガイドに記載されているサンプリングに関する情報は、Analytics におけるサンプリングの理論についてよく理解されているユーザーへの支援を目的に提供しています。監査サンプリング理論は複雑であるため、ここでは詳しく説明していません。

監査サンプリングの詳細については、AICPA の[監査ガイド：監査サンプリング](#)などのリソースを参照してください。

サンプリングの種類

Analytics には、次の3つのサンプリングの種類があります。

- レコード サンプリング(属性サンプリング)
- 金額単位サンプリング
- 従来の変数サンプリング

選択するサンプリングの種類は、行う分析の性質とデータの性質によって決まります。

Analytics のサンプリングは統計サンプリングです

Analytics のサンプリングは統計サンプリングです。Analytics によって抽出されたサンプルは統計的に有効であるか、代表的です。それは、承諾された統計式を使用して計画、抽出、および評価されているためです。

式は確率分布に基づいています。レコード サンプリングと金額単位サンプリングはポアソン分布であり、従来の変数サンプリングは正規分布に基づいています。

どの種類のサンプリングを使用すればよいのでしょうか？

使用するサンプリングの種類に関するガイドラインを次の表に示します。

メモ

代表的でないランダムなレコード選択のみが必要な場合は、「ランダムに選択したレコードの生成」ページ 236を参照してください。代表的でない選択に基づいた結果を適用/推定しても、統計的有効性はありません。

サンプリングの種類	その使用に適する場面
"レコード サンプリング(属性サンプリング)" ページ 1035	<p>統制をテストしようとしている場合を考えます。</p> <p>レコード サンプリングは、規定した統制からの逸脱率を監査する場面に適していません。</p> <p>分析対象のレコードごとにOK/NGまたは合格/不合格を出力する場合には、レコード サンプリングを使用することをお勧めします。</p>
"金額単位サンプリング" ページ 1060	<p>金額の虚偽表示がないか取引の勘定またはクラスを分析する場合に、その財務データに次の特性があることが必要な場面：</p> <ul style="list-style-type: none"> ◦ 虚偽表示がまったくないか、少ない数の虚偽表示しかない。 たとえば、科目数の5%未満にしか虚偽表示がないなど。 ◦ 過少計上数より多くの過剰計上数があるように思われる。 ◦ 0ドルの科目がない。
"従来の変数サンプリング" ページ 1093	<p>金額の虚偽表示がないか取引の勘定またはクラスを分析する場合に、その財務データに次の特性があることが必要な場面：</p>

サンプリングの種類	その使用に適する場面
	<ul style="list-style-type: none"> 虚偽表示の数があまり多くないか、過少計上数より多い。 たとえば、5%以上の科目が虚偽表示である場合など。 過剰計上または過少計上が存在する。 0ドルの科目が存在する。

サンプリングには専門的な判断が必要

Analytics のサンプリング機能では、監査のサンプリングに関わる多数の計算およびプロセスが自動化されます。ただし、信頼性が高く、効果的なサンプリングには、複数の領域で専門的な判断を適用することも必要です。

メモ

信頼できる方法で監査サンプリングを実行するために必要な専門的な判断に精通していない場合は、監査サンプリング担当者または監査サンプリングスペシャリストの助言を得てから、本番環境で Analytics サンプリングを使用することをお勧めします。

領域	必要な判断
信頼度	サンプルが代表的であるという信頼性が必要とされる度合い(必要な信頼度)の評価
Materiality 金額精度	取引の勘定またはクラスの虚偽表示で許容されるレベルの決定
許容逸脱率	規定された統制からの許容される逸脱率の決定
選択方法	適切なサンプル選択方法の選択
評価方法	従来の変数サンプリングに対する適切な評価方法の選択

用語に関する単語

監査サンプリングに精通していない場合の支援として、このガイドにあるいくつかの用語が以下で非常に簡単に定義されています。

詳しく表示

メモ

これらは専門的な定義ではなく、監査および保証において専門的に使用されるより正確な定義を意図的に簡素化したものです。

統制	保証を提供する義務づけられたプロセス。 例:すべての伝票は管理者によって承認され、署名される必要があります。
逸脱、統制の逸脱	統制を遵守できないこと。 例:伝票が管理者の署名なく処理された。
誤った文	不正確な数。一般的には不正確な金額。
Materiality	何かが重大になる点。
重大な虚偽表示	問題になるほど十分に大きい不正確さ。 単一の数値または勘定全体を指すことがあります。たとえば、"この勘定には重大な虚偽表示があります"。
母集団	サンプルが抽出される、ファイルのレコードの全体のセット、または勘定または取引のクラスの金額全体。
プロジェクト	推定すること。観察された値に基づいて不明な値を外挿すること。
代表する	より大きいグループと同じ特性を持つこと。
サンプリング、サンプル	母集団の100%未満を選択する統計的に有効な処理。このサブセットは"サンプル"として知られています。
シード値	Analytics 乱数ジェネレーターを初期化する数値。ユーザーが指定するか、Analytics によってランダムに選択されます。
汚染	虚偽表示金額では、虚偽表示が表す簿価(記録された値)の割合です。 例:実際には\$180である\$200の簿価は\$20の虚偽表示があるため、10%の汚染があります。
許容可能	許容できること。許容度の境界内であること。
ユニバース	別の用語で「母集団」

Analytics の用語と業界用語

Analytics のサンプリング ダイアログ ボックスのさまざまなラベルは Analytics の用語を使用します。監査 サンプリングと関連付けられた用語を既に理解している場合は、用語に不一致があると思われる可能性があります。Analytics の用語を業界用語にマッピングするには、「監査 サンプリング用語」 ページ 1030を参照してください。

サンプルの選択方法

サンプルの選択方法は、サンプルに含めるレコードの選択に使用する特定の手法です。

レコード サンプルングと金額単位 サンプルングについては、Analytics は 3 つのサンプル選択方法をサポートします。

- 固定間隔
- セル
- ランダム

従来の変数 サンプルングについては、ランダム選択方法のみが可能です。

サンプル選択方法とサンプリングタイプ

サンプル選択方法とサンプリングタイプの違いを理解することが重要です。

サンプリングタイプは母集団に関する推計に達するために使用される全体的な統計方法を指します。

サンプル選択方法は、サンプルに含める母集団からレコードを抽出する方法を指します。

サンプリングの種類	使用可能なサンプル選択方法	詳細
レコード サンプルング	<ul style="list-style-type: none"> ◦ 固定間隔 ◦ セル ◦ ランダム 	サンプルに含まれるレコードが直接選択されます
金額単位 サンプルング	<ul style="list-style-type: none"> ◦ 固定間隔 ◦ セル ◦ ランダム 	サンプルに含まれるレコードは、選択した金額単位に対応するレコードです。
従来の変数 サンプルング	<ul style="list-style-type: none"> ◦ ランダム 	サンプルに含まれるレコードが直接選択されます

固定間隔選択方法

固定間隔選択方法では、初期金額単位またはレコードが選択され、すべての後続の選択は固定間隔または離れた距離です。たとえば、初期選択の後のすべての 5000 番目の金額単位またはすべての 20 番目のレコードです。

詳しく表示

固定間隔選択方法を使用するには、次の項目を指定します。

- サンプルサイズを生成するときに Analytics が生成する間隔値
- ゼロより大きく間隔値以下の開始数

開始数と間隔値は、サンプルに含まれるレコードを選択するために使用されます。

メモ

Analytics でランダムに開始番号を選択する場合は、開始番号 '0' を入力するか、開始番号を空欄にします。

例

62 が Analytics によって生成された間隔で、開始数として 17 を選択した場合、次の金額単位またはレコード数が選択されます。

- 17
- 79 (17+62)
- 141 (79+62)
- 203 (141+62)
- など

各選択は同じ距離または離れた固定間隔です。

金額単位 サンプリングでは、選択された実際のレコード数は、選択した金額単位に対応するレコード数です。詳細については、"金額単位 サンプリングによるレコードの選択方法" ページ 1061 を参照してください。

考慮事項

固定間隔選択方法を使用するときには、データのパターンに注意する必要があります。サンプルの抽出に固定間隔が使用されるため、データのパターンが、指定した間隔と一致するような場合には、結果のサンプルが代表的でない可能性があります。

たとえば、間隔 \$10,000 を使用して費用をサンプリングし、同じ費用カテゴリがファイルに 1 万ドル間隔で表示されます。これは、単一の費用カテゴリから発生するすべての選択されたレコードになります。このような状況が起こることはまれですが、可能性があることに留意してください。

セル選択方法

セル選択方法では、データセットが複数の均等なセルまたはグループに分割され、各セルから 1 つの金額単位または 1 つのレコードがランダムに選択されます。

詳しく表示

セル選択方法を使用するには、次の項目を指定します。

- サンプルサイズを生成するときに Analytics が生成する間隔値
- Analytics で乱数ジェネレーターを初期化するために使用されるシード値。

各セルのサイズは間隔値によって指定します。乱数ジェネレーターは、各セルから選択される金額単位またはレコード数を指定します。

メモ

Analytics でランダムにシード値を選択する場合は、シード値 '0' を入力するか、シード値を空白のままにします。

例

Analytics によって生成される間隔が62の場合、1つの金額単位または1つのレコード数が次のセルからランダムに選択されます。

- セル1(1 ~ 62)
- セル2(63 ~ 124)
- セル3(125 ~ 186)
- など

各選択はランダムな離れた距離ですが、セル内に制約されます。

金額単位サンプリングでは、選択された実際のレコード数は、選択した金額単位に対応するレコード数です。詳細については、"金額単位サンプリングによるレコードの選択方法" ページ 1061を参照してください。

シード値

シード値を指定する場合、任意の数字を指定することができます。シード値がそれぞれ一意であると、異なる乱数一式になります。同じシード値を指定すると、同じ乱数一式が生成されます。特定のサンプル選択を複製したい場合は、シード値を明示的に指定し、これを保存します。

考慮事項

固定選択方法と比較したセル選択方法の主な利点は、データのパターンに関連する問題を回避できることです。

金額単位サンプリングでは、2つの問題点があります。

- 金額は2つのセルの間の区切り点にまたがることのできるため、2回選択でき、固定間隔方法で生成されるサンプルよりも一貫性が低いサンプルになります。
- また、項目が大きくても最上層のカットオフ基準よりも小さい場合、抽出される可能性が若干減少してしまうことも示します。

ランダム選択方法

ランダム選択方法では、すべての金額単位またはレコードがデータセット全体から、または従来の変数サンプルを使用している場合は書く層からランダムに選択されます。

詳しく表示

メモ

Analytics を使用して結果 サンプルで検出された虚偽の表示を評価する場合は、金額単位 サンプルでランダム選択方法を使用しないでください。金額単位 サンプルの評価には、固定間隔またはセル選択方法を使用する必要があります。

乱数選択方法を使用するには、次の項目を指定します。

- Analytics によって計算されるサンプルサイズ。選択するサンプル数
- Analytics で乱数ジェネレーターを初期化するために使用されるシード値。
- 母集団サイズ。サンプルフィールドの絶対値またはデータセットのレコードの合計数

従来の変数サンプリングでは、サンプルサイズと母集団サイズは Analytics によって自動的に入力されず。

乱数ジェネレーターは、データセットから選択される金額単位またはレコード数を指定します。各選択は離れたランダム距離です。

メモ

Analytics でランダムにシード値を選択する場合は、シード値 '0' を入力するか、シード値を空白のままにします。

シード値

シード値を指定する場合、任意の数字を指定することができます。従来の変数サンプリングでは、シード値は 2,147,483,647 以下の正の数である必要があります。

シード値がそれぞれ一意であると、異なる乱数一式になります。同じシード値を指定すると、同じ乱数一式が生成されます。特定のサンプル選択を複製したい場合は、シード値を明示的に指定し、これを保存します。コマンド ログからシード値を取得することもできます。

考慮事項

大きい金額は金額単位サンプルから除外できます。

乱数選択方法では、各金額単位の選択の確率が等しくなり、結果のサンプルが均等に分布される保証はありません。結果として、選択した単位間の距離またはギャップが場合によっては大きくなる場合があります。大きい金額に関連付けられたすべての金額単位がギャップに当たる場合、金額はサンプルに含まれません。また、乱数選択方法を使用するときには使用できる最上層のカットオフはありません。

固定間隔とセル選択方法では、選択した単位が均等に分散されるか、相対的に均等に分散されることが保証されます。最上層のカットオフを使用できます。

金額は複数回金額単位サンプルに含まれる場合があります。

Analytics は同じ乱数を 2 回生成しませんが、近い連続する乱数が発生することがあります。

金額単位サンプリングでは、近い連続する乱数が選択されている近い連続する金額単位と等しくなり、関連付けられた金額が複数回選択されることがあります。

レコード サンプルングと従来の変数 サンプルングでは、各乱数が別のレコードと等しいため、同じ問題は発生しません。

ランダム数アルゴリズム

レコード サンプルングと金額単位 サンプルングでは、Analytics の乱数ジェネレーターには以下の2つのアルゴリズム オプションがあります。

- メルセンヌ・ツイスタ
- デフォルト Analytics アルゴリズム

メルセンヌ・ツイスタは、広く使用されている乱数アルゴリズムであり、デフォルトの Analytics アルゴリズムより優れた統計のプロパティを持っています。バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプルングの結果との後方互換性が必要な場合は、単純にデフォルトのアルゴリズムを使用してください。

従来の変数 サンプルングでは、メルセンヌ・ツイスタは選択できず、デフォルトの Analytics アルゴリズムが使用されます。

レコード数フィールドの追加

レコード数フィールドを、サンプルを抽出している Analytics テーブルに追加すると便利な場合があります。サンプルを抽出した後、ソース テーブルから選択された特定のレコード数がサンプルを含む出力テーブルに表示されます。

メモ

従来の変数 サンプルングを使用するときには、レコード番号フィールドが自動的に出力テーブルに含まれます。

方法の説明

1. ソーステーブルで式

```
RECNO( )
```

を使用する演算フィールドを作成します。

詳細については、「条件付き演算フィールドを定義する」ページ 806を参照してください。

2. データをサンプルングするときには、レコードではなく、フィールドで出力します。

計算されたレコード数フィールドをソース テーブルからのレコード数を保持する物理フィールドに変換するには、フィールドで出力する必要があります。

3. 指定する出力フィールドに計算されたレコード数フィールドを含めます。

監査 サンプルング用語

通常、Analytics サンプルング用語は、監査と保証で一般的に使用される用語とは異なります。以下のセクションは Analytics 用語と同等の業界用語の概要を示します。

一部の共通監査 サンプルング用語の簡易定義については"用語に関する単語" ページ 1023用語を参照してください。

Analytics のサンプルングのタイプの名前

Analytics には、次の3つのサンプルングの種類があります。

- レコード サンプルング
- 金額単位 サンプルング
- 従来の変数 サンプルング

以下の表は、これらのタイプのサンプルングで使用する他の用語の一覧を示します。

Analytics 用語	業界用語
金額単位 サンプルング	金額単位 サンプルング ドル単位 サンプルング 金額比例確率 累積金額単位 サンプルング
レコード サンプルング	属性 サンプルング 属性 サンプルング
従来の変数 サンプルング	従来の変数 サンプルング

金額単位 サンプルング用語

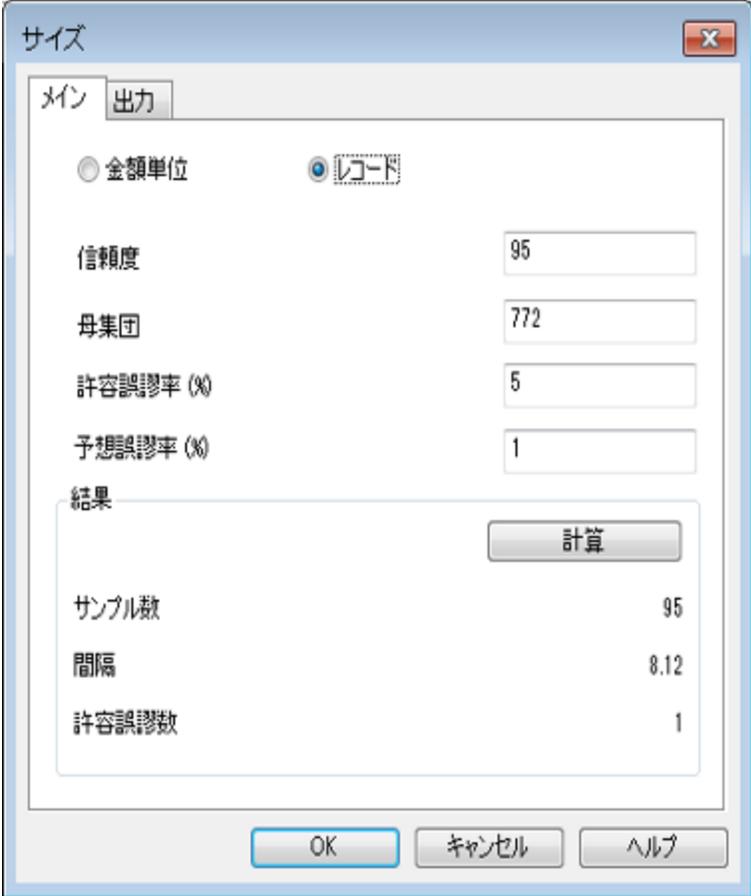
以下の表は、Analytics で使用される金額単位 サンプルング用語の一覧を示し、同等の業界用語を示します。

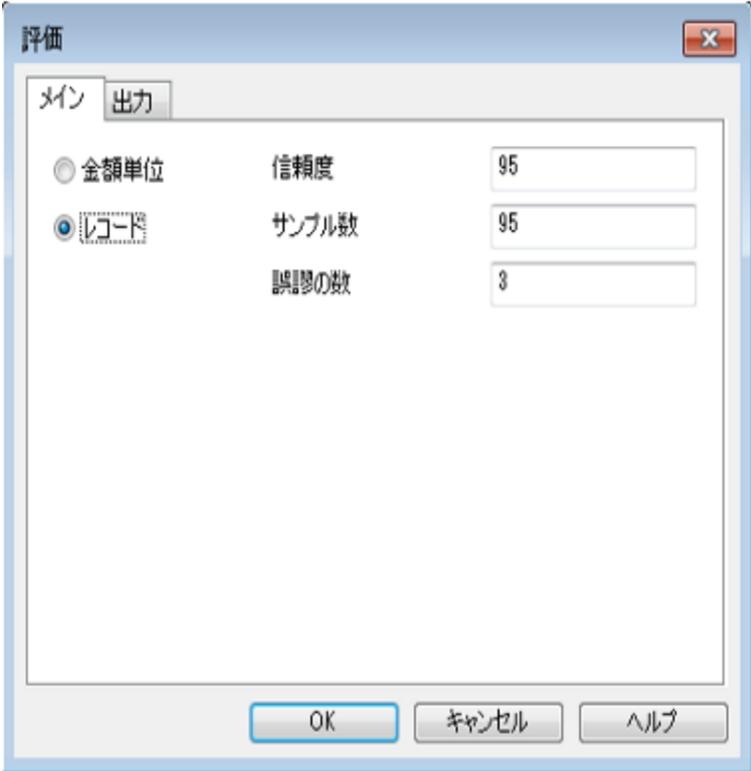
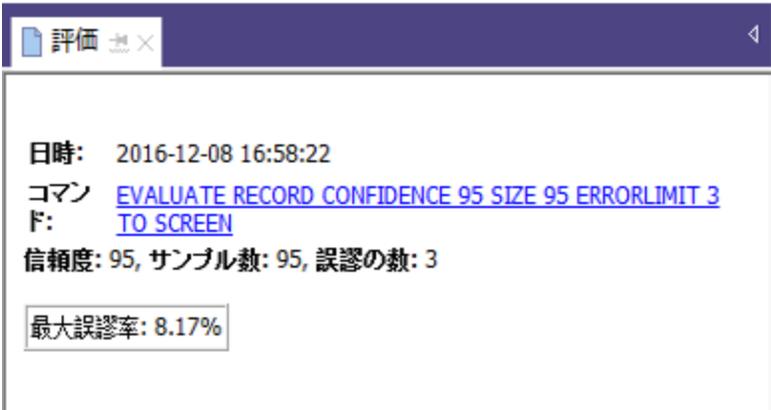
サイズ]ダイアログ ボックス	Analytics 用語	業界用語
	信頼度	信頼度 信頼レベル 信頼性
	母集団	母集団 母集団 サイズ
	Materiality	許容虚偽表示
	想定される合計誤謬数	想定される誤った文 推定誤った文 想定される母集団誤った文
	サンプルサイズ	サンプルサイズ
	間隔	サンプリング間隔
	最大許容汚染 (%) (Analytics 固有の用語)	
評価]ダイアログ ボックス		

サイズ]ダイアログ ボックス	Analytics 用語	業界用語																																								
	誤謬	誤った文																																								
結果の評価																																										
 <table border="1" data-bbox="219 1423 743 1780"> <thead> <tr> <th></th> <th>項目の計上額</th> <th>誤謬額</th> <th>推定誤謬額</th> <th>最大誤謬額</th> </tr> </thead> <tbody> <tr> <td>基本精度</td> <td></td> <td></td> <td></td> <td>18,850.00</td> </tr> <tr> <td></td> <td>1,839.33</td> <td>1,163.35</td> <td>3,974.12</td> <td>6,954.71</td> </tr> <tr> <td></td> <td>3,409.82</td> <td>900.00</td> <td>1,658.44</td> <td>2,570.58</td> </tr> <tr> <td></td> <td>392.41</td> <td>12.85</td> <td>205.76</td> <td>300.41</td> </tr> <tr> <td></td> <td>6,730.56</td> <td>27.00</td> <td>27.00</td> <td>27.00</td> </tr> <tr> <td></td> <td>206.69</td> <td>-3.00</td> <td>-91.20</td> <td>0.00</td> </tr> <tr> <td>合計</td> <td></td> <td></td> <td>5,774.12</td> <td>28,702.70</td> </tr> </tbody> </table>		項目の計上額	誤謬額	推定誤謬額	最大誤謬額	基本精度				18,850.00		1,839.33	1,163.35	3,974.12	6,954.71		3,409.82	900.00	1,658.44	2,570.58		392.41	12.85	205.76	300.41		6,730.56	27.00	27.00	27.00		206.69	-3.00	-91.20	0.00	合計			5,774.12	28,702.70	<p>基本精度</p> <p>最尤誤謬</p> <p>上限誤謬</p>	<p>基準となる精度 サンプリングリスクの基本的な許容誤謬</p> <p>想定された誤った文</p> <p>上限誤った文</p>
	項目の計上額	誤謬額	推定誤謬額	最大誤謬額																																						
基本精度				18,850.00																																						
	1,839.33	1,163.35	3,974.12	6,954.71																																						
	3,409.82	900.00	1,658.44	2,570.58																																						
	392.41	12.85	205.76	300.41																																						
	6,730.56	27.00	27.00	27.00																																						
	206.69	-3.00	-91.20	0.00																																						
合計			5,774.12	28,702.70																																						

レコード サンプルング用語

以下の表は、Analytics で使用されるレコード サンプルング用語の一覧を示し、同等の業界用語を示します。

サイズ]ダイアログ ボックス	Analytics 用語	業界用語
	<p>信頼度</p> <p>母集団</p> <p>上限誤差 (%)</p> <p>想定誤差率 (%)</p> <p>サンプルサイズ</p> <p>間隔</p> <p>許容誤差数 (Analytics 固有の用語)</p>	<p>信頼度 信頼レベル 信頼性</p> <p>母集団 母集団サイズ</p> <p>許容偏差率</p> <p>想定される母集団偏差率 推定母集団偏差率</p> <p>サンプルサイズ</p> <p>サンプリング間隔</p>
評価]ダイアログ ボックス		

サイズ] ダイアログ ボックス	Analytics 用語	業界用語
	誤謬数	偏差数
結果の評価		
	上限誤謬頻度	計算された上限偏差率

レコード サンプルング(属性サンプルング)

レコード サンプルングは、取引の勘定またはクラスで規定された統制からの逸脱率を推定するための統計サンプルング方法です。

サンプルングされたデータの分析がレコードごとにはいいいいえまたは合格/不合格を出力する場合には、レコード サンプルングを使用することをお勧めします。

ヒント

Analytics のレコード サンプルングのエンドツーエンド プロセスに関するすぐに使える導入については、"レコード サンプルングのチュートリアル" ページ 1037を参照してください。

機能の仕組み

レコード サンプルングにより、小さい母集団のサブセットを選択および分析できます。また、結果に基づいて、母集団全体の内部統制からの誤謬または逸脱率を推定できます。

次に、推定された率を許容可能と判断される率と比較し、統制に関する決定を行います。

レコード サンプルングは、この種の記述をサポートします。

- 規定された統制からの逸脱率が4.35%を超えない確率は95%です。これは許容可能な逸脱率の5.0%を下回ります。このため、規定された統制は効果的に機能しています

レコード サンプルング処理の概要

注意

有効なサンプルサイズの計算をスキップしないでください。

すぐにレコードのサンプルを抽出し、サンプルサイズで推定する場合は、分析結果の推定が無効になる確率が高く、最終結果に欠陥が生じます。

レコードのサンプルング処理には、以下の一般的な手順があります。

1. [必要なサンプル数を計算します。](#)
2. サンプルの選択方法を選択します。
 - [固定間隔](#)
 - [セル](#)
 - [ランダム](#)
3. [レコードのサンプルを描画する](#)
4. サンプルングされたデータで意図した監査手順を実行します。

5. サンプルされたデータで観察された統制逸脱率が母集団全体の許容できない逸脱率を表しているかどうかを[評価](#)します。

レコード サンプルングによるレコードの選択方法

レコード サンプルングでは Analytics テーブルからサンプルレコードを選択する次のプロセスを使用します。

- レコード サンプルングを指定する場合、サンプルング単位は一意のレコード番号を持つ個別のレコードです。特定のフィールドは指定しません。
- サンプルの選択方法の1つを使用して、Analytics はレコード番号の中からサンプルを選択します。選択されたレコードは、サンプルング出力テーブルに含まれます。

例

100 レコードあるテーブルでは、Analytics が次のレコード番号を選択する場合があります。

- 9
- 13
- 40
- 52
- 78
- 91
- 99

バイアスのないサンプルの選択

レコード サンプルングは、バイアスがかかっておらず、レコードに含まれる金額に基づいていません。サンプルの対象として選択される可能性はどのレコードについても等しくなります。金額を 1000 ドル含むレコード、250 ドル含むレコード、および 1 ドル含むレコードの選択されるチャンスは同等です。

つまり、指定したレコードが選択される確率は、そのレコードに含まれる金額とは無関係です。

考慮事項

レコード サンプルングは、はい/いいえ、または合格/失敗という結果になる結果をテストする統制で使用するのに適しています。統制テストでは、エラーの累計金額よりも、合計母集団でのエラー率に、より関心があります。

レコード サンプルングではレコードに含まれる金額は考慮されないため、高額取引はサンプルから除外されるチャンスが高くなります。

レコード サンプルングのチュートリアル

このチュートリアルでは、Analytics におけるレコード サンプルングのエンドツーエンド プロセスの概要を説明します。

予定所要時間 - 20 分

サマリー - Vouchers(伝票) テーブルから、レコードで構成されるサンプルを抽出し、そのサンプルから統制からの逸脱を検出します。サンプルからの検出結果に基づいて、テーブル全体の逸脱率を統計的に推定します。

この統計的推定値を使って、伝票統制プロセスが有効に運用されているかどうかを判定します。

主なタスク -レコード サンプルングを正しく実行するには、主なタスクとして以下の4つを実行する必要があります。



レコード サンプルングについて、オプションの使用方法は省略して、オプションなしでの使用方法のみを説明しているため、Analytics のレコード サンプルングの使用方法の基本をすばやく理解できます。

ヒント

サンプルング用語については、"用語に関する単語" ページ 1023に簡潔に定義されていますので参照してください。

レコード サンプルングのシナリオ

伝票統制プロセスのテスト

シナリオ

5000 レコードを超える Vouchers テーブルを検査するとします。伝票のサンプルのハードコピーを抽出して、システム入力と一致し、伝票統制プロセスが効果的に機能していることを確認しようとしています。

ハードコピーを確認し、次の点を確認します。

- 伝票が承認された
- 同じ人が伝票を作成して承認していない

どのように進めるか?

いくつかのハードコピーを抽出するか?どれを抽出するかをどのように決定するか?サンプルから検出した統制からの逸脱が伝票の母集団全体ではどのように分布しているか?

Analytics レコード サンプルングを使用して、これらの質問に対する回答を得ることができます。

このシナリオで使用される Analytics テーブル

このシナリオは Analytics に含まれる `ACL_Rockwood.ac1` サンプルデータファイルの `Vouchers` テーブルを使用します。

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1 有効なサンプルサイズの計算

メモ

本番環境では、有効なサンプルサイズを計算するために指定する値は、専門家の判断によって異なります。

1. `ACL_Rockwood.ac1` で、`Acquisitions_Payment` にある `Vouchers` テーブルを開きます。
2. `サンプルング > レコード/金額単位 サンプルング > サイズの計算` を選択します。
3. `レコード` オプションを選択します。
4. 以下の画面に表示されている入力値を指定し、`計算` をクリックしてサンプルサイズを計算します。

結果を確認した後、`OK` をクリックしてサンプルサイズ計算を確定するか、異なる値を指定して実験します(以下を参照)。

入力値の意味

信頼度	抽出するサンプルが母集団全体を表している信頼度が95% 必要です。 別の方法：サンプルを 100 回抽出した場合、95 回表し、5 回のみ表しません。
母集団	Vouchers テーブルのレコードの総数
上限誤謬 (%)	伝票の最大 2% は適切な承認がないことがあり、統制はまだ効果的であると見なします。
想定誤謬率 (%)	伝票の 1% が適切な承認がないと想定します。

結果の意味

サンプルサイズ	593 のハードコピー伝票を抽出します。
間隔	サンプル選択の間隔方法のいずれかを使用する場合、選択したレコードは以下のいずれかです。 <ul style="list-style-type: none"> 9 レコードごと

	<ul style="list-style-type: none"> 各 9 レコードのブロックからランダムに選択したレコード <p>メモ 8.93 は最大 9 に四捨五入されます。</p>
許容誤謬数	<p>メモ この数字は、統制の逸脱を評価するための代替方法であり、このレコード サンプルング チュートリアルでは使用しません。</p> <p>この数字の意味：</p> <ul style="list-style-type: none"> サンプルのハードコピー伝票を検査するときに、7 以上の伝票が統制に違反する場合、統制が非効果的だと見なすことができます。

詳細：異なる値を指定した実験

サイズ] ダイアログ ボックスの次のフィールドのいずれかで値を変更し、**計算]** をクリックして、結果がどのように変わるのかを観察します。1 度に 1 つの値のみを変更し、変更がどのように結果に影響するのかをわかりやすくします。

- 信頼度
- 上限誤謬 (%)
- 想定誤謬率 (%)

より厳密な要件により、サンプル サイズが大きくなります。より緩い要件により、サンプル サイズが小さくなります。

上の画面に合うように値をリセットし、**OK]** をクリックします。サンプル サイズ計算の結果で **サイズ]** タブを固定します。

2 レコードのサンプルを抽出する

1. **Vouchers** テーブルに戻ります。
2. **サンプリング > レコード/金額単位 サンプリング > サンプル]** を選択します。
3. **レコード]** オプションを選択します。

4. 以下の画面に表示されている値を指定し、[OK]をクリックしてレコードのサンプルを抽出します。

入力値の意味

セル	サンプルレコードを抽出するためにセル選択方法を使用しています。 セル選択方法では、同じサイズのセルまたはレコードのブロック内からランダムに選択されるレコード選択方法。
間隔	各セルのサイズは9レコードです。8.93は最大9に四捨五入されます。
シード値	Analyticsで乱数ジェネレーターを初期化するために使用されるシード値456654。任意のシード値を指定できます。 乱数ジェネレーターは、各セルから選択されるレコード数を指定します。
アルゴリズム	乱数ジェネレーターはメルセンヌ・ツイスタアルゴリズムを使用して、乱数を生成します。
変換先	Vouchers テーブルから抽出されたレコードのサンプルは、新しいテーブル Vouchers_sample へ出力されます。

メモ

Analytics は間隔を9に切り上げ、抽出された実際のレコード数は計算されたサンプルサイズ593より少し小さくなります。

3 サンプルで分析を実行する

このチュートリアルでは、以下を行うものとします。

1. `Vouchers_sample` テーブルに出段する伝票番号のハードコピーを抽出します。
2. 各伝票を検査し、適切な承認がない伝票を記録します。

4 母集団全体に対する分析結果の適用

1. [サンプリング>レコード/金額単位 サンプリング>評価]を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. [レコード]オプションを選択します。
3. 以下の画面に表示されている入力値を指定し、[OK]をクリックして分析結果を適用します。

項目	値
金額単位	<input type="radio"/>
信頼度	95
レコード	<input checked="" type="radio"/>
サンプル数	588
誤謬の数	5

入力値の意味

信頼度	サンプルサイズの計算時に指定したのと同じ信頼度。
サンプルサイズ	抽出したサンプルの実際のレコード数。つまり <code>Vouchers_sample</code> テーブルのレコード数。
誤謬数	ハードコピーを検査したときに、適切な承認がなかった伝票数。

推定/適用される分析結果の意味



上限誤謬頻度	<p>95% の信頼度で推定された伝票の母集団全体の最大逸脱率。</p> <p>別の方法: <code>Vouchers</code> テーブルの適切な承認がない伝票数が 1.79% または 95 伝票を超えないという 95% の確率があります。</p> <p>1.79% はサンプルサイズを計算したときに上限誤謬 (%) に指定した 2% より小さいため、伝票統制が有効に運用されていると結論付けることができます。</p> <p>詳細については、「上限誤謬頻度」の意味 ページ 1059を参照してください。</p>
--------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

詳細:異なる値を指定した実験

異なる誤謬数で評価コマンドを再実行し、結果がどのように変わるのかを確認します。

以下の表は一部の異なる結果を要約しています。

誤謬数 (サンプル)	上限誤謬頻度 (推定された最大 値)	<code>Vouchers</code> テーブル で承認がない伝票 (推定された最大 値)	結論
5	1.79%	95	伝票の統制は有効に運用されています。

誤謬数 (サンプル)	上限誤謬頻度 (推定された最大 値)	Vouchers テーブル で承認がない伝票 (推定された最大 値)	結論
		(0.0179 x 5298)	1.79% < 上限誤謬 2.00%
6	2.02%	107 (0.0202 x 5298)	<p>厳密には、伝票の統制は有効に運用されてい ません。</p> <p>ただし、2.02% は上限誤謬 2.00% に非常に近い です。</p> <p>メモ</p> <p>この例は、上限誤謬頻度と許容誤 謬数の間の差異を示し、統制逸脱 を評価します。</p> <p>より緩い許容誤謬数方法を使用す る場合、伝票の統制が有効に運用 されています。</p> <p>左の"誤謬数"列の観察された誤謬 数は6です。これは、サンプルサイズ を計算したときに報告された許容誤 謬数の6を超えていません。</p>
7	2.24%	119 (0.0224 x 5298)	<p>伝票の統制は有効に運用されていません。</p> <p>2.24% > 上限誤謬 2.00%</p>
10	2.89%	153 (0.0289 x 5298)	<p>伝票の統制は有効に運用されていません。</p> <p>2.89% > 上限誤謬 2.00%</p>

レコード サンプルのサンプルサイズを計算

データセットのサンプリングを実施する前に、統計的に適切なサンプル数と後続のサンプルおよび評価処理に必要な他の値を計算する必要があります。

Analytics のサンプルサイズの計算機能は、ユーザーが提供する入力値に基づき必要な値を計算します。

サンプルサイズを計算することの重要性

後続のサンプルの有効性を計るには、適切なサンプルサイズを計算することが重要になります。サンプルが有効でない、または代表を表してない場合は、全母集団に対してサンプルで実行する監査手続きの結果を信頼して予測することはできません。

サンプルサイズの計算を省略したり、サンプルサイズを推定しないでください。

サンプルサイズの計算に使用するほとんどの入力値は、専門的な判断に基づきます。運用環境でサンプリングの結果を信頼する前に、値が示す意味を十分理解するようにしてください。不明な点がある方は、監査サンプリング担当者、または監査サンプリングスペシャリストにお問い合わせください。

入力値がサンプルサイズに影響する方法

入力値は Analytics で計算されるサンプルサイズに影響します。[サイズ] ダイアログボックスの [計算] ボタンを使用すると、異なる入力値がサンプルサイズに影響する方法を実験できます。

以下の表は、サンプルサイズに対する入力値の影響をまとめています。

注意

本番環境では、サンプルサイズを小さくする目的だけで、入力値を操作しないでください。入力値は、サンプリングされるデータと監査目的に対して最も適切であるという専門的な判断に基づいているべきです。

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
信頼度		✓
母集団	サンプルサイズには影響がありません	
上限誤謬 (%)	✓	

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
想定誤謬率 (%)		✓

手順

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. [サンプリング > レコード/金額単位サンプリング > サイズの計算]を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. [メイン]タブで、[レコード]を選択します。
3. サンプルサイズを計算するために使用する入力値を入力します。

- 信頼
- 母集団
- 上限誤謬 (%)
- 想定誤謬率 (%)

メモ

入力値は以下で詳細に説明します。

4. (省略可能) [計算]をクリックすると、出力結果のプレビューが表示されます。

ヒント

[OK]の代わりに [計算]をクリックすると、結果を出力する前に別の入力値で実験できます。

メモ

出力結果は以下で詳細に説明します。

5. 出力]タブ

- a. [To] パネルで、次のいずれかを選択します。
 - 画面 -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

- **ファイル-**は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。

- b. 出力タイプとして **ファイル]**を選択した場合、次のいずれかを実行します。

- **名前]**テキスト ボックスにファイル名を入力します。
- **名前]**ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**のいずれかのみです。

6. **OK]**をクリックします。
7. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

サイズ ダイアログ ボックスの入力と結果

次の表は、**サイズ]**ダイアログ ボックスの入力値と出力結果の詳細を示します。

メインタブ - 入力値

入力値 - サイズダイアログボックス	説明
信頼度	<p>必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。</p> <p>たとえば、95 を指定した場合は、サンプルが実際に 95% の確率で母集団を代表しているとお客様が信頼したいということを意味します。信頼度は "サンプリングリスク" の補数です。信頼度が 95% ということはサンプリングリスクが 5% ということと同じです。</p>

入力値 - サイズダイアログボックス	説明
母集団	<p>サンプリングするデータセットのレコード数</p> <p>メモ レコード サンプリングでは、母集団は結果のサンプルサイズに影響しません。たとえば、他の入力値が同じままである場合は、統計的に同じ有効なサンプルサイズは150,000または1,000,000レコードの母集団に対して計算されます。 結果として得られる間隔値は、母集団のサイズに直接比例して大きくなります。</p>
上限誤謬 (%)	<p>発生し得るが、発生した場合でも統制が有効であると見なすことのできる、規定した統制からの逸脱率の上限です。</p> <p>たとえば、5を指定するとは、逸脱率が5%を超えたときに統制を無効と見なすことができるということです。</p>
想定誤謬率 (%)	<p>検出する規定の統制からの逸脱率。</p> <p>たとえば、1を指定するとは、この逸脱率が1%以内であることを指定することです。</p> <p>メモ 指定する推定誤謬率 (%)は上限誤謬 (%)より小さい必要があります。これらの相違が小さすぎる場合は、計算するには誤謬率が高すぎますというエラーメッセージが表示されます。 監査サンプリング用語では、相違で代表されるサンプリングの精度の度合いは、指定した信頼度のレベルに対して計算するには小さすぎます。</p>

メインタブ - 出力結果

出力結果 - サイズダイアログボックス	説明
サンプルサイズ	必要なサンプル数です。
間隔	間隔値-固定間隔とセルの選択方法に必要。
許容誤謬数	<p>上限逸脱率 (%)を超えることなく結果から得られるサンプルで発生しうる誤謬または逸脱の最大数のことです。</p> <p>詳細については、"許容誤謬数" 次のページを参照してください。</p>

入力および結果の例

伝票テーブルのレコード サンプルのサイズの計算

次の図は、レコード サンプルのサンプルサイズを計算するときの入力値と出力結果の例です。テーブルには 5298 レコードあります。他の入力値に基づくと、必要なサンプルサイズは 593 レコードです。

項目	値
信頼度	95
母集団	5298
許容誤謬率 (%)	2
予想誤謬率 (%)	1
サンプル数	593
間隔	8.93
許容誤謬数	6

計算は、**Vouchers** テーブルに基づきます。このテーブルは **ACL_Rockwood.ac1** (ACL DATA\Sample Data Files\ACL_Rockwood\ACL_Rockwood.ac1) にあります。

許容誤謬数

メモ

Analytics で評価機能を使用する予定がある場合は、**最大許容汚染**によって報告される値を使用する必要はありません。代わりに、評価機能で計算される**上限誤謬頻度**を使用します。詳細については、「レコード サンプルの誤謬を評価する」ページ 1056を参照してください。

許容誤謬数は、母集団の逸脱を評価する1つの方法です。

この方法を使用する場合は、サンプリングされたデータで監査手順を開始する前に、Analytics が報告したしきい値が事前に分かっています。手順の実行中に観察された累積誤謬が Analytics が報告した値を超える場合、規定された統制からの逸脱率が許容できないほど高いことがその時点で分かっています。

サンプリングされたデータで統制テストを実行した後、検出した誤謬または逸脱数を許容誤謬数と比較できます。観察されたエラー数が許容誤謬数以下の場合、指定された信頼度レベルに対して統制が効果的であると見なすことができます。

Analytics によって生成されるサンプルサイズの統計上の有効性

Analytics は、ほとんどの分析に対して、統計上有効なサンプル数のサンプルを抽出できます。ただし、以下の状況は例外となる可能性があります。

- 1000 レコードより少ないデータセットのサンプリングを行なっている場合
- 組織内に、必要に応じて正確なサンプル数を定義できるサンプリングの専門家がいる場合
- 組織の方針により、ほかのサンプリングツールまたはサンプリング手法を使用する場合

ポワソン分布と二項分布

サンプル数を生成する際、一般的に使用される2つの方法は、ポワソン分布と二項分布です。Analytics では、ポワソン分布によりサンプル数が生成されます。

データセットの大きさが標準的な 1000 件以上のレコードである場合は、ポワソン分布および二項分布により生成されるサンプル数はほぼ同じになります。データセットが 1000 件未満のレコードである場合、ポワソン分布により決定されるサンプル数は二項分布により決定される数と比べてやや大きくなる傾向があり、そのためより控えめになります。データセットが小さい場合、二項分布ではサンプル数が下方修正されるのに対し、ポワソン分布では下方修正されません。データセットが非常に小さい場合には、ポワソン分布により生成されるサンプル数が実際にデータセットのサイズを超過する場合があります。

Analytics でサンプル数を計算するときに、小さいデータセットのレコード サンプリングを行う場合は、サンプル数が必要以上に大きくなる場合があります。このことは分析の障害にはなりません。これは、小さい母集団に対して手動でオーバー サンプリングを行う場合にはよくあることです。

レコード サンプルングの実行

アクティブなテーブルにあるデータの代表 サンプルを含むテーブルを新たに作成することができます。レコード サンプルングは、規定した統制からの逸脱率に関心がある場合に適しています。

メモ

この手順には、フィルタリング(IF 文)または範囲パラメーターは含まれません。これらのオプションを適用すると、サンプルの有効性が損なわれます。

手順

メモ

手順の後に詳細情報が表示されます。"サンプル ダイアログ ボックスのオプション" 見開きページを参照してください。

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. ナビゲーターで、サンプルを抽出するテーブルを開きます。
2. [サンプルング > レコード/金額単位 サンプルング > サンプル] を選択します。
3. [メイン] タブで、[レコード] を選択します。
4. [サンプルパラメーター] パネルで、サンプルの選択方法を指定します。

- 固定間隔
- セル
- ランダム

5. 選択した選択方法のサンプルパラメーターを入力します。

選択方法	サンプルパラメーター
固定間隔	<ul style="list-style-type: none"> ◦ 間隔 ◦ 開始(省略可能)
セル	<ul style="list-style-type: none"> ◦ 間隔 ◦ シード値(省略可能) ◦ アルゴリズム - メルセンヌ・ツイスタを選択します
ランダム	<ul style="list-style-type: none"> ◦ サイズ ◦ シード値(省略可能) ◦ 母集団(省略可能) - テーブルのレコード数が入力されます ◦ アルゴリズム - メルセンヌ・ツイスタを選択します

メモ

サンプルパラメーターは、以下で詳細に説明されます。

6. **保存先**]テキスト ボックスで、出力結果を含める Analytics テーブルの名前を指定します。
7. **詳細**]タブで、次のいずれかを選択します。
 - **レコード** -レコード全体が出力テーブルに含まれます。
 - **フィールド** - 選択したフィールドのみが出力テーブルに含まれます。
8. **フィールド**]を選択した場合は、**抽出フィールド**]リストから出力テーブルに含めるフィールドを選択します。
9. 省略可能。出力結果にORDER フィールドを追加する場合は、**抽出順序をレポートする**]を選択します。

メモ

抽出順序をレポートする]は、**ランダム**選択方法および**フィールド**出力の両方が選択された場合にのみ利用可能です。

10. **OK**]をクリックします。

サンプルダイアログ ボックスのオプション

次の表は、**サンプル**]ダイアログ ボックスのオプションの詳細を示します。

[メイン] タブ

オプション - サンプルダイアログ ボックス	説明
MUS レコード	サンプルの種類： <ul style="list-style-type: none"> ○ MUS - 金額単位 サンプリング ファイルの虚偽表示の合計額に関心がある場合に適しています。 ○ レコード -レコード サンプリング 規定した統制からの逸脱率に関心がある場合に適しています。
サンプルの対象	レコード サンプリングでは使用されません。
固定間隔	固定間隔がサンプル選択で使用されることを指定します。 サンプルは、間隔値と指定した開始番号に基づいて選択されます。詳細については、"固定間隔選択方法" ページ 1025を参照してください。 固定間隔]を選択した場合は、次の値を入力します。 <ul style="list-style-type: none"> ○ 間隔(必須) - サンプル数の計算によって生じた間隔の値

オプション - サンプルダイアログボックス	説明
	<p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ]をクリックして サイズ]ダイアログボックスを開きます。詳細については、「レコード サンプルのサンプルサイズを計算」 ページ 1045を参照してください。</p> <ul style="list-style-type: none"> ◦ 開始 (省略可能) - ゼロより大きく間隔の値より小さいランダム開始値。 <p>ヒント</p> <p>ト開始番号 '0' を入力するか、Analytics でランダムに開始番号を選択する場合は、開始番号を空欄にします。</p>
セル	<p>セル方法をサンプル選択で使用することを指定します。</p> <p>データセットが複数の均等なセルまたはグループに分割され、各セルから1つのサンプルがランダムに選択されます。各セルのサイズは間隔値によって指定します。詳細については、「セル選択方法」 ページ 1026を参照してください。</p> <p>セル]を選択した場合は、次の値を入力します。</p> <ul style="list-style-type: none"> ◦ 間隔 (必須) - サンプル数の計算によって生じた間隔の値 <p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ]をクリックして サイズ]ダイアログボックスを開きます。詳細については、「レコード サンプルのサンプルサイズを計算」 ページ 1045を参照してください。</p> <ul style="list-style-type: none"> ◦ シード (省略可能) - 任意の数値にできます <p>Analytics で乱数ジェネレーターを初期化するために使用される数値です。</p> <p>ヒント</p> <p>シード値を指定したり、'0' のシード値を入力したりできます。あるいは、シード値を Analytics にランダムに選択させたい場合は、シード値をブランクのままにします。</p> <ul style="list-style-type: none"> ◦ アルゴリズム (必須) - メルセンヌ・ツイスタを選択します。 <p>バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、デフォルトを選択してください。</p>
ランダム	<p>ランダム方法をサンプル選択で使用することを指定します。</p> <p>サンプルはデータセット全体からランダムに選択されます。詳細については、「ランダム選択方法」 ページ 1027を参照してください。</p> <p>ランダム]を選択した場合は、次の値を入力します。</p> <ul style="list-style-type: none"> ◦ サイズ (必須) - Analytics によって計算されたサンプルサイズ <p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ]をクリックして サイズ]ダイアログボックスを開きます。詳細については、「レコード サンプルのサンプルサイズを計算」 ページ 1045を参照してください。</p>

オプション - サンプルダイアログボックス	説明
	<ul style="list-style-type: none"> シード(省略可能) - 任意の数値にできます Analytics で乱数ジェネレーターを初期化するために使用される数値です。 <p>ヒント シード値を指定したり、'0' のシード値を入力したりできます。あるいは、シード値を Analytics にランダムに選択させたい場合は、シード値をブランクのままにします。</p> <ul style="list-style-type: none"> 母集団(省略可能) - テーブルのレコード数があらかじめ入力されます。これは、サンプルが選択される母集団です。 アルゴリズム(必須) - メルセンヌ・ツイスタを選択します。 <p>バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、デフォルトを選択してください。</p>
もし次の	<p>注意 サンプルの過程で IF 文を作成したり、レコードをフィルタリングしないでください。このようにすると、サンプルの有効性が損なわれます。 詳細については、「条件付きサンプリング」ページ 1148を参照してください。</p>
変換先	<p>出力テーブルの名前と場所。</p> <ul style="list-style-type: none"> 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 プロジェクト フォルダー以外の場所に出力テーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
ローカル	<p>サーバーテーブルに接続している場合は、出力テーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ローカルを選択 -]では、Analytics プロジェクトと同じ場所、または移動先の場所に出力テーブルを保存します。 AX サーバーの Prefix フォルダーに出力テーブルを保存する場合は、ローカル -]を選択解除されたままにしておきます。
出力テーブルを使用する	<p>操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。</p>

詳細]タブ

オプション - サンプル ダイアログボックス	説明
範囲パネル	<p>注意</p> <p>サンプルの過程で処理されるレコードを制限しないでください。このようにすると、サンプルの有効性が損なわれます。</p> <p>詳細については、「条件付きサンプリング」ページ 1148を参照してください。</p>
レコード フィールド	<p>レコード全体、または選択したフィールドを出力テーブルに含めるかどうかを指定します。</p> <p>フィールド]を選択する場合は、次のいずれかを実行します。</p> <ul style="list-style-type: none"> ○ フィールドの抽出]リストから抽出するフィールドを選択します。 ○ フィールドの抽出]をクリックして、フィールドを選択するか、または式を作成します。 <p>フィールドを選択する順序が、結果に表示する列の順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。</p>
レポート選択順序	<p>(省略可能) 出力結果に ORDER フィールドを追加します。</p> <p>このフィールドには、各レコードの選択順序がランダムであることが表示されます。</p> <p>メモ</p> <p>抽出順序をレポートする]は、ランダム選択方法およびフィールド出力の両方が選択された場合にのみ利用可能です。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加する]を選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、「既存テーブルへの出力結果の追加」ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <p>上書きを確認するメッセージが表示されたら、適切なオプションを選択します。</p> <p>最後に追加]オプションが表示されることを期待していたのに表示されない場合は、いいえ]をクリックして操作をキャンセルし、「既存テーブルへの出力結果の追加」ページ 207を参照してください。</p>

レコード サンプルの誤謬を評価する

サンプリングされたデータのセットに対して監査手順を実行した後は、Analytics を使用して次のことができます。

- 検出したすべての誤謬を母集団に対して投影する
- 逸脱率の上限の計算

誤謬が見つからない場合でも、評価機能を使用して、サンプリングリスクの基本的な許容度を計算します。

メモ

誤謬の評価には、あらかじめサンプル数の計算によって生成された変数を用いる必要があります。

評価と比較の方法

評価するときには、Analytics は統計式を使用して、サンプルで見つかった誤謬を母集団全体に対して推定し、**上限誤謬頻度**(計算された逸脱率の上限)を計算します。

サンプルサイズを計算したときには以前に決定した**上限誤謬 (%)**と計算された値を比較します。比較に基づき、規定された統制が効果的に機能しているかどうかを決定します。

比較	結論
上限誤謬頻度は上限誤謬 (%)以下です	規定された統制は効果的に機能しています
上限誤謬頻度は上限誤謬 (%)より大きいです	規定された統制は効果的に機能していません

手順

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. [サンプリング > レコード/金額単位 サンプリング > 評価] を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. **メイン** タブで、**レコード** を選択します。
3. 誤謬の評価で使用する入力値を入力します。

- 信頼
- サンプル サイズ

メモ

抽出した実際のサンプル サイズを指定します。これは、Analytics によって当初計算されたサンプル サイズと異なってもかまいません。

- 誤謬数

メモ

入力値は以下で詳細に説明します。

4. 出力]タブ

- a. **To** パネルで、次のいずれかを選択します。
 - **画面** -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソース テーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

- **ファイル** - は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。

- b. 出力タイプとして **ファイル** を選択した場合、次のいずれかを実行します。

- **名前**]テキスト ボックスにファイル名を入力します。
- **名前**] ボタンをクリックして、**保存**] または **ファイルを保存する名前**] ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル** または **Unicode テキスト ファイル** のいずれかのみです。

5. **OK**] をクリックします。
6. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

評価]ダイアログボックスの入力値

次の表は、**評価]**ダイアログボックスの入力値の詳細を示します。

メインタブ - 入力値

入力値 - 評価ダイアログボックス	説明
信頼度	サンプル数の計算時に入力した同じ信頼度。 詳細については、「レコード サンプルのサンプル サイズを計算」 ページ 1045を参照してください。
サンプル サイズ	サンプルのレコード数。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>サンプル サイズは抽出した実際のサンプル サイズをです。これは、Analytics によって当初計算されたサンプル サイズと異なってもかまいません。</p> </div>
誤謬数	サンプルで見つかった誤謬または逸脱の総数。

以下の図は、レコード サンプルの誤謬を評価するための入力値の例を示します。

The screenshot shows a dialog box titled "評価" (Evaluation) with two tabs: "メイン" (Main) and "出力" (Output). The "メイン" tab is active. It contains three input fields:

- 金額単位** (Amount Unit): Radio button, not selected.
- レコード** (Record): Radio button, selected.
- 信頼度** (Confidence): Input field with value 95.
- サンプル数** (Sample Size): Input field with value 95.
- 誤謬の数** (Number of Errors): Input field with value 3.

At the bottom of the dialog box, there are three buttons: "OK", "キャンセル" (Cancel), and "ヘルプ" (Help).

結果

レコード サンプルで見つかった誤謬の評価は次の結果になります。

入力値	説明
上限誤謬頻度 (計算された上限偏差率)	Analytics が計算する調整された逸脱率は、指定した信頼度に対してデータセット全体で超過されません。

以下の図は、レコード サンプルで見つかった誤謬を評価した結果を示します。



「上限誤謬頻度」の意味

サンプルサイズを計算したときに決定した**上限誤謬(%)**と比較した場合の**上限誤謬頻度**の意味

- 検査している統制が有効に運用されているかどうか
- 統制が運用状況は有効されている場合、運用の効果

例

レコード サンプルで見つかった誤謬を評価し、Analytics は**上限誤謬頻度 4.35%**を返します。この割合は、サンプルサイズを計算し、信頼度 95% を指定したときに前の手順で指定した**上限誤謬(%)**(許容される逸脱率)の5%未満です。

この情報に基づいて、次のことが言えます。

母集団全体で規定された統制からの実際の逸脱率が4.35%を超える確率は95%です。

上の図のように、**上限誤謬頻度**が5%より大きい場合、規定された統制は運用状況は有効されていない可能性があります。さらに適切な手順を決定し、監査目的を満たす必要があります。

金額単位サンプリング

金額単位サンプリングは、取引の勘定またはクラスの金額虚偽表示の合計金額を推定するための統計サンプリング方法です。

金額単位サンプリングは、次の特性がある財務データで最も効果的に機能します。

虚偽表示がないか、少数の虚偽表示しかない。

たとえば、5%未満の数の項目が虚偽表示である場合など。

過少計上数より多くの過剰計上数があるように思われる。

0ドルの科目がない。

金額単位サンプリングの別名は次のとおりです。

- ドル単位 サンプル
- 確率比例 サンプル

ヒント

Analytics のエンドツーエンド プロセスに関するすぐに使える導入については、「金額単位サンプリングのチュートリアル」 ページ 1063を参照してください。

機能の仕組み

金額単位サンプリングでは、勘定のレコードの小さいサブセットを選択および分析し、結果に基づいて、勘定の金額虚偽表示の合計額を推定します。

次に、推定された虚偽表示を重大と判断される虚偽表示金額と比較し、勘定に関する決定を行います。

金額単位サンプリングは、この種の記述をサポートします。

- 勘定残高の虚偽表示が\$28,702.70を超える確率が95%あります。これは許容虚偽表示\$29,000.00未満です。このため、勘定の金額表示は公正であると見なされます。

金額単位サンプリングプロセスの概要

注意

有効なサンプルサイズの計算をスキップしないでください。

すぐにレコードのサンプルを抽出し、サンプルサイズで推定する場合は、分析結果の推定が無効になる確率が高く、最終結果に欠陥が生じます。

金額単位サンプリング処理には、以下の一般的な手順があります。

1. [必要なサンプル数を計算します。](#)
2. サンプルの選択方法を選択します。
 - [固定間隔](#)
 - [セル](#)
 - [ランダム](#)
3. 任意で次の1つ以上のオプションを指定します。
 - "最上層のカットオフ" ページ 1084
 - "サブサンプル" ページ 1085
 - "繰り返しなしのサンプル選択" ページ 1086
4. [レコードのサンプルを描画する](#)
5. サンプルングされたデータで意図した監査手順を実行します。
6. [評価](#) サンプルングされたデータの金額虚偽表示の観察されたレベルが全体として勘定で虚偽表示の許容可能な金額を表すかどうか。

金額単位 サンプルングによるレコードの選択方法

金額単位 サンプルングでは Analytics テーブルからサンプルレコードを選択する次のプロセスを使用します。

- サンプルングの基準として、数値フィールドに金額を指定します。
- フィールドの金額の絶対値合計は、金額単位の流れとして扱われ、各単位は絶対値の1セントを表します。
- サンプルの選択方法の1つを使用して、Analytics は金額単位の中からサンプルを選択します。選択された金額単位に相当するレコードは、サンプルング出力テーブルに含まれます。

例

テーブルには以下の値の "Amount" フィールドがあります。フィールドには絶対値 \$11.75 があるため、1,175 の金額単位があります。

サンプルング処理で金額単位の 399 と 1,007 が選択されると、レコードの 2 と 5 が出力テーブルに含まれます。レコードの 1、3、および 4 は含まれません。

レコード数	金額	累積残高 (絶対)	金額単位	Analytics によって 選択された単位
1	\$3.50	\$3.50	1 ~ 350	
2	(\$0.75)	\$4.25	351 ~ 425	399

レコード数	金額	累積残高 (絶対)	金額単位	Analytics によって 選択された単位
3	\$1.25	\$5.50	426 ~ 550	
4	\$0.75	\$6.25	551 ~ 625	
5	(\$5.50)	\$11.75	626 ~ 1,175	1,007

より大きい金額に対するバイアス

金額単位サンプリングは、より大きい金額を含むレコードの選択について、値が正、負のいずれかによらず、意図的にバイアスを作り出します。各金額単位が選択されるチャンスは同等です。そのため、金額単位を 100,000 含む 1000ドルの金額は、金額単位を 25,000 含む 250ドルの金額よりも選択される可能性が4倍高くなります。

つまり、指定したレコードが選択される確率は、そのレコードに含まれる金額に正比例します。

考慮事項

金額単位サンプリングは、実証性または虚偽表示の検証で使用するのに適しています。金額単位サンプリングは、より大きい金額にバイアスをかけることによって、母集団での重要な金額のすべてが検証される確実性が高くなります。誤った文のテストでは、重大な誤謬が発生するリスクが高いのは、高額な金額です。

高額決済に偏ったサンプリング手法を選択すると、小額決済に関する潜在的な問題を見逃してしまう可能性があります。少額取引の問題は、集約されると、重要になる場合があります。

金額単位サンプリングのチュートリアル

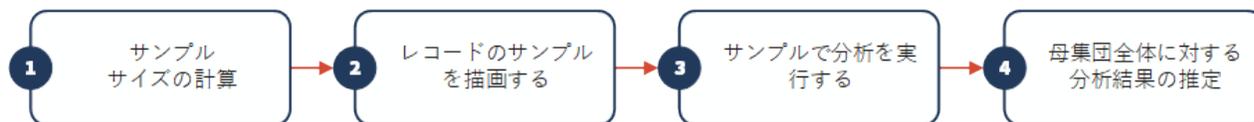
このチュートリアルでは、Analytics の金額単位 サンプリングのエンドツーエンド プロセスの概要を説明します。

予定所要時間 - 20 分

サマリー - invoices テーブルから、レコードで構成されるサンプルを抽出し、そのサンプルの虚偽表示を検出します。サンプルからの検出結果に基づいて、テーブル全体の虚偽表示総額を統計的に推定します。

次に、統計的推定を使って、請求書レコードの表示が全体として公正であるかどうかを判定します。

主なタスク - 金額単位 サンプリングを正しく実行するには、主なタスクとして以下の 4 つを実行する必要があります。



金額単位 サンプリングについて、オプションの使用方法は省略して、オプションなしでの使用方法のみを説明しているため、Analytics の金額単位 サンプリングの使用方法の基本をすばやく理解できます。

ヒント

サンプリング用語については、"用語に関する単語" ページ 1023に簡潔に定義されていますので参照してください。

金額単位 サンプリングのシナリオ

売掛金の虚偽表示の検出

シナリオ

売掛金の確認として、4000レコード以上がある Invoice テーブルを検査しています。請求された顧客のサンプルに連絡し、勘定の未払金額を確認し、虚偽表示を検出しようとしています。

顧客の連絡先を使用して、次のことを確認します。

- 売掛金が存在する
- 売掛金が正しく記録されている

どのように進めるか?

いくつかの顧客に連絡するか?どの顧客に連絡すると決定するか?サンプルから検出した虚偽表示が勘定全体にどのように分布しているか?

Analytics 金額単位 サンプリングを使用して、これらの質問に対する回答を得ることができます。

このシナリオで使用される Analytics テーブル

このシナリオは Analytics に含まれる `ACL_Rockwood.ac1` サンプル データ ファイルの `Invoices` テーブルを使用します。

メモ

`ACL_Rockwood.ac1` の `Invoices` テーブルのほとんどの金額の状態は "支払い済み" です。このシナリオでは、状態が "未払い" で支払金額が \$0.00 であると想定します。

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1 有効なサンプルサイズの計算

メモ

本番環境では、有効なサンプルサイズを計算するために指定する値は、専門家の判断によって異なります。

1. `ACL_Rockwood.ac1` で `Invoices` テーブル (`Sales_and_collection` フォルダー) を開きます。
2. `Invoice Amount` 見出しをクリックし、列を選択します。
3. `分析 > プロファイル` を選択し、`Invoice Amount` フィールドの絶対値を計算します。計算の結果で `プロファイル` タブを固定します。
4. `サンプリング > レコード/金額単位サンプリング > サイズの計算` を選択します。
5. `金額` を選択したままにします。
6. 以下の画面に表示されている入力値を指定し、`計算` をクリックしてサンプルサイズを計算します。

結果を確認した後、`OK` をクリックしてサンプルサイズ計算を確定するか、異なる値を指定して実験します(以下を参照)。

入力値の意味

信頼度	抽出するサンプルが母集団全体を表している信頼度が95% 必要です。 別の方法：サンプルを 100 回抽出した場合、95 回表し、5 回のみ表しません。
母集団	Invoices テーブルの Invoice Amount フィールドの絶対値。
Materiality	重大な虚偽表示と見なされるには、勘定の虚偽表示の合計金額が \$1,392,005.96 (3%) を超える必要があります。
想定される合計誤謬数	勘定の虚偽表示の合計金額は \$464,001.99 (1%) であると想定します。

結果の意味

サンプルサイズ	219 顧客に連絡する必要があります。
間隔	サンプル選択の間隔方法のいずれかを使用する場合、選択したレコードは以下のいずれかに対応します。

	<ul style="list-style-type: none"> 21,140,918 単位ごとに発生する金額単位 各ブロックの 21,140,918 単位からランダムに選択された金額単位 <p>メモ Analytics では、1 金額単位 = 1 セントです</p> <p>詳細については、「金額単位 サンプリングによるレコードの選択方法」ページ 1061を参照してください。</p>
最大許容汚染 (%)	<p>メモ この数字は、虚偽表示を評価するための代替方法であり、この金額単位 サンプリングのチュートリアルでは使用しません。</p> <p>この数字の意味：</p> <ul style="list-style-type: none"> サンプルの請求金額を確認するときに、個別の汚染割合の合計が 219.48% を超える場合、勘定に重大な虚偽表示があると見なすことができます。 <p>虚偽表示金額では、汚染は、虚偽表示が表す簿価の割合です。</p> <p>詳細については、「最大許容汚染 (%)」ページ 1076を参照してください。</p>

詳細：異なる値を指定した実験

サイズ] ダイアログ ボックスの次のフィールドのいずれかで値を変更し、**計算]** をクリックして、結果がどのように変わるのかを観察します。1 度に 1 つの値のみを変更し、変更がどのように結果に影響するのかをわかりやすくします。

- 信頼度
- Materiality
- 想定される合計誤謬数

より厳密な要件により、サンプル サイズが大きくなります。より緩い要件により、サンプル サイズが小さくなります。

上の画面に合うように値をリセットし、**OK]** をクリックします。サンプル サイズ計算の結果で **サイズ]** タブを固定します。

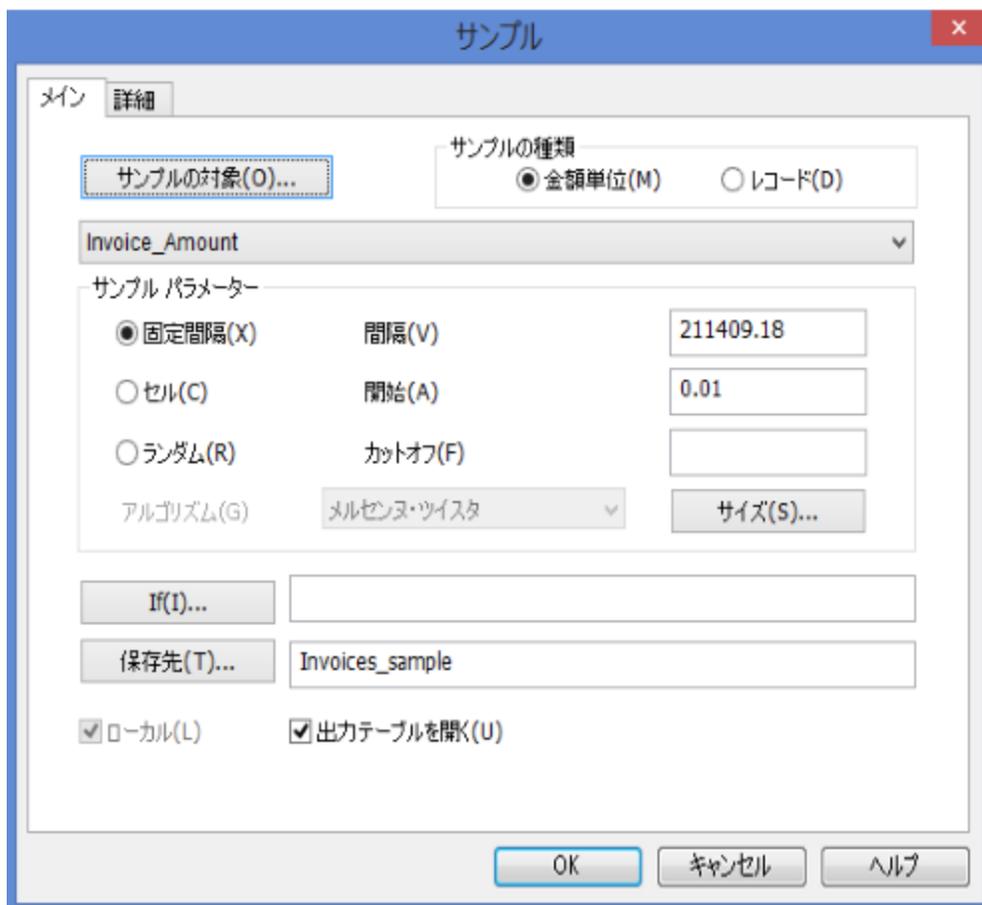
2 レコードのサンプルを抽出する

- Invoices** テーブルに戻ります。

Invoice Amount 列が選択されている場合は、テーブルビューの左上端をクリックして選択解除します。(最初の列ヘッダーの左の空白エリア。)

- サンプリング > レコード/金額単位 サンプリング > サンプル]** を選択します。
- MUS** を選択したままにします。
- 以下の画面に表示されている値を指定し、**OK]** をクリックしてレコードのサンプルを抽出します。

サンプリング基準] ドロップダウン リストで必ず **invoice_amount** フィールドを選択します。



入力値の意味

サンプルの対象	<code>invoice_amount</code> フィールドには、監査対象となる簿価が格納されています。
固定間隔	サンプルレコードを抽出するために固定間隔選択方法を使用しています。 固定間隔選択方法では、選択される初期金額単位を指定します。すべての後続の選択は固定間隔または離れた距離です。 詳細については、「固定間隔選択方法」ページ 1025を参照してください。
間隔	選択した金額単位間の間隔は \$211,409.18 または 21,140,918 単位です。
開始位置	選択した初期金額単位は \$0.01 または 単位 1 です。
変換先	<code>Invoices</code> テーブルから抽出されたレコードのサンプルは、新しいテーブル <code>Invoices_sample</code> に出力されます。

3 サンプルで分析を実行する

このチュートリアルでは、以下を行うものとします。

1. `Invoices_sample` テーブルに出現する顧客に連絡します。
2. 売掛金を確認し、虚偽表示を記録します。

4 母集団全体に対する分析結果の適用

1. [サンプリング>レコード/金額単位 サンプリング>評価]を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. [金額]を選択したままにします。
3. 以下の画面に表示されている入力値を指定し、[OK]をクリックして分析結果を適用します。

メモ

商品金額と誤謬の間にはカンマを使用し、金額にはカンマを使用しません。各金額と誤謬は別の行に入力します。

評価

メイン 出力

金額単位 信頼度 95

レコード 間隔 211409.18

項目の計上額、誤謬

誤謬 6002.16,6002.16 ^
31997.46,18000

OK キャンセル ヘルプ

入力値の意味

信頼度	サンプルサイズの計算時に指定したのと同じ信頼度。
間隔	サンプルの抽出時に使用した間隔。
誤謬	<p>売掛金を確認したときに入力した虚偽表示の形式： 簿価, 虚偽表示金額</p> <p>この例では以下のことが考察されます。</p> <ul style="list-style-type: none"> ○ 1つの請求された顧客には金額 \$6,002.16 のレコードがありませんでした。 ○ 別の請求された顧客はハードコピーの請求書の金額が 13,997.46 であり、31,997.46 ではないため、Invoices テーブルのデータ入力エラーの可能性を示しています。

推定/適用される分析結果の意味

	項目の計上額	誤謬額	推定誤謬額	最大誤謬額
基本精度				634,228.00
	6,002.16	6,002.16	211,409.18	369,966.07
	31,997.46	18,000.00	118,927.10	184,337.01
合計			330,336.28	1,188,531.07

基本精度	<p>サンプリングリスクの基本許容度: \$634,228.00</p> <p>Analytics はサンプリングリスクの基本許容度を計算します。サンプルに虚偽表示がない場合でも、全体として勘定に虚偽表示があることを確認できません。</p>
最尤誤謬	<p>勘定の推定された合計虚偽表示: \$330,336.28</p> <p>サンプルで見つかった実際の虚偽表示金額の勘定全体に対する推定。</p>
上限誤謬 (合計)	<p>95% の信頼度で推定された勘定全体の最大虚偽表示金額: \$1,188,531.07</p> <p>別の方法: Invoices テーブルの合計虚偽表示金額が \$1,188,531.07 を超えないと確率が 95% です。</p> <p>\$1,188,531.07 はサンプル サイズを計算したときに Materiality に対して指定した</p>

\$1,392,005.96 より小さいため、売掛金に重大な虚偽表示がないと結論付けることができません。

詳細については、「「上限誤謬」の意味」ページ 1091を参照してください。

詳細：異なる値を指定した実験

誤謬フィールドの異なる値で評価コマンドを再実行し、結果がどのように変わるのかを確認します。

以下の表は異なる結果を要約しています。

誤った文 (サンプル)	上限誤謬 (推定された最大値)	結論
6,002.16, 6,002.16 31,997.46, 18,000.00	1,188,531.07	勘定には重大な虚偽表示がありません。 \$1,188,531.07 < 重大しきい値 \$1,392,005.96
6,002.16, 6,002.16 31,997.46, 18,000.00 13,225.50, 8,644.34	1,392,005.84	厳密には、勘定には重大な虚偽表示がありません。 ただし、\$1,392,005.84 は重大しきい値 \$1,392,005.96 に非常に近いです メモ この例は、 上限誤謬 と 最大許容汚染 (%) の差異を示し、虚偽表示を評価します。 より厳密な 最大許容汚染 (%) 方法を使用する場合、勘定に重大な虚偽表示があります。 左の "Misstatements" 列の汚染割合の合計は 221.61% (100% + 56.25% + 65.36%) です。これは、サンプルサイズを計算したときに報告される 最大許容汚染 (%) の 219.48% よりも少し大きいです。
6,002.16, 6,002.16 31,997.46, 18,000.00 13,225.50, 13,225.50	1,505,511.86	勘定には重大な虚偽表示があります。 \$1,505,511.86 > 重大しきい値 \$1,392,005.96

金額単位サンプルのサンプルサイズを計算

データセットのサンプリングを実施する前に、統計的に適切なサンプル数と後続のサンプルおよび評価処理に必要な他の値を計算する必要があります。

Analytics のサンプルサイズの計算機能は、ユーザーが提供する入力値に基づき必要な値を計算します。

サンプルサイズを計算することの重要性

後続のサンプルの有効性を計るには、適切なサンプルサイズを計算することが重要になります。サンプルが有効でない、または代表を表してない場合は、全母集団に対してサンプルで実行する監査手続きの結果を信頼して予測することはできません。

サンプルサイズの計算を省略したり、サンプルサイズを推定しないでください。

サンプルサイズの計算に使用するほとんどの入力値は、専門的な判断に基づきます。運用環境でサンプリングの結果を信頼する前に、値が示す意味を十分理解するようにしてください。不明な点がある方は、監査サンプリング担当者、または監査サンプリングスペシャリストにお問い合わせください。

入力値がサンプルサイズに影響する方法

入力値は Analytics で計算されるサンプルサイズに影響します。[サイズ] ダイアログボックスの [計算] ボタンを使用すると、異なる入力値がサンプルサイズに影響する方法を実験できます。

以下の表は、サンプルサイズに対する入力値の影響をまとめています。

注意

本番環境では、サンプルサイズを小さくする目的だけで、入力値を操作しないでください。入力値は、サンプリングされるデータと監査目的に対して最も適切であるという専門的な判断に基づいているべきです。

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
信頼度		✓
母集団		✓
Materiality	✓	

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
想定される合計誤謬数		✓

手順

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. **サンプリング > レコード/金額単位サンプリング > サイズの計算**]を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. **メイン**]タブで、**金額単位**]を選択します。
3. サンプルサイズを計算するために使用する入力値を入力します。

- 信頼
- 母集団
- Materiality
- 想定される合計誤謬数

メモ

入力値は以下で詳細に説明します。

4. (省略可能) **計算**]をクリックすると、出力結果のプレビューが表示されます。

ヒント

OK]の代わりに **計算**]をクリックすると、結果を出力する前に別の入力値で実験できます。

メモ

出力結果は以下で詳細に説明します。

5. 出力]タブ

- a. [To] パネルで、次のいずれかを選択します。
 - 画面 -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

- **ファイル-**は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。

- b. 出力タイプとして **ファイル]**を選択した場合、次のいずれかを実行します。

- **名前]**テキスト ボックスにファイル名を入力します。
- **名前]**ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**のいずれかのみです。

6. **OK]**をクリックします。
7. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

サイズ ダイアログ ボックスの入力と結果

次の表は、**サイズ]**ダイアログ ボックスの入力値と出力結果の詳細を示します。

メインタブ - 入力値

入力値 - サイズダイアログボックス	説明
信頼度	<p>必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。</p> <p>たとえば、95 を指定した場合は、サンプルが実際に 95% の確率で母集団を代表しているとお客様が信頼したいということを意味します。信頼度は "サンプリングリスク" の補数です。信頼度が 95% ということはサンプリングリスクが 5% ということと同じです。</p>

入力値 - サイズダイアログボックス	説明
母集団	<p>数値 サンプルフィールドの絶対値</p> <p>メモ 絶対値を取得するには、サンプルフィールドで統計情報をプロファイリングするか生成します。</p>
Materiality	<p>サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。重大な虚偽表示額とまでは見なされません。</p> <p>たとえば、29000 を指定するとは、虚偽表示合計金額が \$29,000 を上回る場合に重大な虚偽表示額と見なすということです。</p>
想定される合計誤謬数	<p>サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。</p> <p>たとえば、5800 を指定するとは、虚偽表示合計金額として \$5,800 までを許容するということです。</p> <p>メモ 指定する 推定定合計誤謬 は Materiality 未満でなければなりません。これらの相違が小さすぎる場合は、計算するには誤謬率が高すぎます というエラーメッセージが表示されます。</p> <p>監査 サンプル用語では、相違で代表されるサンプリングの精度の度合いは、指定した信頼度のレベルに対して計算するには小さすぎます。</p>

メインタブ - 出力結果

出力結果 - サイズダイアログボックス	説明
サンプルサイズ	必要なサンプル数です。
間隔	間隔値-固定間隔とセルの選択方法に必要。
最大許容汚染 (%)	<p>Materiality を超えずに、結果のサンプリングの虚偽表示金額で発生しうる最大累積汚染割合。</p> <p>メモ Analytics が報告した 最大許容汚染 (%) 値は 100% を超えることができます。</p> <p>詳細については、"最大許容汚染 (%)" ページ 1076 を参照してください。</p>

入力および結果の例

Invoices テーブルの金額単位 サンプルのサイズを計算

次の図は、金額単位 サンプリングのサンプル サイズを計算するときの入力値と出力結果の例です。

- 取引金額フィールドの絶対値は、\$46,400,198.71ドルです
- **Materiality** は絶対値の3% 未満に設定されます
- **推定合計誤謬**は絶対値の1% 未満に設定されます
- 必要なサンプルサイズは219レコードです

The screenshot shows a dialog box titled "サイズ" (Size) with a close button (X) in the top right corner. It has two tabs: "メイン" (Main) and "出力" (Output). The "メイン" tab is active, showing input fields for "信頼度" (Confidence) set to 95, "母集団" (Population) set to 46400198.71, "許容誤謬額" (Tolerable Error) set to 1392005.96, and "予想誤謬額" (Estimated Error) set to 464001.99. There are radio buttons for "金額単位" (Amount) and "レコード" (Record), with "金額単位" selected. A "計算" (Calculate) button is highlighted. The "出力" tab shows the results: "サンプル数" (Sample Size) is 219, "間隔" (Interval) is 211,409.18, and "最大許容誤謬率 (%)" (Maximum Tolerable Error Rate (%)) is 219.48. At the bottom are "OK", "キャンセル" (Cancel), and "ヘルプ" (Help) buttons.

計算は、Invoices テーブルに基づきます。このテーブルは `ACL_Rockwood.ac1` (`ACL DATA\Sample Data Files\ACL_Rockwood\ACL_Rockwood.ac1`) にあります。

最大許容汚染(%)

メモ

Analytics で評価機能を使用する予定がある場合は、**最大許容汚染(%)**によって報告される値を使用する必要はありません。代わりに、評価機能で計算される**上限逸脱率**を使用します。詳細については、「金額単位 サンプルの誤謬を評価する」ページ 1087 を参照してください。

最大許容汚染(%)は、母集団の虚偽表示を評価する1つの方法です。

この方法を使用する場合は、サンプリングされたデータで監査手順を開始する前に、Analytics が報告したしきい値が事前に分かっています。手順の実行中に観察された累積誤謬がしきい値を超える場合、サンプルフィールドに重大な虚偽表示があることがその時点で分かっています。

例

売掛金テーブルでは、簿価 \$1000 が実際には \$930 であることがわかります。虚偽表示金額では、汚染は、虚偽表示が表す簿価の割合です。

簿価	監査値	過大申告	汚染
\$1,000	\$930	\$70	7% (70/1000)

サンプリングされたデータで独立した手順を実行した後、任意の虚偽表示金額からすべての個別の汚染割合を合計できます。汚染割合の合計が Analytics が報告した**最大許容汚染(%)**以下の場合、サンプルフィールドの金額全体が、指定した信頼度レベルに対して重大な虚偽表示があると見なすことができます。

例

売掛金テーブルで3つの虚偽表示金額を見つけます。これは、次の汚染と合計汚染割合になります。

簿価	監査値	過大申告	汚染
\$1,000	\$930	\$70	7% (70/1000)
\$2,500	\$1,500	\$1000	40% (1000/2500)
\$2,750	\$2,695	\$55	2% (55/2750)

簿価	監査値	過大申告	汚染
			49%(合計汚染割合)

テーブルのサンプルサイズを計算したときに Analytics が報告した**最大許容汚染 (%)**が**92.30%**であったとします。汚染割合の合計が最大許容汚染 49% は 92.30% 未満であるため、サンプルフィールドの金額全体が、指定した信頼度レベルに対して重大な虚偽表示があると見なすことができます。

メモ

最大許容汚染 (%)を使用した評価は、Analytics の評価機能よりも少し厳密です。汚染割合の合計がわずかに**最大許容汚染 (%)**値を超えている場合は、評価機能を使用し、サンプルフィールドが実際に重大な虚偽表示であることを確認してください。詳細については、「金額単位サンプルの誤謬を評価する」ページ 1087を参照してください。

Analytics によって生成されるサンプルサイズの統計上の有効性

Analytics は、ほとんどの分析に対して、統計上有効なサンプル数のサンプルを抽出できます。ただし、以下の状況は例外となる可能性があります。

- 1000 レコードより少ないデータセットのサンプリングを行なっている場合
- 組織内に、必要に応じて正確なサンプル数を定義できるサンプリングの専門家がいる場合
- 組織の方針により、ほかのサンプリングツールまたはサンプリング手法を使用する場合

ポワソン分布と二項分布

サンプル数を生成する際、一般的に使用される2つの方法は、ポワソン分布と二項分布です。Analytics では、ポワソン分布によりサンプル数が生成されます。

データセットの大きさが標準的な 1000 件以上のレコードである場合は、ポワソン分布および二項分布により生成されるサンプル数はほぼ同じになります。データセットが 1000 件未満のレコードである場合、ポワソン分布により決定されるサンプル数は二項分布により決定される数と比べてやや大きくなる傾向にあり、そのためより控えめになります。データセットが小さい場合、二項分布ではサンプル数が下方修正されるのに対し、ポワソン分布では下方修正されません。データセットが非常に小さい場合には、ポワソン分布により生成されるサンプル数が実際にデータセットのサイズを超過する場合があります。

Analytics でサンプル数を計算するときに、小さいデータセットのレコード サンプリングを行う場合は、サンプル数が必要以上に大きくなる場合があります。このことは分析の障害にはなりません。これは、小さい母集団に対して手動でオーバー サンプリングを行う場合にはよくあることです。

金額単位サンプリングの実行

アクティブなテーブルにある金額データの代表サンプルを含むテーブルを新たに作成することができます。金額単位サンプリングファイルの虚偽表示の合計額に関心がある場合に適しています。

メモ

この手順には、フィルタリング(IF文)または範囲パラメーターは含まれません。これらのオプションを適用すると、サンプルの有効性が損なわれます。

手順

メモ

手順の後に詳細情報が表示されます。"サンプルダイアログボックスのオプション" 次のページを参照してください。

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. ナビゲーターで、サンプルを抽出するテーブルを開きます。
2. 省略可能。[ランダム] 選択方法を使用する場合は、サンプルフィールドで統計をプロファイリングまたは生成します。
3. [サンプリング > レコード/金額単位サンプリング > サンプル] を選択します。
4. [メイン] タブで、[MUS] を選択します。
5. [サンプルの対象] ドロップダウンリストからサンプリングの対象となるフィールドを選択します。
6. [サンプルパラメーター] パネルで、サンプルの選択方法を指定します。

- 固定間隔
- セル
- ランダム

メモ

Analytics を使用して結果サンプルで検出された虚偽の表示を評価する場合は、金額単位サンプルでランダム選択方法を使用しないでください。

金額単位サンプルの評価には、固定間隔またはセル選択方法を使用する必要があります。

7. 選択した選択方法のサンプルパラメーターを入力します。

選択方法	サンプルパラメーター
固定間隔	◦ 間隔

選択方法	サンプルパラメーター
	<ul style="list-style-type: none"> ○ 開始(省略可能) セル ○ カットオフ(省略可能)
セル	<ul style="list-style-type: none"> ○ 間隔 ○ シード値(省略可能) ○ カットオフ(省略可能) ○ アルゴリズム-メルセンヌ・ツイスタを選択します
ランダム	<ul style="list-style-type: none"> ○ サイズ ○ シード値(省略可能) ○ 母集団 ○ アルゴリズム-メルセンヌ・ツイスタを選択します

メモ

サンプルパラメーターは、以下で詳細に説明されます。

8. **保存先**]テキスト ボックスで、出力結果を含める Analytics テーブルの名前を指定します。
9. **詳細**]タブで、次のいずれかを選択します。
 - **レコード** -レコード全体が出力テーブルに含まれます。
 - **フィールド** - 選択したフィールドのみが出力テーブルに含まれます。
10. **フィールド**]を選択した場合は、**抽出フィールド**]リストから出力テーブルに含めるフィールドを選択します。
11. 省略可能。以下のオプションを必要に応じて選択します。
 - サブサンプル
 - レポート選択順序
 - 繰り返しなし

メモ

オプションは以下で説明します。 [

サブサンプル]は、出力に"フィールド"を選択した場合のみ選択可能です。

抽出順序をレポートする]は、**ランダム**選択方法および**フィールド**出力の両方が選択された場合にのみ利用可能です。

12. **OK**]をクリックします。

サンプル ダイアログ ボックスのオプション

次の表は、**サンプル**]ダイアログ ボックスのオプションの詳細を示します。

メイン] タブ

オプション - サンプルダイアログボックス	説明
MUS レコード	<p>サンプルの種類:</p> <ul style="list-style-type: none"> ◦ MUS - 金額単位 サンプリング ファイルの虚偽表示の合計額に関心がある場合に適しています。 ◦ レコード -レコード サンプリング 規定した統制からの逸脱率に関心がある場合に適しています。
サンプルの対象	<p>数値サンプルフィールド。</p> <ul style="list-style-type: none"> ◦ サンプルの対象] ドロップダウン リストからフィールドを選択します。 ◦ サンプルの対象] ボタンをクリックしてフィールドを選択するか、式を作成します。
固定間隔	<p>固定間隔がサンプル選択で使用されることを指定します。</p> <p>サンプルは、間隔値と指定した開始番号に基づいて選択されます。詳細については、"固定間隔選択方法" ページ 1025を参照してください。</p> <p>固定間隔] を選択した場合は、次の値を入力します。</p> <ul style="list-style-type: none"> ◦ 間隔 (必須) - サンプル数の計算によって生じた間隔の値 <p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ] をクリックして サイズ] ダイアログボックスを開きます。詳細については、"金額単位 サンプルのサンプル サイズを計算" ページ 1071を参照してください。</p> <ul style="list-style-type: none"> ◦ 開始 (省略可能) - ゼロより大きく間隔の値より小さいランダム開始値。 <p>ヒント</p> <p>ト開始番号 '0' を入力するか、Analytics でランダムに開始番号を選択する場合は、開始番号を空欄にします。</p> <ul style="list-style-type: none"> ◦ カットオフ (省略可能) - カットオフ値の最上層 <p>カットオフ値以上のサンプルフィールドの金額が自動的に選択され、サンプルに取り込まれます。カットオフ値を空白にした場合は、間隔値に等しいデフォルトのカットオフ値が使用されます。</p> <p>詳細については、"最上層のカットオフ" ページ 1084を参照してください。</p>
セル	<p>セル方法をサンプル選択で使用することを指定します。</p> <p>データセットが複数の均等なセルまたはグループに分割され、各セルから1つのサンプルがランダムに選択されます。各セルのサイズは間隔値によって指定します。詳細については、"セル選択方法" ページ 1026を参照してください。</p> <p>セル] を選択した場合は、次の値を入力します。</p> <ul style="list-style-type: none"> ◦ 間隔 (必須) - サンプル数の計算によって生じた間隔の値

オプション - サンプルダイアログボックス	説明
	<p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ]をクリックしてサイズ]ダイアログボックスを開きます。詳細については、「金額単位 サンプルのサンプルサイズを計算」 ページ 1071を参照してください。</p> <ul style="list-style-type: none"> シード(省略可能) - 任意の数値にできます Analytics で乱数ジェネレーターを初期化するために使用される数値です。 <p>ヒント</p> <p>シード値を指定したり、'0' のシード値を入力したりできます。あるいは、シード値を Analytics にランダムに選択させたい場合は、シード値を空白のままにします。</p> <ul style="list-style-type: none"> カットオフ(省略可能) - カットオフ値の最上層 カットオフ値以上のサンプルフィールドの金額が自動的に選択され、サンプルに取り込まれません。カットオフ値を空白にした場合は、間隔値に等しいデフォルトのカットオフ値が使用されます。 詳細については、「最上層のカットオフ」 ページ 1084を参照してください。 アルゴリズム(必須) - メルセンヌ・ツイスタを選択します。 バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、デフォルトを選択してください。
ランダム	<p>ランダム方法をサンプル選択で使用することを指定します。</p> <p>サンプルはデータセット全体からランダムに選択されます。詳細については、「ランダム選択方法」 ページ 1027を参照してください。</p> <p>ランダム]を選択した場合は、次の値を入力します。</p> <ul style="list-style-type: none"> サイズ(必須) - Analytics によって計算されたサンプルサイズ <p>メモ</p> <p>サンプル数をまだ計算していなかった場合は、サイズ]をクリックしてサイズ]ダイアログボックスを開きます。詳細については、「金額単位 サンプルのサンプルサイズを計算」 ページ 1071を参照してください。</p> <ul style="list-style-type: none"> シード(省略可能) - 任意の数値にできます Analytics で乱数ジェネレーターを初期化するために使用される数値です。 <p>ヒント</p> <p>シード値を指定したり、'0' のシード値を入力したりできます。あるいは、シード値を Analytics にランダムに選択させたい場合は、シード値を空白のままにします。</p> <ul style="list-style-type: none"> 母集団(必須) - 絶対値額。この合計額が、サンプルの選択元の母集団になります。

オプション - サンプルダイアログボックス	説明
	<p>ヒント</p> <p>サンプルフィールドで統計を以前にプロファイリングまたは生成した場合は、このフィールドに正しい値が入力されます。</p> <ul style="list-style-type: none"> ◦ アルゴリズム(必須) - メルセンヌ・ツイスタを選択します。 <p>バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、デフォルトを選択してください。</p>
もし次の	<p>注意</p> <p>サンプルの過程で IF 文を作成したり、レコードをフィルタリングしないでください。このようにすると、サンプルの有効性が損なわれます。</p> <p>詳細については、"条件付きサンプリング" ページ 1148を参照してください。</p>
変換先	<p>出力テーブルの名前と場所。</p> <ul style="list-style-type: none"> ◦ 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 ◦ プロジェクト フォルダー以外の場所に出力テーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先]をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ</p> <p>Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
ローカル	<p>サーバーテーブルに接続している場合は、出力テーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ◦ ローカルを選択 -]では、Analytics プロジェクトと同じ場所、または移動先の場所に出力テーブルを保存します。 ◦ AX サーバーの Prefix フォルダーに出力テーブルを保存する場合は、ローカル-]を選択解除されたままにしておきます。
出力テーブルを使用する	<p>操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。</p>

詳細]タブ

オプション - サンプルダイアログボックス	説明
範囲パネル	<p>注意</p> <p>サンプルの過程で処理されるレコードを制限しないでください。このようにすると、サンプルの有効性が損なわれます。</p> <p>詳細については、「条件付きサンプリング」ページ 1148を参照してください。</p>
レコードフィールド	<p>レコード全体、または選択したフィールドを出力テーブルに含めるかどうかを指定します。</p> <p>フィールド]を選択する場合は、次のいずれかを実行します。</p> <ul style="list-style-type: none"> ○ フィールドの抽出]リストから抽出するフィールドを選択します。 ○ フィールドの抽出]をクリックして、フィールドを選択するか、または式を作成します。 <p>フィールドを選択する順序が、結果に表示する列の順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。</p>
サブサンプル レポート選択順序 繰り返しなし	<p>(省略可能) 追加のサンプリングオプション。</p> <p>以下のオプションを必要に応じて選択します。</p> <ul style="list-style-type: none"> ○ サブサンプル] - 出力結果に SUBSAMPLE フィールドを追加します。 このフィールドを使用して、合計金額から個別の取引をランダムに選択できます。 詳細については、「サブサンプル」ページ 1085を参照してください。 ○ レポート選択順序] - 出力結果に ORDER フィールドを追加します。 このフィールドには、各レコードの選択順序がランダムであることが表示されます。 ○ 繰り返しなし] - レコードが2回以上選択されないようにします。 <p>メモ</p> <p>サブサンプル]は、出力に「フィールド」を選択した場合のみ選択可能です。</p> <p>抽出順序をレポートする]は、ランダム選択方法およびフィールド出力の両方が選択された場合にものみ利用可能です。</p> <p>繰り返しなし]では、選択したレコードが後から選択できなくなり、サンプルのサイズが縮小することがあります。それを補うため、データセットのオーバーサンプリングすることを検討してください。</p> <p>詳細については、「繰り返しなしのサンプル選択」ページ 1086を参照してください。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加する]を選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、「既存テーブルへの出力結果の追加」ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <p>上書きを確認するメッセージが表示されたら、適切なオプションを選択します。</p>

オプション - サンプルダイアログボックス	説明
	最後に追加]オプションが表示されることを期待していたのに表示されない場合は、 [いい] をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

金額単位サンプリングのオプション

金額単位サンプリングの場合に限り、以下のオプションを指定することができます。

- 最上層のカットオフ
- サブサンプル
- 繰り返しなしのサンプル選択

最上層のカットオフ

メモ

最上層のカットオフは、固定間隔またはセルの選択方法を使用する金額単位のサンプリングにのみ利用できます。

最上層のカットオフは、多額のお金単位のサンプリングにバイアスをつけるために Analytics が使用する追加的な方法です。デフォルトでは、間隔値以上のサンプルフィールドのお金が最上層のお金と見なされ、自動的にサンプルに取り込まれます。

負のお金と正のお金を自動で取り込むことができます。見なされる金額が絶対値であるためです。

自動選択のお金が大きければ大きいほど、サンプルサイズも大きくなります。

間隔値より大きいまたは小さい最上層のカットオフの値を任意で指定することができます。

間隔値よりも大きい最上層のカットオフ	より大きい金額がサンプルに自動的に含まれる確率が下がります。 サンプルフィールドの最大の正または負のお金よりも大きいカットオフ値を指定する場合、金額は自動的に選択されません。
間隔値よりも小さい最上層のカットオフ	より大きい金額がサンプルに自動的に含まれる確率が上がります。 デフォルトの最上層のカットオフを使用して金額が自動的に選択されない場合は、カットオフ値を下に調整し、サンプルフィールドでより大きい金額の一部を自動的に選択できます。 注意 サンプルフィールド金額と比較して小さすぎるカットオフ値を指定した場合、自動的に選択される金額が多くなりすぎ、サンプルの目的に沿わなくなります。

ログに記録された最上層の選択と金額

金額単位サンプルを実行するときには、最上層の選択数と合計最上層のお金がログに表示されます。

例

ログは、93 の選択したレコードのうち8 が最上層であり、数値 サンプル フィールドの絶対値 \$33,153.55 を占めるのを示します。

サンプル サイズ = サンプルングされた 733 レコード中、93(8 最上位の階層)
母集団: 585674.41、最上位の階層: 33153.55、その他: 552520.86

サブサンプル

メモ

サブサンプルリングは、金額単位 サンプルングでフィールドの出力を使用した場合にのみ使用することができます。

場合によっては、サンプルフィールドの各金額が複数の個別の取引の合計を表します。サンプルングした各合計金額のうち1つの取引に対してのみ監査手続きを実行したいときには、サブサンプルリングを使って任意に個々の取引を選択できます。

サンプル]ダイアログボックスで **サブサンプルリング]**を選択すると、結果のサンプルに SUBSAMPLE フィールドが含まれます。このフィールドには、合計金額と合計金額を選択するために使用される実際の金額単位との間の差異を表す金額が含まれます。

例

\$12,455	(合計金額)
-\$4,620	(選択した金額単位)
= \$7,835	(SUBSAMPLE フィールドに表示された金額)

処理を完了するには、その特定の合計金額の累積残高に7,835番目のドルを含む取引を選択します。

メモ

サンプルのすべての最上層のカットオフ金額は“0.00”を SUBSAMPLE フィールドに表示します。これらはサンプルに自動的に含まれ、金額単位はその選択に関係しないためです。

繰り返しなしのサンプル選択

金額単位サンプリングは同じレコードを複数回選択する場合があります。サンプルフィールドの各金額には複数の金額単位が含まれ、同じ金額に属する2つ以上の金額単位を選択できます。これにより、金額を含むレコードが複数回選択されます。

サンプル]ダイアログボックスで **繰り返しなし]**を選択すると、同じレコードが複数回選択されないようにできます。結果のサンプルには重複が含まれません。ただし、サンプリングされたレコード数はAnalyticsによって計算されたサンプルサイズよりも小さくなる場合があります。補正するには、次の方法のいずれかを使用して、サンプルサイズを増やし、オーバーサンプリングできます。

- **固定間隔またはセル選択方法:**
 - 間隔のサイズを下げる
 - 最上層のカットオフ値を調整し、自動的に選択するレコード数を増やす
- **ランダム選択方法** -は指定されたサンプルサイズを増やします

金額単位サンプルの誤謬を評価する

サンプリングされたデータのセットに対して監査手順を実行した後は、Analytics を使用して次のことができます。

- 勘定全体に対して見つけたサンプルの虚偽表示を推定する
- 虚偽表示金額の上限を計算する

誤謬が見つからない場合でも、評価機能を使用して、サンプリングリスクの基本的な許容度を計算します。

メモ

誤謬の評価には、あらかじめサンプル数の計算によって生成された変数を用いる必要があります。

金額単位サンプルの結果で評価機能を使用する場合は、固定間隔またはセル選択方法を使用して、サンプルを抽出している必要があります。

評価と比較の方法

評価するときには、Analytics は統計式を使用して、サンプルで見つかった虚偽表示を勘定全体に対して推定し、**合計誤謬上限**(虚偽表示の上限)を計算します。

サンプルサイズを計算したときには以前に決定した**Materiality**と計算された値を比較します。比較に基づき、金額データが正しく表記されているかを決定します。

比較	結論
上限誤謬は Materiality 以下です	サンプルフィールドの金額全体は正しく表記されています
上限誤謬は Materiality より大きいです	サンプルフィールドの金額全体には重大な虚偽表示があります

手順

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. **サンプリング > レコード/金額単位 サンプリング > 評価]**を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

2. **メイン]**タブで、**金額単位]**を選択します。
3. 虚偽表示の評価で使用する入力値を入力します。

- 信頼
- 間隔
- 誤謬

メモ

入力値は以下で詳細に説明します。

4. **出力]**タブ
 - a. **[To]**パネルで、次のいずれかを選択します。
 - **画面** -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

- **ファイル** - は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。

- b. 出力タイプとして **ファイル]**を選択した場合、次のいずれかを実行します。
 - **名前]**テキスト ボックスにファイル名を入力します。
 - **名前]**ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**のいずれかのみです。

5. **OK]**をクリックします。
6. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

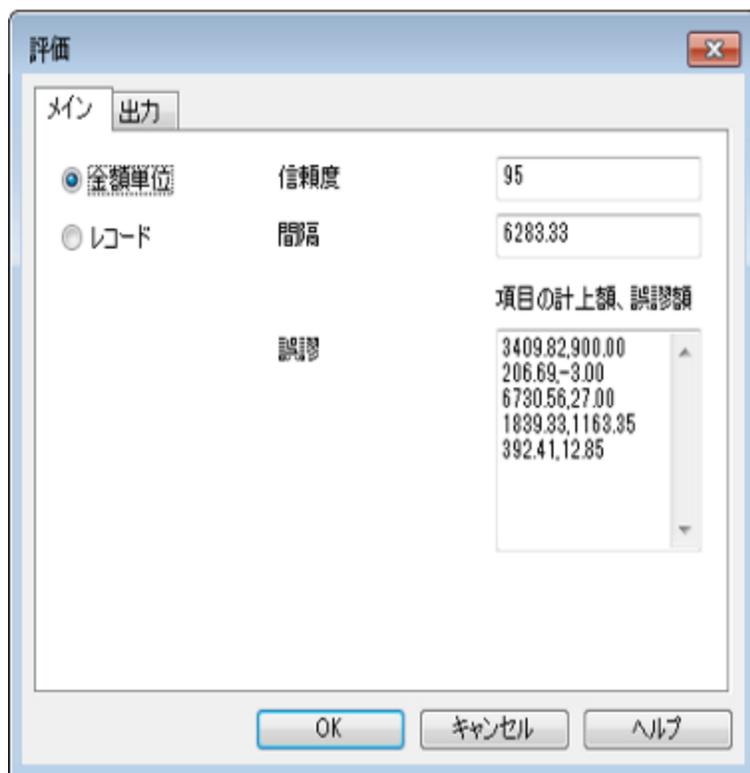
評価]ダイアログボックスの入力値

次の表は、**評価]**ダイアログボックスの入力値の詳細を示します。

メインタブ - 入力値

入力値 - 評価ダイアログボックス	説明
信頼度	<p>サンプル数の計算時に入力した同じ信頼度。 詳細については、"金額単位サンプルのサンプルサイズを計算" ページ 1071を参照してください。</p>
間隔	<p>サンプルの抽出時に使用した間隔値。</p> <p>メモ 使用した間隔値は、Analytics によって当初計算された間隔値と異なっていてもかまいません。</p>
誤謬 (Item 金額、誤謬)	<p>サンプルで見つかった、虚偽表示の全誤謬のリスト。 虚偽表示額と金額の簿価をカンマで区切って指定します。過剰計上は正の金額として、過少計上は負の金額として入力します。</p> <p>ヒント 虚偽表示誤謬リストが長い場合は、別のアプリケーションからリストをコピーして貼り付ける方が簡単な場合があります。</p> <div data-bbox="456 1188 1414 1587" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>例</p> <p>金額の簿価が\$1,000で、金額の監査額が\$930の場合は、1000,70と入力します。 金額の簿価が\$1,250で、金額の監査額が\$1,450の場合は、1250,-200と入力します。</p> <p>各誤謬は別の行に入力します。</p> <p>1000,70 1250,-200</p> </div>

以下の図は、金額単位サンプルの誤謬を評価するための入力値の例を示します。



結果

金額単位サンプルで見つかった誤謬の評価は次の結果になります。

結果値	説明
項目	虚偽表示誤謬のサンプル金額の入力したリスト。
エラー	虚偽表示額の入力したリスト。
基本精度	サンプリングリスクの基本的な許容誤謬(以下の図の18,850.00)
最尤誤謬 (想定された誤った文)	サンプリング金額を含む間隔に投影された各誤謬の虚偽表示金額。 最上層ではない 最大尤度誤謬 金額は降順で一覧表示されます。最上層の虚偽表示金額は投影された過剰表示と投影された過小表示の間に一覧表示されます。 投影計算は最上層の虚偽表示金額では実行されません。
上限誤謬 (上限誤った文)	サンプリングリスクに調整された 最大尤度誤謬 。 調整計算は最上層の虚偽表示金額または投影された過小評価では実行されません。投影された過小表示は"0.00"と一覧表示されるため、 上限誤謬 には影響しません。
合計	母集団全体または勘定残高の 最大尤度金額 、および 上限誤謬金額 。

以下の図は、金額単位サンプルで見つかった誤謬を評価した結果を示します。

Item 列	投影計算	調整計算
最初の3つの金額は最上位階層ではありません <ul style="list-style-type: none"> 1,839.33 3,409.82 392.41 	✓	✓
4番目の金額が最上位階層です <ul style="list-style-type: none"> 6,730.56 	✗	✗
5番目の金額は過小表示されています <ul style="list-style-type: none"> 206.69 	✓	✗

評価

日時: 2016/07/12 09:01:47

コマンド: [EVALUATE MONETARY CONFIDENCE 95 ERRORLIMIT 3409.82,900.00,206.69,-3.00,6730.56,27.00,1839.33,1163.35,392.41,12.85 INTERVAL 6283.33 TO SCREEN](#)

信頼度: 95, 間隔: 6283

	項目の計上額	誤謬額	推定誤謬額	最大誤謬額
基本精度				18,850.00
	1,839.33	1,163.35	3,974.12	6,954.71
	3,409.82	900.00	1,658.44	2,570.58
	392.41	12.85	205.76	300.41
	6,730.56	27.00	27.00	27.00
	206.69	-3.00	-91.20	0.00
合計			5,774.12	28,702.70

「上限誤謬」の意味

サンプルサイズを計算したときに決定した **Materiality** と比較したときの**上限誤謬**の合計金額は次の意味があります。

- 検査している勘定残高の金額が正しく表示されているかどうか
- 金額が正しく表示されている場合は、虚偽表示の最大金額と考えられるもの

例

金額単位サンプリングで見つかった誤謬を評価し、Analytics は**上限誤謬** \$28,702.70 を返します。この金額は、サンプルサイズを計算し、信頼度 95% を指定したときに前の手順で指定した **Materiality**(許容虚偽表示) の \$29,000 未満です。

この情報に基づいて、次のことが言えます。

勘定残高の実際の虚偽表示が\$28,702.70を超える確率が95%あります。

上限誤謬が\$29,000より大きい場合、勘定残高に重大な虚偽表示がある可能性があります。さらに適切な手順を決定し、監査目的を満たす必要があります。

金額単位サンプリングの上限誤謬を計算する方法

Analyticsによって計算される**上限誤謬**はサンプリングリスクに調整された複合数値です。つまり、サンプリングされた金額が、検査している勘定残高の虚偽表示の真の合計金額を過小表示するリスクです。

詳しく表示

上限誤謬は次の金額の合計です。

金額	説明
基本精度	<p>サンプリングされた金額に虚偽表示の誤謬がない場合、統計式を使用して Analytics によって計算されたサンプリングリスクの基本許容度。</p> <p>サンプリングリスクの基本許容値が必要です。サンプリングされた金額に誤謬がない場合でも、母集団全体に誤謬が存在しないことを確認できないためです。</p>
各虚偽表示の誤謬の調整された金額	<p>次の計算の結果：</p> <p>汚染割合 (虚偽表示金額 / サンプル金額の簿価) * 間隔金額 * サンプリング リスクの増分許容度</p> <ul style="list-style-type: none"> 汚染割合 * 間隔金額は、1つのサンプル金額の虚偽表示の観察された金額をサンプル金額を含む間隔に投影します。 <p>Analyticsはこの投影された虚偽表示最大尤度誤謬を呼び出します。すべての投影された虚偽表示の合計は、勘定残高の最大尤度誤謬です。</p> <ul style="list-style-type: none"> 最大尤度誤謬 * サンプリング リスクの増分許容度は Analytics によって統計式を使用して計算されます。 <p>最大尤度誤謬は勘定残高の虚偽表示の真の金額を過小表示する可能性があるため、この追加の調整は必要です。</p>

従来の変数サンプリング

従来の変数サンプリングは、以下を推定するための統計的サンプリング方法です。

- 取引の勘定またはクラスの合計監査済み金額
- 取引の勘定またはクラスの金額虚偽表示の合計金額

従来の変数サンプリングは、次の特性がある財務データで最も効果的に機能します。

通常量～大量の虚偽表示

たとえば、5%以上の数の項目が虚偽表示である場合など。

過剰計上または過少計上が存在する。

0ドルの科目が存在する。

ヒント

Analyticsにおける従来の変数サンプリングのエンドツーエンドプロセスに関するクイックスタートガイドについては、「従来の変数サンプリングのチュートリアル」ページ 1101を参照してください。

メモ

従来の変数サンプリングは、財務データに使用できるだけでなく、数量や時間単位などの測定単位をはじめとする、可変特性を持つ任意の数値データに対して使用することができます。

機能の仕組み

従来の変数サンプリングでは、アカウントのレコードの小さいサブセットを選択して分析できます。サブセットの分析結果に基づいて、アカウントの合計監査金額と、虚偽表示の合計金額を推定できます。

2つの推定金額は範囲として計算されます。

- **点推定値**は範囲の中間点です。
- **上限と下限**は、範囲に含まれる2つの端点です。

点推定値と上限だけまたは下限だけを使って、片側推定または範囲を計算することもできます。

推定された範囲を、勘定の簿価、または重大と判断される虚偽表示金額と比較し、勘定に関する決定を行います。

従来の変数サンプリングでは、次のような表示がサポートされます。

- 勘定の真の監査金額が、勘定簿価 46,400,198.71 が含まれる 45,577,123.95 ~ 46,929,384.17 の間となる確率が95%あります。このため、勘定の金額表示は公正であると見なされます。

- 勘定残高の虚偽表示額が 813,074.76 ~ 539,185.46 となる確率が 95% あります。これは金額精度 $\pm 928,003.97$ の範囲を逸脱していません。このため、勘定の金額表示は公正であると見なされます。

従来の変数サンプリング処理の概要

注意

有効なサンプルサイズの計算をスキップしないでください。

すぐにレコードのサンプルを抽出し、サンプルサイズで推定する場合は、分析結果の推定が無効になる確率が高く、最終結果に欠陥が生じます。

従来の変数サンプリング処理には、以下のステージがあります。

1. [従来の変数サンプリングの準備\(計画\)](#)
2. [レコードのサンプルを描画する](#)
3. サンプリングされたデータで意図した監査手順を実行します。
4. 以下を[評価](#)します。

- 全体として勘定に対して推定されるときに、サンプリングされたデータの監査済み金額が記録された簿価の許容可能な範囲内に入るかどうか
- サンプリングされたデータの金額虚偽表示の観察されたレベルが全体として勘定で虚偽表示の許容可能な金額を表すかどうか

各値は保持され、次の段階に移動すると入力されます。

Analytics での従来の変数サンプリングには、3つの異なるダイアログボックスに情報を入力し、関連するコマンドを次の順に実行します。

1. **CVS 準備** ダイアログボックス
2. **CVS サンプル** ダイアログボックス
3. **CVS 評価** ダイアログボックス

この処理を進める過程で、1つのダイアログボックスに入力した情報が次のダイアログボックスに自動的にあらかじめ入力されます。値があらかじめ入力されることで、手間が省かれるとともに、誤って不適切な値を入力してサンプルが無効とされるリスクを防ぐことができます。

なお、**CVS サンプル** および **CVS 評価** ダイアログボックスに自動的に入力された値は、一時的に保存されるだけであり、Analytics プロジェクトを閉じれば削除されます。

従来の変数サンプリングの値の再生成

実稼働環境では、タイミングが異なれば、従来の変数サンプリングのプロセスの異なる段階を実行するのが一般的です。Analytics をクローズしたときに失われた従来の変数サンプリングの値を再生成するには、

次のいずれかの方法を使用できます。

最初の方法は最も簡単です。

- **あらかじめ入力されたコマンドを保存**

CVS 準備段階と CVS サンプル段階の結果には、従来の変数サンプリングのプロセス内の後続のコマンドが含まれます。これらのコマンドには、必要な値があらかじめ入力されています。あらかじめ入力されたこれらのコマンドを、後で使用できるように、独立したスクリプトに保存します。

詳細については、「従来の変数サンプリングのチュートリアル」ページ 1101を参照してください。

- **実行したコマンドをスクリプトに保存**

fCVS 準備段階と CVS サンプル段階を実行した後で、Analytics の表示領域に表示されている `CVSPREPARE` および `CVSSAMPLE` コマンドをコピーし、独立したスクリプトに保存します。このようなスクリプトを後で実行することで、従来の変数サンプリングの値を再生成することができます。

この方法の欠点は、レコードの冗長なサンプルが抽出されることです。

- **実行したコマンドをログから取得**

ログに記録されている `CVSPREPARE` および `CVSSAMPLE` コマンドをコピーし、これらのコマンドをコマンドラインで再実行して、従来の変数サンプリングの値を再生成します。

この方法の欠点は、ログ内のコマンドの正しいインスタンスを見つけるのが困難な場合があることであり、その場合はレコードの冗長なサンプルが抽出されます。

あらかじめ入力された値の変更

従来の変数サンプリングのあらかじめ入力された値は通常、変更しないでください。あらかじめ入力された値を変更すると、サンプリングプロセスの統計の有効性が否定される可能性があります。

注意

あらかじめ入力された値を更新するのは、変更の影響を理解するための統計に関する知識がある場合に限りです。

数値の長さ制限

従来の変数サンプリングの準備段階では、いくつかの内部計算が行われます。これらの計算では、最大 17 桁の数値がサポートされます。計算結果が 17 桁を超える場合には、その計算結果が出力に含まれなくなるため、サンプリング処理を続行できなくなります。

注意: 17 桁未満のソースデータの数値から、17 桁を超える内部計算結果が生成される場合もあります。

階層化

従来の変数サンプリングでは、サンプルを抽出する前に、母集団のレコードを数値的に階層化することができます。

階層化のメリットは、必要なサンプルサイズが通常、大幅に小さくなるとともに、統計的有効性が維持されることです。サンプルサイズが小さくなるとは、目標を達成するのに必要なデータ分析作業が少なくなるということです。

機能の仕組み

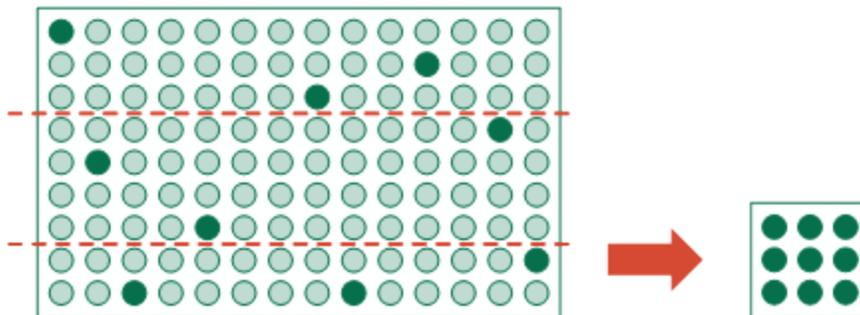
詳しく表示

階層化とは、層と呼ばれるいくつかの小グループに母集団を分割することです。各層内の値は、比較的に同質であるのが理想です。

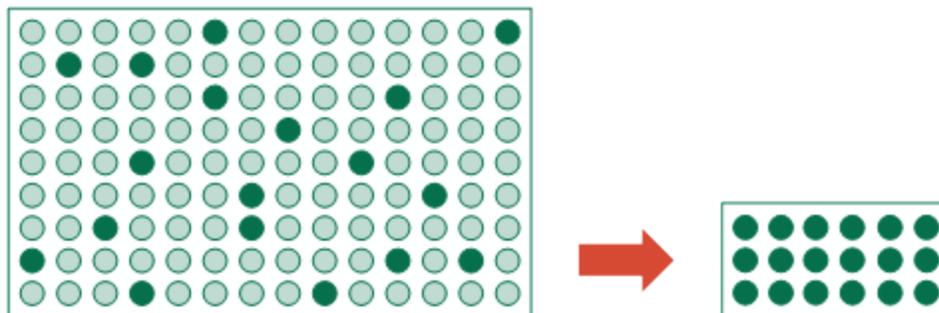
層間の境界は統計的アルゴリズム(ネイマン配分法)によって設定されます。このアルゴリズムでは、各層内の値の散布度が最小化されるように境界の位置が決定されるため、母分散を使用する効果が減少します。分散または'散らばり具合'が小さくなれば、必要なサンプルサイズも小さくて済みます。設計上、各層の範囲は均一ではありません。

必要なサンプル数は、階層化されていない母集団全体に対して計算されるのではなく、層別に計算されて合計されます。通常、非階層化アプローチより階層化アプローチの方が、データセットに対するサンプルサイズがはるかに小さくて済みます。

階層化された母集団のサンプリング



階層化されていない母集団のサンプリング

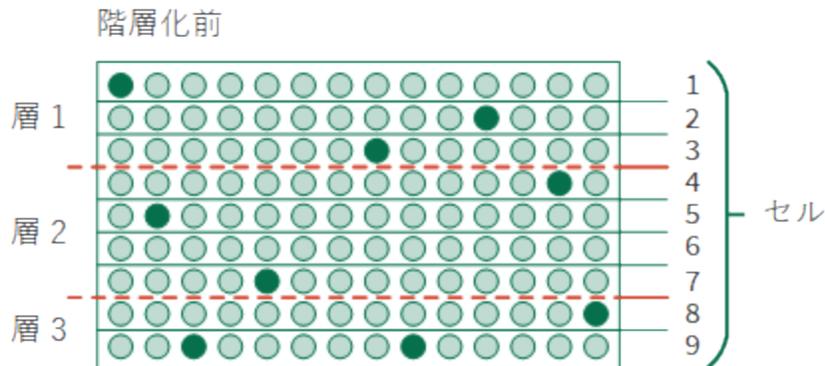


セルを使用した、階層化の前処理

階層化処理の一部として、母集団階層化の前処理に使用するセル数を指定します。セルは分割間隔が均等であり、層より幅が狭くなります。

統計的アルゴリズムでは、層間の最適な境界を割り当てる計算の一部として各層内のレコード数が使用されます。最終的に階層化された出力には、セルは保持されません。

指定するセルの数は、指定する層の数の2倍以上である必要があります。



メモ

階層化の前処理に使用するセル数と、セル方式のサンプル選択で使用するセル数は、同じものではありません。

過ぎたるは及ばざるがごとし

階層化はサンプルサイズを管理できる強力なツールですが、層数とセル数は慎重に指定してください。

手始めに以下を試してください。

- 層数: 4 ~ 5
- セル数: 50

ある層数またはセル数より大きい値を指定すると、ほとんどあるいはまったくサンプルサイズに影響しなくなる値があります。ただし、これらの値も、大きなデータセットを階層化する場合には、サンプルの設計や Analytics のパフォーマンスに悪影響を及ぼす可能性があります。

サンプル設計については、評価ステージに入ったら、母集団全体の虚偽表示を高い信頼性で推定するために、各層に最低限の数の虚偽表示が含まれていることを確認する必要があります。虚偽表示数に比べて層数が多すぎる場合は、推定時に問題が発生する可能性があります。

確実性層

階層化で使用可能なもう1つのオプションとして、**確実性層**があります。上位確実性層、下位確実性層、またはその両方を定義できます。

確実性層の使用には以下の2つのメリットがあります。

- **自動包含** - 単独で重要な項目や高額項目は、自動的にサンプルに含まれるため、ランダム選択方法によって除外されるリスクがありません。
- **分散減少法** - 確実性層項目はサンプルサイズ計算の対象から除外されます。高額項目はその性質上、母分散(および、高額項目が計算対象になる場合は、必要なサンプルサイズ)の値を大幅に大きくする可能性があります。

確実性層の定義

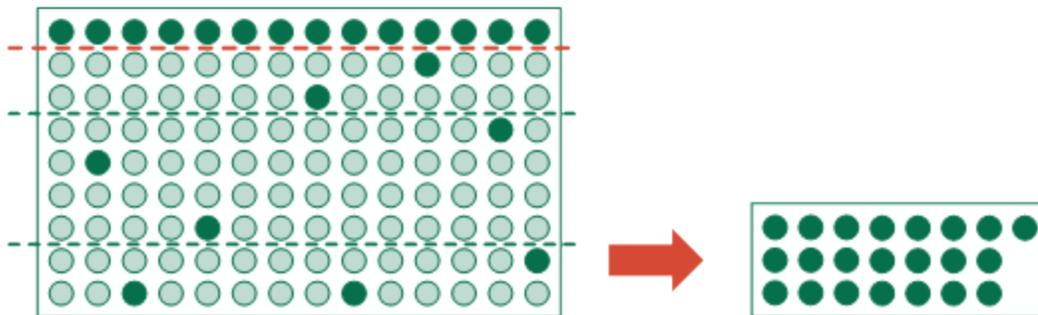
確実性層を定義するには、カット オフ数値を指定します。

- **上位確実性層カットオフ** - カットオフ値以上のキーフィールドの簿価がすべて自動的に選択され、サンプルに取り込まれます。
- **下位確実性層カットオフ** - カットオフ値未満のキーフィールドの簿価がすべて自動的に選択され、サンプルに取り込まれます。

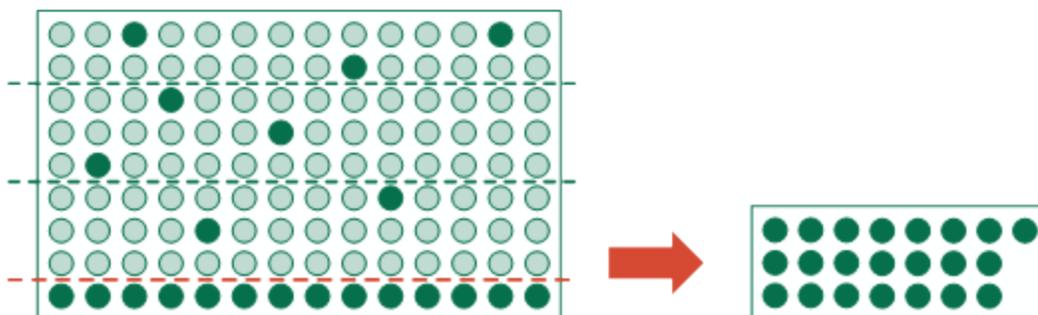
下位確実性層を使用すると便利なのは、大量のマイナス値が母集団に存在し、かつそれらを自動で含めたい場合です。

確実性層によって捕捉されない母集団の部分は、ランダム選択方法を使用してサンプリングされます。

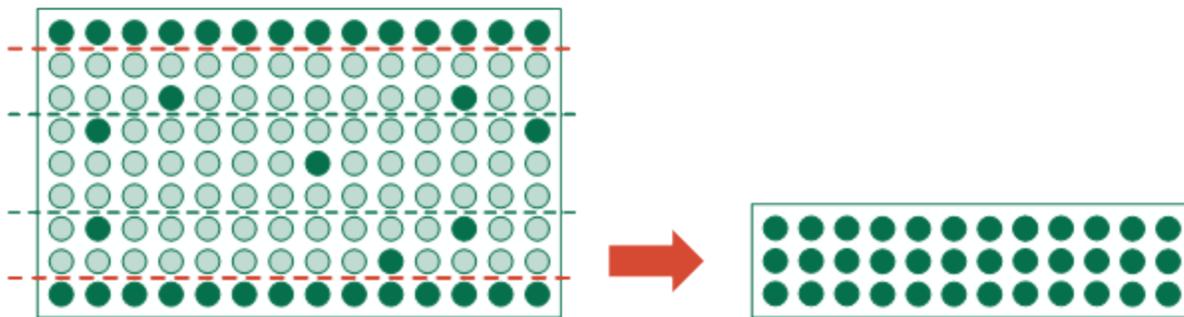
上位の確実性層を使用したサンプリング



下位の確実性層を使用したサンプリング



両方の確実性層を使用したサンプリング



メモ

上位確実性層のカットオフ値を低くしたり、下位確実性層のカットオフ値を高くすると、データの性質に応じてサンプルサイズ全体が増える場合があります。

そのような極端なカットオフ値を設定しないようにしてください。カットオフ値の設定値がわからない場合は、サンプリングのスペシャリストにお問い合わせください。

上位確実性層と下位確実性層の調整

サンプル抽出時に上位確実性層と下位確実性層を両方とも使用する場合は、上位確実性層カットオフ値と下位確実性層カットオフ値の関係を考慮する必要があります。

- 上位確実性層と下位確実性層は重なることができない - 下位確実性層カットオフ値よりも小さい上位確実性層カットオフ値を指定すると、エラーが発生します。
- 十分に差がある上位確実性層カットオフ値と下位確実性層カットオフ値を選択する - 互いに近すぎる上位確実性層カットオフ値と下位確実性層カットオフ値を指定すると、母集団の大部分が自動的にサンプルに含まれてしまうので、サンプリングの目的が達せられなくなります。

従来の変数サンプリングでレコードが選択される仕組み

従来の変数サンプリングでは、次の手順を使って、Analytics テーブルからサンプルレコードを選択します。

- サンプリングの基準として数値フィールドを指定します。サンプリング単位はテーブル内のレコードです。
- Analytics では、ランダム選択方法を使ってテーブル内のレコードからサンプルが選択されます。
- 階層化を使用する場合は、各層からほぼ等しい数のレコードがランダムに選択されます。
- 階層化を使用しない場合は、母集団全体からレコードがランダムに選択されます。
- 選択されたレコードは、サンプリング出力テーブルに含まれます。

例

300 レコードあるテーブルが、Analytics により、3 つの層に分割され、以下のレコード番号が選択されます。

層 1	層 2	層 3
<ul style="list-style-type: none"> ○ 9 ○ 13 ○ 40 ○ 52 ○ 78 ○ 91 ○ 99 	<ul style="list-style-type: none"> ○ 104 ○ 119 ○ 132 ○ 144 ○ 153 ○ 186 	<ul style="list-style-type: none"> ○ 211 ○ 229 ○ 236 ○ 248 ○ 278 ○ 295 ○ 296

300 レコードある階層化されていないテーブルから、Analytics により下記のレコード番号が選択されます。選択されたレコード番号が、階層化されているテーブルほど均一に分布していないことがわかります。

メモ

下記のレコード番号は比較しやすいように3つの列にグループ化してありますが、列が層を表しているわけではありません。

<ul style="list-style-type: none"> ○ 25 ○ 64 ○ 79 ○ 104 ○ 122 ○ 127 ○ 138 	<ul style="list-style-type: none"> ○ 143 ○ 175 ○ 179 ○ 184 ○ 191 ○ 201 ○ 234 	<ul style="list-style-type: none"> ○ 241 ○ 257 ○ 259 ○ 281 ○ 289 ○ 299
------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------

バイアスのないサンプルの選択

従来の変数サンプリングは、バイアスがかかっておらず、レコードに含まれる金額に基づいていません。サンプルの対象として選択される可能性はどのレコードについても等しくなります。金額を1000ドル含むレコード、250ドル含むレコード、および1ドル含むレコードの選択されるチャンスは同等です。

つまり、指定したレコードが選択される確率は、そのレコードに含まれる金額とは無関係です。

最大級の金額が含まれるレコードを確実に選択したい場合は、「確実性層」ページ 1097を参照してください。

従来の変数サンプリングのチュートリアル

このチュートリアルでは、Analytics における従来の変数サンプリングのエンドツーエンド プロセスの概要を説明します。

予定所要時間 - 30 分

サマリー - invoices テーブルから、レコードで構成されるサンプルを抽出し、そのサンプルの虚偽表示を検出します。サンプルからの検出結果に基づいて、以下の2つの金額を統計的に推定します。

- テーブル全体の監査額合計
- テーブル全体の虚偽表示額合計

次に、統計的推定を使って、請求書レコードの表示が全体として公正であるかどうかを判定します。

主なタスク - 従来の変数サンプリングを正しく実行するには、主なタスクとして以下の4つを行う必要があります。



このチュートリアルでは、従来の変数サンプリングについて、オプションの使用方法は省略して、オプションなしでの使用方法のみを説明しているため、Analytics における従来の変数サンプリングの使用方法の基本をすばやく理解できます。

ヒント

サンプリング用語については、「用語に関する単語」ページ 1023に簡潔に定義されていますので参照してください。

従来の変数サンプリングのシナリオ

売掛金の虚偽表示の検出

シナリオ

売掛金の確認として、4000レコード以上がある Invoice テーブルを検査しています。請求された顧客のサンプルに連絡し、勘定の未払金額を確認し、虚偽表示を検出しようとしています。

顧客の連絡先を使用して、次のことを確認します。

- 売掛金が存在する
- 売掛金が正しく記録されている

どのように進めるか?

いくつの顧客に連絡するか?どの顧客に連絡すると決定するか?サンプルから検出した虚偽表示が勘定全体にどのように分布しているか?

上記の質問に対する回答を得るために、Analytics の従来の変数サンプリングを使用することができます。

このシナリオで使用される Analytics テーブル

このシナリオは Analytics に含まれる `ACL_Rockwood.ac1` サンプルデータファイルの `Invoices` テーブルを使用します。

メモ

`ACL_Rockwood.ac1` の `Invoices` テーブルのほとんどの金額の状態は "支払い済み" です。このシナリオでは、状態が "未払い" で支払金額が \$0.00 であると想定します。

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1 母集団を階層化し、有効なサンプルサイズを計算する

メモ

実稼働環境では、母集団を階層化して有効なサンプルサイズを計算するために指定する値は、専門家としてのあなたの判断によって異なります。

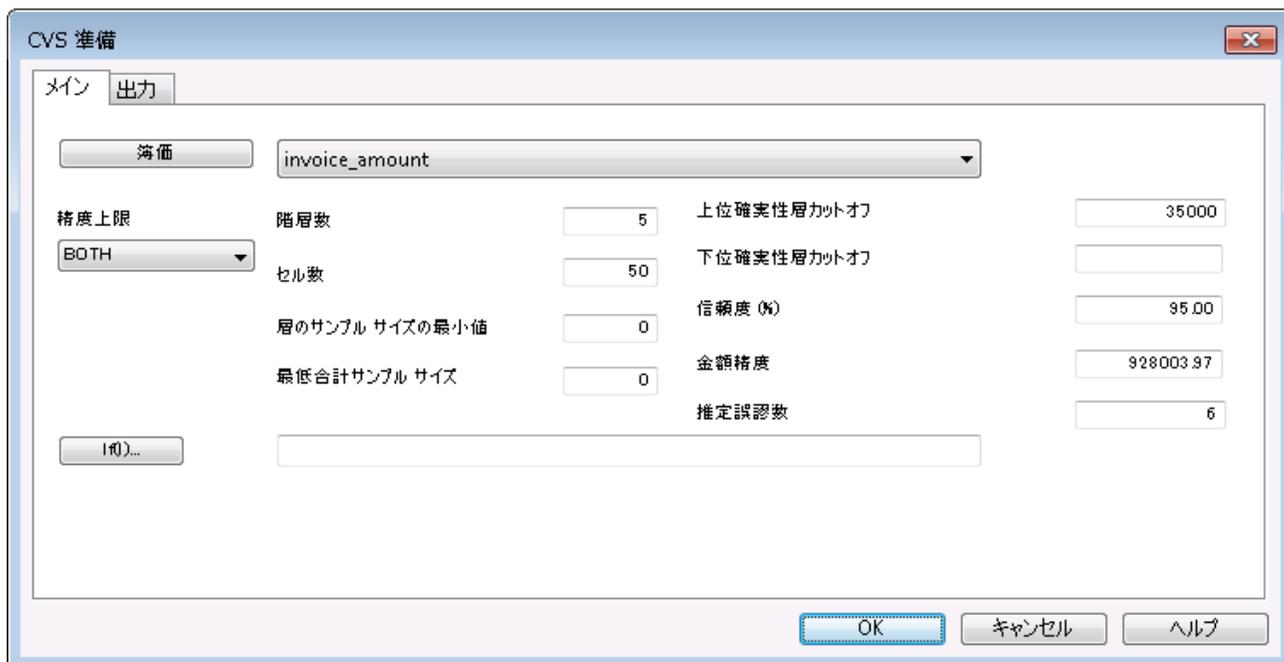
1. `ACL_Rockwood.ac1` で `Invoices` テーブル (`Sales_and_collection` フォルダ) を開きます。
2. `サンプリング > 従来の変数サンプリング (CVS) > 準備` を選択します。
3. 以下の画面に表示されている入力値を指定し、`[OK]` をクリックします。

`簿価` ドロップダウンリストで必ず `invoice_amount` フィールドを選択します。

Analytics により、母集団が階層化され、各層および母集団全体のサンプルサイズが計算されます。

4. 省略可能。階層化およびサンプルサイズ計算の出力結果を表示した状態でタブをピン留めします。

従来の変数サンプリング処理を進めるときに出力結果をピン留めしておくと、処理が完了後に処理全体を見直すことができます。



入力値の意味

簿価	invoice_amount フィールドには、監査対象となる簿価が格納されています。
精度制限	両方を選択します(デフォルト設定)。理由： <ul style="list-style-type: none"> 勘定全体が過剰表示または過小表示である可能性がある 方向の虚偽表示が金額精度を超えるかどうかを推定することに関心があります。
階層数	サンプルサイズを大幅に小さくする手段として、母集団を5つの層、つまり小グループに分割する必要があります。
セル数	母集団を階層化する前処理用に50個のセルを指定します。 セルの数は、層の数より分割可能な数が少なくなります。階層化の前処理は、層の境界の位置を最適化する内部処理の一部です。最終的に階層化された出力には、セルは保持されません。 セル数は層数の2倍(層数 × 2)以上である必要があります。
層のサンプルサイズの最小値	省略可能。 デフォルト値ゼロ(0)をそのまま使用すると、サンプルとして採取されるレコード数の下限を各層に適用しないということです。
最低合計サンプルサイズ	省略可能。 デフォルト値ゼロ(0)をそのまま使用すると、最低合計サンプルサイズを適用しないというこ

	とです。
上位確実性層カットオフ	\$35,000 以上の簿価科目をすべてサンプリングし、テストする必要があります。 上位確実性層にあるすべての項目がサンプル出力カテゴリーに出力されます。
下位確実性層カットオフ	省略可能。 このフィールドを空のままにすることは、下位確実性層を指定しないという意味になります。
信頼度 (%)	抽出するサンプルが母集団全体を表している信頼度が95% が必要です。 別の方法: サンプルを 100 回抽出した場合、95 回表し、5 回のみ表しません。
金額精度	金額精度は、許容虚偽表示から推定虚偽表示を差し引いたものです。 勘定の簿価の3% までの合計虚偽表示を許容し、この虚偽表示が勘定の簿価の1% になると推定されるため、2% の金額精度、つまり \$928,003.97 が得られます。
推定誤謬数	サンプル内の虚偽表示の誤謬数が6 以上であると推定します。

結果の意味

階層番号	階層境界	母集団項目	パーセントカウント	金額のパーセント	母集団値	サンプル項目	差異	標準偏差	平均
1	4376.88	1,279	31.33	7.29	3,382,131.93	37	1,022,929.96	1,011.40	2,644.36
2	9248.74	898	22.00	12.27	5,693,215.11	36	1,962,436.76	1,400.87	6,339.88
3	16904.52	763	18.69	21.52	9,987,014.57	49	5,030,465.84	2,242.87	13,089.14
4	23864.32	627	15.36	27.28	12,657,163.59	36	4,011,127.73	2,002.78	20,186.86
5	< 35000.00	479	11.73	28.76	13,346,354.63	39	7,757,896.09	2,785.30	27,862.95
小計		4,046	99.12%	97.12%	45,065,879.83	197	19,784,856.37	9,443.22	70,123.20
下	< 201.00	0	0.00	0.00	0.00	0	0.00	0.00	0.00
上	=> 35000.00	36	0.88	2.88	1,334,318.88	36	4,514,605.06	2,124.76	37,064.41
合計		4,082	100%	100%	46,400,198.71	233	24,299,461.43	11,567.98	107,187.61

日時: 2019/12/05 10:34:35
 コマンド: [CVSPREPARE ON INVOICE AMOUNT NUMSTRATA 5 MINIMUM 0 PRECISION 928003.97 CONFIDENCE 95.00 CUTOFF 35000 NCELLS 50 PLIMIT BOTH ERRORLIMIT 6 MINSAMPLESIZE 0 TO SCREEN](#)
 テーブル: Invoices

金額精度	928,003.97
信頼度	95%
標準誤謬	474,610.19

関連付けられた CVSSAMPLE コマンド
[CVSSAMPLE ON invoice amount NUMSTRATA 5 CUTOFF 35000.00 STRATA 4376.88,9248.74,16904.52,23864.32,35000.00 SAMPLESIZE](#)

<p>サンプル項目</p>	<p>合計 233 の顧客に問い合わせる必要があります。</p> <p>母集団の各層と上位確実性層について、指定した数の顧客に問い合わせる必要があります。</p> <p>たとえば、層 3 については、49 の顧客に問い合わせる必要があります。</p>
<p>層のブレイクダウン</p>	<p>Analytics によって行われる母集団の階層化。</p> <p>各層と上位確実性層について、以下の記述値が提供されます。</p> <ul style="list-style-type: none"> ○ 階層番号 - 各層に割り当てられる連続番号 上位確実性層には番号 '0' が割り当てられます(この画面には表示されていません)。 ○ 階層境界 - 確実性層のカットオフ値と各層の上限の境界。 ○ 母集団項目 (数) - テーブルのレコード数。上位確実性層などの層にブレイクダウンされません。 ○ サンプル項目 (数) - 必要なサンプルの合計サイズ。層にブレイクダウンされます。上位確実性層のすべての項目を含みます。
<p>記述統計</p>	<p>記述統計により、母集団の層の統計的特性に対する洞察を得ることができます。</p> <ul style="list-style-type: none"> ○ 標準誤謬 ○ 差異 ○ 標準偏差 ○ 相加平均
<p>関連付けられた CVSSAMPLE コマンド</p>	<p>CVS 準備の結果には、このコマンドのバージョンとして、従来の変数サンプリングプロセスの CVS サンプル段階(次の段階)で使用される、値があらかじめ入力されたバージョンが含まれます。</p> <p>このコマンドにあらかじめ入力された値は、Analytics の終了後も保存されることはありません。値を保存することでそれらを後で再生成する手間を回避するには、値があらかじめ入力されたコマンドを手動で保存します。</p> <p>実稼働環境では、CVS 準備段階は、母集団の階層化とサンプルサイズを最適化する際に、何回でも実行できます。CVS CVSSAMPLE コマンドを手動で保存できます。</p>

CVSSAMPLE コマンドの保存(省略可能)

メモ

このチュートリアルでは、Analytics を閉じない限り、本コマンドを保存する必要はありません。実稼働環境では、本コマンドを保存しておくことをお勧めします。

詳しく表示

CVSSAMPLE コマンドに指定した値を再現しなければならない場合に備えて、このコマンドを保存してください。

1. CVS 準備の表示領域の下部にある **CVSSAMPLE** へのリンクをクリックすることで、コマンドをコマンドラインに読み込みます。
2. コマンドラインに読み込んだコマンドの全体をコピーし、**CVS_Sample** という名前の Analytics スクリプトに保存します。

2 レコードのサンプルを抽出する

1. **Invoices** テーブルに戻ります。
2. **サンプリング > 従来の変数サンプリング (CVS) > サンプル** を選択します。
3. 以下の画面に表示されている値を指定し、**[OK]** をクリックしてレコードのサンプルを抽出します。

メモ

大部分の値は、CVS 準備段階の出力結果から自動的に入力されます。

自動入力される値がいくつか欠けている場合は、"CVSSAMPLE コマンドの使用 (省略可能)" 次のページを参照してください。

次に示されているとおりのシード値を指定してください。12345

このシード値は、サンプルレコードのランダム選択の結果を初期化するのに使用されます。これと異なるシード値を使用すると、異なるレコードグループが選択されるため、サンプル数が以下の例と一致しなくなります。

4. 省略可能。サンプリング処理結果の概要を表示した状態でタブをピン留めします。
従来の変数サンプリング処理を進めるときに出力結果をピン留めしておくと、処理が完了後に処理全体を見直すことができます。

CVS サンプル

メイン

簿価

invoice_amoun ▼

階層数 5 シード値 12345

上位確実性層カットオフ 35000.00

下位確実性層カットオフ

階層境界	サンプルサイズ	母集団(カウント、値)
4376.88	37	1279,3382131.93
9248.74	36	898,5693215.11
16904.52	49	763,9987014.57
23864.32	36	627,12657163.59
35000.00	39	479,13346354.63

If...

保存先(T)... Invoices_sample

OK キャンセル ヘルプ

CVSSAMPLE コマンドの使用(省略可能)

メモ

本チュートリアルの実行中に Analytics を開いたままにしておく場合は、このセクションをスキップしてください。

詳しく表示

[CVS サンプル] ダイアログボックスにいくつかの自動入力値が欠けている場合は、CVS サンプル段階から CVS 評価段階までの間に Analytics を閉じていたために値が失われた可能性があります。

この場合、レコードのサンプルを抽出するのに、[CVS サンプル] ダイアログボックスを使用する代わりに、スクリプトに保存しておいた `CVSSAMPLE` コマンドを使用することもできます。

1. **CVS_Sample** スクリプトを開き、**CVSSAMPLE** コマンドを次のように更新します。

```
NUMSTRATA 5 SEED 12345 CUTOFF 35000.00
```

```
TO Invoices_sample
```

2. **Invoices_sample** テーブルを開きます。
3. スクリプトを実行するか、本コマンド全体をコマンドラインにコピーして Enter キーを押します。

スクリプトを実行する場合は、ログ内の **CVSSAMPLE** コマンドをダブルクリックして CVS サンプルの表示領域を開きます。

ヒント

CVSSAMPLE コマンドを保存していなかった場合は、CVS 準備段階を再実行することで、CVS 評価段階に必要な値を再生成します。**Invoices_sample** テーブルを開き、ログ内の **CVSPREPARE** コマンドをすばやく再実行します。

入力値の意味

簿価 階層数 上位確実性層カットオフ 階層境界 サンプルサイズ 母集団(カウント、値)	CVS 準備段階では、ユーザーが入力した値または Analytics で計算された値に基づいて、入力値が自動的に入力されます。
シード値	省略可能。 シード値の指定は省略可能ですが、サンプルに含まれるレコードがチュートリアルのサンプルと一致するように同じシード値を指定してください。
変換先	Invoices テーブルから抽出されたレコードのサンプルは、新しいテーブル Invoices_sample へ出力されます。

結果の意味

Invoices CVS 準備 CVS サンプル

日時: 2019/12/05 10:37:58

コマンド: [CVSSAMPLE ON INVOICE AMOUNT NUMSTRATA 5 SEED 12345 CUTOFF 35000.00 STRATA 4376.88,9248.74,16904.52,23864.32,35000.00 SAMPLESIZE 37,36,49,36,39 POPULATION 1279,3382131.93,898,5693215.11,763,9987014.57,627,12657163.59,479,13346354.63 TO "INVOICES SAMPLE"](#)

テーブル: Invoices

シード値	12345
簿価フィールド	invoice_amount
選択方法	RANDOM

階層番号	階層境界	母集団項目	母集団値	サンプル項目	サンプル値
1	4376.88	1,279	3,382,131.93	37	98,112.90
2	9248.74	898	5,693,215.11	36	206,801.51
3	16904.52	763	9,987,014.57	49	623,529.81
4	23864.32	627	12,657,163.59	36	723,926.60
5	< 35000.00	479	13,346,354.63	39	1,069,044.86
小計		4,046	45,065,879.83	197	2,721,415.68
上	>= 35000.00	36	1,334,318.88	36	1,334,318.88
合計		4,082	46,400,198.71	233	4,055,734.56

関連付けられた CVSEVALUATE コマンド:
[CVSEVALUATE BOOKED invoice amount AUDITED invoice amount ETYPE MPU STRATA 4376.88,9248.74,16904.52,23864.32 POPULATI](#)

シード値 簿価フィールド	ユーザーが提供した入力値
選択方法	Analytics では、ランダム選択方法を使って、指定された数のレコードを各層から抽出します。 上位確実性層内のレコードはすべて自動的に選択されます。
層のブレイクダウン	CVS 準備段階の出力に表示されるブレイクダウン(内訳、分解)は常に同じです。 実際のサンプルレコードが抽出されたため、Analytics により、サンプル内の各層およびサンプル全体の サンプル値 (サンプル金額)が計算されます。 サンプル値と母集団値 (母集団金額)の違いに注意してください。
関連付けられた CVSEVALUATE コマンド	CVS サンプルの結果には、このコマンドのバージョンとして、従来の変数サンプリングプロセスの CVS 評価段階(最終段階)で使用される、値があらかじめ入力されたバージョンが含まれます。 このコマンドにあらかじめ入力された値は、Analytics の終了後も保存されることはありません。値を保存することでそれらを後で再生成する手間を回避するには、値があらかじめ入力されたコマンドを手動で保存します。

実稼働環境では、サンプルデータに対して監査手続きを実行する、CVS サンプル段階から CVS 評価段階までの期間は、数週間となる可能性があります。

CVSEVALUATE コマンドの保存(省略可能)

メモ

このチュートリアルでは、Analytics を閉じない限り、本コマンドを保存する必要はありません。実稼働環境では、本コマンドを保存しておくことをお勧めします。

詳しく表示

`CVSEVALUATE` コマンドに指定した値を再現しなければならない場合に備えて、このコマンドを保存してください。

1. CVS サンプルの表示領域の下部で、**CVSEVALUATE** リンクをクリックし、コマンドをコマンドラインに読み込みます。
2. コマンドラインに読み込んだコマンドの全体をコピーし、**CVS_Evaluate** という名前の Analytics スクリプトに保存します。

Audit_Value(監査額) フィールドを追加し、サンプルテーブルをエクスポートします。

Analytics は読み取り専用アプリケーションであるため、サンプルレコードから成るテーブルを Excel にエクスポートし、監査額を追加できるようにする必要があります。

テーブルをエクスポートする前に、**invoice_amount** フィールドを複製したフィールドを追加しておく必要があります。複製したフィールドを Excel で編集します。

監査額フィールドを追加します。

1. **Invoices** テーブルを閉じます。
2. **Invoices_sample** テーブルを開きます。
3. このコマンドをコピーし、コマンドラインに貼り付けます。

```
DEFINE FIELD AUDIT_VALUE COMPUTED invoice_amount
```

コマンドラインが表示されない場合、**[ウィンドウ> コマンドライン]**を選択します。

4. Enter を押し、新しいフィールドを作成します。

ヒント

AUDIT_VALUE(監査額) フィールドが作成されたことを確認する場合は、コマンドラインで「`display`」を入力し、Enter を押してテーブルのフィールドのリストを表示します。

サンプルテーブルを Excel にエクスポートする

1. **データ > エクスポート**]を選択します。
2. **メイン**]タブで **フィールド**]を選択します。
3. **エクスポート フィールド > すべて追加**]をクリックします。
4. 以下の各フィールドを選択し、上矢印  を使ってそれらを **選択したフィールド**]リストの最上部に移動します。
 - SAMPLE_RECORD_NUMBER
 - STRATUM
 - invoice_amount
 - AUDIT_VALUEここに表示される順序は変更しないでください。
5. **OK**]をクリックします。
6. **エクスポート形式**]ドロップダウンリストから "Excel(*.xlsx) " を選択します。
7. **保存先**]フィールドに「**Invoices_sample_audited**」を入力し、**OK**]をクリックします。
8. **保存先**]リンクをクリックして Excel ファイルを開きます。

メモ

Analytics は閉じないでください。

3 サンプルで分析を実行する

このチュートリアルでは、以下を行うものとします。

1. **Invoices_sample_audited** テーブルにある顧客に問い合わせます。
2. 売掛金を確認し、虚偽表示を記録します。

Excel ファイルの更新

1. 次の表内の **AUDIT_VALUE** フィールドにリストされた値を使って、Excel ファイルを更新します。
2. Excel ファイルを保存して閉じます。

以下を指定します。

- 層ごとに3つの虚偽表示
- 上位確実性層に1つの虚偽表示
- 過剰計上と過少計上

メモ

簡単にExcel ファイルを更新し、後続の評価の効果を明確にするには

- ファイルの先頭の数レコードのみを更新します。
- 過少計上、または過少計上と過剰計上のミックスをそれぞれ特定の層にグループ化します。

実稼働環境では、過剰計上と過少計上は複数の層およびファイル全体に散在している可能性があります。

ヒント

次の表全体をコピーし、空のExcel ワークシートに貼り付けます。次に、監査額を **Invoices_sample** ワークシートの **AUDIT_VALUE** 列にコピーします。

あるいは別の方法として、[監査額の列](#)を内容とするテキスト ファイルをダウンロードすることもできます。

SAMPLE_RECORD_NUMBER	STRATUM	請求金額	AUDIT_VALUE
1	3	9,394.55	9,494.55
2	5	27,033.66	17,033.66
3	4	22,617.90	22,917.90
4	2	4,576.24	4,575.83
5	1	4,039.67	0.00
6	3	13,753.12	31,753.12
7	4	23,633.12	23,433.12
8	5	33,663.50	33,660.00
9	2	7,136.79	6,136.79
10	2	4,495.13	0.00
11	1	1,575.87	1,075.87
12	0	44,379.67	34,379.67
13(監査額は不変)	0	35,159.99	35,159.99
14	5	27,204.08	27,200.00
15	4	20,156.50	20,000.00
16(監査額は不変)	0	37,448.07	37,448.07

SAMPLE_RECORD_NUMBER	STRATUM	請求金額	AUDIT_VALUE
17	3	11,879.05	11,889.05
18	1	994.98	964.98

4 母集団全体に対する分析結果の適用

更新済みサンプルテーブルのインポート

メモ

更新した監査額が含まれた Excel ファイルを閉じます。

1. Analytics で [インポート > ファイル] を選択します。
2. **定義するファイルの選択** ダイアログボックスで、`..\SampleDataFiles\ACL_Rockwood\Invoices_sample_audited` を特定して選択し、**開く** をクリックします。
3. **ファイル形式** ページで、**Excel ファイル** が選択されていることを確認し、**次へ** をクリックします。
4. **データソース** ページで、"Invoices_sample" ワークシートを選択します。
5. これらのオプションが指定したとおりに設定されていることを確認し、**次へ** をクリックします。

- **先頭を行をフィールド名として使用する** チェックボックスがオンになっている。
- **開始する行** に "1" が設定されている
- **すべてのフィールドを文字タイプとしてインポートします** がオフになっている

6. **Excel Import(Excel の全 ワークシートのインポート)** ページで、`invoice_amount` フィールドと `AUDIT_VALUE` フィールドの **型** に "数値" を指定し、**次へ** をクリックします。

フィールドのヘッダーを選択し、それらに割り当てられている型を確認します。 **型** の値が "数値" でなければ "数値" に更新し、 **小数位** フィールドに "2" を指定します。

7. **データファイルを別名で保存** ダイアログボックスで、 **ファイル名** フィールドに `Invoices_sample_audited` を入力し、 **保存** をクリックします。
8. **最終** ページで **完了** をクリックします。
9. テーブルレイアウト名を指定するダイアログボックスで **OK** をクリックします。

新しい Analytics テーブルは、インポートされた Excel ファイルからのデータを使って作成されます。

サンプル分析の結果の評価

1. `Invoices_sample_audited` テーブルがまだ開かれていない場合は、それを開きます。
2. **サンプリング > 従来の変数サンプリング(CVS) > 評価** を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

3. 以下の画面に表示されている値を指定し、[OK]をクリックしてサンプルの分析結果を母集団全体に適用します。

メモ

大部分の値は、CVS 準備段階の出力結果から自動的に入力されます。

自動入力される値がいくつか欠けている場合は、"CVSEVALUATE コマンドの使用(省略可能)" 次のページを参照してください。

4. 省略可能。評価結果を表示した状態のまま、タブをピン留めします。

従来の変数サンプリング処理を進めるときに出力結果をピン留めしておくと、処理が完了後に処理全体を見直すことができます。

CVSEVALUATE コマンドの使用(省略可能)

メモ

本チュートリアルの実行中に Analytics を開いたままにしておく場合は、このセクションをスキップしてください。

詳しく表示

CVS 評価 ダイアログボックスにいくつかの自動入力値が欠けている場合は、CVS サンプル段階から CVS 評価段階までの間に Analytics を閉じていたために値が失われた可能性があります。

この場合、結果を評価するのに、**CVS 評価** ダイアログボックスを使用する代わりに、スクリプトに保存しておいた CVSEVALUATE コマンドを使用することもできます。

1. **CVS_Evaluate** スクリプトを開き、CVSEVALUATE コマンドを次のように更新します。

```
AUDITED AUDIT_VALUE ETYPE DIFFERENCE
```

2. **Invoices_sample_audited** テーブルを開きます。
3. スクリプトを実行するか、本コマンド全体をコマンドラインにコピーして Enter キーを押します。

スクリプトを実行する場合は、ログ内の CVSEVALUATE コマンドをダブルクリックして評価結果を開きます。

ヒント

CVSEVALUATE コマンドを保存していなかった場合は、CVS 準備段階と CVS s サンプル段階を再実行することで、CVS 評価段階に必要な値を再現します。**Invoices** テーブルを開き、ログ内の CVSPREPARE および CVSSAMPLE コマンドをすばやく再実行します。

入力値の意味

推定タイプ	推定タイプ、 差異(推定) を指定する場合です。 Analytics の従来の変数サンプリングには、サンプル分析の結果を母集団全体に対して適用した場合の結果を推定する方法が4つあります。
監査値	AUDIT_VALUE フィールドには監査値(監査額)が格納されます。
信頼度(%) 推定誤謬数 簿価 精度制限 階層境界 母集団(カウント、値) 上位確実性層(カットオフ、	CVS 準備段階では、ユーザーが入力した値または Analytics で計算された値に基づいて、入力値が自動的に入力されます。

カウント、値)	
---------	--

推定/適用される分析結果の意味

CVS準備 CVSサンプル CVS評価 Invoices_sample_audited

日時: 2019/12/05 10:40:31

コマンド: CVSEVALUATE BOOKED INVOICE_AMOUNT AUDITED AUDIT_VALUE ETYPE DIFFERENCE STRATA 4376.88,9248.74,16904.52,23864.32 POPULATION 1279,3382131,93,898,5693215.11,763,9987014.57,627,12657163.59,479,13346354.63 CONFIDENCE 95.00 CUTOFF 35000.00,36,1334318.88 ERRORLIMIT 6 PLIMIT BOTH TO SCREEN

テーブル: Invoices_sample_audited

評価方法:	DIFFERENCE
信頼度:	95%
点推定値	46,253,254.06
標準推定誤差	344,964.34
t分布	1.96
精度	676,130.11
推定の%としての精度	1.46

推定合計監査値		
下限	点推定値	上限
45,577,123.95	46,253,254.06	46,929,384.17
-676,130.11		676,130.11

推定誤差総額		
下限	点推定値	上限
-823,074.76	-146,944.65	529,185.46
-676,130.11		676,130.11

統計サマリー

計算	サンプル	高価値項目	合計
推定合計監査値	44,928,935.18	1,324,318.88	46,253,254.06
推定合計監査誤差	-136,944.65	-10,000.00	-146,944.65
実現された精度	676,130.11	0.00	676,130.11
合計監査値上限	45,605,065.29	1,324,318.88	46,929,384.17
合計監査値下限	44,252,805.07	1,324,318.88	45,577,123.95
合計誤差上限	539,185.46	-10,000.00	529,185.46
合計誤差下限	-813,074.76	-10,000.00	-823,074.76

層のブレイクダウン

階層番号	階層境界	母集団項目	母集団簿価	サンプル項目	サンプル簿価	サンプル監査値	サンプル差異値	推定合計監査値	標準推定誤差	推定標準偏差	平均エラー	エラーカウント
1	4376.88	1,279	3,382,131.93	37	98,112.90	93,543.23	4,569.67	3,224,169.55	138,158.63	666.78	-123.50	3
2	9248.74	898	5,693,215.11	36	206,801.51	201,305.97	5,495.54	5,556,131.92	111,859.54	762.84	-152.65	3
3	16904.52	763	9,987,014.57	49	623,529.81	641,639.81	-18,110.00	10,269,013.14	271,106.07	2,571.14	369.59	3
4	23864.32	627	12,657,163.59	36	723,926.60	723,870.10	56.50	12,656,179.55	6,738.62	66.42	-1.57	3
5	< 35000.00	479	13,346,354.63	39	1,069,044.86	1,059,037.28	10,007.58	13,223,441.02	117,712.04	1,601.25	-256.60	3
小計		4,046	45,065,879.83	197	2,721,415.68	2,719,396.39	2,019.29	44,928,935.18	344,964.34	5,668.43	-164.74	15
上	>= 35000.00	36	1,334,318.88	36	1,334,318.88	1,324,318.88	10,000.00	1,324,318.88	0.00	0.00	-277.78	1
合計		4,082	46,400,198.71	233	4,055,734.56	4,043,715.27	12,019.29	46,253,254.06	344,964.34	5,668.43	-442.52	16

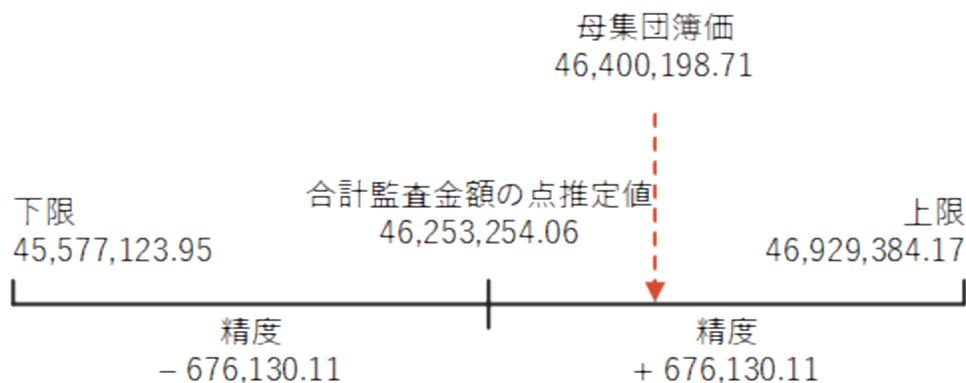
<p>点推定値 精度</p>	<p>点推定値(46,253,254.06) - Invoices テーブル全体の推定監査額を統計的に推定した値</p> <p>精度(676,130.11) - 点推定値が取り得る金額の幅を統計的に推定した値</p> <ul style="list-style-type: none"> これらの統計的推定は Invoices_sample_audited テーブルの監査額をベースにしています。 点推定値は推定された範囲の midpoint です。 この範囲の上限と下限は、点推定値±精度によって得られます。
<p>推定合計監査値</p>	<p>推定合計監査値(推定監査総額)の範囲を視覚的に表したものの 範囲の使用方法</p> <ul style="list-style-type: none"> Invoices テーブルの母集団簿価が範囲内にある場合： この勘定の表示が公正である可能性が非常に高くなります。 Invoices テーブルの母集団簿価が範囲の上限または下限の外部にある場合： この勘定には重大な虚偽表示がある可能性があります。
<p>推定誤謬総額</p>	<p>推定誤謬総額の範囲を視覚的に表したものの 推定誤謬総額の範囲を計算する方法</p> <ul style="list-style-type: none"> 誤謬総額の点推定値は、推定合計監査値(推定監査総額)から母集団簿価を差し引いたものです。 この範囲の上限と下限は、点推定値±精度によって得られます。 マイナスの誤謬額は過剰計上の誤謬を表し、プラスの誤謬額は過少計上の誤謬を表します。 <p>範囲の使用方法</p> <ul style="list-style-type: none"> 推定誤謬総額の範囲がゼロ(0)±928,003.97(CVS準備段階で指定した金額精度)の範囲内の場合： この勘定の表示が公正である可能性が非常に高くなります。 推定誤謬総額範囲の上限または下限が金額精度範囲の上限または下限の外部にある場合： 勘定の表示に重大な虚偽表示がある可能性があります。

請求書レコード全体の表示が公正であるかどうかを判定する

請求書レコード全体の表示が公正であるかどうかを判定するには、2つの方法があります。1つは推定監査総額(推定合計監査値)を見る方法で、もう1つは推定誤謬総額を見る方法です。

推定合計監査値

母集団簿価 46,400,198.71 が推定合計監査値の範囲内に含まれるため、この勘定全体の表示は公正である可能性が非常に高くなります。

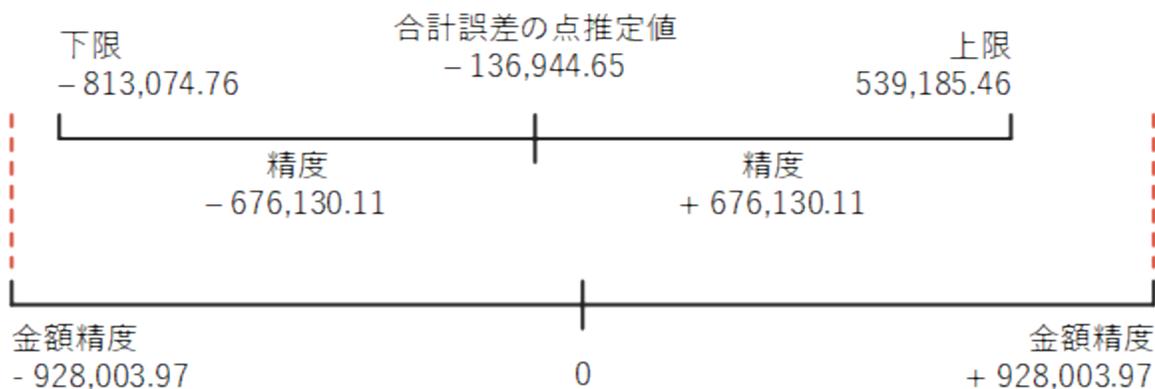


評価結果に次のような意見を作成できるサポートが行われています。

勘定の真の監査金額が、勘定簿価 46,400,198.71 が含まれる 45,577,123.95 ~ 46,929,384.17 の間となる確率が95%あります。このため、勘定の金額表示は公正であると見なされます。

推定誤謬総額

推定誤謬総額の範囲が金額精度の範囲内に含まれるため、この勘定全体の表示が公正である可能性が非常に高くなります。



評価結果に次のような意見を作成できるサポートが行われています。

勘定残高の虚偽表示額が - 813,074.76 ~ 539,185.46 となる確率が95%あります。これは金額精度 $\pm 928,003.97$ の範囲を逸脱していません。このため、勘定の金額表示は公正であると見なされます。

従来の変数サンプリングの準備

データセットをサンプリングする前に、レコードが格納されているテーブルを階層化し、各層の統計的に有効なサンプルサイズを計算します。

Analytics の CVS 準備機能は、ユーザーが提供する入力値に基づき必要な値を計算します。

サンプルサイズを計算することの重要性

後続のサンプルの有効性を計るには、適切なサンプルサイズを計算することが重要になります。サンプルが有効でない、または代表を表していない場合は、全母集団に対してサンプルで実行する監査手続きの結果を信頼して予測することはできません。

サンプルサイズの計算を省略したり、サンプルサイズを推定しないでください。

サンプルサイズの計算に使用するほとんどの入力値は、専門的な判断に基づきます。運用環境でサンプリングの結果を信頼する前に、値が示す意味を十分理解するようにしてください。不明な点がある方は、監査サンプリング担当者、または監査サンプリングスペシャリストにお問い合わせください。

数値の長さ制限

従来の変数サンプリングの準備段階では、いくつかの内部計算が行われます。これらの計算では、最大 17 桁の数値がサポートされます。計算結果が 17 桁を超える場合には、その計算結果が出力に含まれなくなるため、サンプリング処理を続行できなくなります。

注意：17 桁未満のソースデータの数値から、17 桁を超える内部計算結果が生成される場合もあります。

精度上限の仕組み

従来の変数サンプリングを準備するときには、**精度上限**について、以下の一覧のオプションを 1 つ選択する必要があります。

選択したオプションは、サンプリング処理の CVS 評価ステージ中に生成する推定範囲の種類を指定します。

準備ステージ中にはオプションを選択する必要があります。選択するオプションは、サンプルサイズの計算の要件の 1 つであるためです。

精度制限	CVS 評価中にこの種類の推定の場合：
BOTH: 上限および	点推定と上下限のある両面範囲

精度制限	CVS 評価中にこの種類の推定の場合：
下限	
UPPER: 上限	点推定と上限のある片面範囲
LOWER: 下限	点推定と下限のある片面範囲

詳しく表示

範囲の面がないこと

範囲の面がないことは、従来の変数サンプリングの基礎を構成する正規分布またはベル曲線から発生します。

両面範囲

全体として過剰表示または過小表示されている可能性がある勘定を検査する場合は、一般的に、いずれかの方向の虚偽表示が許容可能または承認可能であると判断される虚偽表示金額を超えているかどうかということに関心があります。

両面範囲または推定が必要です。

- 下限は、指定する信頼度について勘定に存在しうる過剰表示の最大金額の推定です
- 上限は、指定する信頼度について勘定に存在しうる過小表示の最大金額の推定です

片面範囲

全体として過剰表示または過小表示されている可能性が高い勘定を検査する場合は、一般的に、1つの方向の虚偽表示が許容可能または承認可能であると判断される虚偽表示金額を超えているかどうかということにのみ関心があると考えられます。

片面範囲または推定を使用できます。

- 下限のみの範囲は、指定する信頼度について勘定に存在しうる過剰表示の最大金額の推定です
- 上限のみの範囲は、指定する信頼度について勘定に存在しうる過小表示の最大金額の推定です

両面範囲または片面範囲を使用すべきですか。

CVS 評価ステージ中には、両面範囲の使用は、より慎重な選択です。両面範囲では、勘定の虚偽表示の全体的な方向に関係なく、勘定が公正に表示されているかどうかを判断できます。

片面範囲を使用する利点は、必要なサンプルサイズを削減できることです。これにより、サンプルデータの分析にかかる作業とコストが削減されます。削減金額はさまざまですが、一般的には50%未満です。

片面範囲を使用するリスクは、勘定の虚偽表示の全体的な方向が間違っている場合に、検査されていない方向における重大な虚偽表示を見逃し、勘定残高の公正さについて誤った判断を行う可能性があるということです。

入力値がサンプルサイズに影響する方法

入力値は Analytics で計算されるサンプルサイズに影響します。[CVS 準備]ダイアログボックスで異なる入力値を実験し、サンプルサイズへの影響を確認できます。

詳しく表示

注意

本番環境では、サンプルサイズを小さくする目的だけで、入力値を操作しないでください。入力値は、サンプリングされるデータと監査目的に対して最も適切であるという専門的な判断に基づいているべきです。

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
階層数	 特定の点の額、層数を増やしても、ほとんどあるいはまったくサンプルサイズに影響しません。	
セル数	 データの本質に応じてサンプルサイズを増やすか減らし、全体的にサンプルサイズを小さくします 特定の点の額、セル数を増やしても、ほとんどあるいはまったくサンプルサイズに影響しません。	
層のサンプルサイズの最小値		 最小しきい値が1つ以上の層に適用される場合はサンプルサイズを大きくします
最低合計サンプルサイズ		 最小しきい値が適用される場合はサンプルサイズを大きくします
上位確実性層カットオフ	 データの本質に応じてサンプルサイズを増やすか減らします データの値が比較的均等に分布している場合は、上位確実性層カットオフを大きくすると、サンプルサイズが小さくなります	

この入力値を増やす:	サンプルサイズを小さくします	サンプルサイズを大きくします
下位確実性層カットオフ		<p>✓</p> <p>データの本質に応じてサンプルサイズを増やすか減らします</p> <p>データの値が比較的均等に分布している場合は、下位確実性層カットオフを大きくすると、サンプルサイズが大きくなります</p> <p>メモ マイナスのカットオフ値の場合、"大きくする"とはゼロ(0)に近づけることを意味します。</p>
信頼度 (%)		✓
金額精度	✓	
推定誤差数	サンプルサイズには影響がありません	
精度制限	両方は上または下よりも大きいサンプルサイズが必要です	

手順

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. サンプルサイズを指定する予定の簿価を含むテーブルを開きます。
2. [サンプルサイズ > 従来の変数 サンプルサイズ (CVS) > 準備] を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

3. [メイン] タブで [簿価] ドロップダウンリストから簿価フィールドを選択します。
4. [精度制限] ドロップダウンリストで、該当するオプションを選択します。

- BOTH: 上限および下限
- UPPER: 上限
- LOWER: 下限

メモ

オプションは以下で詳細に説明します。

5. サンプルデザインを準備するために使用する入力値を入力します。

- 階層数
- セル数
- 層のサンプルサイズの最小値
- 最低合計サンプルサイズ
- 上位確実性層カットオフ
- 下位確実性層カットオフ
- 信頼度 (%)
- 金額精度
- 推定誤差

メモ

入力値は以下で詳細に説明します。

6. 省略可能。現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用してIF ステートメントを作成します。

注意

条件式を指定する場合、サンプルサイズの計算時とサンプルの抽出時とで、同じ条件式を使用する必要があります。

一方の段階で使用した条件を他方の段階で使用しない場合、つまり2つの条件が同一でない場合、サンプリング結果が通常、統計的に無効になります。

7. **出力]タブ**

- a. **[o]** パネルで、次のいずれかを選択します。
- **画面** -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

- **ファイル** - は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。

b. 出力タイプとして **ファイル]** を選択した場合、次のいずれかを実行します。

- **名前]** テキスト ボックスにファイル名を入力します。
- **名前]** ボタンをクリックして、**保存]** または **ファイルを保存する名前]** ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりする

こともできます。たとえば、`C:\Results\Output.txt` または `Results\Output.txt` のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、ASCII テキスト ファイルまたは Unicode テキスト ファイルのいずれかのみです。

8. [OK] をクリックします。

CVS 準備出力結果が画面表示されるか、ファイルに保存されます。

値が入力済みの CVSSAMPLE コマンドのバージョンは表示に含まれます。

メモ

出力結果は以下で詳細に説明します。

CVSEVALUATE コマンドを後で使用するために保存しておきます(任意)。

従来の変数サンプリングによるサンプルを準備したら、CVSSAMPLE コマンドを再利用のために保存しておく便利です。

1. CVS 準備の表示領域の下部にある CVSSAMPLE へのリンクをクリックすることで、コマンドをコマンドラインに読み込みます。
2. コマンドラインに読み込んだコマンドの全体をコピーし、Analytics スクリプトに保存します。

サンプルを準備したら、[CVS サンプル] ダイアログ ボックスまたは CVSSAMPLE コマンドを使ってレコードのサンプルを抽出できます。

メモ

CVSSAMPLE コマンドを実行する場合は、そのコマンドに出力テーブルの名前(と、オプションでシード値)を追加する必要があります。

詳細については、「CVSSAMPLE コマンド」 ページ 1703を参照してください。

CVS 準備 ダイアログ ボックスの入力値と結果

次の表は、[CSV 準備] ダイアログ ボックスの入力値と出力結果の詳細を示します。

メインタブ - 入力値

従来の変数サンプリングのチュートリアルからの画面例

入力値 - CVS 準備 ダイアログ ボックス	説明
簿価	監査対象となる簿価が格納されているフィールド。
精度制限	<p>使用する精度制限のタイプ。</p> <p>BOTH: 上限および下限</p> <p>次の場合はこのオプションを選択します。</p> <ul style="list-style-type: none"> 勘定全体が過剰表示または過小表示である可能性がある 方向の虚偽表示が金額精度を超えるかどうかを推定することに関心がある <p>UPPER: 上限</p> <p>次の場合はこのオプションを選択します。</p> <ul style="list-style-type: none"> 全体としての勘定が過小評価されている可能性が高い 過小評価の合計金額が指定された金額精度を超過するかどうかのみを推定する <p>LOWER: 下限</p> <p>次の場合はこのオプションを選択します。</p> <ul style="list-style-type: none"> 全体としての勘定が過大評価されている可能性が高い 過大評価の合計金額が指定された金額精度を超過するかどうかのみを推定する <p>注意</p> <p>選択するオプションがわからない場合は、両方を選択します。詳細については、「精度上限の仕組み」ページ 1121を参照してください。</p>
階層数	<p>データセットを数値的に階層化するために使用する層(小グループ)の数。</p> <p>層の数は下限が1、上限が256です。</p> <p>確実性層を指定する場合は、層数に含まれません。</p>

入力値 - CVS 準備 ダイアログ ボックス	説明
	<p>詳細については、「階層化」 ページ 1095を参照してください。</p> <p>メモ 層数はセル数の50%を超えることができません。</p>
セル数	<p>データセットをあらかじめ階層化するために使用するセルの数。 セルの数は下限が2、上限が999です。 詳細については、「階層化」 ページ 1095を参照してください。</p> <p>メモ セル数は少なくとも層の数の2倍である必要があります。n</p>
層のサンプルサイズ の最小値	各層からサンプリングする最小レコード数。
最低合計サンプル サイズ	データセット全体からサンプリングするレコードの最小数。
上位確実性層カット オフ	<p>省略可能。上位確実性層のカットオフ値。 カットオフ値以上の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。 カットオフ値を指定しない場合は、上位確実性層にレコードが追加されないようにする、次のデフォルトのカットオフ値が使用されます。</p> <div data-bbox="505 1098 1344 1163" style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p>簿価]フィールドの最高額より大きい(>)</p> </div> <p>詳細については、「確実性層」 ページ 1097を参照してください。</p>
下位確実性層カット オフ	<p>省略可能。下位確実性層のカットオフ値。 カットオフ値以下の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。 カットオフ値を指定しない場合は、下位確実性層にレコードが追加されないようにする、次のデフォルトのカットオフ値が使用されます。</p> <div data-bbox="505 1451 1344 1516" style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p>簿価]フィールドの最低額より小さい(<)</p> </div> <p>詳細については、「確実性層」 ページ 1097を参照してください。</p>
信頼度 (%)	<p>必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。 たとえば、95 を指定した場合は、サンプルが実際に95%の確率で母集団を代表しているとお客様が信頼したいということを意味します。 信頼度は「サンプリングリスク」の補数です。信頼度が95%ということはサンプリングリスクが5%ということと同じです。</p> <ul style="list-style-type: none"> ○ 精度の限度が上または下の場合、最低信頼度は55%、最大信頼度は99.5%です。 ○ 精度の限度が両方の場合、最低信頼度は10%、最大信頼度は99.5%です。

入力値 - CVS 準備 ダイアログ ボックス	説明
金額精度	<p>許容虚偽表示および勘定で想定される虚偽表示の間の差異である金額。</p> <p>たとえば、許容虚偽表示が\$29,000 であり、想定される虚偽表意が\$5,800 の場合、23200 (\$23,200 の差異)を入力します。</p> <p>金額精度は、勘定が公正に表記されていると見なされる許容度の範囲を確立します。</p>
推定誤謬数	<p>省略可能。サンプルで想定する最低誤謬数。</p> <p>この値はCVS 計算で使用されません。代わりに、次の状況のいずれかで、通知を発生させるために使用されます。</p> <ul style="list-style-type: none"> 推定誤謬数(推定誤謬数)が層数未満である CVS 評価段階において、サンプルで見つかった実際の誤謬数が推定誤謬数(推定誤謬数)未満である <p>このような状況のいずれかでは、使用可能な唯一の評価方法は単位あたりの平均です。</p>

出力結果

従来の変数サンプリングのチュートリアルからの画面例

Invoices
CVS 準備
◀ ▶

日時: 2019/12/05 10:34:35

コマンド: [CVSPREPARE ON INVOICE AMOUNT NUMSTRATA 5 MINIMUM 0 PRECISION 928003.97 CONFIDENCE 95.00 CUTOFF 35000 NCELLS 50 PLIMIT BOTH ERRORLIMIT 6 MINSAMPsize 0 TO SCREEN](#)

テーブル: Invoices

金額精度	928,003.97
信頼度	95%
標準誤謬	474,610.19

階層番号	階層境界	母集団項目	パーセントカウント	金額のパーセント	母集団値	サンプル項目	差異	標準偏差	平均
1	4376.88	1,279	31.33	7.29	3,382,131.93	37	1,022,929.96	1,011.40	2,644.36
2	9248.74	898	22.00	12.27	5,693,215.11	36	1,962,436.76	1,400.87	6,339.88
3	16904.52	763	18.69	21.52	9,987,014.57	49	5,030,465.84	2,242.87	13,089.14
4	23864.32	627	15.36	27.28	12,657,163.59	36	4,011,127.73	2,002.78	20,186.86
5	< 35000.00	479	11.73	28.76	13,346,354.63	39	7,757,896.09	2,785.30	27,862.95
小計		4,046	99.12%	97.12%	45,065,879.83	197	19,784,856.37	9,443.22	70,123.20
下	< 201.00	0	0.00	0.00	0.00	0	0.00	0.00	0.00
上	=> 35000.00	36	0.88	2.88	1,334,318.88	36	4,514,605.06	2,124.76	37,064.41
合計		4,082	100%	100%	46,400,198.71	233	24,299,461.43	11,567.98	107,187.61

関連付けられた CVSSAMPLE コマンド

[CVSSAMPLE ON invoice amount NUMSTRATA 5 CUTOFF 35000.00 STRATA 4376.88,9248.74,16904.52,23864.32,35000.00 SAMPLESIZE](#)

出力結果 - CVS 準備	説明
金額精度	入力として指定した金額精度。
信頼度	入力として指定した信頼度。
階層番号	各層に割り当てられた連続で増える数。 また、確実性層には次の数値が割り当てられます(ただし、この画面には表示されていません)。 <ul style="list-style-type: none"> ○ -1 - 下位確実性層 ○ 0 - 上位確実性層
階層境界	確実性層のカットオフ値と各層の上限の境界。 簿価は以下の場合に層に割り当てられます。 <ul style="list-style-type: none"> ○ 上限の境界より小さいか、上限の境界に等しい ○ すぐ下の境界より大きい 簿価はカットオフ値以上の場合に確実性層に割り当てられます。 簿価はカットオフ値以上の場合に上位確実性層に割り当てられます。
母集団項目	テーブルのレコード数。確実性層などの層にブレイクダウンされます。
パーセントカウント	確実性層などの各層に含まれるレコード数の割合。
金額のパーセント	合計簿価に対する割合。確実性層などの各層に含まれます。
母集団値	テーブルの合計簿価。確実性層などの層にブレイクダウンされます。
サンプル項目	合計必要サンプルサイズ。層にブレイクダウンされます。確実性層のすべての項目を含みます。
関連付けられた CVSSAMPLE コマンド	CVS サンプル段階を実行するためのコマンド。CVS 準備段階の値があらかじめ入力されています。

従来の変数サンプリングの実行

アクティブなテーブルにある金額データの代表サンプルを含むテーブルを新たに作成することができます。従来の変数サンプリングは、次のいずれかに関心がある場合に適しています。

- ファイルの合計監査済み金額
- ファイルの金額の虚偽表示の合計金額

Analytics は読み取り専用アプリケーションであるため、レコードのサンプルを抽出した後は、サンプルテーブルを Excel などデータを入力できるアプリケーションにエクスポートし、監査値を追加できるようにする必要があります。

従来の変数サンプリングの正しいプロセスを実行

レコードのサンプルを抽出する前に、レコードが格納されているテーブルを階層化し、各層の統計的に有効なサンプルサイズを計算します。

詳細については、「従来の変数サンプリングの準備」 ページ 1121を参照してください。

レコードのサンプルを描画する

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. サンプリングする予定の簿価を含むテーブルを開きます。
2. **サンプリング > 従来の変数サンプリング (CVS) > サンプル** を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

[CVS サンプル銀] ダイアログ ボックスが開きます。サンプリング段階の入力として、CVS 準備段階の出力結果を使用する場合は、フィールドのほとんどに必要な値が自動的に入力されます。

自動入力される値がいくつか欠けている場合は、以下を行ってください。

- 値を再生成するために、ログに記録されている CVSPREPARE コマンドを再実行する

- CVS 準備段階で生成された CVSSAMPLE コマンドを保存してある場合は、それを実行する

メモ

保存してある CVSSAMPLE コマンドを実行する場合は、そのコマンドに出力テーブルの名前(と、オプションでシード値)を追加する必要があります。
詳細については、"CVSSAMPLE コマンド" ページ 1703を参照してください。

3. あらかじめ入力された値を使用していない場合、または1つ以上の値を調整する場合は、必要な値を入力または更新します。

- 簿価
- 階層数
- 上位確実性層カットオフ
- 下位確実性層カットオフ
- 階層境界
- サンプルサイズ
- 母集団(カウント、値)

メモ

入力値は以下で詳細に説明します。

注意

通常は、入力された値を変更しないでください。あらかじめ入力された値を変更すると、サンプリングプロセスの統計の有効性が否定される可能性があります。

4. 省略可能。[シード]を選択し、数値を入力します。
シード値については、以下で説明します。
5. CVS 準備段階で条件式を使用した場合は、必ず [f] テキスト ボックスにも同じ式を指定してください。

メモ

CVS 準備段階で指定した If 条件は、CVS サンプル段階に自動的に伝達されません。

条件式を使用する場合は、両方の段階で同じ式を使用することで、サンプリング結果が統計的に有効になるようにする必要があります。

6. 保存先]テキスト ボックスで、出力結果を含める Analytics テーブルの名前を指定します。

ヒント

簿価テーブル名を使用し、サフィックス `_sample` を追加します。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

7. [OK] をクリックします。

レコードのランダム サンプルが簿価テーブルから抽出され、指定した出力テーブルに保存されます。

サンプルプロセスと出力結果のサマリーは画面に表示されます。

入力済みバージョンの CVSEVALUATE コマンドは表示に含まれます。

メモ

出力結果は以下で詳細に説明します。

CVSEVALUATE コマンドを後で使用するために保存しておきます(任意)。

サンプリングしたレコードを一度分析したら、使用した CVSEVALUATE コマンドを保存しておくで便利です。

1. CVS サンプルの表示領域の下部にある CVSEVALUATE リンクをクリックすることで、このコマンドをコマンドラインに読み込みます。
2. コマンドラインに読み込んだコマンドの全体をコピーし、Analytics スクリプトに保存します。

サンプリングされたデータを分析し、監査値を追加した後は、[CSV 評価] ダイアログボックスまたは CVSEVALUATE コマンドを使用して、分析結果を母集団全体に対して推定します。

メモ

CVSEVALUATE コマンドを使用する場合は、監査値フィールドの名前と評価タイプを更新する必要があります。

詳細については、「CVSEVALUATE コマンド」ページ 1694 を参照してください。

CVS サンプル ダイアログ ボックスの入力値と結果

次の表は、[CSV サンプル] ダイアログ ボックスの入力値と出力結果の詳細を示します。

メインタブ - 入力値

従来の変数サンプリングのチュートリアルからの画面例

CVS サンプル

メイン

簿価 階層数 シード値

上位確実性層カットオフ

下位確実性層カットオフ

階層境界 サンプル サイズ 母集団(カウント、値)

4376.88	37	1279,3382131.93
9248.74	36	898,5693215.11
16904.52	49	763,9987014.57
23864.32	36	627,12657163.59
35000.00	39	479,13346354.63

入力値 - CVS サンプルダイアログボックス	説明
簿価	監査対象となる簿価が格納されているフィールド。
階層数	データセットを数値的に階層化するために使用する層(小グループ)の数。 確実性層を指定する場合は、層数に含まれません。
上位確実性層カットオフ	上位確実性層のカットオフ値。 カットオフ値以上の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。
下位確実性層カットオフ	下位確実性層のカットオフ値。 カットオフ値以下の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。

入力値 - CVS サンプルダイアログボックス	説明
階層境界	データセットを階層化するために使用する境界値。
サンプルサイズ	各層からサンプリングするレコード数。
母集団(カウント、値)	各層のレコード数と各層の合計値。
シード値	<p>省略可能。任意の数値。 Analytics で乱数ジェネレーターを初期化するために使用される数値です。</p> <p>ヒント シード値を Analytics にランダムに選択させたい場合は、シード値をブランクのままにします。</p>

出力結果

従来の変数サンプリングのチュートリアルからの画面例

Invoices
CVS 準備
CVS サンプル

日時: 2019/12/05 10:37:58

コマンド: [CVSSAMPLE ON INVOICE_AMOUNT NUMSTRATA 5 SEED 12345 CUTOFF 35000.00 STRATA 4376.88,9248.74,16904.52,23864.32,35000.00 SAMPLESIZE 37,36,49,36,39 POPULATION 1279,3382131.93,898,5693215.11,763,9987014.57,627,12657163.59,479,13346354.63 TO "INVOICES_SAMPLE"](#)

テーブル: Invoices

シード値	12345
簿価フィールド	invoice_amount
選択方法	RANDOM

階層番号	階層境界	母集団項目	母集団値	サンプル項目	サンプル値
1	4376.88	1,279	3,382,131.93	37	98,112.90
2	9248.74	898	5,693,215.11	36	206,801.51
3	16904.52	763	9,987,014.57	49	623,529.81
4	23864.32	627	12,657,163.59	36	723,926.60
5	< 35000.00	479	13,346,354.63	39	1,069,044.86
小計		4,046	45,065,879.83	197	2,721,415.68
上	>= 35000.00	36	1,334,318.88	36	1,334,318.88
合計		4,082	46,400,198.71	233	4,055,734.56

関連付けられた CVSEVALUATE コマンド:
[CVSEVALUATE BOOKED invoice_amount AUDITED invoice_amount ETYPE MPU STRATA 4376.88,9248.74,16904.52,23864.32 POPULATI](#)

出力結果 - CVS サンプル	説明
シード値	Analytics 乱数ジェネレーターを初期化するために使用されるシード値。 シード値は自分が指定するか、Analytics によってランダムに選択されます。
簿価フィールド	入力として指定した簿価フィールド。
選択方法	RANDOM - Analytics によって使用されるサンプル選択方法。
階層番号	各層に割り当てられた連続で増える数。 また、確実性層には次の数値が割り当てられます(ただし、この画面には表示されていません)。 <ul style="list-style-type: none"> ◦ -1 - 下位確実性層 ◦ 0 - 上位確実性層
階層境界	確実性層のカットオフ値と各層の上限の境界。 簿価は以下の場合に層に割り当てられます。 <ul style="list-style-type: none"> ◦ 上限の境界より小さいか、上限の境界に等しい ◦ すぐ下の境界より大きい 簿価はカットオフ値以上の場合に確実性層に割り当てられます。 簿価はカットオフ値以上の場合に上位確実性層に割り当てられます。
母集団項目	層でブレイクダウンされ、確実性層を含む、ソーステーブルのレコードの数。
母集団値	層でブレイクダウンされ、確実性層を含む、ソーステーブルの合計簿価。
サンプル項目	層で分解された、サンプルで抽出された合計レコード数。 確実性層のすべてのレコードは出力テーブルで自動的に抽出され、含まれます。
サンプル値	層でブレイクダウンされ、確実性層レコードの値を含む、サンプルで抽出されたレコードの合計額。
関連付けられた CVSEVALUATE コマンド	CVS 評価段階を実行するためのコマンド。CVS 準備段階と CSV サンプル段階の値があらかじめ入力されています。

システム生成フィールド

Analytics により、自動的に4つのフィールドが生成され、サンプル出力テーブルに追加されます。サンプルに追加される各レコードのフィールドには、以下の記述的情報が格納されます。

- 層 - レコードが割り当てられる層の数
- 元のレコード番号 - ソースデータテーブル内の元のレコード番号
- 選択順序 - 各層においてレコードがランダムに選択された順序
- サンプルレコード番号 - サンプル出力テーブル内のレコード番号

Audit_Value(監査額)フィールドを追加し、サンプルテーブルをエクスポートします。

Analytics は読み取り専用アプリケーションであるため、サンプルレコードから成るテーブルを Excel などデータを入力できるアプリケーションにエクスポートする必要があります。サンプリングされたデータを分析する過程で検出した監査額をすべて外部アプリケーションに追加できます。

監査額が追加された時点で、テーブルを Analytics にインポートします。

テーブルをエクスポートする前には、簿価フィールドを複製するフィールドを追加します。必要に応じて、この複製されたフィールドを外部アプリケーションで編集できます。

監査額フィールドを追加します。

1. 簿価テーブルを閉じます。
2. サンプリングされたレコードを含む新しく作成されたテーブルを開きます。

例: <簿価テーブル>_sample

3. このコマンドをコピーし、コマンドラインに貼り付けます。

```
DEFINE FIELD AUDIT_VALUE COMPUTED 簿価フィールド
```

コマンドラインが表示されない場合、[ウィンドウ>コマンドライン]を選択します。

4. 簿価フィールドを簿価を含むテーブルのフィールドの実際の名前で置換します。

例:

```
DEFINE FIELD AUDIT_VALUE COMPUTED 請求金額
```

メモ

2つの名前が異なる場合は、表示名ではなく、フィールドの物理名を使用します。

5. Enter を押し、新しいフィールドを作成します。

ヒント

AUDIT_VALUE(監査額)フィールドが作成されたことを確認する場合は、コマンドラインで「display」を入力し、Enter を押してテーブルのフィールドのリストを表示します。

サンプルテーブルを Excel にエクスポートする

サンプルテーブルを Excel にエクスポートするか、または他の適した外部アプリケーションで使用するために区切り文字付きファイルにエクスポートします。

エクスポートを実行するとき:

- **[フィールド]** オプションを選択し、エクスポートするすべてのフィールドを選択します
- 追加されたサフィックス **_audited** が付いたサンプルテーブル名など、後からわかりやすいエクスポートされたテーブルの名前を指定します

例: **<簿価テーブル>_sample_audited**

詳細については、"データのエクスポート" ページ 210を参照してください。

従来の変数サンプリングの誤謬の評価

サンプリングされたデータのセットに対して監査手続きを実行した後は、Analytics を使用して次のことができます。

- 勘定全体に対してサンプルの監査金額を推定する
- 勘定全体に対して見つけたサンプルの虚偽表示を推定する
- 推定された合計監査金額と推定された合計虚偽表示金額の上限と下限を計算する

誤謬が見つからない場合でも、評価機能を使用して、サンプリングリスクの基本的な許容度を計算します。この場合、推定タイプとして平均推定を使用する必要があります。

従来の変数サンプリングの正しいプロセスを実行

従来の変数サンプリングのプロセスにおける最終段階は、エラーを評価することです。エラーを評価できるようになるには、これまでの段階を実行し終わっている必要があります。

詳細については、次を参照してください。

- "従来の変数サンプリングの準備" ページ 1121
- "従来の変数サンプリングの実行" ページ 1131

どの推定タイプを使用すればよいですか？

CVS 評価段階では、4 つの異なる推定タイプ(評価方法)から選択できます。

- 平均推定
- 差異
- 比率分離
- 比率結合

使用しなければならない推定タイプは、データの性質、サンプル簿価、サンプル監査金額、およびそれらの関係によって決まります。

ガイドライン

推定タイプの選択に役立つガイドラインを次に示します。

ヒント

様々な推定タイプによって生成された結果同士を比較するには、**推定タイプ]**ドロップダウンリストで**"すべて"**を選択します。**"すべて"**を選択することで、評価出力にすべての推定タイプが含まれます。

詳しく表示

推定タイプ	虚偽表示の存在	虚偽表示のサイズ	簿価の符号	層間の比率の比較
平均推定	<p>虚偽表示がないか、非常に少ない虚偽表示しかない。</p> <p>監査したサンプル母集団に虚偽表示がないか、非常に少ない場合に、唯一有効な推定タイプ</p>	該当なし	該当なし	該当なし
差異	<p>虚偽表示が必要</p> <p>監査したサンプル母集団にはいくつかの虚偽表示が必要です。</p> <p>たとえば、サンプルの5%以上に虚偽表示が含まれているなどの場合です。</p>	<p>虚偽表示が比例しない場合</p> <p>虚偽表示が関連する簿価に比例しない場合、つまり虚偽表示のサイズが関連する簿価のサイズに比例しない場合に適しています。</p> <p>つまり、簿価の大小に関係なく、それに含まれる虚偽表示が大きかったり小さかったりする場合があります。</p>	該当なし	該当なし
比率分離		<p>虚偽表示が関連する簿価に比例する場合</p> <p>虚偽表示が関連する簿価に比例する場合、つまり虚偽表示のサイズが関連する簿価のサイズに比例する場合に適しています。</p> <p>つまり、簿価が小さければ虚偽表示が小さく、簿価が大きければ虚偽表示が大きくなる場合です。</p>	<p>すべての簿価が同じ符号を持つ</p> <p>サンプルの全簿価が同じ符号を持つ、つまりすべてプラスか、すべてマイナスかのどちらかである必要があります。</p>	<p>比率が異なる</p> <p>サンプルの監査金額の平均とサンプル簿価の平均との比率が層間で大きく異なる場合に適しています。</p>
比率結合				<p>比率間の差異があまりない</p> <p>サンプルの監査金額の平均とサンプル簿価の平均との比率が層間であまり差異がない場合に適しています。</p>

更新済みサンプルテーブルのインポート

Excel から、または監査額の追加に使用する任意の外部アプリケーションから、更新済みサンプルテーブルを Analytics にインポートします。

テーブルは次を含む必要があります。

- **簿価フィールド** - サンプルを抽出したときに存在していた元の記録された簿価
- **監査金額フィールド** - 監査済み金額。分析の結果に基づいて必要な場合に更新されます

詳細については、"Microsoft Excel データのインポート" ページ 250を参照してください。

サンプル分析の結果の評価

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。これらの文字を使用すると、コマンドを実行できないか、エラーが発生します。

1. インポートした更新済みサンプルテーブルを開きます。
2. **サンプリング > 従来の変数サンプリング (CVS) > 評価**] を選択します。

メモ

メニューオプションは、テーブルが開いていない場合は無効です。

CVS 評価] ダイアログボックスが開きます。評価段階の入力として、CVS 準備段階と CVS サンプル段階の出力結果を使用する場合は、フィールドのほとんどに必要な値が自動的に入力されます。

自動入力される値がいくつか欠けている場合は、以下を行ってください。

- 値を再生成するために、ログに記録されている CVSSAMPLE コマンドを再実行する
- CVS サンプル段階で生成された CVSEVALUATE コマンドを保存してある場合は、それを実行する

メモ

保存してある CVSEVALUATE コマンドを実行する場合は、監査値フィールドの名前と評価タイプを更新する必要があります。

詳細については、"CVSEVALUATE コマンド" ページ 1694を参照してください。

3. **メイン**] タブで、**推定タイプ**] ドロップダウンリストから次のオプションのいずれかを選択します。
 - MPU
 - 差異
 - 比率分離
 - 比率結合

- すべて

メモ

オプションは上記で詳細に説明します。

- あらかじめ入力された値を使用していない場合、または1つ以上の値を調整する場合は、必要な値を入力または更新します。

- 信頼度 (%)
- 推定誤差数
- 簿価
- 監査値
- 精度上限
- 上位確実性層(カットオフ、カウント、値)
- 下位確実性層(カットオフ、カウント、値)
- 階層境界
- 母集団(カウント、値)

メモ

入力値は以下で詳細に説明します。

注意

通常は、入力された値を変更しないでください。あらかじめ入力された値を変更すると、評価プロセスの統計の有効性が否定される可能性があります。

- 出力]タブ

- 【o】パネルで、次のいずれかを選択します。
 - 画面 -- Analytics の表示領域に結果を表示します

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

- ファイル-は結果をテキスト ファイルに保存または追加します
ファイルは Analytics の外部に保存されます。
- 出力タイプとして **ファイル]**を選択した場合、次のいずれかを実行します。
 - 名前]テキスト ボックスにファイル名を入力します。
 - 名前]ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。

Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりする

こともできます。たとえば、`C:\Results\Output.txt` または `Results\Output.txt` のように指定します。

メモ

ファイルタイプ オプションは、使用している Analytics のエディションに応じて、ASCII テキスト ファイルまたは Unicode テキスト ファイルのいずれかのみです。

6. **OK**] をクリックします。

CVS 評価出力結果が表示されるか、ファイルに保存されます。

メモ

出力結果は以下で詳細に説明します。

出力結果の解釈の詳細については、「請求書レコード全体の表示が公正であるかどうかを判定する」ページ 1119を参照してください。

CVS 評価ダイアログ ボックスの入力値と結果

次の表は、**CSV 評価**]ダイアログ ボックスの入力値と出力結果の詳細を示します。

メイン タブ - 入力値

従来の変数サンプリングのチュートリアルからの画面例

CVS 評価

メイン 出力

推定タイプ: DIFFERENCE 信頼度 (%): 95.00 推定誤謬数: 6

簿価: invoice_amount 監査値: AUDIT_VALUE 精度上限: BOTH

上位の確実性層(カットオフ、カウント、値): 35000.00,36,1334318.88

下位の確実性層(カットオフ、カウント、値):

階層境界

階層境界	母集団(カウント、値)
4376.88	1279,3382131.93
9248.74	898,5693215.11
16904.52	763,9987014.57
23864.32	627,12657163.59
	479,13346354.63

OK キャンセル ヘルプ

入力値 - CVS 評価ダイアログボックス	説明
推定タイプ	使用する推定タイプ(評価方法)。
信頼度 (%)	<p>必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。</p> <p>たとえば、95 を指定した場合は、サンプルが実際に95%の確率で母集団を代表しているとお客様が信頼したいということを意味します。</p> <p>信頼度は"サンプリングリスク"の補数です。信頼度が95%ということはサンプリングリスクが5%ということと同じです。</p>
推定誤謬数	<p>サンプルで想定する最低誤謬数。</p> <p>この値はCVS評価計算で使用されません。サンプルで見つかった実際の誤謬数が推定誤謬数を下回る場合に、通知をトリガーするために使用されます。</p> <p>実際の誤謬数が推定誤謬数未満の場合、使用可能な唯一の評価方法は平均推定です。</p>

入力値 - CVS 評価 ダイアログ ボックス	説明
簿価	記録された簿価を含むサンプルテーブルの数値フィールド。
監査値	監査金額を含むサンプルテーブルの数値フィールド。
精度制限	使用する精度制限のタイプ。 詳細については、"従来の変数サンプリングの準備" ページ 1121を参照してください。
上位確実性層(カットオフ、カウント、値)	CVS プロセスで使用された上位確実性層カットオフ値、上位確実性層のレコード数、および合計値。
下位確実性層(カットオフ、カウント、値)	CVS プロセスで使用された下位確実性層カットオフ値、下位確実性層のレコード数、および合計値。
階層境界	データセットを階層化するために使用された境界値。
母集団(カウント、値)	各ソース テーブルのレコード数と各層の合計値。

出力結果

従来の変数サンプリングのチュートリアルからの画面例

CVS準備 CVSサンプル CVS評価 Invoices_sample_audited

日時: 2019/12/05 10:40:31

コマンド: CVSEVALUATE BOOKED INVOICE_AMOUNT AUDITED AUDIT_VALUE ETYPE DIFFERENCE STRATA 4376.88,9248.74,16904.52,23864.32 POPULATION 1279,3382131,93,898,5693215.11,763,9987014.57,627,12657163.59,479,13346354.63 CONFIDENCE 95.00 CUTOFF 35000.00,36,1334318.88 ERRORLIMIT 6 PLIMIT BOTH TO SCREEN

テーブル: Invoices_sample_audited

評価方法:	DIFFERENCE
信頼度:	95%
点推定値	46,253,254.06
標準推定誤差	344,964.34
t分布	1.96
精度	676,130.11
推定の%としての精度	1.46

推定合計監査値		
下限	点推定値	上限
45,577,123.95	46,253,254.06	46,929,384.17
-676,130.11		676,130.11

推定誤差総額		
下限	点推定値	上限
-823,074.76	-146,944.65	529,185.46
-676,130.11		676,130.11

統計サマリー

計算	サンプル	高価値項目	合計
推定合計監査値	44,928,935.18	1,324,318.88	46,253,254.06
推定合計監査誤差	-136,944.65	-10,000.00	-146,944.65
実現された精度	676,130.11	0.00	676,130.11
合計監査値上限	45,605,065.29	1,324,318.88	46,929,384.17
合計監査値下限	44,252,805.07	1,324,318.88	45,577,123.95
合計誤差上限	539,185.46	-10,000.00	529,185.46
合計誤差下限	-813,074.76	-10,000.00	-823,074.76

層のブレイクダウン

階層番号	階層境界	母集団項目	母集団簿価	サンプル項目	サンプル簿価	サンプル監査値	サンプル差異値	推定合計監査値	標準推定誤差	推定標準偏差	平均エラー	エラーカウント
1	4376.88	1,279	3,382,131.93	37	98,112.90	93,543.23	4,569.67	3,224,169.55	138,158.63	666.78	-123.50	3
2	9248.74	898	5,693,215.11	36	206,801.51	201,305.97	5,495.54	5,556,131.92	111,859.54	762.84	-152.65	3
3	16904.52	763	9,987,014.57	49	623,529.81	641,639.81	-18,110.00	10,269,013.14	271,106.07	2,571.14	369.59	3
4	23864.32	627	12,657,163.59	36	723,926.60	723,870.10	56.50	12,656,179.55	6,738.62	66.42	-1.57	3
5	< 35000.00	479	13,346,354.63	39	1,069,044.86	1,059,037.28	10,007.58	13,223,441.02	117,712.04	1,601.25	-256.60	3
小計		4,046	45,065,879.83	197	2,721,415.68	2,719,396.39	2,019.29	44,928,935.18	344,964.34	5,668.43	-164.74	15
上	>= 35000.00	36	1,334,318.88	36	1,334,318.88	1,324,318.88	10,000.00	1,324,318.88	0.00	0.00	-277.78	1
合計		4,082	46,400,198.71	233	4,055,734.56	4,043,715.27	12,019.29	46,253,254.06	344,964.34	5,668.43	-442.52	16

出力結果 - CVS 評価	説明
評価方法	選択した推定タイプ。
信頼度	入力として指定した信頼度。
点推定値	ソーステーブル内のデータセット全体の推定監査額を統計的に推定した値。 点推定値は推定された範囲の midpoint です。
精度	点推定値が取り得る金額の幅を統計的に推定した値。 この範囲の上限と下限は、点推定値±精度によって得られます。
推定合計監査値	<p>推定合計監査値(推定監査総額)の範囲を視覚的に表したものの範囲の使用方法</p> <ul style="list-style-type: none"> ソーステーブルの母集団簿価が範囲内にある場合： この勘定の表示が公正である可能性が非常に高くなります。 ソーステーブルの母集団簿価が範囲の上限または下限の外部にある場合： この勘定には重大な虚偽表示がある可能性があります。
推定誤謬総額	<p>推定誤謬総額の範囲を視覚的に表したものの推定誤謬総額の範囲を計算する方法</p> <ul style="list-style-type: none"> 誤謬総額の点推定値は、推定合計監査値(推定監査総額)から母集団簿価を差し引いたものです。 この範囲の上限と下限は、点推定値±精度によって得られます。 マイナスの誤謬総額は過剰計上の誤謬を表し、プラスの誤謬総額は過少計上の誤謬を表します。 <p>範囲の使用方法</p> <ul style="list-style-type: none"> 推定誤謬総額の範囲がゼロ(0)±(CVS準備段階で指定した金額精度)の範囲内の場合： この勘定の表示が公正である可能性が非常に高くなります。 推定誤謬総額範囲の上限または下限が金額精度範囲の上限または下限の外部にある場合： 勘定の表示に重大な虚偽表示がある可能性があります。

条件付きサンプリング

注意

サンプル時にコマンド フィルタリングまたは範囲パラメーターを適用すると、サンプルの有効性が損なわれます。これを行う場合は、サンプルの結果が無効になる可能性があるという注記がログに生成されます。

[サンプル] ダイアログ ボックスでは、コマンド フィルターや範囲パラメーターを適用できるオプションがありますが、本ガイドではサンプリング手順にこれらの手順を含めていません。

条件付きサンプリングは、特定の場所で発生する取引、特定の製造元で製造された製品など、指定された条件を満たすレコードにサンプル選択を制限するために使用されます。

条件付きサンプリングを実行する場合は、必ず正確なデータセットで作業する必要があります。サンプル時にコマンド フィルターを使用してデータを絞り込むと、予期しない結果が生成される可能性があります。最初に、任意の条件を満たすデータを新しいテーブルに抽出してから、フィルターを使用せずに、新しいテーブルでサンプリングを実行することをお勧めします。

フィルタリングされたデータのサンプリングと、サンプリングされたデータのフィルタリング

条件付きサンプリングを実行するときには、次の差異に注意してください。

- フィルタリングされたデータのサンプリング
- サンプリングされたデータのフィルタリング

ベスト プラクティス: フィルタリングされたデータのサンプリング

1000 件のレコード中 150 件が条件 “Dept 03” を満たすテーブルを考えてください。“Dept 03” から 10 レコードのサンプルを抽出するとします。

最善の方法は “Dept 03” レコードをフィルタリングし、新しいテーブルに先に抽出してから、サンプルを抽出します。次に新しいテーブルをサンプリングし、“Dept 03” レコードのみから抽出します。この方法を使用すると、フィルタリングされたデータをサンプリングします。

回避する方法: サンプリングされたデータのフィルタリング

1000 件のレコード中 150 件が条件 “Dept 03” を満たすテーブルを考えてください。“Dept 03” から 10 レコードのサンプルを抽出するとします。

1000 レコードある元のテーブルから 10 レコードのサンプルを抽出し、処理でコマンド フィルター `IF Dept = "03"` を適用する場合は、サンプリングされたデータをフィルタリングします。

この方法の問題は、まずフィルタリングされていないデータセットから 10 件のレコードが選択されます。その中から “Dept 03” に一致するレコードが選択されるので、サンプルに必要なレコード数が 10 件よりも少なくなってしまいます。サンプルが代表的ではなく、無効です。

同様の理由から、サンプリングされたレコードを含む出力テーブルに対しフィルタリングした場合もサンプルが無効になります。

データの分析

「データの分析」は、さまざまなプロセスと手法を網羅する幅広い概念です。同じデータ分析目標を達成するためには、複数の方法がある場合があります。多くの場合、全体的なプロセスは反復的であり、作業途中で検出した情報に基づいて、初期アプローチを修正していくことが必要になります。

効果的なデータ分析

基本的に、データの分析とは、データに関する情報を見つけ出す処理です。Analytics は、分析を実行し、調査しているデータに関する一般的な洞察を得て、特定の問題に対して回答するために使用できる多数のコマンドおよび他のツールを提供します。ただし、Analytics で 2 つまたは 3 つのボタンをクリックするだけで、すべてのデータ分析の回答が魔法のように得られるということはありません。

効果的なデータ分析の要件：

- データの本質を理解する
- 特定の分析目標を策定する
- 十分な知識を持ってツールを応用する

Analytics は、データ分析機能を大幅に強化することはできますが、それらを置き換えることはしません。

Analytics のデータ分析コマンドとツール

以下の表は、データ分析領域別に、Analytics コマンドとツールを分類します。この分類は絶対的なものではありません。分類外でもコマンドを有効に使用できる場合があります。並べ替えや結合などの一部のコマンドは、主に分析的ではない主要な目的に対応しますが、状況によっては分析の洞察を提供することができます。

メモ

一般的に、基本事項以外に、データ分析では、単一のコマンドを分離して使用するのではなく、一連のコマンドを使用して、分析目標に向かって作業を進める必要があります。

データ分析領域	コマンドまたはツール	説明
一般的な特性	検証 カウント 合計 プロファイル 統計 異常値 並べかえ	<p>これらのコマンドを使用して、次のようなデータセットの一般特性を検出します。</p> <ul style="list-style-type: none"> ○ データの妥当性 ○ レコード数 ○ 合計金額 ○ 最低、最高、および平均金額 ○ 標準偏差、中央値、最頻値、四分位値 ○ 異常値 ○ 範囲 ○ 正の値と負の値の分布

データ分析領域	コマンドまたはツール	説明
	インデックス	<ul style="list-style-type: none"> パターン
信頼性/正確性	演算フィールド	演算フィールドを使用して、税込み合計金額などのデータセットの計算された金額の正確性を再計算およびテストします。
分離	フィルターリング 検索	フィルターリングを使用して、データセットまたはデータ処理を、関心があるレコードのサブセットに制限します 検索を使用して、データセットの特定の値を検索します
順次的な順序	順番検査	データが連続しているかどうかをテストし、順序から外れた項目を特定します
完全性	Gaps/ギャップ	順次的な順序の一連の小切手など、連続するすべてのレコードが存在するかどうかを検証し、順番のギャップの場所を特定します
一意性チェック	Duplicates/重複	フィールドの重複値または項目、あるいは重複するレコード全体を特定します
不正確さ	あいまい重複	同じ現実のエンティティを参照する可能性があるほぼ同一の値を特定します
頻度分布 Materialityの集合化	階層化 年齢調べ 分類化 要約 クロス集計 ヒストグラム クラスター	レコードをグループ化し、数値範囲またはクラスター、期間、ロケーションコード、業者/顧客番号、または製品 ID などのレコード識別子によって集合化されるレコード数と値を決定します 異常値の特定でも役立ちます
比較	結合 曖昧結合 関連付け	テーブルを結合し、請求書テーブルや PO テーブルなどのテーブル全体でレコードが一致しているか一致していないかどうかを検出します。
数値の例外	Benford(ベンフォード分析の実行)	想定されるベンフォード分布からの差異の先頭の数字をテストし、例外の数値データを検出します

データのプロフィール分析

テーブルにある1つ以上の数値フィールドについて、以下の要約統計を表示できます。

- **合計値** - フィールドの値すべての合計。
- **絶対値** - 数値の符号を無視した、フィールドの値の合計です。絶対値は、金額単位サンプリングによってデータをサンプリングするときに、母集団パラメーターのデフォルト値として使用されます。
- **最小値** - フィールドの値の中の最小値。
- **最大値** - フィールドの値の中の最大値。

メモ

最小値と最大値は、データを階層化したりヒストグラムを生成したりするときに、それぞれのパラメーターのデフォルト値としてよく使用されます。

Analytics には自動的にプロフィールを行うオプションがあります。このオプションを有効にすると、テーブルを開くたびにすべての数値型フィールドのプロフィールが自動的に行われます。

手順

データのプロフィール分析を行うと、テーブルにある1つ以上の数値フィールドについて、以下の要約統計を生成することができます。

方法の説明

1. **分析 > プロフィール** の順に選択します。
2. **メイン** タブで、次のいずれかを行います。
 - **プロフィールフィールド** リストからプロフィール対象のフィールドを選択します。
 - **プロフィールフィールド** をクリックして、フィールドを選択するか、または式を作成します。
フィールドを選択する順序が、結果に表示する列の順序になります。
3. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

4. **詳細** タブをクリックします。
5. **適用範囲** パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While テキスト ボックスに条件を入力するか、または While ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

6. **OK**] をクリックします。

自動プロファイルを有効にする

Analytics では、テーブルを開くときに、自動的にプロファイルを実行するよう設定できます。プロファイルの結果を表示するため、表示領域で単独の [プロファイル] タブが生成されます。このオプションが有効な場合は、すべての Analytics テーブルに適用されます。

方法の説明

1. [ツール > オプション] の順にクリックします。
2. **テーブル**] タブをクリックします。
3. **開くときに自動的にプロファイルを実行する**] を選択します。
4. **OK**] をクリックします。

統計の生成

テーブルにある数値フィールドおよび日付時刻フィールドについて、詳細な統計を生成できます。統計はテーブルの概要を示し、データの異常値をハイライトして、後続の分析につなげることができます。

統計を生成するときには、標準出力オプションの他に、出力結果を含む様々なシステム変数が自動的に作成されます。詳細については、"Analytics コマンドによって作成されたシステム変数" ページ 1582を参照してください。

統計の生成結果は次の表で説明します。

メモ

すべての統計は数値フィールドに対して生成されます。日付時刻フィールドにはサブセットの統計のみが生成されます。

統計名	結果
範囲	数値フィールド： <ul style="list-style-type: none"> フィールド内の最高値と最低値の差 日付時刻フィールド： <ul style="list-style-type: none"> 最も古い日付と最新の日付の間の日数
正	<ul style="list-style-type: none"> 正の値の数 正の値の合計(日付時刻フィールドには特に意味がない) 正の値の平均
負	<ul style="list-style-type: none"> 負の値の数 負の値の合計 負の値の平均
ゼロ	ゼロの値の数
合計	<ul style="list-style-type: none"> 正、負およびゼロの値の数の合計 正、負およびゼロのすべての値の合計 正、負およびゼロのすべての値の平均
絶対値	数値の符号を無視した、値すべての合計
標準偏差(オプション)	平均値からの標準偏差
中央値(省略可能)	中央値 <ul style="list-style-type: none"> 奇数の値セット: 中央値 偶数の値セット: 中央にある2つの値の平均
Q25(省略可能)	最初の四分位数値(下四分位数値)

統計名	結果
	<ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
Q75(省略可能)	3 番目の四分位数(下四分位数) <ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
最頻値(省略可能)	最も頻繁に発生する値 <ul style="list-style-type: none"> 2 回以上出現する値がない場合は、"N/A" が表示される 関連付けの場合は、最も低い値が表示される
最高値	最高値から 5 番目までの値
最低値	最低値から 5 番目までの値
<p>ヒント</p> <p>統計 ダイアログボックスの 詳細 タブの 高/低数 を使用して、結果に含まれる高い値と低い値の数を指定できます。</p>	

手順

テーブルにある数値フィールドと日付時刻フィールドについて、記述統計を生成できます。

方法の説明

1. **分析] > 統計]** を選択します。
2. **メイン** タブで、次のいずれかを行います。
 - **統計の対象]** リストから、統計を生成するフィールドを選択します。
 - **統計の対象]** をクリックして、フィールドを選択するか、または式を作成します。
フィールドを選択する順序が、結果に表示する列の順序になります。
3. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

4. 選択したフィールドの標準偏差を計算する場合は、**標準偏差]** を選択します。

5. 選択されたフィールドの中央値、最頻値、および第 1 と第 3 の四分位数の値を計算する場合は、**中央値、最頻値、Q25、Q75** を選択します。

メモ

これらの追加の統計を計算するには、追加のコンピューターメモリが必要です。非常に大きいデータセットの追加の統計を計算すると、コンピューターのメモリを超過し、エラーメッセージが表示される場合があります。

6. **出力]** タブをクリックします。
7. **出力先]** パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果をテキスト ファイルに保存または追加するには、このオプションを選択します。ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

8. 出力の種類として **ファイル]** を選択した場合は、**出力指定]** パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは、使用している Analytics のエディションに応じて、"**ASCII テキスト ファイル**" または "**Unicode テキスト ファイル**" のいずれかのみです。結果を新しいテキスト ファイルに保存するか、または既存のテキスト ファイルに追加します。
 - **名前** - **名前]** テキスト ボックスにファイル名を入力します。もしくは、**名前]** ボタンをクリックして、**保存]** または **ファイルを保存する名前]** ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。
 - **ローカル** - 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。
9. **詳細]** タブをクリックします。
10. **適用範囲]** パネルで、適切な出力オプションを選択します。

- すべて
- 先頭
- 次へ
- While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While テキスト ボックスに条件を入力するか、または While ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>	

11. 結果に含まれるフィールドの最高値と最低値の数を変更したい場合は、**高値/低値の数**]に数字を入力します。
12. 出力の種類として **ファイル**]を選択した場合、出力結果を既存のテキスト ファイルの末尾に追加するときは、**既存のファイルに追加する**]を選択します。
13. **OK**]をクリックします。
14. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

異常値の特定

Analytics で異常値機能を使用すると、正常ではなく、詳細な調査が必要な可能性があるレコードを特定できます。

異常値とは何か

異常値とは、グループのレコードの数値金額とは大幅に異なる数値金額を含むレコードです。

グループの異常値の例

買掛金ファイルで、一般的に、特定の会社からの請求書は、\$500 から \$1,000 の範囲です。しかし、\$8,500 の請求書が 1 つあります。

メモ

正当な理由で、レコードが異常値になることがあります。一般的に、Analytics で特定された異常値をさらに調査し、実際に問題があるかどうかを判断する必要があります。

レコードのグループ化は任意です

データの異常値を調べるときには、レコードをグループ化する必要はありません。特定のグループ内ではなく、テーブル全体で異常値を検出できます。

レコードの全体セットの異常値の例

買掛金ファイルで、一般的に、請求書の全体セットは、\$40 から \$5,000 の範囲です。しかし、\$20,000 を超える請求書が 3 つあります。

異常値が特定される方法

レコードの各グループまたはレコードの全体セットに対して、Analytics は特定の数値フィールドの標準偏差または標準偏差の乗数を使用して、異常値の上限と下限を設定します。

上限より大きいか下限より小さい数値フィールドの値の任意のレコードが、異常値として出力結果に追加されます。

標準偏差はデータセットの分布の測定です。つまり、値の拡散方法を測定します。異常値計算は母集団標準偏差を使用します。

数値のセットの異常値の特定

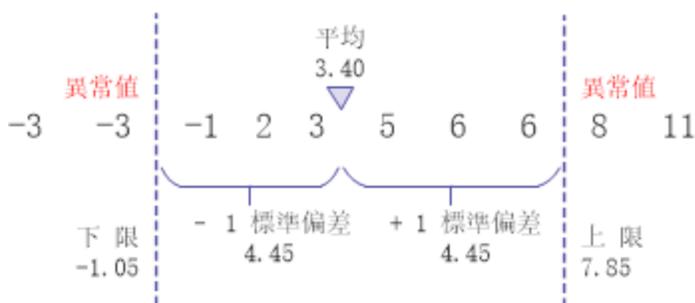
次のセットの数値で異常値を特定します。

-3, -3, -1, 2, 3, 5, 6, 6, 8, 11

数値の平均は3.40です。平均は、セットの標準偏差 (SD) を計算するために使用されず:4.45。

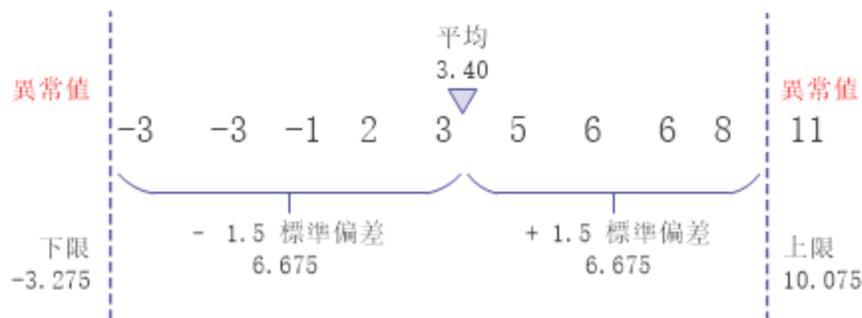
平均 ± 1 標準偏差

最初の例では、平均 ± 1 標準偏差を使用して、異常値の上限と下限を設定します。4つの値が異常値として特定されます。



平均 ± 1.5 標準偏差

2番目の例では、平均 ± 1.5 標準偏差を使用して、異常値の上限と下限を設定します。1つの値のみが異常値として特定されます。



異常値の境界の設定

必要に応じて、異常値の境界を設定するか、異なる位置をテストし、結果を比較できます。

境界を設定するには、異常値フィールドの標準偏差の正の乗数を指定します: 0.5、1、1.5 など。たとえば、1.5 の乗数を指定する場合、異常値の境界は、異常値フィールドの値の平均または中央値の上下 1.5 標準偏差です。

同じデータセットに対し、標準偏差の乗数の値を大きくすると、出力結果の異常値の数が減る可能性があります。

データの分布

数値データのセットの値は、通常、最も小さい値から最も大きい値の範囲で分散します。正規分布では、値はデータの中心点付近に均等に分布し、ベル形状の曲線を描きます。一般的に、中心点は、値の平均として定義されますが、中央値または最頻値のこともあります。

詳しく表示

正規分布の標準偏差

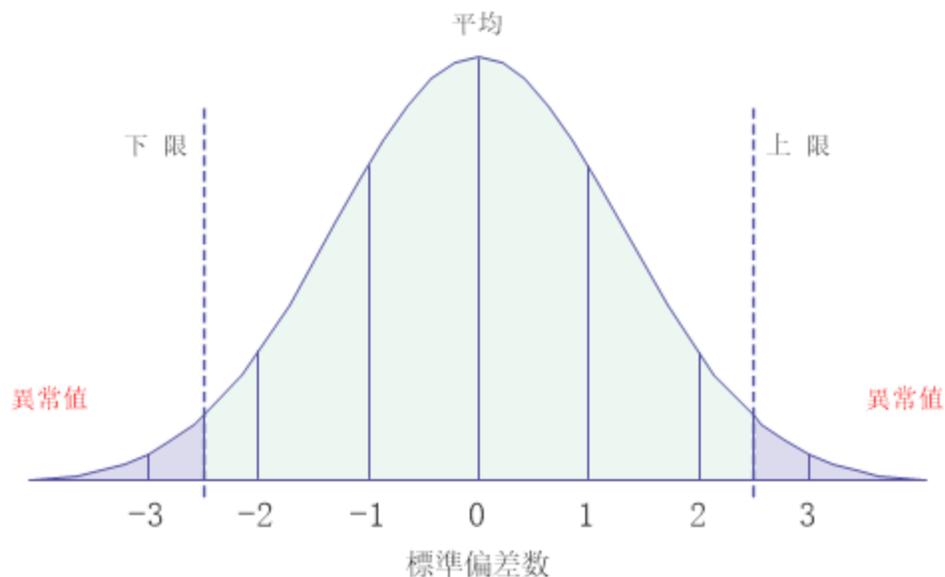
正規分布値のセットの標準偏差を計算する場合、値の 68% が平均 (\pm) の 1 つの標準偏差内に入り、値の 99.7% が平均 (\pm) の 3 つの正規分布内に入ります。ごく一部の値のみが平均からの 3 つの標準偏差を超えます。

Analytics で分析するデータセットの値の分布は、一般的に、正規分布よりも歪んでいる場合があります。たとえば、取引ファイルに、相対的に小さい数千件の取引と数件の大きい取引があることがあります。ただし、異常値の境界が Analytics で動作する方法を簡潔に示すために、正規分布を使用できません。

以下の例のように、標準偏差乗数を大きくすると、異常値の上限と下限が分布曲線の最後近くに移動します。境界が最後に近づくにつれ、境界外の値が減っていきます。

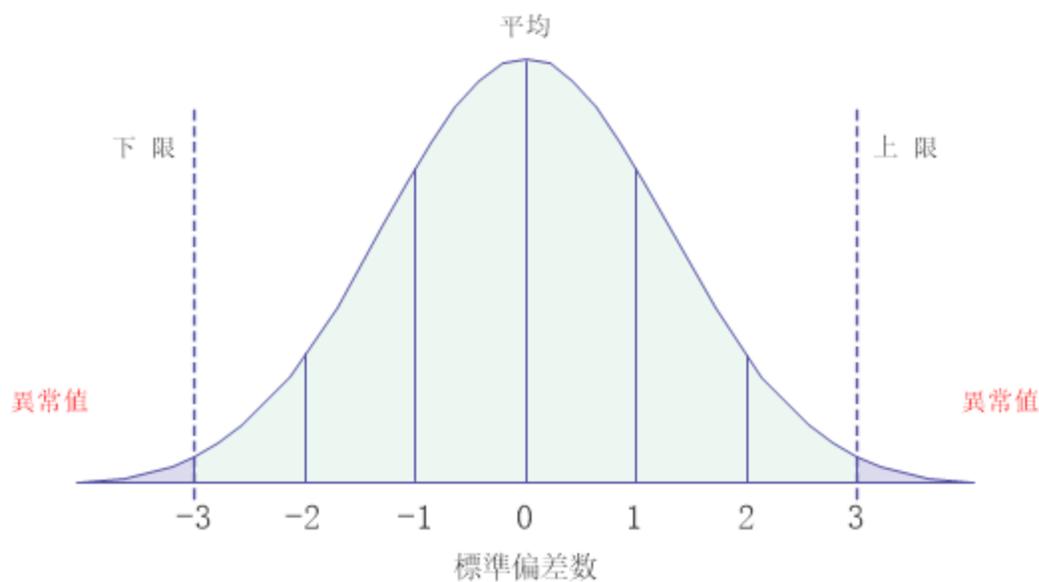
平均から異常値 ± 2.5 標準偏差

平均から +2.5 標準偏差または -2.5 標準偏差未満は、出力結果に異常値として含まれます。



平均から異常値 ± 3 標準偏差

平均から +3 標準偏差または -3 標準偏差未満は、出力結果に異常値として含まれます。



ガイドライン

異常値機能の設定を指定するときには、分析しているデータの本質を考慮してください。

データの本質	設定ガイドライン
値が小さい範囲でクラスター化される	小さい標準偏差乗数を使用します。1 から始めます。1.25 などの小数の乗数を使用し、正確な調整を行います。
値が大きい範囲で分散される	大きい標準偏差乗数を使用します。3 から始めます。
データが歪んでいる。データの残りと比較して、大きい値または小さい値の割合が少ない	検査している値の中央点を計算する方法として、 平均 ではなく、 中央値 を使用します。

出力結果に基づく調整

- **結果が多すぎる** - 標準偏差の乗数を大きくします
- **結果が少なすぎるか、結果がない** - 標準偏差の乗数を小さくします

数値の乗数を使用でき、乗数は1未満です。例: 0.75。

手順

メモ

手順の後に詳細情報が表示されます。" 異常値]ダイアログ ボックス オプション" 見開き ページを参照してください。

1. 異常値をテストするテーブルを開きます。
2. Analytics メインメニューから、**分析 > 異常値]**を選択します。
3. **方法]**の下で、調査している数値フィールドの値の中心点を計算する方法を選択します。

- **平均**
- **中央値**

4. **標準偏差の回数]**で、異常値の境界に使用する標準偏差の乗数を指定します。

任意の正の整数または10進数(0.5、1、1.5、2、...)を指定できます。

5. 次のいずれかを実行します。

- **主キー]**リストから、テーブルのレコードをグループ化するために使用する1つ以上のキーフィールドを選択します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

- **キーなし]**を使用すると、特定のグループ内ではなく、テーブル全体の異常値を特定します。
6. **フィールド]**リストから、異常値を調査する数値フィールドを選択します(異常値フィールド)。
 7. 省略可能。 **他のフィールド]**リストで、出力テーブルに含める1つ以上の追加フィールドを選択します。

メモ

キーフィールドと異常値フィールドは、自動的に出力テーブルに追加されるため、選択する必要はありません。

8. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または**[f]**ボタンをクリックし、**式ビルダー**を利用してIFステートメントを作成します。

メモ

If条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IFステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

9. 次のいずれかを実行します。
 - a. **保存先**テキストボックスに、出力テーブルの名前を指定します。
 - b. **画面**を選択し、Analytics表示領域に結果を出力します。
10. 必要に応じて、**事前並べ替え**を選択解除します。

メモ

ガイダンスは以下に示します。

11. **詳細**タブ
 - a. 省略可能。レコードのサブセットのみが処理されることを指定するには、**範囲**パネルでオプションのいずれかを選択します。
 - b. 省略可能。出力テーブルを自動的に開くようにするには、**出力テーブルの使用**をオンにします。
 - c. **OK**をクリックします。

異常値]ダイアログボックスオプション

次の表は、**異常値**]ダイアログボックスのオプションの詳細を示します。

メイン]タブ

異常値 - 異常値] ダイアログボックス	説明
平均 中央値	異常値フィールドの値の中心点を計算する方法。 <ul style="list-style-type: none"> ○ 平均-を指定すると、フィールドの値の平均(平均値)が使用されます。

異常値 - 異常値] ダイアログ ボックス	説明
	<p>○ 中央値 -はフィールドの値の中央値を使用します 中心点は、異常値フィールドの値の標準偏差を計算するときに使用されます。</p> <p>メモ 中央値]を選択する場合、異常値フィールドを並べ替える必要があります。異常値フィールドが並べ替えられていない場合は、事前並べ替え]を選択します。</p> <p>ヒント 異常値がないかどうかを調べるデータに大きな偏りがある場合は、中央値を指定した方が、データの大勢をより正しく表す結果を生成することができます。</p>
標準偏差の回数	<p>異常値フィールドにおいて、平均または中央値から異常値の上限および下限までに含まれる標準偏差の数。任意の正の整数または小数値(0.5, 1, 1.5, 2...)を指定できます</p> <p>Fたとえば、2と指定すると、各キーフィールドグループまたはフィールドが全体として次のように設定されます。</p> <ul style="list-style-type: none"> ○ 平均または中央値より標準偏差の2倍だけ大きい、異常値の上限 ○ 平均または中央値未満の異常値下限2標準偏差 <p>上限より大きいか下限より小さい異常値フィールドの任意の値が、異常値として出力結果に追加されます。</p> <p>メモ 同じデータセットに対し、標準偏差の数の値を大きくすると、出力結果の異常値の数が減る可能性があります。</p>
主キー 省略可能	<p>テーブルのデータをグループ化するために使用するフィールド。</p> <p>キーフィールドグループごとに、異常値フィールドにおけるそのグループの数値に関する標準偏差が計算されます。このグループの標準偏差は、グループの異常値を検出するための基準値として使用されます。</p> <p>キーフィールドには、文字、数値、または日付時刻があります。複数のフィールドはデータ型を任意に組み合わせることができます。</p> <p>複数のフィールドを選択すると、入れ子のグループが作成されます。入れ子でのフィールド間の順序は、フィールドを選択した順になります。</p> <p>メモ 1つまたは複数のキーフィールドを基準にして並べ替えを行っておく必要があります。1つまたは複数のフィールドを基準にした並べ替えがまだ行われていない場合は、事前並べ替えを使用します。</p>
キーなし 省略可能	<p>テーブルのデータをグループ化しません。</p> <p>異常値フィールドの標準偏差は全体として計算されます。このフィールドの標準偏差は、フィールドの異常値を検出するための基準値として使用されます。</p>
フィールド ("異常値フィールド")	<p>異常値がないかどうかを調べる数値型フィールド。一度に1つのフィールドしか調べることができません。</p> <p>キーフィールドを指定した場合は、グループレベルの異常値が検出されます。キーなし]を指定し</p>

異常値 - 異常値] ダイアログ ボックス	説明						
	<p>た場合は、フィールド レベルの異常値が検出されます。</p>						
その他のフィールド 省略可能	<p>出力に含める1つ以上の追加フィールド。</p> <p>メモ キーフィールドと異常値フィールドは、自動的に出力テーブルに追加されるため、選択する必要はありません。</p>						
もし次の 省略可能	<p>レコードを処理から除外する条件を作成できます</p> <p>[F]テキスト ボックスに条件を入力するか、または [F] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p>						
変換先 省略可能	<p>出力テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 プロジェクト フォルダー以外の場所出力テーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>						
画面 省略可能	<p>出力テーブルではなく、Analytics 表示領域に結果を表示します。</p>						
事前並べ替え 省略可能	<p>当該のコマンドを実行する前に並べ替え操作を実行します。</p> <table border="1" data-bbox="451 1461 1414 1843"> <thead> <tr> <th data-bbox="451 1461 932 1524">事前並べ替えと以下を指定した場合：</th> <th data-bbox="932 1461 1414 1524">並べ替えの基準：</th> </tr> </thead> <tbody> <tr> <td data-bbox="451 1524 932 1766"> <ul style="list-style-type: none"> 1つ以上のキー 平均 </td> <td data-bbox="932 1524 1414 1766"> <ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールド、次に異常値フィールド(異常値フィールドが計算される場合) <p>メモ 計算された異常値フィールドの並べ替えは内部的な Analytics の技術要件です。</p> </td> </tr> <tr> <td data-bbox="451 1766 932 1843"> <ul style="list-style-type: none"> 1つ以上のキー 中央値(Median) </td> <td data-bbox="932 1766 1414 1843"> <p>キーフィールド、次に異常値フィールド</p> </td> </tr> </tbody> </table>	事前並べ替えと以下を指定した場合：	並べ替えの基準：	<ul style="list-style-type: none"> 1つ以上のキー 平均 	<ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールド、次に異常値フィールド(異常値フィールドが計算される場合) <p>メモ 計算された異常値フィールドの並べ替えは内部的な Analytics の技術要件です。</p>	<ul style="list-style-type: none"> 1つ以上のキー 中央値(Median) 	<p>キーフィールド、次に異常値フィールド</p>
事前並べ替えと以下を指定した場合：	並べ替えの基準：						
<ul style="list-style-type: none"> 1つ以上のキー 平均 	<ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールド、次に異常値フィールド(異常値フィールドが計算される場合) <p>メモ 計算された異常値フィールドの並べ替えは内部的な Analytics の技術要件です。</p>						
<ul style="list-style-type: none"> 1つ以上のキー 中央値(Median) 	<p>キーフィールド、次に異常値フィールド</p>						

異常値 - 異常値] ダイアログ ボックス	説明	
	事前並べ替えと以下を指定した場合： <ul style="list-style-type: none"> ○ キーなし ○ 平均 	並べ替えの基準： <ul style="list-style-type: none"> 並べ替えなし
	<ul style="list-style-type: none"> ○ キーなし ○ 中央値(Median) 	異常値フィールド
	<p>ヒント</p> 入力テーブル内の1つまたは複数の該当フィールドを基準にした並べ替えが既に行われている場合は、 事前並べ替え を選択しないことで処理時間を短縮できません。	

詳細]タブ

異常値 - 異常値] ダイアログ ボックス	説明	
範囲パネル	処理されるレコードを指定します。 <ul style="list-style-type: none"> ○ すべて-(デフォルト) テーブルのすべてのレコードが処理されます。 ○ 先頭--このオプションを選択してテキスト ボックスに数を入力すると、テーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 ○ 次--このオプションを選択してテキスト ボックスに数を入力すると、テーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。 ○ While--WHILE ステートメントを使用して、条件に基づいてテーブル内のレコードの処理を制限するには、このオプションを選択します。 <ul style="list-style-type: none"> • [While]テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 • WHILE ステートメントでは、指定した条件がtrue と評価される間のみ、レコードを処理することができます。 • While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。 	

異常値 - 異常値] ダイアログ ボックス	説明
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
出力テーブルを開く	操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。
OK	処理を実行します。 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

並べ替え、フィルタリング、検索

並べ替え、フィルタリング、検索は、Analytics でデータに対して実行できる最も基本的な操作のうちの一つです。メインの分析テストの準備としてこれらの操作を1つ以上実行する必要がある場合や、これらの操作だけでも有用な分析を示すこともあります。

並べ替えとフィルタリングでは、他の Analytics コマンドの実行に欠かせない要素として、これらの操作を実行するオプションもあります。たとえば、年次テーブルで第三四半期の取引のみを集計するために、データフィルターを集計コマンドに取り込むことができます。

可能な別のアプローチ

以下の表は、並べ替え、フィルタリング、検索で取ることができるさまざまなアプローチについて概説します。

操作	アプローチ
並べ替え	<ul style="list-style-type: none"> ◦ クイックソート - ビューのレコードを一時的に並べ替えるために、列の値を並べ替えます ◦ 並べ替えコマンド - レコードを並べ替え、新しい物理的に並べ替えられた Analytics テーブルに出力します ◦ インデックスコマンド - レコードのインデックスを作成し、現在のテーブルで一時的に並べ替えます ◦ あらかじめ並べ替える - Analytics コマンドの実行の不可欠な要素としてレコードを並べ替えます
フィルタリング	<ul style="list-style-type: none"> ◦ クイックフィルター - ビューでマウスを使用して、データを並べ替えます ◦ グローバルフィルター - は、ビューで表示するレコード、Analytics 操作で処理するレコードを制限します。 ◦ ローカルフィルター - 単一の Analytics 操作の1回の実行中に処理されるレコードを制限します
検索	<ul style="list-style-type: none"> ◦ クイック検索 - テーブルの単語、語句、数値、日付を検索します ◦ 一致するすべてのレコードを分離する - 検索条件と一致するレコードのみを含めるために、簡易検索または詳細検索を実行します ◦ 一致する最初のレコードを選択する - 検索条件と一致するテーブルの最初のレコードを選択します

ビューのデータでクイックソートを使用する

ビュー内の列にクイックソートを実行すると、選択している列の値を基準にして、レコードを昇順または降順に並べ替えることができます。

列をクイックソートすると、[インデックス]ダイアログボックスでインデックスを作成する処理を通さずに、選択した列を基に、ビューの一時的なインデックスが作成されます。

メモ

クイックソートは最大 247 文字のフィールドで動作します。フィールドが 247 文字を超える場合、クイックソートメニューオプションは無効です。

手順

1. 並べ替えで使用する列の列見出しを右クリックします。
2. **クイックソート 昇順**]または **クイックソート 降順**]を選択し、その列を昇順または降順に並べ替えます。
3. クイックソートを解除する場合は、列を右クリックして **クイックソート オフ**]を選択します。
クイックソートは、テーブルを閉じるときに自動的に解除されます。

ビューのデータでクイックフィルターを使用する

クイックフィルターを使用すると、マウスを使ってビュー内の値を選択することで、データを簡単かつ迅速にフィルターリングできます。1つの値、または隣接している複数の値を選択できます。クイックフィルターは、どのデータ型でも使用できます。

クイックフィルターはフィルター式の構文を自動的に生成します。その後、それを編集して別のフィルターを作成したり複雑なフィルターを作成したりすることができます。

クイックフィルターは、**式ビルダー**や手動入力よりも簡単にフィルターを作成できますが、制限も多くあります。このフィルターはほかのタイプのフィルターと同様、後で再利用するために名前を付けて保存することができます。

クイックフィルターはグローバルフィルターです

作成するクイックフィルターは**グローバルフィルター**です。グローバルフィルターは、ビューで表示するレコード、Analytics操作で処理するレコードを制限します。詳細については、「[グローバルフィルター\(ビューフィルター\)](#)」ページ 1213を参照してください。

隣接しない値を使用したクイックフィルターリング

最初のクイックフィルターでは、隣接しない値を使用することはできません。フィルターの複雑さによっては、ビュー内のフィールドの配置を変えて値が隣り合うようにすることができます。

最初のクイックフィルターで作成されたデータのサブセットに対して、さらに(2番目の)クイックフィルターを適用する場合は、隣接していない値を使用することができます。

フィルターリングされていないデータセットに対して単独フィルターを適用するときに、隣接していない値を使用する必要がある場合、また、フィールドの配置を変えて値が隣り合うようにすることができない場合は、フィルター式を手動で入力するか、あるいは**式ビルダー**を使用してフィルター式を作成する必要があります。

空白または空白以外の値によるクイックフィルターリング

クイックフィルターリング文字フィールドのオプションのうち2つは、**空白**と**空白以外**です。これらの条件のいずれかを使用するには、まずフィールドの値を選択する必要がありますが、実際に選択された値は無視さ

れます。この動作により、空白の値を最初に検索せずに、非常に長い列のデータの空白にフィルターを適用できます。

空白または空白以外の値による数値または日付時刻フィールドの詳細については、「データの検索」ページ 1224を参照してください。

日付、日付時刻、または時刻によるクイックフィルタリング

日付または時刻値の表示書式を指定したときに、その書式がソースデータで利用できるすべてのデータ要素を表示しないものであった場合は、日付、日付時刻または時刻でクイックフィルターを実行すると不正な結果が生成されます。たとえば、時、分、秒を含む時刻データに対して `hh:mm` という書式を指定した場合などです。

日付と時刻の表示書式の詳細については、「日付と時刻オプション」ページ 135を参照してください。

クイックフィルターオプション

[クイックフィルター]メニューのオプションは、フィルタリングするフィールドのデータ型、および単一の値を選択するか、隣接する複数の値を選択するかどうかによって異なります。

詳しく表示

選択したデータ	クイックフィルターメニューオプション
単独の文字値	<ul style="list-style-type: none"> ○ 等しい ○ 等しくない ○ 次の値より大きい ○ 以上 ○ より小さい ○ 以下 ○ 空白 ○ 空白以外
単独の数値	<ul style="list-style-type: none"> ○ 等しい ○ 等しくない ○ 次の値より大きい ○ 以上 ○ より小さい ○ 以下
単独の日付値	<ul style="list-style-type: none"> ○ 当日 ○ 当日以外 ○ より後 ○ 以降 ○ より前 ○ 以前
単独の日付時刻値または時刻値	<ul style="list-style-type: none"> ○ 同時刻 ○ 同時刻外

選択したデータ	クイックフィルター メニュー オプション
	<ul style="list-style-type: none"> ○ より後 ○ 以降 ○ より前 ○ 以前
単独の論理値	<ul style="list-style-type: none"> ○ 等しい ○ 等しくない
同じレコード内で隣接している複数の値	<ul style="list-style-type: none"> ○ 等しい 選択したすべての値に適合するレコードをフィルターリング後のテーブルに抽出します。 ○ 等しくない 選択したすべての値に適合しないレコードをフィルターリング後のテーブルに抽出します。 ○ 一部が等しくありません 選択したいいずれかの値に適合しないレコードをフィルターリング後のテーブルに抽出します。
同じ列内で隣接している複数の文字値	<ul style="list-style-type: none"> ○ 等しい ○ 等しくない ○ 範囲内 ○ 範囲外 ○ 空白 ○ 空白以外
同じ列内で隣接している複数の数値	<ul style="list-style-type: none"> ○ 等しい ○ 等しくない ○ 範囲内 ○ 範囲外
同じ列内で隣接している複数の日付値	<ul style="list-style-type: none"> ○ 当日 ○ 当日以外 ○ 範囲内 ○ 範囲外
同じ列内で隣接している複数の日付時刻値または時刻値	<ul style="list-style-type: none"> ○ 同時刻 ○ 同時刻外 ○ 範囲内 ○ 範囲外
2 つ以上のレコードまたは 2 つ以上の列におよぶ値のブロック	<ul style="list-style-type: none"> ○ 等しい(表示されません)

手順

初期クイックフィルターの適用

1. クイックフィルターの基準となる、1つの値、または隣接する複数の値を選択します。
隣接している複数の値を選択するには、クリックしてから選択対象の範囲までドラッグします。
空白または空白以外の値で文字フィールドにクイックフィルターを適用する場合は、フィールドの任意の値を選択します。
2. ビューのデータ領域内を右クリックして **クイックフィルター**]を選択し、サブメニューから適切なオプションを選択します。
テーブル内のレコードは、選択に基づいてフィルターリングされます。ビュータブの上部にある **フィルター**]テキスト ボックスに、自動生成されたフィルター式の構文が表示されます。
複数のレコードおよび複数の列におよぶ値のブロックを選択した場合、サブメニュー オプションは利用できません。クイックフィルターは、選択した値(複数)と一致するレコードのみを自動的にフィルターリングしてビューに表示します。

フィルターされたレコードの数をカウントします。

クイックフィルターを適用した後、以下の手法を用いてフィルターに含まれるレコードの数をカウントします。クイックフィルターを変更または拡張した後は同じ手法を用います。

1. Analytics メインメニューから、**[カウント 

クイックフィルターの修正**

クイックフィルターを編集したい場合は、**フィルター**]テキスト ボックスでフィルター式を手動で編集し、**フィルターの設定**] 

クイックフィルターの拡張

さらに1つ以上のクイックフィルターを追加適用して、クイックフィルターを拡張したい場合は、以下の手順を実行します。

1. 1つの値、または隣接している複数の値を選択します。
2. ビューのデータ領域内を右クリックして、**クイックフィルター > AND**] または **[OR]** を選択し、サブメニューから適切なオプションを選択します。

例

Customer = '795401' でフィルターリングしたレコードを、さらに取引タイプが'IN'のレコードに絞り込むには、INの値を選択し、**クイックフィルター > AND > 等しい**の順にクリックします。フィルター式は次のように修正されます。

```
(Customer = '795401') AND (Type = 'IN')
```

AND さらにデータのフィルターリングされたセットを制限します。拡張されたフィルター式の詳細にもよりますが、通常、OR は最初にフィルターリングされたデータセットを拡張します。

現在のクイックフィルターの置換

現在のクイックフィルターを新しいクイックフィルターに置き換えたい場合は、以下の手順を実行します。

1. 1つの値、または隣接している複数の値を選択します。
2. ビューのデータ領域内を右クリックして **クイックフィルター > 置換** を選択し、サブメニューから適切なオプションを選択します。

クイックフィルターの削除

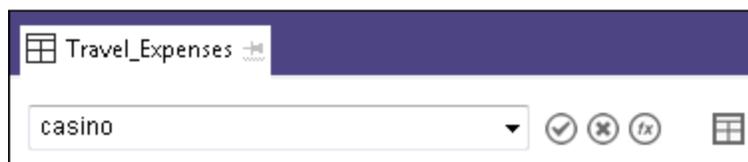
クイックフィルターを削除する場合は、フィルターツールバーにある **フィルターの削除**  をクリックします。

クイックフィルターを名前付きフィルターとして保存する

クイックフィルターを、テーブルに関連付けられる名前付きフィルターとして保存する場合は、フィルターツールバーにある **ビューフィルターの編集**  をクリックし、**次の名前**で保存]テキストボックスにフィルターの名前を入力して、**OK** をクリックします。

テーブルのクイック検索

[ビュー]タブの上部にある [フィルター]テキスト ボックスに1つ以上の検索語を入力して、テーブル内のデータのクイック検索を実行することができます。



検索の範囲

現在のビューに表示されているデータだけではなく、テーブルのすべてのソース データが検索されます。データソース、テーブル、およびビューについては、"Analytics テーブルの構造" ページ 118を参照してください。

特定のフィールドではなく、レコードの未定義の部分を含むレコード全体が検索されます。たとえば、**casino** と入力すると、レコード内のどこかに「casino」を含んでいるレコードが検索されます。特定のフィールドを検索する必要がある場合は、後から検索を修正できます(文字データの検索のみ)。

演算フィールドと関連フィールドは、続いて、その検索を修正して特定のフィールドを指定しない限り、検索されません。

検索されたデータ型

文字データの検索は、クイック検索の最も簡単な使い方です。数値データや日付時刻データも検索できますが、考慮すべきいくつかの事項があります。これについては、次のセクションで説明します。

検索語のフィルターへの自動変換

入力した検索語は、FIND() 関数を使用するグローバルフィルターに自動的に変換されます。

たとえば、**casino** を入力すると、フィルター `FIND("casino")` という結果になります。



フィルターにより [フィルター]テキスト ボックスに自動的に入力が行われますが、必要に応じて値を変更できます。たとえば、`FIND("casino")` を修正し、結果を特定のフィールドである `FIND("casino", Merchant)` に限定することができます。

また、フィルターはフィルタ履歴とコマンド ログに追加されます。後で、ここからフィルターを再適用することができます。

検索語とフィルター構文の自動識別

[フィルター]テキスト ボックスは、検索語とフィルター構文を自動的に識別します。たとえば、[フィルター]テキスト ボックスに `match` と入力すると、文字列「`match`」を検索しますが、`match(City, "New York", "Washington")` と入力すると、`MATCH()` 関数を使用するフィルターを作成します。

手順

メモ

数値または日付時刻値を検索する場合は、ビューにおける書式ではなく、ソースデータの書式と一致させる必要があります。詳細については、「数値または日付時刻データのクイック検索」ページ 1179を参照してください。

1 つ以上の検索語の検索

[ビュー]タブの上部にある [フィルター]テキスト ボックスで、1 つ以上の検索語を入力し、Enter を押しします。

複数の検索語を入力した場合、クイック検索は論理 OR 演算を実行し、指定された検索語のうち少なくとも 1 つを含むレコードを検出します。

正確な語句の検索

[ビュー]タブの上部にある [フィルター]テキスト ボックスで、語句を入力し、二重引用符で囲み、Enter を押しします。

クイック検索は、正確な語句を含むレコードのみを検出します。

検索を特定のフィールド(文字フィールドのみ)に制限する

1. [フィルター]テキスト ボックスで自動生成されたフィルターを修正します。検索語の後にカンマを入力し、その後にフィールドの名前を追加します。

たとえば、`FIND("casino")` を `FIND("casino", Merchant)` に変更します。

2. Enter キーを押しします。

検索は指定したフィールドに制限されます。

メモ

フィールドの物理名を使用する必要があります。物理名は、ビューにおけるフィールドの表示名とは異なる場合があります。

物理名を確認するには、該当する列ヘッダーを右クリックし、**プロパティ**]を選択します。必要に応じて、**列の変更**]ダイアログボックスの上部にあるテキストボックスから物理名をコピーします。**代替列見出し**]を使用しないでください。

関連付けられたフィールドで検索するには、フィールドの完全修飾名(つまり、テーブル.フィールド名)を指定する必要があります。たとえば、**FIND("casino", Vendor.Vendor_Name)**

のように指定します。

文字データのクイック検索

文字データをクイック検索するときは、単語全体、単語の一部、または正確な語句を入力できます。

詳しく表示

- 複数の単語を入力した場合、クイック検索は論理 OR 演算を実行し、指定された単語のうちの少なくとも1つを含むレコードを検出します。
- 正確な語句を検索するには、語句全体を二重引用符で囲みます。
- 検索語を識別するには、語の後に末尾のスペースを入れ、語とそのスペースを二重引用符で囲みます。

たとえば、**"cash "** は "cash" を返しますが、"cashier" は返しません。これは、データ内の文字列 "cash" の後ろに1つ以上のスペースが続いていると見なされます。

検索語	次を含むレコードが返されます
cas	<ul style="list-style-type: none"> ◦ casino ◦ cash ◦ Americas ◦ Lancashire ◦ etcetera . . .
casino	<ul style="list-style-type: none"> ◦ casino ◦ casinos
casino liquor	<ul style="list-style-type: none"> ◦ casino ◦ casinos ◦ liquor ◦ liquors ◦ casino(および) liquor(順序は考慮されない) ◦ etcetera . . .
"Diamond Casino"	<ul style="list-style-type: none"> ◦ Diamond Casino

検索語	次を含むレコードが返されます
"Diamond Casino" "Golden Casino"	<ul style="list-style-type: none"> ◦ Diamond Casino ◦ Golden Casino ◦ Diamond Casino(および) Golden Casino(順序は考慮されない)
casino, "ABC Liquors"	<ul style="list-style-type: none"> ◦ casino ◦ casinos ◦ ABC Liquors ◦ casino(および) ABC Liquors(順序は考慮されない) ◦ etcetera . . .
"ABC L"	<ul style="list-style-type: none"> ◦ ABC Liquors ◦ ABC Limousine ◦ ABC Learning ◦ etcetera . . .
"cash " (単語「cash」の後ろにスペースが1 つあります)	<ul style="list-style-type: none"> ◦ cash (データ内で、文字列「cash」の後ろに1 つ以上のスペースが続いている必要があります) ◦ 「cashier」や「Lancashire」を返しません

数値または日付時刻データのクイック検索

メモ

特定のフィールドの数値データまたは日付時刻データを検索する場合は、クイックフィルターを使用します。詳細については、「ビューのデータでクイックフィルターを使用する」ページ 1171を参照してください。

数値データまたは日付時刻データをクイック検索するときには、検索対象はビューに表示されているデータではなく、元となるソースデータであることを覚えておく必要があります。

多くの場合、ソースデータの数値、日付、および時刻は、ビューの項目とは異なる方法で書式設定されています。検索語は、ビューにおける書式ではなく、ソースデータの書式と一致する必要があります。

編集 > テーブルレイアウト]を選択すると、テーブルのソースデータを表示できます。

数値データのクイック検索

どのレコードが特定の検索後に返されるかに、ソースデータの数値書式が影響します。

詳しく表示

検索語	ビューの数値書式	ソースデータの数値書式	次を含むレコードが返されま す
1234.00	9999.99	9999.99	1234.00
		9,999.99	レコードは返されません
1,234.00	9,999.99	9999.99	レコードは返されません
		9,999.99	1,234.00
		9.999,99	レコードは返されません
(1234.00)	(9999.99)	(9999.99)	(1234.00)
		-9999.99	レコードは返されません
1234.01	9999.99 (丸められた数値) 例: 1234.01	9999.9999	レコードは返されません
1234.0085		例: 1234.0085	1234.0085
123 456	9999.99	9999.99	<ul style="list-style-type: none"> ○ 123 ○ 456 ○ 123(および) 456(順序は考慮されない)

日付時刻データのクイック検索

どのレコードが特定の検索後に返されるかに、ソースデータの日付時刻書式が影響します。

詳しく表示

検索語	ビューの日付時刻書式	ソースデータの日付時刻書式	次を含むレコードが返されません
12/31/2015	MM/DD/YYYY	MM/DD/YYYY	12/31/2015
		DD/MM/YYYY	レコードは返されません
		YYYYMMDD	レコードは返されません
31/12/2015	MM/DD/YYYY	MM/DD/YYYY	レコードは返されません
		DD/MM/YYYY	31/12/2015
		YYYYMMDD	レコードは返されません
20151231	MM/DD/YYYY	MM/DD/YYYY	レコードは返されません
		DD/MM/YYYY	レコードは返されません
		YYYYMMDD	20151231
2015-12-31	YYYY-MM-DD	YYYY-MM-DD	エラーメッセージ レコードは返されません
FIND("2015-12-31")			2015-12-31
23:59:59	hh:mm:ss	hh:mm:ss	23:59:59
		hhmmss	レコードは返されません
20151231.235959	MM/DD/YYYY hh:mm:ss	YYYYMMDD.hhmmss	20151231.235959
		MM/DD/YYYY hh:mm:ss	レコードは返されません

クイック検索のその他の特性

クイック検索には、ほかにも次のような特性があります。

特徴	説明
大文字と小文字の区別	検索は大文字と小文字を区別しません。
ワイルドカード	検索語内のワイルドカード文字はサポートされません。
スペース	複数の検索語における先頭、末尾、および中間のスペースは、検索語とスペースが二重引用符で囲まれている場合にのみ考慮されます。 スペースを二重引用符で囲んだ場合、そのスペースは文字として扱われ、データ内で正確に一致する必要があります。

特徴	説明
二重引用符	語句を囲むために使用できるのは二重引用符だけです。一重引用符はこの目的ではサポートされず、通常の文字のように扱われます。
演算フィールド	演算フィールドは検索されません。
関連付けられたフィールド	関連フィールドは検索されません。
フィールド別検索の限定	自動入力されたフィルターを修正して、検索対象を特定のフィールドに限定するときは、文字フィールドのみを指定できます。 数値フィールドまたは日付時刻フィールドを指定すると、エラーになります。
サポートされていない文字	次に挙げる文字がクイック検索の検索語で使用された場合は、一貫しない結果になるか、エラーメッセージが発生する可能性があります。これらの文字は、Analytics 式で使用される演算子です。 <code>^ * () - + = < ></code> これらの文字のいずれかを検索する場合は、[フィルター]テキスト ボックスで FIND() 関数を手動で入力します。例： <code>FIND("a+b")</code> または <code>FIND("2015-12-31")</code>
フィールドの境界 末尾のスペース	レコード内のフィールドの境界は無視されます。つまり、検索語は、フィールドの境界を越えて文字列と一致する可能性があるということです。フィールド内の末尾のスペースは文字のように扱われます。 このタイプの検索を明確に意図しない限り、フィールドの境界を越えて検出された検索結果は有効な結果ではありません。たとえば、口座番号の最後の数字と隣接するフィールドの金額の最初の数字は数値の検索語と一致する可能性があります。おそらく誤検出です。 メモ ここで言う「フィールド境界」は、レイアウト内のフィールドの物理的な順序に基づいて、テーブルレイアウトに表示される境界です。 レイアウト内のフィールドの順序は、関連付けられているビュー内の列の順序とは異なる場合があるため、レイアウトとビューでは隣接するものが異なる状況が生じます。 クイック検索で特定のレコードが返される理由がわからない場合は、 編集 > テーブルレイアウト を選択して、検索するソース データを表示します。

検索およびフィルターリングの追加情報

- ワイルドカードの使用を含む、その他のさまざまなデータ検索オプションについては、「データの検索」ページ 1224を参照してください。
- この操作の詳細については、「データのフィルターリング」ページ 1209を参照してください。
- FIND() 関数の詳細については、「FIND() 関数」ページ 2267を参照してください。

ソートとインデックス

並べ替えとインデックスの作成は、テーブル内のデータを順番どおりに並べるための2つの異なる方法です。一部の Analytics コマンドは、入力が最初に並べ替えるか、インデックスを付ける必要があります。データを順序付けることは、パターンと異常値に焦点を当て、それ自体が有用な分析操作として役立ちます。

操作	説明
並べ替え	テーブルを並べ替えると、物理的にデータを順番に整列し直して、その結果を新しい Analytics テーブルに出力します。
インデックス	インデックスは、元となるデータの物理的な順序に変更を加えません。その代わりに、テーブル内のレコードを参照する別個のインデックスファイルを作成して、物理的な順序ではなく順番でレコードにアクセスできるようにします。ビュー内のデータは、インデックスがアクティブである間のみ、そのインデックスに従って並べ替えられます。

他の操作の前提条件としてのデータの並べ替え

コンピューターは先頭レコードから順にファイルを処理するため、データが順番どおりに並んでいることが、Analytics のいくつかの分析テストやその他の操作における前提条件となっています。結合や関連付けなどの、複数テーブルを用いる操作では、キーフィールドの並べ替えやインデックス付けを必要とすることがあります。

それ以外のテストや操作では順序付けられたデータを必要としないかもしれませんが、データが最初に並べ替えられたりインデックス付けされたりした場合は、より迅速に実行されます。

並べ替えまたはインデックス

並べ替えるかインデックスかの決定は、実行する特定のタスクによって異なる場合があります。例：

- **並べ替え** -は、以降の分析の基準となる新しいテーブルを出力するため、調査作業にとってのより良い選択であるかもしれません。
- **インデックス** -は、アクティブなテーブル内のデータの異なる表現をすばやく切り替えることができるため、情報作業や準備作業にとってのより良い選択であるかもしれません。

並べ替えとインデックスの利点と問題点

次の表は、並べ替えとインデックスの利点と問題点を比較し、前提条件として並べ替えまたはインデックスが必要である処理を一覧で示します。

	並べ替え	インデックス
物理的に別個の新規 Analytics テーブルに結果を出力する	はい	いいえ
データを物理的に並べ替える	はい	いいえ
操作の速度	遅い	速い
処理に必要なディスクの空き領域	多い	少ない
コマンド実行後のファイルサイズ	大きい	小さい
並べ替えられたファイルまたはインデックスが付いたファイルの後続の処理	速い	遅い
文字フィールドの検索	遅い	速い
前提条件	<ul style="list-style-type: none"> ○ 結合 (お勧めしますが、主テーブルでは強制ではありません) ○ マージ ○ Duplicates/重複 ○ Gaps/ギャップ 	<ul style="list-style-type: none"> ○ 関係の定義 (子テーブルのキーフィールドのインデックス付けは、Analytics によって自動的に実行されます) ○ 結合 (副テーブルへのインデックスの適用は、コマンドラインまたはスクリプト内のみで行うことができます) ○ マージ (副テーブルへのインデックスの適用は、コマンドラインまたはスクリプト内のみで行うことができます) ○ Duplicates/重複 ○ Gaps/ギャップ ○ Find/検索 ○ [文字検索] 検索オプション ○ Seek/検索 ○ [式検索] 検索オプション

並べ替え順]オプションとソート順序

並べ替え順]オプション(`ソール > オプション > テーブル`) は、文字データのソート順序(照合順序)を指定します。選択したオプションは、文字フィールドを用いた、レコードの並べ替えやインデックス付け、または順番の検査を行うときに、どのソート順序を使用するかを定義します。

ソート順序

ソート順序は、Analytics が並べ替え、インデックスの作成、順番の検査、またはクイックソートを実行するときに、文字フィールド内の各値の最初の文字または文字列と比較するテンプレートのようなものです。

以下の表は、デフォルトの Analytics ソート順序設定と関連付けられたソート順序を示します。

<p style="writing-mode: vertical-rl; text-orientation: upright;">Analytics の デ フォ ルト</p>	<p style="writing-mode: vertical-rl; text-orientation: upright;">[並べ替え順序]のデフォルト</p> <p>関連付けられているソート順序</p>
<p style="writing-mode: vertical-rl; text-orientation: upright;">非 ユニ コード フォ ルト (- A- S- C- I- T-)</p>	<ul style="list-style-type: none"> 数字、大文字、小文字 0, 1, 2... A, B, C... a, b, c... たとえば、"Z" は "a" の前にソートされます。 特殊文字は、文字に応じて、シーケンス内の別のポイントで発生します。 分音記号の付いた文字はシーケンスの最後に発生し、小文字の内部シーケンスの前に同じ大文字を使用します。 <p>すべての並べ替えられた文字を表示する(非 Unicode エディション)</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <pre>! " # \$ % & ' () * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _ ` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~ € , f „ … ^ Š < Ž ‘ ’ “ ” – – ~ ™ š > ž Ÿ ¡ ¢ £ ¥ ¦ § ¨ © « ® ± ´ , » ¿ À Á Â Ã Ä Å Ç È É Ê Ë Ì Í Î Ï Ñ Ò Ó Ô Õ Ö Ø Ù Ú Û Ü Ý Þ ß à á â ã ä å ç è é ê ë ì í î ï ð ñ ò ó ô õ ö ø ù ú û ü ý þ ÿ</pre> </div>
<p style="writing-mode: vertical-rl; text-orientation: upright;">U- n- i- c- o- d- e (- U- C- A-)</p>	<ul style="list-style-type: none"> 数字、小文字と大文字の混合 0, 1, 2... a, A, b, B, c, C... たとえば、"a" は "Z" の前にソートされます。 特殊文字は、数字の前に発生します。 分音記号の付いた文字は、分音記号の付いていない文字と混合されます。 例: e, E, é, É, f, F <p>すべての並べ替えられた文字を表示する(Unicode エディション)</p>

A-n-a-l-y-t-i-c-s の エ デ イ シ ョ ン	{並べ替え順}のデフォルト 関連付けられているソート順序
(-U-n-i-c-o-d-e 照 合 ア ル ゴ リ ズ ム)	<pre> - - - , ; : ! i ? ¿ ' ‘ ’ , < > “ ” „ « » () [] { } @ * / \ & # % ` ´ ~ ^ ¨ . ^ © ® + ± < = > ¡ ~ ¢ \$ £ ¥ € 0 1 2 3 4 5 6 7 8 9 a A á Á à À â Â ä Ä ä ä ã Ã b B c C ç Ç d D ð e E é É è È ê Ê ë Ë f F g G h H i I í ì Ï î Ë j J k K l L m M n N ñ Ñ o ó Ó ò ô Ô ö Ö ø Ø p P q Q r R s S š Š ß t T ™ u U ú Ú ù Û Ü ü Û Ü v V w W x X y Y ý Ý ÿ Ÿ z Z ž Ž þ Þ </pre>

並び替え順序を変更する

別の言語の方が分析しているデータとより一致する場合は、**並び替え順**]をその言語に変更することができます。また、Unicode 版の Analytics では、この変更をコマンドに基づき、コマンドラインまたはスクリプトで ISOLOCALE パラメーターを使用することによって行うこともできます。

ソート順序の変更

非 Unicode 版の Analytics では、別の言語を選択したときに、**並び替え順**]テキストボックスで文字の並び順を変更することによって、関連付けられているソート順序を変更するオプションがあります。

また、独自のソート順序を作成するオプションもあります。**並び替え順**]フィールドで "カスタム" を選択してシーケンス(順序付け)を指定するか、あるいは、コマンドラインまたはスクリプトで SET ORDER <TO> **値** を入力してシーケンスを指定します。指定した文字は、その他すべての文字の前に、指定した順序で

並べ替えられます。たとえば、値 `aAbBcC...` を入力して、小文字と大文字が混合されるように指定することができます。`SET ORDER` を指定すると、並べ替え順序をデフォルト設定に戻します。

バイト順序に基づくデフォルトのソート順序

個々の言語のデフォルトのソート順序は、その文字セット内の各文字のバイト順序から導出されます。文字セット内の文字のバイト順は、Windows の文字コード表で見ることができます。

レコードの並べ替え

レコードを昇順または降順に並べ替えて、その結果を新しい、物理的に整列し直された Analytics テーブルに出力することができます。Analytics テーブルへの出力は唯一の出力オプションです。

レコードの並べ替えは複数の Analytics 操作の前提条件です。詳細については以下を参照 "明示的な並べ替えを実行するか、事前並べ替えを使用すべきか" ページ 1190

並べ替えは、パターンと異常値に焦点を当て、それ自体が有用な分析操作として役立ちます。

メモ

レコードのインデックスは、レコードを並べ替えることの代替方法です。状況によっては、インデックスがより良い選択となることがあります。詳細については以下を参照 "並べ替えまたはインデックス" ページ 1183

レコード全体または指定したフィールドのみを出力すべきか

並べ替えるときには、並べ替えられた出力テーブルのレコード全体または指定されたフィールドのみを含めることができます。以下に要約されているように、各オプションに関連付けられた意味があります。

選択するオプションは並べ替えの速度にも影響することがあります。詳細については、"並べ替えを高速化する方法" 次のページを参照してください。

ヒント

フィールドで出力する特性を維持しながら、レコード全体が必要な場合は、フィールドで出力し、すべてのフィールドを選択します。

出力タイプ	意味
レコード	<ul style="list-style-type: none"> レコード全体が並べ替えられた出力テーブルに含まれます。 演算フィールドは演算式として保持されます。 関連付けられたフィールドを含めることはできません。ただし、新しい出力テーブルは元の子テーブルに自動的に関連付けられ、子テーブルのフィールドを出力テーブルビューに追加できます。
フィールド	<ul style="list-style-type: none"> 指定したフィールドのみが並べ替えられた出力テーブルに含まれます。キーフィールドは自動的に含まれ、指定する必要はありません。 演算フィールドを、実際に計算された値が格納された物理フィールドに変換できます。 関連付けられたフィールドを含めることができます。出力テーブルの永久物理フィールドになります。新しい出力テーブルは元の子テーブルに関連付けられません。

複数のキーフィールドでの並べ替え

1つのキーフィールドを使用してレコードを並べ替えることができます。また、複数のキーフィールド(主キーフィールド、副キーフィールドなど)を基にして並べ替えることで、入れ子の並べ替えの構成を作成することができます。入れ子の並べ替えでは、キーフィールド全体で昇順と降順が混在しても、データ型が混在してもかまいません。

例

取引テーブルを Date キーフィールドで昇順に並べ替え、それぞれの日にち内を Amount キーフィールドで降順に並べ替えることができます。

以下の結果は、データ型が混在(日付時刻と数値)したネストされた並べ替えを昇順および降順で示します。

Date フィールド (昇順)	Amount フィールド (降順、入れ子)
15 Jan 2011	\$2300.00
15 Jan 2011	\$1200.00
15 Jan 2011	\$600.00
16 Jan 2011	\$900.00
16 Jan 2011	\$100.00
17 Jan 2011	\$4700.00
17 Jan 2011	\$900.00
17 Jan 2011	\$500.00

並べ替えを高速化する方法

数百万件のレコードがある非常に大きいテーブルを並べ替えると、時間がかかる場合があります。並べ替えは大量のシステムリソースを必要とするため、ほかのタスクが同時に実行されている場合は、処理速度が低下する可能性があります。

並べ替え速度の改善

2つのオプションで並べ替え速度を改善できます。

- **フィールドのサブセットの出力** - 期間に含まれるデータの一部のみが必要な場合は、並べ替えられた出力テーブルにレコード全体を含めないでください。必要なフィールドのみを選択します。多くの場合、これにより、並べ替え処理が高速化します。

合計フィールド数の割合としてフィールドのサブセットが小さいほど、パフォーマンスが改善します。

- **並べ替えで利用できるメモリを増やす** - 並べ替え用の特定のメモリ量を最大 2000 MB に割り当てることができます。[ソール | オプション | テーブル | 並べ替えメモリ]に移動するか、[SET SORTMEMORY](#) コマンドを使用します。

追加の提案

大きいテーブルを並べ替える時間が問題になる場合は、次の点を考慮してください。

- コンピューターのハードウェアをアップグレードする
- スクリプトを作成して、スケジュールで夜間にデータを並べ替える

明示的な並べ替えを実行するか、事前並べ替えを使用すべきか

次の処理の前にレコードを並べ替えることは前提条件が推奨されます。

- テーブルの結合
- テーブルのマージ
- 集計 (キーフィールドの同じ値の各セットで単一グループが必要な場合)
- 重複の検査
- ギャップの検査

これらすべての操作には **あらかじめ並べ替える** オプションが含まれており、操作の一部としてレコードを順番に並べ替える予備の手段を組み込むことができます。

1つのテーブルに対して複数の操作を実行している場合は、**あらかじめ並べ替える** を繰り返し使用するよりも、最初に明確にテーブルを並べ替えた方がより効率的であるかもしれません。テーブルに含まれるレコード数が多い場合は特にそうです。

すべてのソースレコードが出力テーブルにあることを確認する

テーブル内のすべてのレコードを並べ替えて出力する場合は、数値フィールドにコントロール合計を設定することによって、実際にすべてのレコードが新しいテーブルへ出力されたかどうかを検証できます。

テーブルレイアウト]ダイアログボックスで、フィールドにコントロール合計を設定します。レコードを並べ替えて出力したら、新しいテーブルで **ソールテーブル履歴**]を選択して、入力と出力のコントロール合計を比較します。詳細については、「物理フィールドを定義する」ページ 790を参照してください。

手順

アクティブなテーブルの1つまたは複数のキーフィールドによってレコードを並べ替え、その結果を新しい Analytics テーブルに出力することができます。並べ替えられた出力テーブルのレコード全体または指定されたフィールドのみを含めることができます。

方法の説明

メモ

手順の後に詳細情報が表示されます。「**並べ替え**]ダイアログボックスオプション」見開きページを参照してください。

並べ替えの処理中に使用される一時ファイルを作成するために、並べ替えられるファイルのサイズの少なくとも 2.5 倍の空き領域が必要です。

1. ナビゲーターで、並べ替えるテーブルを開きます。
2. **データ > 並べ替え**]の順に選択します。
3. **メイン**]タブで、次のいずれかを行います。
 - **並べ替えの対象**]リストから、並べ替えるフィールドを選択します。
 - **並べ替えの対象**]をクリックして、キーフィールドを選択するか、または式を作成します。

ヒント

並べ替えの対象]をクリックした場合は、任意で並べ替えの矢印  (デフォルト並べ順は昇順) をクリックすることにより、1つまたは複数のキーフィールドの出力結果を降順にするよう指定することができます。

4. レコード全体または指定されたフィールドのみを出力するには、次のいずれかを実行します。
 - 並べ替えられた出力テーブルにレコード全体を含める場合は、**レコード**]を選択します。
 - 並べ替えられた出力テーブルにフィールドのサブセットのみを含める場合は、**フィールド**]を選択します。

メモ

レコードに含まれるデータの一部のみが必要な場合は、特にテーブルが大きい場合は、**フィールド**]を選択します。

詳細については、「並べ替えを高速化する方法」ページ 1189を参照してください。

5. **フィールド]**を選択した場合は、次のいずれかを行います。
 - **他のフィールド]**リストから適切な非キーフィールドを選択します。
 - **その他のフィールド]**をクリックして、フィールドを選択するか、または式を作成します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

6. **保存先]**テキスト ボックスに、出力テーブルの名前を指定します。
7. **詳細]**タブ
 - a. (省略可能)レコードのサブセットのみが処理されることを指定するには、**範囲]**パネルでオプションのいずれかを選択します。
 - b. **OK]**をクリックします。

並べ替え]ダイアログ ボックス オプション

次の表は、**並べ替え]**ダイアログ ボックスのオプションの詳細を示します。

メイン]タブ

オプション - 並べ替え]ダイアログ ボックス	説明
並べ替え基準	<p>テーブルを並べ替えるために使用するキーフィールドを指定します。</p> <ul style="list-style-type: none"> ◦ 並べ替え基準]リストからフィールドを選択できます。 ◦ 並べ替え基準]をクリックして、フィールドを選択するか、または式を作成するか、降順を指定し、OK]をクリックします。 <p>メモ 論理フィールドでの並べ替えでは、フィールド リストにフィルターを含める]を選択する必要があります(ルール] > オプション] > インターフェイス])。</p> <p>キーフィールドのガイドライン:</p> <ul style="list-style-type: none"> ◦ 単一キーフィールド -1 つのキーフィールドのみを選択する場合は、各並べ替えられたグループ内のレコードは相互に相対的な元の並べ順を保持します。 ◦ 複数のキーフィールド -複数のキーフィールドを選択する場合、フィールドを選択する順序が入れ子になった並べ替えの優先度を示します。レコードが最初に選択したフィールドで並べ替えられ、最初のフィールドに複数の値の出現が存在する場合は、次にそのグループ内のレコードが2番目に選択したフィールドで並べ替えられる、というようになります。 <p>詳細については、「複数のキーフィールドでの並べ替え」 ページ 1189を参照してください。</p>

オプション- 並べ替え]ダイアログボックス	説明
	<ul style="list-style-type: none"> ○ キーフィールド列の順序 <ul style="list-style-type: none"> • 出力テーブルにレコード全体を含める場合は、結果テーブルのキーフィールドの列順は、キーフィールドを選択する順序に関係なく、ソーステーブルの列順と同じです。 • 出力テーブルにフィールドのサブセットを含める場合は、結果テーブルのキーフィールドの列順は選択した順序と同じです。グループとして、出力テーブルの非キーフィールドの前にキーフィールドが表示されます。 ○ 関連付けられたキーフィールド - テーブル関係で子テーブルのフィールドで並べ替える場合： <ul style="list-style-type: none"> • 並べ替え基準]をクリックします。選択済みのフィールド]ダイアログボックスの対象テーブル]ドロップダウンリストを使用して、適切な子テーブルを選択できます。 <p>注意</p> <p>レコード]オプションを使用するときに関連するキーフィールドで並べ替える場合は、関連するキーフィールドは、並べ替えられた出力テーブルには含まれず、混乱する可能性があります。</p>
レコードフィールド	<p>並べ替えられた出力テーブルのレコード全体またはフィールドのサブセットのみを含めるかどうかを指定します。</p> <ul style="list-style-type: none"> ○ レコード -]はレコード全体を含めます。出力テーブルのフィールドは、ソーステーブルと同じ並べ順です。 ○ フィールド -]はフィールドを個別に含めます。出力テーブルのフィールドは選択した順序で表示されます。 <p>1つ以上の演算フィールドを含める場合：</p> <ul style="list-style-type: none"> ○ レコード]を選択すると、フィールドは演算式として保持されます。 ○ フィールド]を選択すると、抽出されたフィールドは適切なデータ型の物理フィールドに変換され、実際に計算された値が格納されます。 <p>テーブルの関係における子テーブルからフィールドを含める場合：</p> <ul style="list-style-type: none"> ○ フィールド]を選択します <p>レコード]オプションを使って子テーブルのフィールドを含めることはできません。</p>
その他のフィールド	<p>フィールド]を選択した場合は、並べ替えられた出力テーブルに含める非キーフィールドを指定します。</p> <ul style="list-style-type: none"> ○ 他のフィールド]リストから適切なフィールドを選択します。 ○ その他のフィールド]をクリックして、該当するフィールドを選択するか、または式を作成して、OK]をクリックします。 <p>メモ</p> <p>キーフィールドは自動的に出力テーブルに含まれます。他のフィールド]として指定されると無視されます。</p> <p>グループとして、出力テーブルの他のフィールドの前にキーフィールドが表示されます。</p> <p>テーブルの関係における子テーブルからフィールドを選択したい場合：</p> <ul style="list-style-type: none"> ○ その他のフィールド]をクリックします。選択済みのフィールド]ダイアログボックスの対象テーブル]ドロップダウンリストを使用して、適切な子テーブルを選択できます。

オプション - 並べ替え]ダイアログボックス	説明
もし次の	<p>(省略可能)レコードを処理から除外する条件を作成できます</p> <p>[F]テキストボックスに条件を入力するか、または [F]ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p>
変換先	<p>出力テーブルの名前と場所を指定します。</p> <ul style="list-style-type: none"> 出力テーブルを Analytics プロジェクト フォルダーに保存するには、-テーブル名のみを入力します。 プロジェクト フォルダー以外の場所に出力テーブルを保存するには、絶対ファイルパスや相対ファイルパスを指定したり、保存先]をクリックし、別のフォルダーに移動します。 <p>たとえば、C:\Results\Output.fil または Results\Output.fil のように指定します。</p> <p>保存する出力テーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。</p> <p>Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。</p> <p>メモ</p> <p>Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
ローカル	<p>サーバーテーブルに接続している場合は、出力テーブルを保存する場所を指定します。</p> <ul style="list-style-type: none"> ローカルを選択 -]では、Analytics プロジェクトと同じ場所、または移動先の場所に出力テーブルを保存します。 AX サーバーの Prefix フォルダーに出力テーブルを保存する場合は、ローカル-]を選択解除されたままにしておきます。
出力テーブルを開く	<p>操作の完了時に、出力結果を含んでいる Analytics テーブルを自動的に開くかどうかを指定します。</p>

詳細]タブ

オプション - 並べ替え]ダイアログボックス	説明
範囲パネル	<p>処理されるソーステーブルのレコードを指定します。</p> <ul style="list-style-type: none"> すべて-(デフォルト) ソーステーブルのすべてのレコードが処理されます。 先頭 -- このオプションを選択してテキストボックスに数を入力すると、ソーステーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 次 -- このオプションを選択してテキストボックスに数を入力すると、ソーステーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。

オプション- 並べ替え]ダイアログ ボックス	説明
	<p>行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。</p> <ul style="list-style-type: none"> ○ While -- WHILE ステートメントを使用して、条件に基づいてソーステーブル内のレコードの処理を制限するには、このオプションを選択します。 <ul style="list-style-type: none"> ● [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。 ● WHILE ステートメントでは、指定した条件が true と評価される間のみ、レコードを処理することができます。 ● While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。 <p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>
既存のファイルに追加する	<p>出力結果が既存の Analytics テーブルの最後に追加されることを指定します。</p> <p>結果として生じる結合されたテーブルは、並べ替えられていないテーブルと見なされます。これは、並べ替えられたレコードを出力先テーブルの末尾に追加するときに、出力先テーブルの既存の並べ替え順が何も考慮されないからです。</p> <ul style="list-style-type: none"> ○ レコードまたはフィールドと出力先テーブルの構造がまったく同じであることがわかっている場合は、既存のファイルに追加する]を選択します。 ○ 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、既存のファイルに追加する]を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。 <p>メモ</p> <p>出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、既存のファイルに追加する]を選択解除されたままにしておくことをお勧めします。</p> <p>追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>
OK	<p>処理を実行します。</p> <p>上書きを確認するメッセージが表示されたら、適切なオプションを選択します。</p> <p>最後に追加] オプションが表示されることを期待していたのに表示されない場合は、[いい]をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。</p>

レコードのインデックス付け

インデックスは、独立したインデックスファイル(.inx ファイル)を作成します。このファイルによって、物理的な順序(つまり、未処理データの順序)ではなく、シーケンシャルな順序で Analytics テーブル内のレコードにアクセスできます。

インデックスは、テーブル内のデータを物理的に並べ替えません。ただし、テーブルのインデックスがアクティブであるときは、ビュー内のデータは、インデックスで指定された順序に従って並べ替えられ、分析は、この順序に基づいてデータを処理します。テーブルに複数のビューがある場合は、すべてのビューがアクティブなインデックスの対象になります。

インデックスがアクティブなときは、ステータスバーのレコード数の前に、「インデックス」という語が表示されます。たとえば、「インデックスレコード: 500」と表示されます。

インデックスがアクティブでないときは、ビュー内のレコードは元の物理的な順序に戻ります。Analytics テーブルを開くとき、既存のすべてのインデックスは、デフォルトで非アクティブになっています。

メモ

レコードの並べ替えは、インデックスを作成することの代替方法です。状況によっては、並べ替えがより良い選択となることがあります。詳細については以下を参照 "並べ替えまたはインデックス" ページ 1183

インデックスとフィールド型

演算フィールドや一時的に作成した式など、データ型に関係なく、あらゆる種類のフィールドのインデックスを作成することができます。

論理フィールドでのインデックスでは、**[フィールド リストにフィルターを含める]**を選択する必要があります (**ツール > オプション > インターフェイス**)。

単一テーブルの複数のインデックス

単一のテーブルに対して複数のインデックスを作成し、必要に応じてインデックス間を切り替えることができます。これは、データのセットを一次審査する場合に役立ちます。一度にアクティブにできるインデックスは1つだけです。

入れ子のインデックス

1つのキーフィールドを使用してレコードのインデックスを作成することができます。また、複数のキーフィールド(主キーフィールド、副キーフィールドなど)を基にしてインデックスを作成し、入れ子のインデックス構

成を作成することができます。

入れ子のインデックスでは、キーフィールド全体で昇順と降順が混在しても、データ型が混在してもかまいません。

昇順と降順が混在する入れ子のインデックス

並べ替えられていない取引テーブルで、日ごとの最大取引金額を表示できます。日付キーフィールドで昇順に、各日の金額キーフィールドで降順に、テーブルにインデックスを作成します。

Date フィールド (昇順)	Amount フィールド (降順、入れ子)
15 Jan 2011	\$2300.00
15 Jan 2011	\$1200.00
15 Jan 2011	\$600.00
16 Jan 2011	\$900.00
16 Jan 2011	\$100.00
17 Jan 2011	\$4700.00
17 Jan 2011	\$900.00
17 Jan 2011	\$500.00

インデックスは Analytics テーブルに制限されま す

インデックスは Analytics テーブル、つまり、.fil ソース データ ファイルを持つテーブルに限定されています。ローカルベースとサーバーベースの Analytics テーブルは、.fil ファイルを持っている場合は、どちらもインデックスを作成できます。

データベース プロファイルを使用して接続しているデータベース テーブルのインデックスを作成することはできません。これらには .fil ファイルがありません。データは、データベースから直接読み取られます。このような状況でデータを順序付けるには、データベースへのアクセス時に、**データ定義ウィザード**で SQL の ORDER 句を使用することができます。

一部の Analytics コマンドに必要なインデックス

インデックスは、Analytics テーブルを検索するときに、**検索** ダイアログボックスで **文字検索** および **式検索** オプションを使用するための前提条件です(それぞれのオプションは、FIND コマンドと SEEK コマンドに相当します)。

次の場合にのみこれらのオプションを使用できます。

- テーブルにインデックスが作成されている
- インデックスが有効である
- インデックスの主キーフィールドが昇順にインデックスが作成された文字フィールドである

テーブルには入れ子のインデックスを持つことができますが、検索されるのは主キーフィールドだけです。

条件付きインデックス

インデックスには、**[f]**、**先頭**、**次**、および **[While]** パラメーターを含めることができます。これらが指定されている場合、インデックスは条件付きインデックスになります。条件付きインデックスがアクティブなときは、条件に一致するレコードのみがインデックス付けされます。それらが表示されて、分析に利用できません。

詳しく表示

インデックスをアクティブにするたびに、条件が自動的に再適用されます。条件付きインデックスを使って大きなテーブルのサブセットを作成することにより、特定のタイプの分析を円滑にすることができます。

If パラメーターを持つ条件付きインデックスがアクティブなときは、ステータスバーのレコード数の前に、「**フィルターされたインデックス**」という語が表示されます。たとえば、「**フィルターされたインデックスレコード: 500**」と表示されます。**先頭**、**次**、および **[While]** パラメーターを持つ条件付きインデックスがアクティブなときは、条件の付いていないインデックスと同様、レコード数の前には「**インデックス**」という語が表示されません。

インデックスとフィルター

If パラメーターまたはフィルターを持つ条件付きインデックスを作成する際には、グローバルフィルター(ビューのフィルター)、ローカルフィルター(コマンド内のフィルター)、またはその両方を含めることができます。

次の表は、インデックスにフィルターを含める例と、サンプルデータに対するインデックスの効果を示します。

フィルターはインデックス構文の一部なので、ログや、インデックスの詳細で見ることができます。詳細については、「[インデックスの詳細の表示](#)」ページ 1202を参照してください。

フィルターの種類	説明/インデックス構文	vendor_ID	trans_amount
なし	インデックスなし (物理的な順序)	212 108 359 108 359 212 359 212 359 108	1400.00 3400.00 1600.00 1100.00 3400.00 1200.00 2200.00 1700.00 1400.00 2300.00
グローバル	インデックスには、業者番号 359 のレコードのみが含まれます <code>INDEX ON trans_amount TO "vendor 359 transactions"</code> グローバルフィルター: <code>vendor_ID = "359"</code>	359 359 359 359	1400.00 1600.00 2200.00 3400.00
ローカル	インデックスには、取引金額が \$2000 以上のレコードのみが含まれます <code>INDEX ON trans_amount IF trans_amount >= 2000 TO "trans amount 2000 or greater"</code>	359 108 108 359	2200.00 2300.00 3400.00 3400.00
グローバルローカル	インデックスには、業者番号 359 のうち取引金額が \$2000 以上のレコードのみが含まれます <code>INDEX ON trans_amount IF trans_amount >= 2000 TO "vendor 359 transactions 2000 or greater"</code> グローバルフィルター: <code>vendor_ID = "359"</code>	359 359	2200.00 3400.00

手順

レコードのインデックス付け

アクティブなテーブルの 1 つまたは複数のキーフィールドによって、レコードをインデックス付けすることができます。また、その結果作成されるインデックスを利用して、元となるデータの物理的な順序に影響を与ることなく、一時的にレコードを並べ替えることができます。

方法の説明

1. **データ > インデックス**の順に選択します。
2. **メイン**タブで、次のいずれかを行います。

- **インデックスの対象**リストから、インデックスを作成するフィールドを選択します。
- **インデックスの対象**をクリックして、フィールドを選択するか、または式を作成します。

複数のフィールドを選択する場合、フィールドを選択する順序が入れ子になったインデックスの優先度を示します。レコードが最初に選択したフィールドでインデックス付けられ、最初のフィールドに複数の値の出現が存在する場合は、次にそのグループ内のレコードが2番目に選択したフィールドでインデックス付けられる、というようになります。追加のフィールドを選択していない場合は、グループ内のレコードは元と同じ並び順が保持されます。

式や演算フィールドを使用したインデックスの詳細については、「演算キーフィールドを使った並べ替えとインデックスの作成」ページ 1204を参照してください。

メモ

インデックスが作成されるフィールドの合計の長さは、247文字を超えてはいけません。

3. **インデックスの対象**をクリックした場合は、任意で並べ替えの矢印  (デフォルトは昇順) をクリックすることにより、1つまたは複数の選択したフィールドのインデックスを降順にするよう指定することができます。
4. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用してIFステートメントを作成します。

メモ

If条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IFステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

5. 次のいずれかを実行します。
 - **保存先**テキストボックスに、インデックスファイルの名前を指定します。
 - **保存先**ボタンをクリックして、**保存**または **ファイルを保存する名前**ダイアログボックスでインデックスファイル名を指定するか、または既存のインデックスファイルに上書きする場合は、そのファイルを選択します。

Analyticsによってインデックスファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

メモ

インデックスの名前は、64文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

ヒント

インデックスには、そのインデックスによって設定される順序付けの性質を説明する、意味のある名前を付けることが最良の策です。たとえば、テーブルをDateによって昇順で並べ、日にちごとではAmountによって降順で並べるインデックスならば、"Date_Amount_D"という名前にします。

6. インデックスを直ちにアクティブにしたいかどうかに応じて、**出力テーブルを開く**]をオンまたはオフにします。

テーブルのインデックスはいつでも、ビューの右上にある **インデックス**]ドロップダウンリストから選択することで、アクティブにすることができます。

7. **詳細**]タブをクリックします。
8. **適用範囲**]パネルで、適切な出力オプションを選択します。

- **すべて**
- **先頭**
- **次へ**
- **While**

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキストボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキストボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While]テキストボックスに条件を入力するか、または While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>

メモ

"**先頭**" または "**次**" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。

ビューでクイックソートが実行されている場合、"**次**" は "**先頭**" のように動作します。

9. **OK**]をクリックします。
10. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

インデックスのエントリが、ビュータブの **[インデックス]**ドロップダウンリストに追加されます。 **出力テーブルを開く]**を選択した場合、インデックスがアクティブ化され、インデックスに従ってテーブルが並べ替えられます。

インデックスを非アクティブにする

インデックスは、作成時、または作成後いつでもアクティブにすることができます。Analytics テーブルを開くとき、既存のすべてのインデックスは、デフォルトで非アクティブになっています。

方法の説明

- インデックスをアクティブにするには、次のいずれかを行います。
 - インデックスの作成時に、すぐにインデックスをアクティブにするには、 **[インデックス]**ダイアログボックスで **出力テーブルを開く]**をオンにします。
 - ビューの右上隅にある **[インデックス]**ドロップダウンリストからインデックスを選択します。
- インデックスを非アクティブにするには、次のいずれかを行います。
 - ビューの右上隅にある **[インデックス]**ドロップダウンリストで **"(なし)"**を選択します。
 - 別のインデックスに切り替えます。
 - テーブルを閉じます。

インデックスの詳細の表示

インデックスの詳細(つまり、特定の Index コマンドの実際の構文)を表示することができます。コマンド構文には、キーフィールド、およびすべてのパラメーター、フィルター、式が含まれます。インデックスの詳細は、特定のインデックスがテーブル内のレコードをどのように処理するのかを明らかにします。

方法の説明

1. インデックスを含んでいるテーブルを開きます。
2. **ナビゲーター**でテーブルを右クリックし、 **プロパティ]**を選択します。
3. **[インデックス]**タブをクリックし、インデックス名を選択してから **詳細]**をクリックします。
インデックス プロパティ]ダイアログボックスに、インデックスの詳細が表示されます。
 - **[コマンド -]**には、Index コマンドの構文が、ローカルフィルターも含めて表示されます。
 - **[フィルター -]**には、インデックスの一部であるグローバルフィルターの構文が表示されます。
4. **OK]**をクリックし、もう一度 **OK]**をクリックして **テーブルプロパティ]**ダイアログボックスを終了します。

インデックスの管理

テーブルプロパティ ダイアログボックスの **インデックス** タブで、インデックスのコピー、名前変更、または削除を行うことができます。また、ここでインデックスを追加することもできます。

方法の説明

メモ

このようなメンテナンスタスクは、Analytics を介してのみ実行できます。直接 Windows フォルダでインデックスファイル (.inx ファイル) の名前を変更すると、Analytics でそのインデックスを次回アクティブにしたとき、元の名前でインデックスファイルが自動的に再作成されます。直接インデックスファイルを削除すると、そのインデックスを次回アクティブにしたとき、インデックスファイルが自動的に再作成されます。

1. インデックスを含んでいるテーブルを開きます。
2. **ナビゲーター**でテーブルを右クリックし、**プロパティ**を選択します。
3. **インデックス** タブをクリックし、インデックス名を選択してから、次のいずれかを行います。
 - インデックスをコピーするには、**コピー**をクリックします。
インデックス名の末尾に、増分数が付加されたインデックスがコピーされます。
 - インデックスの名前を変更するには、**名前変更**をクリックし、新しい名前を入力したら **OK** をクリックします。

メモ

インデックスの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- インデックスを削除するには、**削除**をクリックし、もう一度 **削除**をクリックします。
4. 新しいインデックスを追加する場合は、**追加**をクリックします。
インデックス ダイアログボックスが表示され、通常の方法でインデックスを作成することができます。
5. **OK** をクリックして **テーブルプロパティ** ダイアログボックスを終了します。

演算キーフィールドを使った並べ替えとインデックスの作成

キーフィールド内のデータの書式が、テーブルの正確な並べ替えやインデックス付けを妨げている場合は、演算キーフィールドを作成することにより、正確な結果を得られる場合があります。並べ替えやインデックス付けを妨げるデータ書式の例として、数値に含まれる数字以外のプレフィックスや先頭のスペースに一貫性がない、大文字で始める名前に一貫性がない、などが挙げられます。演算キーフィールドを作成して、データの一貫性を確保してから、演算キーフィールドを使用して並べ替えやインデックスの作成ができます。

また、厳密にキーフィールドの値の順番どおりではない方法でテーブルを整列させるために、演算キーフィールドを使用して並べ替えやインデックス付けを行うこともできます。たとえば、演算キーフィールドを使用して、テーブル内のレコードの物理順序を逆にしたり、偶数の金額値をグループ化したり、特定の文字列を含んでいる名前をグループ化したりすることができます。

演算キーフィールド値を表示する

演算キーフィールドを使用してインデックス付けを行う最良の実践は、ビューに演算フィールドを追加することです。そうすると、演算フィールドを使用してどのようにテーブルのインデックスが作成されるかを正確に見ることができます。演算キーフィールドを使用して並べ替えを行った場合、新しい並べ替え済みのテーブルには、その演算フィールドが自動的に含まれます。

並べ替えおよびインデックス処理で直接式を取り込み、演算キーフィールドと同じ結果を得ることができます。ただし、このアプローチは推奨されません。テーブルを並べ替えるために使用されている演算値が表示されないためです。

演算キーフィールドを使った並べ替えとインデックスの作成の例

演算キーフィールドを使った並べ替えとインデックスの作成の複数の例は以下のとおりです。比較のため、物理キーフィールドの非演算、順次的な順序も含まれます。

詳しく表示

説明/演算キーフィールド式	Analytics 関数	キーフィールド (物理的な順序)	キーフィールド (順次的な順序、 Analytics 非 Unicode)	演算キー フィールド (順次的な順序)	キーフィールド (演算キーフィールドの順序に基づく順序)
数字以外の文字とスペースを無視し、数字のみを並	INCLUDE()	92 12	20 #85	12 20	12 20

データの分析

説明/演算キー フィールド式	Analytics 関 数	キーフィールド (物理的な順序)	キーフィールド (順次的な順序、 Analytics 非 Unicode)	演算キー フィールド (順次的な順 序)	キーフィールド (演算キーフィー ルドの順序に基づ く順序)
べ替え/インデックス の対象とします INCLUDE(dept_ID, "1234567890")		T-38 20 #85	12 92 T-38	38 85 92	T-38 #85 92
大文字と小文字 の違いによって並 べ替え/インデックス の相違が生じない ようにします UPPER(last_name)	UPPER()	Smythe JONES Smith JOHNSON SMYTHE Jones SMITH Johnson	JOHNSON JONES Johnson Jones SMITH SMYTHE Smith Smythe	JOHNSON JOHNSON JONES JONES SMITH SMITH SMYTHE SMYTHE	JOHNSON Johnson JONES Jones Smith SMITH Smythe SMYTHE
テーブル内のレコー ドの物理的な順 序を逆にします <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> メモ 演算 キー フィー ルドは 降順 である 必要 があり ます。 レコー ド番 号は、 降順 が適 用され ても常 に昇 順で 表示 されま す </div> RECNO()	RECNO()	82.12 87.00 62.79 97.47 43.00	43.00 62.79 82.12 87.00 97.47	1 2 3 4 5	43.00 97.47 62.79 87.00 82.12
偶数の金額をグ ループにし、グルー	MOD()	82.12 87.00	43.00 62.79	0.00 0.00	43.00 87.00

説明/演算キー フィールド式	Analytics 関 数	キーフィールド (物理的な順序)	キーフィールド (順次的な順序、 Analytics 非 Unicode)	演算キー フィールド (順次的な順 序)	キーフィールド (演算キーフィー ルドの順序に基づ く順序)
<p>プ内の金額を順番 に並べます</p> <p>メモ 演算 キー フィー ルド内 で、 キー フィー ルドの 並べ 替え/ イン デック スを入 れ子に する必 要があ ります</p> <p>MOD(trans_ amount, 1.00)</p>		<p>62.79</p> <p>97.47</p> <p>43.00</p>	<p>82.12</p> <p>87.00</p> <p>97.47</p>	<p>0.12</p> <p>0.47</p> <p>0.79</p>	<p>82.12</p> <p>97.47</p> <p>62.79</p>
<p>特定の文字列を 含んでいる値をグ ループにし、グルー プ内の値を順番に 並べます</p>	FIND()	Lilydale Hardware	Binford Tools	T	Global Trade Hardware

説明/演算キーフィールド式	Analytics 関数	キーフィールド (物理的な順序)	キーフィールド (順次的な順序、 Analytics 非 Unicode)	演算キー フィールド (順次的な順 序)	キーフィールド (演算キーフィー ルドの順序に基づ く順序)
<p>メモ 演算キーフィールドは降順である必要があります。演算キーフィールド内で、キーフィールドの並べ替え/インデックスを入れ子にする必要があります</p> <pre>FIND("Hardware", vendor_name)</pre>					
		Triathlon Group	Bolton Distribution	T	Lilydale Hardware
		Wholesome Hardware	Global Trade Hardware	T	Wholesome Hardware
		Steel Case Manufacturing	Industrial Equipment Co- Op	F	Binford Tools
		Industrial Equipment Co- Op	Lilydale Hardware	F	Bolton Distribution
		Global Trade Hardware	Steel Case Manufacturing	F	Industrial Equipment Co- Op
		Binford Tools	Triathlon Group	F	Steel Case Manufacturing
		Bolton Distribution	Wholesome Hardware	F	Triathlon Group

手順

演算キーフィールドを使った並テーブルのべ替えとインデックスの作成の手順については、下記の一般的な用語で説明されています。

演算フィールドの作成や、並べ替えまたはインデックスの実行に関する詳細については、このガイドの関連するセクションを参照してください。

- 適切な式を使用して、並べ替えまたはインデックスの物理キーフィールドに基づいた演算キーフィールドを作成します。
- インデックス操作を実行しようとしている場合は、ビューに演算キーフィールドを追加します。
- 演算フィールドをキーフィールドとして使用して、通常の並べ替えまたはインデックスの手順を実行します。

- 必要であれば、演算キーフィールドに降順を指定します。一部の式では、テーブルの上部に結果を配置するために、降順であることを必要とします。
- 演算キーフィールド式によって作成されたグループ内の結果を順番に並べたい場合は、並べ替えやインデックスを構成するときに、演算キーフィールドと物理キーフィールドの両方を選択します。演算キーフィールドが物理キーフィールドより優先されるよう、最初に演算キーフィールドを選択するようにしてください。

データのフィルターリング

フィルターは、データを分析する際に不可欠なツールです。テーブル内の現在不要なデータを除外できるので、ターゲットのレコード サブセットのみを分析できます。

除外されたデータは [ビュー] タブに表示されず、抽出などの Analytics 操作によって処理されません。除外されたデータは非表示になるだけで、削除されません。また、フィルターを削除すると、いつでも表示できます。フィルターは、プロジェクト全体ではなく Analytics プロジェクトの特定のテーブルに関連付けられません。

フィルターの仕組み

フィルターは、テーブル内のすべてのレコードを評価して、True (T) または False (F) を返す論理式です。たとえば、`Invoice_Amount > 1000.00` のような式です。

True として評価されたデータは、フィルターリングされたテーブルに含まれたり Analytics 操作の対象として含まれたりするのに対し、False として評価されたデータはそれらから除外されます。

フィルターは、Analytics 内のいくつかの異なる場所から適用でき、さらに、それらを互いに組み合わせて使用することができます。

フィルターのタイプ

Analytics プロジェクトで作成できるフィルターにはさまざまな種類があります。

- グローバルフィルター。ビューフィルターとも呼ばれます。
- クイックフィルター
- ローカルフィルター。コマンド フィルターとも呼ばれます。
- データフィルター

フィルターは、「**その場限りで使用**する」、つまりテーブルに永続的に保存されないようにするか、あるいは後で再利用できるよう、**名前を付けて**テーブルに保存することができます。また、名前付きフィルターはワークスペースに保存すると、複数のテーブル間で共有することもできます。

グローバルフィルター(ビューフィルター)

グローバルフィルターは、テーブルレイアウトに関連付けられている 1 つまたは複数のビューに適用して、表示または処理されるレコードを制限します。グローバルフィルターが適用されると、テーブルで実行されるあらゆる操作は、そのフィルターに含まれるレコードに対してのみ実行されます。

グローバルフィルターは、削除する、別のグローバルフィルターに置き換える、またはテーブルを閉じるまではアクティブなままです。グローバル フィルターを、あるテーブルのデフォルト フィルターとし、そのテーブルを開くときには自動的にグローバルフィルターが適用されるようにすることもできます。

詳細については、「**グローバルフィルター(ビューフィルター)**」 ページ 1213を参照してください。

クイックフィルター

クイックフィルターはグローバルフィルターの一種です。ビューで右クリックし、ショートカットメニューの **クイックフィルター** オプションを使用することにより適用されます。クイックフィルターを使用すると、フィルターの値や条件を手作業で指定しなくてもマウスで選択することができ、有効なフィルター構文を使った式が [フィルター] テキストボックスに自動的に生成されるため、便利です。

フィルターの値と条件はマウスを使って選択する必要があるため、クイックフィルターには特定の制限があります。通常、複数の条件を含む複雑なフィルターを作成する場合には使用できません。

詳細については、「ビューのデータでクイックフィルターを使用する」 ページ 1171を参照してください。

ローカルフィルター(コマンド フィルター)

ローカルフィルターは、単一の Analytics コマンドの単一の実行に適用して、そのコマンドが処理するテーブル内のレコードを制限します。コマンドがその処理を終えた時点で、ローカルフィルターはアクティブでなくなります。

詳細については、「ローカルフィルター(コマンド フィルター)」 ページ 1222を参照してください。

データ フィルター

データフィルターは特定の目的を果たします。これは、印刷イメージ(レポート) ファイルや複数レコードタイプファイルなど、複数のレコードタイプを含んでいるデータソースのデータを選択的に定義するための方法を提供します。ほかの種類フィルターとは異なり、Analytics でのデータ分析における一般的利用を目的としていません。

詳細については、「データフィルターについて」 ページ 835を参照してください。

一時的フィルターと名前付きフィルター

一時的なフィルター

グローバルフィルターやローカルフィルターは、たとえば `Invoice_Amount > 1000.00` のように、フィルター構文を単独で使用して適用することができます。この場合、フィルターはその場限りです。その場限りの一時的フィルターは、テーブルに永続的に保存されません。

一時的なグローバルフィルターは、特定のテーブルに関連付けられているフィルター履歴に表示される間に限り、保持されます。

一時的なローカルフィルターは、単一の Analytics 操作の間しか保持されません。ただし、必要であれば、コマンド ログから取得することができます。

名前付きフィルター

後で再利用するために、グローバルフィルターやローカルフィルターに名前を付けて保存することができます。この場合は、関連付けられた Analytics テーブルに永続的に保存されます。たとえば、一時的フィルターの `Invoice_Amount > 1000.00` に `Inv_greater_than_1K` という名前を付けて保存することができます。

フィルターを再適用するときは、構文を再作成するのではなくフィルター名を指定するので、手間が省けます。また、フィルターに名前を付けることは、多数保存されているフィルターを識別しやすくします。例：

- `Inv_less_than_1K`
- `Inv_1K_to_5K`
- `Inv_greater_than_5K`

フィルターの名前付けと保存は、フィルターを作成した時点、あるいはそれ以降、その一時的フィルターがフィルター履歴に表示されている任意の時点に行えます。名前を付けて保存されると、そのフィルターは、テーブルに関連付けられたすべてのビューでグローバルフィルターとして、また、テーブルで実行されるすべての操作のローカルフィルターとして利用できるようになります。

フィルターに名前を付けて保存する方法や、一時的フィルターを名前付きフィルターに変換する方法については、「ビューへのグローバルフィルターの適用」 ページ 1218を参照してください。

フィルター履歴

テーブルにグローバルフィルターを適用すると、そのテーブルに関連付けられたフィルター履歴にフィルターが保存されます。そのフィルターがフィルター履歴に残っている限り、ビューの上部にある [フィルター] ドロップダウンリストからそれを選択して、再適用することができます。

フィルター履歴には、一時的と名前付きの両方のグローバルフィルターが保存されます。ローカルフィルターは、フィルター履歴には保存されません。

フィルター履歴の詳細：

テーブル、ビュー、およびフィルター履歴	テーブルごとに別々のフィルター履歴を持ちます。同じテーブルの複数のビューは、同じフィルター履歴を共有します。
フィルター履歴の保持	テーブルを閉じて、プロジェクトを閉じて、また Analytics を閉じて、フィルター履歴は保持されます。
リストにおけるフィルターの順序	最近適用したフィルターが、[フィルター] ドロップダウンリストの先頭に表示されます。
保存されるフィルターの最大数	最大 10 個までのフィルターが保存されます。最大数を超えると、最も古いフィルターがリストの下から削除され、最新のフィルターが一番上に追加されます。
重複したフィルター	フィルター履歴内のフィルターは一意です。フィルターを複数回適用しても、重複したフィルター履歴のエントリが発生することはありません。
削除した名前付きフィルター	削除した名前付きフィルターは、フィルター履歴からは削除されませんが、機能しなくなります。

フィルター履歴の消去	[フィルター]テキスト ボックスを右クリックして 履歴のクリア を選択すると、フィルター履歴全体を消去できます。フィルター履歴からフィルターを選択的に削除することはできません。フィルター履歴を消去しても、名前付きフィルターは削除されません。
------------	---------------------------------------------------------------------------------------------------------------------------------

フィルター保持のまとめ

フィルター保持について、次の表にまとめます。

	テーブルへの永続的な保存	フィルター履歴への追加
一時的グローバルフィルター	いいえ	はい
一時的ローカルフィルター	いいえ	いいえ
名前付きグローバルフィルター	はい	はい
名前付きローカルフィルター	はい	いいえ

構成可能なフィルターオプション

2つの構成可能なオプションにより、フィルターの動作のさまざまな側面を制御できます。

フィールド一覧にフィルターを含める	フィールド一覧に名前付きフィルターを表示するかどうかを制御します。 詳細については、"インターフェイス オプション" ページ 123を参照してください。
フィルターされたレコードを表示しない	フィルターされたレコードをビューに表示しないか、あるいは表示はするが視覚的に目立たないようにするかを制御します。 詳細については、"ビュー オプション" ページ 130を参照してください。

グローバルフィルター(ビューフィルター)

グローバルフィルターは、ビューで表示するレコード、Analytics 操作で処理するレコードを制限します。レコードを大まかにフィルターリングするための、条件が1つの単純なフィルターや、非常に特定のデータのサブセットを分離するための、条件が複数ある複雑なフィルターを構築することができます。

シンプルなフィルターと複雑なフィルター

シンプルなフィルター

以下などの特定のエンティティに関連するレコードは、条件が1つのシンプルなフィルターを作成すれば抽出できます。

- 名前
- 日付
- 勘定科目番号

たとえば、買掛金テーブルで、特定の業者に関するレコードのみを表示または処理できるようにするため、その業者の業者番号でフィルターすることができます。

```
Vendor_No = "14438"
```

複雑なフィルター

より限定されたデータサブセットを抽出したい場合は、複数の条件を使って複雑なフィルターを作成します。

たとえば、買掛金テーブルから次の3つの条件をすべて満たす請求書のみを抽出するフィルターを作成できます。

- ベンダー 14438
- 2014年に送信
- \$1000.00以上

```
(Vendor_No = "14438") AND (BETWEEN(Invoice_Date,`20140101`,`20141231`)) AND  
(Invoice_Amount >= 1000.00)
```

ビューへは一度に1つしかフィルターを適用できませんが、上の例で示したように、AND や OR などの論理演算子を使用すれば1つのフィルター内で複数の条件を組み合わせることができます。

論理演算子の詳細については、"Analytics 式の演算子" ページ 868を参照してください。

フィルター式で抽出の条件を指定する

`Vendor_No = "14438"`などのフィルター式ビルダーを作成するには、レコードがフィルターリング後のテーブルに抽出されるための条件を指定します。

フィルター式でブール論理の **True** と評価されたレコードが、フィルター後のテーブルに抽出されます。**False** と評価されたレコードは除外されます。

したがって、この例の場合は、次のようになります。

- ベンダー番号 14438 のレコードはすべて True と評価されるため、抽出される
- ベンダー番号 90215 のレコードはすべて False と評価されるため、除外される

ヒント

どのようなレコードをフィルターで包含するのかわかりやすくするため、フィルター式の前に "レコード抽出条件:" というフレーズを付けてみてください。このテクニックは、複雑な式を作成したり、NOT や Not Equal To (<>) などの否定を行う論理演算子を使用する場合に役立つことがあります。

フィルター式の例

次の例は、同じフィルター値グループと同じデータセットを使用するフィルターリングのバリエーションです。

レコード抽出条件:

- すべての値に適合する
- いずれかの値に適合する
- すべての値に適合しない
- いずれかの値に適合しない

すべての値に適合するレコードを抽出する

次のフィルター式によってフィルターリング後のテーブルに抽出されるレコードとは、ベンダーが 14438、日付が 2014 年 7 月 15 日、かつ請求金額が \$1,000 であるレコードです。

つまり、これら 3 つの条件のすべてに適合するレコードがフィルターリング後のテーブルに抽出されます。

```
(Vendor_No = "14438") AND (BETWEEN(Invoice_Date, `20140101`, `20141231`)) AND (Invoice_Amount >= 1000.00)
```

含まれているか?	ベンダー番号	請求日付	請求金額
はい	14438	2014 年 7 月 15 日	\$1000

含まれているか?	ベンダー番号	請求日付	請求金額
いいえ	90215	2014年7月15日	\$1000
いいえ	14438	2015年5月25日	\$1000
いいえ	14438	2014年7月15日	\$500
いいえ	90215	2015年5月25日	\$500

いずれかの値に適合するレコードを抽出する

次のフィルター式によってフィルターリング後のテーブルに抽出されるレコードとは、ベンダーが14438、日付が2014年7月15日、または請求金額が\$1,000であるレコードです。

つまり、これら3つ条件のいずれかに適合するレコードがフィルターリング後のテーブルに抽出されます。

```
(Vendor_No = "14438") OR (Invoice_Date = `20140715`) OR (Invoice_Amount = 1000.00)
```

含まれているか?	ベンダー番号	請求日付	請求金額
はい	14438	2014年7月15日	\$1000
はい	90215	2014年7月15日	\$1000
はい	14438	2015年5月25日	\$1000
はい	14438	2014年7月15日	\$500
いいえ	90215	2015年5月25日	\$500

すべての値に適合しないレコードを抽出する

次のフィルター式によってフィルターリング後のテーブルに抽出されるレコードとは、ベンダーが14438以外、日付が2014年7月15日以外、かつ請求金額が\$1,000以外であるレコードです。

つまり、これら3つの条件のすべてに適合するレコードがフィルターリング後のテーブルに抽出されます。

```
(Vendor_No <> "14438") AND (Invoice_Date <> `20140715`) AND (Invoice_Amount <> 1000.00)
```

含まれているか?	ベンダー番号	請求日付	請求金額
いいえ	14438	2014年7月15日	\$1000
いいえ	90215	2014年7月15日	\$1000
いいえ	14438	2015年5月25日	\$1000
いいえ	14438	2014年7月15日	\$500
はい	90215	2015年5月25日	\$500

いずれかの値に適合しないレコードを抽出する

次のフィルター式によってフィルターリング後のテーブルに抽出されるレコードとは、ベンダーが14438以外、日付が2014年7月15日以外、または請求金額が\$1,000以外であるレコードです。

つまり、これら3つ条件のいずれかに適合するレコードがフィルターリング後のテーブルに抽出されます。

```
(Vendor_No <> "14438") OR (Invoice_Date <> `20140715`) OR (Invoice_Amount <> 1000.00)
```

含まれているか?	ベンダー番号	請求日付	請求金額
いいえ	14438	2014年7月15日	\$1000
はい	90215	2014年7月15日	\$1000
はい	14438	2015年5月25日	\$1000
はい	14438	2014年7月15日	\$500
はい	90215	2015年5月25日	\$500

部分一致

文字データをフィルターリングするときは部分一致がサポートされます。つまり、フィルターリングに使用するフィールドに含まれる長い値の一部をフィルター値に指定できます。

例：

- `Vendor_Name = "R"` は、テーブルを "R" で始まる名前の業者に制限します。
- `Address = "PO Box"` は、テーブルを "PO Box" で始まる住所に制限します。

メモ

フィルター値は、一致を成すフィールドの先頭に現れる必要があります。

正確な文字比較を行う] オプションがオフ(デフォルト 設定) の場合、部分一致は有効になります。オプションをオンにすると、部分一致は無効になり、フィルター値は、一致を成すフィールドの値と正確に一致しなければなりません。詳細については、"テーブルオプション" ページ 125を参照してください。

フィルターの保持

グローバルフィルターは、削除する、別のグローバルフィルターに置き換える、またはテーブルを閉じるまではアクティブなままです。グローバルフィルターを、あるテーブルのデフォルト フィルターとし、そのテーブルを開くときには自動的にグローバルフィルターが適用されるようにすることもできます。

グローバルフィルターはローカルフィルターとは異なります。ローカルフィルターは、単独の Analytics 操作の1回の実行時にのみアクティブになるフィルターです。

グローバルフィルターがアクティブになっている場合、ステータス バーには **グローバルフィルター** インジケータが表示され、それに続いて、フィルター構文(一時的なフィルターの場合) またはフィルター名(名前付きフィルターの場合) が表示されます。

- 一時的なフィルター- グローバルフィルター: (Vendor_No = "14438")
- 名前付きフィルター- グローバルフィルター: Vend_14438

グローバルフィルターを作成および適用するさまざまな方法

グローバルフィルターを作成および適用するには、次のようにさまざまな方法があります。

- [フィルター] テキスト ボックスに手動でフィルター構文を入力する
- クイックフィルターを作成する
- **式ビルダー**を使用してフィルターを作成する、または既存のフィルターを選択する
- [フィルター] ドロップダウン リストから既存のフィルターを選択する

ビューへのグローバルフィルターの適用

グローバルフィルターを作成するか、既存のフィルターを選択し、それをビューに適用することによって、表示されるレコードやAnalytics操作で処理されるレコードを制限することができます。

また、グローバルフィルターを、あるテーブルのデフォルトフィルターとして指定し、そのテーブルを開くときには自動的にグローバルフィルターが適用されるようにすることもできます。

単純なフィルターは [フィルター] テキスト ボックスに直接入力するか、クイックフィルターを使用して作成できます。複数の条件を用いる複雑なフィルターの場合は、**式ビルダー**を使用した方が簡単に作成できます。

新しいフィルターを作成する

新しいグローバルフィルターを作成するには、次のいずれかを使用します。

- **[フィルター] テキスト ボックス** - フィルター式 (`Invoice_Amount > 1000.00` など) を入力し、**[フィルターの設定]**  をクリックします。

このフィルターは一時的なもので、テーブルに関連付けられているフィルター履歴に表示される間に限り、保持されます。

- **クイックフィルター** - クイックフィルターを作成します。詳細については、「ビューのデータでクイックフィルターを使用する」 ページ 1171を参照してください。

このフィルターは一時的なもので、テーブルに関連付けられているフィルター履歴に表示される間に限り、保持されます。

ヒント

クイックフィルターを使用して有効なフィルター構文を自動的に作成してから、それを手動で編集することで、必要とするフィルターを作成できます。

- **式ビルダー** - **[ビューフィルターの編集]**  をクリックして**式ビルダー**を開きます。フィルター式を作成して、任意で **[次の名前保存]** テキスト ボックスにフィルターの名前を入力して **[OK]** をクリックします。

フィルターの名前を入力した場合、そのフィルターはテーブルで永続的に保存されます。名前を入力しない場合、このフィルターは一時的なもので、テーブルに関連付けられているフィルター履歴に表示される間に限り、保持されます。

フィルター名は、256文字以内の英数字で指定できますが、名前の先頭に数字は使用できません。

式ビルダーの使用方法の詳細については、「式ビルダーによる式の作成」 ページ 872を参照してください。

フィルターされたレコードの数をカウントします。

グローバルフィルターを適用した後、以下の手法を用いてフィルターに含まれるレコードの数をカウントします。

1. Analytics メインメニューから、[カウント - 2. [OK] をクリックします。

フィルターに含まれるレコードの数とテーブル内のレコードの総数は、Analytics インターフェイスの下部にあるステータスバーに表示されます。たとえば: レコード: 108/772

グローバルフィルターをテーブルのデフォルト フィルターとして指定する

1. [ビューフィルターの編集 フィルター名は、256 文字以内の英数字で指定できますが、名前の先頭に数字は使用できません。
2. [編集 > テーブルレイアウト] の順にクリックします。
3. [フィールド/式の編集] タブでフィルターの名前をダブルクリックします。
4. [デフォルト フィルター] オプションをオンにします。
5. [入力を受け入れる]  をクリックし、[閉じる 

このフィルターはテーブルに保存されたので、今後、このテーブルを開くときには、そのフィルターが自動的に適用されます。1 つのテーブルに対し、一度に指定できるデフォルト フィルターは 1 つのみです。

デフォルト フィルターを削除する場合は、[編集 > テーブルレイアウト] の順に選択し、そのフィルター名をダブルクリックして [デフォルト フィルター] オプションをオフにします。このフィルターはその後もテーブルへ適用されますが、デフォルト フィルターではなくなります。

既存のフィルターを選択する

既存のフィルターを選択する場合は、次のいずれかを使用します。

- [フィルター] ドロップダウン リスト - [フィルター] ドロップダウン リストからフィルターを選択します。
このリストには、テーブルに対して最近適用された 10 個のフィルターが表示されます。
- 式ビルダー - [ビューフィルターの編集 

テーブルで永続的に保存された名前付きフィルターのみが [フィルター] リストに表示されます。名前を付けなかった一時的なフィルターは [フィルター] リストには表示されません。

一時的なフィルターを名前付きフィルターに変換する

1. 一時的なフィルターをテーブルへ適用します。
2. **ビューフィルターの編集**  をクリックして式ビルダーを開きます。 **次の名前で保存** テキストボックスにそのフィルターの名前を入力して **OK** をクリックします。
そのフィルターはテーブルで永続的に保存されます。

現在適用されているフィルターを削除する

フィルターの削除  をクリックします。

これはフィルター自体を完全に削除するわけではありません。一時的なフィルターは、テーブルに関連付けられているフィルター履歴に表示される間に限り、保持されます。名前付きフィルターの場合は、テーブルで永続的に保存されます。

名前付きフィルターの管理

[フィルター] ダイアログボックスを使用すると、テーブルに関連付けられた名前付きフィルターを使用して、次のアクションを実行できます。

- 追加
- 変更
- 重複
- 名前変更
- 削除

変更を行うことなく、名前付きフィルターの構文を見ることもできます。

メモ

現在テーブルに適用されているフィルターについては、名前を変更したり削除したりすることはできません。

1. 管理する名前付きフィルターがあるテーブルを開きます。
2. **編集 > フィルター** の順にクリックします。
Analyticsテーブルに関連付けられている名前付きフィルターが表示されます。
3. 新しいフィルターを追加する場合は、**新規作成** をクリックし、**式ビルダー** を使ってフィルターを作成します。

メモ

フィルターを定義する場合に使用できるのは論理式のみです。

式ビルダーの使用方法的詳細については、「式ビルダーによる式の作成」ページ 872を参照してください。

4. 既存のフィルターについて作業する場合は、一覧の中からフィルターを選択し、次のいずれかを実行します。
 - **OK**]をクリックして、選択したフィルターを**式ビルダー**で表示または変更します。既存のフィルター構文を表示または変更したら、**OK**]をクリックします。
フィルターを変更した場合、**OK**]をクリックして**式ビルダー**を閉じると、そのフィルターが更新されます。
 - **複製**]をクリックして、選択したフィルターの複製を作成します。フィルターの完全なコピーを作成するには、**完了**]をクリックします。複製されたフィルターで使用される式を変更するには、**OK**]をクリックします。

ヒント

複雑なフィルターを複製して変更することで、フィルターを一から作成するよりも作業が簡単になります。

- **名前変更**]をクリックし、テキストボックスに新しい名前を入力したら、**OK**]をクリックします。
フィルター名は、256文字以内の英数字で指定できますが、名前の先頭に数字は使用できません。フィルターの既存の値を使用する場合は **完了**]をクリックし、フィルターで使用される式を編集する場合は **OK**]をクリックします。
- **削除**]をクリックして、フィルターを削除します。確認ダイアログボックスでもう一度 **削除**]をクリックし、**完了**]をクリックしてダイアログボックスを閉じます

ローカルフィルター(コマンド フィルター)

ローカルフィルターは Analytics 操作の一環として、その操作で処理するレコードを制限するために適用します。たとえば、買掛金テーブルに対する合計操作に、特定の業者から提示された請求書の総額を計算するためのローカルフィルターを含めることができます。

ローカルフィルターの作成に使用する式は、グローバルフィルター用の式の場合と同じ方法で機能します。詳細と具体的な例については、"グローバルフィルター(ビューフィルター)" ページ 1213を参照してください。

ローカルフィルターの存続期間

ローカルフィルターは、単独の Analytics 操作の1回の実行に対して適用されるもので、ビュー内のレコードの表示は変更しません。操作が完了すると、ローカルフィルターは非アクティブになります。

ローカルフィルターはグローバルフィルターとは異なります。グローバルフィルターは、削除する、別のグローバルフィルターに置き換える、またはテーブルを閉じるまではアクティブなままです。

ローカルフィルターとグローバルフィルターの同時適用

ローカルフィルターとグローバルフィルターは同時に適用することができます。この場合は、両方のフィルターの条件に一致するレコードのみを対象として Analytics 操作を処理します。

2つのフィルターは論理的に一貫性を持たせる必要があります。一方のフィルターによって除外されるレコードを、もう一方のフィルターで包含することはできません。たとえば、グローバルフィルターには `(請求金額 >= 1000.00)` という条件があり、ローカルフィルターには `(請求金額 >= 500.00)` という条件がある場合は、ローカルフィルターの指定にもかかわらず、\$1000.00 未満の請求金額は処理の対象から除外されます。

ローカルフィルターの作成

Analytics 操作の一環として、ローカルフィルターを作成して、または既存の名前付きフィルターを選択して、表示するレコードを制限することができます。

単純なフィルターであれば、Analytics コマンド ダイアログ ボックスにある **[f]** テキスト ボックスに手動で入力することができます。

複数の条件を用いる複雑なフィルターの場合は、**式ビルダー**を使用した方が簡単に作成できます。ローカルフィルターに名前を付けて保存する場合は、**式ビルダー**で作成する必要があります。

1. Analytics コマンド ダイアログ ボックスで次のいずれかを実行します。

- If テキスト ボックスにフィルター式を入力します。

たとえば、`Invoice_Amount > 1000.00` です。このフィルターは一時的なもので、適用されるのはその Analytics 操作の処理中のみです。

- [f] をクリックしてフィルターを**式ビルダー**で作成、または既存のフィルターを選択します。

2. **式ビルダー**を使用する場合は、以下のいずれかの手順を実行します。

- フィルター式を作成し、任意で **次の名前**で保存] テキスト ボックスにフィルターの名前を入力して **OK**] をクリックします。

フィルターの名前を入力した場合、そのフィルターはテーブルで永続的に保存されます。名前を入力しなかった場合、このフィルターは一時的なもので、適用されるのはその Analytics 操作の処理中のみです。

フィルター名は、256 文字以内の英数字で指定できますが、名前の先頭に数字は使用できません。

式ビルダーの使用方法の詳細については、「**式ビルダー**による式の作成」 ページ 872を参照してください。

- **フィルター**] リストで名前付きフィルターをダブルクリックして、**OK**] をクリックします。

テーブルで永続的に保存された名前付きフィルターのみが **フィルター**] リストに表示されます。名前を付けなかった一時的なフィルターは **フィルター**] リストには表示されません。

データの検索

Analytics テーブルからデータを検索するには、複数の方法を使用することができます。

必要な結果	検索式を入力する場所	方法
一致するすべてのレコードを分離する	[フィルター]テキスト ボックス	<ul style="list-style-type: none"> 基本検索 -: "等しい"(=)、"より小さい"(<)、または "より大きい"(>) などの演算子を使って基本検索を行います。 クイック検索 -: クイック検索機能またはクイックフィルター機能を使って、クイック検索を行います。 関数 -: Analytics 関数を使って検索を行います。
一致する最初のレコードを選択する	検索]ダイアログボックス	<ul style="list-style-type: none"> コマンド -: Analytics のコマンドを使って検索を行います。

ヒント

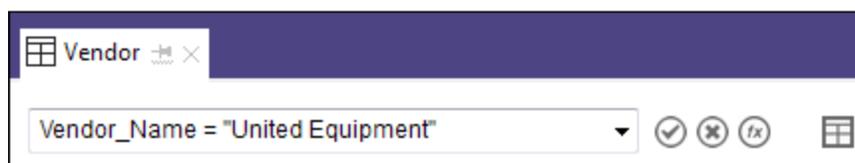
一般に、ユーザーは検索を行い、一致するすべてのレコードを分離して一連の結果を返します。このため、通常は [フィルター]テキスト ボックスを使用するので、検索]ダイアログボックスは無視できます。

一致するすべてのレコードを検索して分離する

基本検索

基本検索を行うには、テーブルビューの上部にある [フィルター]テキスト ボックスに式を入力し、Enter キーを押します。

以下の例では、Vendor_Name フィールドに "United Equipment" という名前が入っているすべてのレコードを分離しています。



演算子を使用した他の基本検索

- \$5,000.00 以上の請求書を分離します。

```
Invoice_Amount >= 5000
```

- 2017 の第 3 四半期の請求書を分離します。

```
(Invoice_Date >= `20170701`) AND (Invoice_Date <= `20170930`)
```

ヒント

検索式は、それがシンプルな場合にのみ、ゼロから作成します。複雑な検索については、クイックフィルター手法か、または関数を使った高度な検索を使用します。

空白、空、または無効な値の検索

空白テキストまたは数値か、空白または無効な日付時刻値を検索できます。空白以外の値を検索するには、式で使用する演算子を変更します。

空白または空白以外のテキスト値

- Vendor_Name フィールドが空白のすべてのレコードを分離します。

```
Vendor_Name = " "
```

```
ISBLANK(Vendor_Name)
```

- Vendor_Name フィールドが空白ではないすべてのレコードを分離します。

```
Vendor_Name <> " "
```

```
NOT(ISBLANK(Vendor_Name))
```

空白または空白以外の数値

- Invoice_Amount フィールドが空白またはゼロ (0) のすべてのレコードを分離します。

```
Invoice_Amount = 0
```

- Invoice_Amount フィールドが空白でもゼロ (0) でもないすべてのレコードを分離します。

```
Invoice_Amount <> 0
```

空白または空白以外の日付時刻値

- Invoice_Date フィールドが空白であるか、値が無効であるすべてのレコードを分離します。

```
Invoice_Date = `19000101`
```

```
NOT VERIFY(Invoice_Date)
```

フィールドで使用される日付形式と一致しないか、日付が存在しない場合は、日付値が無効になることがあります。例: 31 April 2020。

- Invoice_Date が空白ではなく、値が無効であるすべてのレコードを分離します。

```
Invoice_Date <> `19000101`
```

```
VERIFY(Invoice_Date)
```

基本検索のガイドライン

<p>フィールド名</p>	<p>検索するフィールド名としては、テーブルビュー内の表示名でなく、テーブルレイアウト内の物理フィールド名を指定する必要があります。</p> <p>ヒント 物理フィールド名を確認するには、テーブルビュー内の列ヘッダーを右クリックし、プロパティを選択します。</p>
<p>複数のフィールドで検索を行う</p>	<p>複数のフィールドで検索を行う式を作成することができます。レコード内のすべてのフィールドにまたがって検索を行う最も簡単な方法は、関数を使って検索を行うことです。詳細については、「Analytics関数を使って検索とフィルタリングを行う」ページ 1235を参照してください。</p>

部分一致	検索語の部分一致はサポートされていません。 部分一致の使用方法については、"Analytics関数を使って検索とフィルタリングを行う" ページ 1235を参照してください。
二重引用符	テキストの検索語は"二重引用符"で囲む必要があります。
バッククオート	日付時刻の検索語は`バッククオート`で囲む必要があります。
日付時刻書式	<ul style="list-style-type: none"> 日付時刻の検索語にはYYYYMMDDまたはYYMMDDの書式を使用する必要があります。 時刻の部分にはhhmmss書式を使用する必要があります。また、その前には1つの空白スペース、文字'g'、または文字'T'を付ける必要があります。例: `t183000` 日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。
演算子	有効な演算子のリストについては、"Analytics 式の演算子" ページ 868を参照してください。
関連付けられたフィールド	関連付けられたフィールドで検索を行うには、フィールドの完全修飾名(つまり、テーブル名.フィールド名)を指定する必要があります。

クイック検索とクイックフィルター

クイック検索とクイックフィルターは、[フィルター]テキストボックスに検索式を作成することで簡単に検索を行えるようにする、Analytics の2つの機能です。

- ・ **クイック検索** -: [フィルター]テキストボックスにテキスト語を入力します。
- ・ **クイックフィルター** -: マウスを使って検索条件を選択します。

詳細については、次を参照してください。

- ・ "テーブルのクイック検索" ページ 1176
- ・ "ビューのデータでクイックフィルターを使用する" ページ 1171
- ・ "グローバルフィルター(ビューフィルター)" ページ 1213

メモ

クイック検索とクイックフィルターにあるいくつかの制限については、これらの機能についてのトピックを参照してください。

Analytics 関数を使って検索を行う

関数を使った検索により、最高レベルの検索能力と柔軟性が得られます。[フィルター]テキストボックスに検索式を入力するのは基本検索の場合と同じですが、式に関数を含める点が異なります。

下記の例では `FINDMULTI()` 関数を使用しています。この関数は、検索語のうち1つ以上を含むレコードをすべて分離します。



関数を使った検索の詳細については、"Analytics関数を使って検索とフィルタリングを行う" ページ 1235を参照してください。

一致する最初のレコードを選択する

メインメニューから操作できる、Analytics コマンドを使用して、テーブル内で検索条件を満たす最初のレコードを選択することができます。この機能は、特定のタスクを実行するのに他のコマンドと組み合わせた使用が可能である Analytics スクリプト内で主に役立ちます。

あるコマンドでは、特定のレコード番号に直接アクセスできます。これは、Analytics のユーザー インターフェイスで大型テーブルを操作する際に役立ちます。

詳細については、"一致する最初のレコードの選択" 次のページを参照してください。

一致する最初のレコードの選択

Analytics コマンドを使用して、テーブル内で検索条件に一致する最初のレコードを選択することができます。レコードは、選択されますが、Analytics の他のタイプの検索と違って分離されることはありません。他のレコードはテーブルビューに残っています。

スクリプトでの有用性

一致する最初のレコードを選択する機能は、主に Analytics スクリプトで役立ちます。たとえば、下記のコマンドは他のスクリプト作成テクニックと組み合わせることで、選択された各レコードに対して反復アクションを実行するプレカーソルとしてテーブル内のレコード間を順次移動するのに使用することができます。

検索]ダイアログボックス

Analytics ユーザー インターフェイスでは、**検索]ダイアログボックス(データ > 検索)** からコマンドにアクセスできます。



検索]ダイアログボックスの他のオプションを下 の表に示します。また、これらのオプションが主に Analytics スクリプトで役立つと考えて、同等の Analytics スクリプト コマンドも示します。

メモ

コマンドの詳細については、下記のコマンド名をクリックしてください。

検索]ダイアログボックスのオプション	同等の Analytics コマンド	説明
レコード検索	LOCATE RECORD	テーブル内の特定のレコード番号を選択します。
条件検索	LOCATE	あらゆる型のリテラル、任意のデータ型を使用した式、またはデータ型が混在した式の最初の出現を選択します。テーブルは、インデックス付けされている必要はありません。 例： <ul style="list-style-type: none"> ○ Vendor_City = "New York" ○ Invoice_Amount = 296.50 ○ Invoice_Date = `20141231` ○ Vendor_City = v_city ○ Vendor_City = v_city AND Invoice_Amount > 1000
文字検索	FIND	昇順でインデックス付けされている文字フィールドで、文字リテラル(たとえば、New York)の最初の出現を選択します。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>FIND コマンドと FIND() 関数は Analytics の 2 つの別個の機能であり、大きな違いがあります。</p> </div>
式検索	SEEK	昇順でインデックス付けされている文字フィールドで、文字リテラル(たとえば、"New York")または文字式(たとえば、v_city)の最初の出現を選択します。

インデックスの必要性

文字検索]または [式検索]オプションを使用するには、検索する文字フィールドにまず昇順のインデックスを作成する必要があります。どちらのオプションも、インデックス付きのフィールドでのみ検索を行います。

テーブルが2つ以上のフィールドでインデックス付けされている(入れ子のインデックス)場合は、主キーフィールドのみが検索されます。ただし、主キーフィールドは、昇順でインデックス付けされた文字フィールドであるとして、インデックスが条件付きである場合は、ビューから除外されたレコードは検索対象からも除外されます。

ガイドライン

データ型	文字フィールドではすべてのオプションを使用できます。日付時刻フィールドまたは数値フィールドでは、 条件検索]オプションのみを使用できます。
部分一致	文字フィールドを検索する場合には部分一致がサポートされますが、検索文字列はフィールドの先頭に現れる必要があります。 たとえば、Vendor_Name = "Uni" は "United Equipment" を見つけることができますが、Vendor_Name = "Equip" では見つけられません。

大文字と小文字の区別	すべてのオプションは、文字フィールドの検索に使用される場合、大文字と小文字を区別します。
パフォーマンス	条件検索 オプションはテーブルを順次検索するため、 文字検索 や 数値検索 のようなインデックス付きのテーブルを検索するオプションよりも処理速度が遅くなります。ただし、 条件検索 オプションには、テーブルのインデックス付けに費やされる時間はかかりません。
レコードの順序	条件検索 オプションは、テーブル内のレコードの元の順序を保持することになり、分析の性質によっては、このことが望ましい場合もあります。

テーブル内の特定のレコード番号を選択する

1. Analytics メインメニューから、**データ > 検索 > レコード検索**]を選択します。
2. **数値**テキスト ボックスにレコード番号を入力して **OK**]をクリックします。

レコード番号が見つかった場合、そのレコードが選択され、テーブルはそのレコードに位置付けられます。

任意の型のリテラルまたは式の最初のオカレンスを選択する

1. Analytics メインメニューから、**データ > 検索 > 条件検索**]を選択します。
2. 次のいずれかを実行します。

- **数値**]ボックスに式を入力して **OK**]をクリックします。
- **数値**]をクリックして**式ビルダー**を開き、式を作成して **OK**]をクリックし、もう一度 **OK**]をクリックします。

式は、要求に応じて簡単にも複雑にもなります。1つのフィールドまたは複数のフィールドを含むことができるほか、データ型を混在させることもできます。例：

- `Vendor_Name = "United Equipment"`
- `Invoice_Amount > 1000`
- `Vendor_Name = "United Equipment" AND Invoice_Amount > 1000 AND Invoice_Date > "20140930"`

文字リテラル値は二重引用符で、日付時刻値はバッククォートで囲む必要があります。

指定した値が見つかった場合、テーブルはそのレコードに位置付けられます。

指定した値が見つからなかった場合、テーブルは最初のレコードに位置付けられます。

インデックス付きテーブル内の文字リテラルの最初のオカレンスを選択する

1. 検索したいテーブルのインデックスをアクティブ化します。
テーブルには、検索したい文字フィールドでインデックスを作成しておく必要があります。
2. Analytics メインメニューから、**データ > 検索 > 文字検索**]を選択します。
3. **式**]テキスト ボックスに文字リテラル値を入力して **OK**]をクリックします。
引用符がフィールドのデータの一部でない限り、文字リテラル値を引用符で囲まないでください。
例：

-
-

指定した値が見つかった場合、テーブルはそのレコードに位置付けられます。

指定した値が見つからなかった場合は、"キーと一致するインデックスがありません" というメッセージが表示されます。テーブルは、指定した値よりも大きいキーフィールド値を持つ最初のレコードに位置付けられます。指定した値より大きい値がない場合は、テーブルの最初のレコードに位置付けられます。

インデックス付きテーブル内の文字リテラルまたは式の最初のオカレンスを選択する

1. 検索したいテーブルのインデックスをアクティブ化します。
テーブルには、検索したい文字フィールドでインデックスを作成しておく必要があります。
2. Analytics メインメニューから、**データ > 検索 > 式検索**]を選択します。
3. 次のいずれかを実行します。
 - **式**]テキスト ボックスに文字型の式または文字リテラル値を入力して、**OK**]をクリックします。
 - **式**]をクリックして**式ビルダー**を開き、式を作成して **OK**]をクリックし、もう一度 **OK**]をクリックします。

例：

- `v_vendor_name`
-

文字リテラル値は、二重引用符で囲む必要があります。

指定した値が見つかった場合、テーブルはそのレコードに位置付けられます。

指定した値が見つからなかった場合は、"キーと一致するインデックスがありません" というメッセージが表示されます。テーブルは、指定した値よりも大きいキーフィールド値を持つ最初のレコードに位置付けられます。指定した値より大きい値がない場合は、テーブルの最初のレコードに位置付けられます。

Analytics 検索コマンドの比較

以下に、Analytics 検索コマンド同士での主要機能の比較対照表を示します。Analytics スクリプトでコマンドを使用する場合、各コマンドを制御する詳細な規則が異なることを知っておくと役立ちます。

	レコード検索 / 条件検索	文字検索	式検索
検索可能なデータ型	<ul style="list-style-type: none"> 文字 日付時刻 数値 (レコード番号で検索することもできます)	文字	
検索場所	<ul style="list-style-type: none"> フィールド フィールド 	フィールド	
関連フィールドでの検索	はい (完全修飾フィールド名を指定する必要があります)	はい	
インデックスの必要性	いいえ	はい (昇順である必要があります)	
先頭スペースの検索	はい (データまたは検索文字列内のスペースは、文字として扱われます)	いいえ	はい (データまたは検索文字列内のスペースは、文字として扱われます)
大文字と小文字の区別	はい		
部分一致	はい (検索文字列は、フィールドの先頭に現れる必要があります。文字のみ)	はい (検索文字列は、フィールドの先頭に現れる必要がありません)	
検索語を囲む引用符の必要性	<ul style="list-style-type: none"> はい(文字) いいえ(数値) 任意(レコード番号) 逆引用符(日付時刻) 	いいえ(引用符がデータの一部でない限り、検索語を引用符で囲んではいけません)	はい
正確な文字比較を行うオプション	はい	いいえ	

	レコード検索 / 条件検索	文字検索	式検索
(SET EXACT ON/OFF)による影響			
式のサポート	はい	いいえ	はい
付記	検索]ダイアログボックスの レコード検索]および 条件検索]操作と、LOCATE RECORD/LOCATE コマンドは同等です。	検索]ダイアログボックスの 文字検索]操作とFIND コマンドは同等です。	検索]ダイアログボックスの 式検索]操作とSEEK コマンドは同等です。

Analytics関数を使って検索とフィルタリングを行う

Analytics 関数を使用して、テーブルのデータに対して、強力な効果的な検索とフィルタリングを実行することができます。

検索またはフィルタリング関数を使用するには、テーブルビューの上部にある [フィルター] テキスト ボックスでフィルターを作成します。フィルターには、下記の Analytics 関数の 1 つを使用します。

たとえば、次のフィルターでは FINDMULTI() 関数を使用しています。この関数は、検索語のうち 1 つ以上を含むレコードをすべて分離します。



関数を使用して検索またはフィルタリングを実行するためのガイドライン

フィールド名	<p>検索するフィールド名としては、テーブルビュー内の表示名でなく、テーブルレイアウト内の物理フィールド名を指定する必要があります。</p> <p>ヒント 物理フィールド名を確認するには、テーブルビュー内の列ヘッダーを右クリックし、プロパティを選択します。</p>
二重引用符	テキストの検索語は "二重引用符" で囲む必要があります。
バッククオート	日付時刻の検索語は `バッククオート` で囲む必要があります。
日付時刻書式	<ul style="list-style-type: none"> 日付時刻の検索語には YYYYMMDD または YYMMDD の書式を使用する必要があります。 時刻の部分には hhmmss 書式を使用する必要があります。また、その前には 1 つの空白スペース、文字 'z'、または文字 'T' を付ける必要があります。例: `t183000` 日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。
関連付けられたフィールド	関連付けられたフィールドで検索を行うには、フィールドの完全修飾名 (つまり、テーブル名.フィールド名) を指定する必要があります。
関数の規則	<p>どの関数にも、サポートされるデータ型や大文字と小文字の区別など、その動作を制御する特定の規則があります。</p> <p>Analytics の検索関数を制御する規則同士の比較対照表が掲載されていますので、"Analytics 検索関</p>

数の比較" ページ 1245を参照してください。各関数の詳細については、下記のリンク付き関数名をクリックしてください。

検索のタイプ

関数を使用して、テキスト、数値、日付時刻データを検索またはフィルタリングできます。ただし、検索またはフィルタリングしているデータ型に合った正しい関数を使用する必要があります。

- **関数でサポートされるデータ型** - 関数は特定のデータ型で機能するように設計されています。また、場合によっては、複数のデータ型に対応することもあります。

たとえば、ISBLANK() 関数はテキスト データ(文字データ)で使用できますが、数値または日付時刻データでは使用できません。MATCH() または BETWEEN() 関数は、文字、数値、日付時刻データで使用できます。

- **データのデータ型** - 検索またはフィルタリングしているデータのデータ型を認識し、データ型に合った関数を使用する必要があります。一般的に、数字と日付のデータ型は数値または日付時刻です。ただし、文字データ型を使用する場合もあります。

メモ

各関数の詳細については、下記の関数名をクリックしてください。

ヒント

下記の例をコピーして [フィルター] テキスト ボックスに貼り付け、お使いのデータに合わせて検索語とその他の入力値を修正することができます。

テキスト検索(文字データ型)

単一の検索語を検索する

使用方法: "FIND() 関数" ページ 2267

説明: 制限が最も少ない検索関数です。大文字と小文字を区別しません。個々のフィールドまたは複数のフィールドに加え、レコード全体を検索することができます。

例	結果
FIND("United Equipment")	レコード内のどこかに"United Equipment" という名前が入っているすべてのレコードを分離します。
FIND("Equip")	レコード内のどこかに"Equip" という文字列が入っているすべてのレコードを分離します。

例	結果
<code>FIND("United Equipment", Vendor_Name)</code>	Vendor_Name フィールドに "United Equipment" という名前が入っているすべてのレコードを分離します。
<code>FIND("United Equipment", Vendor.Vendor_Name)</code>	関連する Vendor テーブルの Vendor_Name フィールドに "United Equipment" という名前が含まれるすべてのレコードを分離します。

空白のテキスト値を検索する

使用方法: "ISBLANK() 関数" ページ 2315

説明: 文字フィールドで空白値を検索することができます。

例	結果
<code>ISBLANK(First_Name)</code>	First_Name が空白であるすべてのレコードを分離します。

複数のテキスト語を検索する

使用方法: "FINDMULTI() 関数" ページ 2272

説明: FIND() と似ていますが、複数の検索語を指定できる点が異なります。

例	結果
<code>FINDMULTI(RECORD, "United Equipment", "Muller Corp.")</code>	レコードのどこかに "United Equipment" または "Muller Corp" という名前が入っているすべてのレコードを分離します。
<code>FINDMULTI(RECORD, "equip", "supp")</code>	"equip" または "supp" という文字列が含まれるすべてのレコードを分離します。
<code>FINDMULTI(RECORD, "United Equipment", "Muller Corp.")</code>	レコードのどこかに "United Equipment" または "Muller Corp" という名前が入っているすべてのレコードを分離します。
<code>FINDMULTI(RECORD, "United Equipment", "Muller Corp.")</code>	レコードのどこかに "United Equipment" または "Muller Corp" という名前が入っているすべてのレコードを分離します。

使用方法: "MATCH() 関数" ページ 2352

説明: 多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH(Vendor_City, "Phoenix", "Austin", "Los Angeles")</code>	Vendor_City フィールド内の値が "Phoenix"、"Austin"、または "Los Angeles" と正確に一致するか、またはこれらの文字列で始まるすべてのレコードを分離します。
<code>NOT MATCH(Vendor_City, "Phoenix", "Austin", "Los Angeles")</code>	Vendor_City フィールド内の値が "Phoenix"、"Austin"、または "Los Angeles" と正確に一致しないか、またはこれらの文字列で始まらないすべてのレコードを分離します。
<code>MATCH(Product_Code, "A", "D", "F")</code>	Product_Code フィールド内の製品コードが "A"、"D"、または "F" であるか、もしくは "A"、"D"、"F" で始まるすべてのレコードを分離します。
<code>MATCH(Product_Code, "A", "D", "F")</code>	Product_Code フィールド内の製品コードが 1 文字の "A"、"D"、または "F" であるすべてのレコードを分離します。 正確な文字比較を行う] オプションをオンにする必要があります。
メモ	
MATCH() に関する例では、特に指定のない限り、 正確な文字比較を行う] オプション はオフになっているものとします。	

大文字と小文字の区別があるテキスト語を検索する

使用方法: "MATCH() 関数" ページ 2352

説明: 多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH>Last_Name, "SMITH")</code>	Last_Name フィールドの値が "SMITH" (すべて大文字) になっているすべてのレコードを分離します。

例	結果
<code>MATCH>Last_Name, "smith")</code>	Last_Name フィールドの値が"smith"(すべて小文字)になっているすべてのレコードを分離します。
<code>MATCH>Last_Name, "Smith")</code>	Last_Name フィールドの値が"Smith"(単語の先頭文字が大文字)になっているすべてのレコードを分離します。

複数のフィールドでテキスト語を検索する

使用方法: "MATCH() 関数" ページ 2352

説明: 多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH("Phoenix", Vendor_City, City, City_2)</code>	Vendor_City、City、または City_2 フィールドの値のうち少なくとも1つが"Phoenix"と正確に一致するか、またはこの文字列で始まるすべてのレコードを分離します。

複数のテキスト語に一致させる検索を行う

使用方法: "MATCH() 関数" ページ 2352

説明: 多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH(Vendor_Address, Employee_Address)</code>	業者と従業員の住所がまったく同じであるすべてのレコードを分離します。 業者と従業員の住所の形式を統一する必要がある場合は、その他の関数を使用します。

特定の文字または部分文字列の1つまたは複数の出現を検索する

使用方法: "OCCURS() 関数" ページ 2385

説明: 文字フィールドにおける、部分文字列の1つまたは複数の出現を検索することができます。

例	結果
<code>OCCURS(Invoice_Number, "-") > 1</code>	請求書番号にハイフンが2つ以上入っているすべてのレコードを分離します。
<code>OCCURS(Full_Name, ALLTRIM(Last_Name))=1</code>	Last_Name フィールドの値が Full_Name フィールドに現れるすべてのレコードを分離します。 式に ALLTRIM() 関数を含めると、Last_Name フィールドの先頭と末尾のスペースがすべて取り除かれ、テキスト値のみが比較されるようになります。
<code>OCCURS(Vendor_Name, "UNITED EQUIPMENT") > 0</code>	Vendor_Name フィールドに "UNITED EQUIPMENT" という名前が大文字で入っているすべてのレコードを分離します。 FIND() 関数とは異なり、OCCURS() 関数は大文字と小文字を区別します。

特定のバイト位置で始まる部分文字列を検索します。

使用方法: "AT() 関数" ページ 2167

説明: 文字フィールドで、続いて現れる部分文字列を検索して、目的の部分文字列の開始バイト位置を特定することができます。

例	結果
<code>AT(2, "-", Invoice_Number) > 10</code>	請求書番号に2つ以上のハイフンが入っており、2つ目のハイフンが、文字列内の10番目の文字より後に現れるすべてのレコードを分離します。

範囲のテキストの検索

使用方法: "BETWEEN() 関数" ページ 2170

説明: 範囲内に入っているテキスト値を検索することができます。

例	結果
<code>BETWEEN(Last_Name, "C", "K")</code>	Last_Name フィールドの値が、"C" から "K" まで(これらの値も含む)の文字で始まるすべてのレコードを分離します。 正確な文字比較を行う] オプションをオフにする必要があります。

ほぼ同じテキスト値(あいまい重複)を検索する

使用方法: "ISFUZZYDUP() 関数" ページ 2319

説明: 同じ値だけでなく、ほぼ同じ値(あいまい重複)を検索することができます。大文字と小文字を区別しません。

使用方法: "LEVDIST() 関数" ページ 2335

説明: ISFUZZYDUP() と似ていますが、デフォルトで大文字と小文字が区別される点が異なります。

例	結果
<pre>ISFUZZYDUP(Last_Name, "Braun", 2)</pre>	<p>Last_Name フィールドに "Braun" という名前か、または "Braun" のあいまい重複が入っているすべてのレコードを分離します。</p> <p>この例では、レーベンシュタイン距離(あいまいさの度合い)を 2 に設定していますが、この値は増減できます。</p>
<pre>LEVDIST(TRIM(Last_Name), "Braun") < 3</pre>	<p>Last_Name フィールドに "Braun" という名前か、または "Braun" のあいまい重複が入っているすべてのレコードを分離します。</p> <p>この例では、レーベンシュタイン距離(あいまいさの度合い)を < 3 に設定していますが、この値は増減できます。</p> <p>式に TRIM() 関数を含めると、Last_Name フィールドから先頭のスペースがすべて取り除かれ、テキスト値のみが比較されるようになります。</p>

基本的なパターンを検索する

使用方法: "MAP() 関数" ページ 2345

説明: ワイルドカード文字、リテラル文字、またはその両方の組み合わせを使って検索を行うことができます。

例	結果
<pre>MAP(Invoice_Number, "XX99999")</pre>	<p>請求書番号が、2 つの文字の後に 5 つの数字が続く構成であるか、またはこの構成で始まるすべてのレコードを分離します。</p>
<pre>MAP(Invoice_Number, "AB12345")</pre>	<p>請求書番号が "AB12345" であるか、または "AB12345" で始まるすべてのレコードを分離します。</p>
<pre>MAP(Invoice_Number, "AB99999")</pre>	<p>請求書番号が、"AB" の後に 5 つの数字が続く構成であるか、またはこの構成で始まるすべてのレコードを分離します。</p>
<pre>NOT MAP(SSN, "999-99-9999")</pre>	<p>SSN フィールドの値が社会保障番号の標準形式と一致しないすべてのレコードを分離します。</p>

複雑なパターンを検索する

使用方法: "REGEXFIND() 関数" ページ 2447

説明:最も強力な柔軟性のある検索関数です。リテラル文字とメタ文字を組み合わせた正規表現を使用して検索することができます。他の検索関数を使用するより複雑になることがあります。

例	結果
<code>REGEXFIND(Vendor_City, "Phoenix Austin Los Angeles")</code>	Vendor_City フィールドの値に "Phoenix"、"Austin"、または "Los Angeles" を含んでいるすべてのレコードを分離します。
<code>REGEXFIND(Product_Code, "\b\d{3}-[a-zA-Z]{6}\b")</code>	製品コードが、3つの数字で始まり、その後1つのハイフンと6つの文字が続くすべてのレコードを分離します。
<code>REGEXFIND(Product_Code, "\b\d{3,}-[a-zA-Z]{6}")</code>	製品コードが、3つ以上の数字で始まり、その後1つのハイフンと6つ以上の文字が続くすべてのレコードを分離します。

数値検索

数値を検索する

使用方法: "MATCH() 関数" ページ 2352

説明:多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH(Invoice_Amount, 154.00)</code>	\$154.00 の請求金額が含まれるすべてのレコードを分離します。
<code>MATCH(Invoice_Amount, 154.00, 522.00)</code>	\$154.00 または \$522.00 の請求金額が含まれるすべてのレコードを分離します。
<code>NOT MATCH(Inventory_Value_at_Cost, Cost_x_Quantity)</code>	Inventory_Value_at_Cost フィールドの金額と演算フィールド Cost_x_Quantity の金額が異なるすべてのレコードを分離します。

範囲の数字の検索

使用方法: "BETWEEN() 関数" ページ 2170

説明:範囲内に入っている数値を検索することができます。

例	結果
<code>BETWEEN(Invoice_Amount, 1000, 5000)</code>	請求金額が1000ドルから5000ドルまで(これらの値も含む)であるすべてのレコードを分離します。

テーブル全体からの数字の検索

使用方法: "FIND() 関数" ページ 2267

説明: 個々のフィールドまたは複数のフィールドに加え、レコード全体を検索することができます。

使用方法: "FINDMULTI() 関数" ページ 2272

説明: FIND() と似ていますが、複数の検索語を指定できる点が異なります。

メモ

FIND() や FINDMULTI() 関数を使って数値や日付時刻値を検索する際には、注意が必要です。これらの関数はソースデータファイル(.fil)内の厳密に一致する文字を検索しますが、このような文字はテーブルビューでは様々な外観で表示されます。

検索結果に一貫性がないように思われる場合は、**テーブルレイアウト** ダイアログボックスでソースデータを調べてください。

例	結果
<code>FIND("154.00")</code>	ソースデータファイルから、 154.00 という文字がこのまま含まれるすべてのレコードを分離します。

日付時刻検索

日付時刻値を検索する

使用方法: "MATCH() 関数" ページ 2352

説明: 多目的に使える検索関数です。1つのフィールドで同時に複数の検索語を探したり、複数のフィールドで同じ検索語を探したりすることができます。また、2つのフィールドで一致する値を検索することもできます。

例	結果
<code>MATCH(Invoice_Date, `20170731`)</code>	請求日が2017年7月31日であるすべてのレコードを分離します。
<code>MATCH(Invoice_Date, `20170731`, `20170831`, `20170930`)</code>	第3四半期の各月末の日付が入った請求書が含まれるすべての

例	結果
	レコードを分離します。

空白または無効な日付値の検索

使用方法: "VERIFY() 関数" ページ 2562

説明: 日付フィールドで空白値または無効な値を検索することができます。

例	結果
<pre>NOT VERIFY(Invoice_Date)</pre>	Invoice_Date フィールドの日付が空白か無効であるすべてのレコードを分離します。

範囲の日付時刻値の検索

使用方法: "BETWEEN() 関数" ページ 2170

説明: 範囲内に入っている日付時刻値を検索することができます。

例	結果
<pre>BETWEEN(Invoice_Date, `20140930`, `20141030`)</pre>	請求日が2014年9月30日から2014年10月30日まで(これらの日付も含む)であるすべてのレコードを分離します。
<pre>NOT BETWEEN(Invoice_Date, `20140930`, `20141030`)</pre>	請求日が2014年9月30日から2014年10月30日まで(これらの日付も含む)の範囲内でないすべてのレコードを分離します。

テーブル全体からの日付時刻値の検索

使用方法: "FIND() 関数" ページ 2267

説明: 個々のフィールドまたは複数のフィールドに加え、レコード全体を検索することができます。

使用方法: "FINDMULTI() 関数" ページ 2272

説明: FIND() と似ていますが、複数の検索語を指定できる点が異なります。

メモ

FIND() や FINDMULTI() 関数を使って日付時刻値を検索する際には、注意が必要です。これらの関数はソースデータファイル(.fil)内の厳密に一致する文字を検索しますが、このような文字はテーブルビューでは様々な外観で表示されます。

検索結果に一貫性がないように思われる場合は、**テーブルレイアウト**ダイアログボックスでソースデータを調べてください。

例	結果
<pre>FINDMULTI(RECORD, "31/07/2017", "31/08/2017")</pre>	<p>ソースデータファイルから、31/07/2017 または 31/08/2017 という文字がそのまま含まれるすべてのレコードを分離します。</p> <p>日付時刻書式(YYYYMMDD、YYMMDD、hhmmss、hhmm)に関する一般的な制限は、FIND() や FINDMULTI() を使って日付時刻値を検索する際には適用されません。</p>

Analytics 検索関数の比較

次の表に、Analytics 検索関数同士での主要機能の比較対照表を示します。各関数を制御する特定の規則がどのような点で異なっているかを知ると、Analytics 内で検索表現を作成する際に役立ちます。

検索時のデータ型

詳しく表示

サポートされているデータ型	関数
文字	AT() FIND() FINDMULTI() ISFUZZYDUP() LEVDIST() MAP() OCCURS() REGEXFIND()
文字 日付時刻 数値	BETWEEN() MATCH()

検索する場所(フィールド、レコード)

詳しく表示

サポートされている検索場所	関数
単一のフィールド	BETWEEN() ISFUZZYDUP() LEVDIST()
1つ以上のフィールド	AT() MAP() MATCH() OCCURS() REGEXFIND()
1つ以上のフィールド レコード	FIND() FINDMULTI()

先頭スペースの検索

詳しく表示

先頭スペースの検索	関数
はい データ内の先頭スペースは、状況に応じて、検索文字列と一致させることができます	AT() BETWEEN() FIND() FINDMULTI() OCCURS()
はい データ内の先頭スペースは、検索文字列内のスペースと正確に一致している必要があります	MAP() MATCH()
はい データまたは検索文字列内のスペースは、文字として扱われます	ISFUZZYDUP() LEVDIST() REGEXFIND()

大文字と小文字の区別

詳しく表示

この関数は大文字と小文字が区別されます。	関数
はい	AT() BETWEEN() MAP() (リテラル文字) MATCH() OCCURS() REGEXFIND()
いいえ	FIND() FINDMULTI() ISFUZZYDUP() MAP() (ワイルドカード文字)
省略可能	LEVDIST()

部分一致

詳しく表示

部分一致がサポートされる場合とは	関数
はい 検索文字列は、フィールド内のどこに現れてもかまいません	AT() FIND() FINDMULTI() OCCURS() REGEXFIND()
はい 検索文字列は、フィールドの先頭に現れる必要があります。 文字データ型のみ	BETWEEN() MATCH()
はい 検索文字列は、データ値と同じ長さであるか、または短くなければなりません	MAP()
はい	ISFUZZYDUP() LEVDIST()

複数の検索語

詳しく表示

複数の検索語のサポート	関数
はい	FINDMULTI() MATCH() REGEXFIND()
いいえ	AT() BETWEEN() FIND() ISFUZZYDUP() LEVDIST() MAP() OCCURS()

正確な文字比較を行う]オプション(SET EXACT ON/OFF) による影響

詳しく表示

正確な文字比較を行う]オプションによる影響 (SET EXACT ON/OFF)	関数
はい	BETWEEN() MATCH()
いいえ	AT() FIND() FINDMULTI() ISFUZZYDUP() LEVDIST() MAP() OCCURS() REGEXFIND()

順番検査

シーケンシャル順の検査([順番検査] オプション)を使用すると、データが既にソート済みであるか、またはインデックス付けされているかどうかを検証できます。また、特定の分析テストやデータの結合操作を実行する前に、並べ替えやインデックスの作成が必要かどうかを検証できます。

Analytics のいくつかのテストや操作では、結果を有効なものにするため、また操作を正常に実行するために、データが順番どおりに並んでいることを要求します。不必要にテーブルをソートしたりインデックス付けしたりする代わりに、それらの操作が必要かどうかを調べるために、最初に検査することができます。特に大きなテーブルでは、並べ替えにかなりの時間とシステムリソースを要することがあるため、最初に検査することは時間の節約になります。

文字、数値、日付時刻、または演算フィールドの順番を検査できます。また、データが2つ以上のフィールドによって並べ替えられたり、インデックス付けされたりしている場合は、複数のフィールドやデータ型を組み合わせで検査することができます。

メモ

順番に並べられたデータは、データにギャップがないということではありません。たとえば、数値系列 1、3、5 は順番に並べられています。ギャップのテストは別の操作です。詳細については、"ギャップの検査" ページ 1256を参照してください。

レコードを並べ替えない順番検査

順番検査は、レコードを順番どおりに並べたり、並べ替えたりしません。いかなる方法でも、検査するテーブル内のレコードの並び順を変えることはありません。テーブル内の指定されたフィールドが、現在順番どおりに並んでいるかどうかを検証し、順番の誤りをすべて報告します。レコードを並べ替えるには並べ替え操作を、レコードを順番どおりに並べるにはインデックスの作成操作を別々に実行する必要があります。

順番が正しくない項目の検査

また、順番検査では、順番どおりに並んでいることが要求される請求書番号や小切手番号などのデータ内で、順序が正しくない項目を識別することができます。これは不正行為の可能性を示しています。たとえば、特定の業者の請求書データを請求日順に並べ替え、その後、請求書番号の順番を検査することができます。順序が正しくない請求書番号は、詳細な調査を必要とするかもしれません。

文字データの検査の並べ順

文字フィールドの順番は、[並べ替え順] オプション([ソールオプションテーブル]) で文字データに指定されている並べ替え順に対して検査されます。設定を変更していない限り、通常は、デフォルトの並べ替え

順が指定されます(0,1,2... A,B,C...)。非 Unicode 版の Analytics には、並べ替えにいくつかの微妙な違いがあります。たとえば、デフォルトの並べ替え順は、大文字のすべての英文字が小文字のすべての英文字より前に並べ替えられるように指示します。

順番検査エラーが報告される方法

順番検査は、列の最初の値を2番目の値と比較し、2番目の値を3番目の値と、というように、列を下へ進みながらペアとなる値を比較していきます。値のペアが順番を途切れさせると、順番検査エラーが報告されます。

途切れた後、あらためて順番が開始され、2番目のペアとなる値を新しい開始点として使用します。この中断点の後にある値を中断点の前の値と比較した場合に、順序どおりになっていなくても、それらは報告されません。たとえば、次のような値の列を昇順として検査した場合、Analytics は2個の順番検査エラー(4、1)を報告します。5個(4、4、5、1、2)ではありません。

```

1
  3
  6
  4   順番検査エラー
  4
  5
  6
  9
  1   順番検査エラー
  2

```

入れ子になっている並べ替え順の検査

2つ以上のフィールドを組み合わせて順番を検査する場合に、有効な結果を得るためには、検査対象とするフィールドを選択するとき、並べ替え順やインデックス順の優先順位と同じ順序で主キーフィールド、副キーフィールドというように選択する必要があります。また、各フィールドの並び順の方向、つまり昇順または降順も一致している必要があります。

入れ子の並べ替え順を検査した場合の有効な結果と無効な結果

次の例では、検査順序がネストされた並べ順と、日付および時刻キーフィールドの値の方向と一致するかどうかに基づいて、有効な結果と無効な結果を示しています。

日付 (主キーフィールドの昇順)	金額 (ネストされた副キーフィールドの降順)
15 Jan 2011	\$2300.00
15 Jan 2011	\$1200.00
15 Jan 2011	\$600.00
16 Jan 2011	\$900.00
16 Jan 2011	\$100.00
17 Jan 2011	\$4700.00
17 Jan 2011	\$900.00
17 Jan 2011	\$500.00

有効な結果

順番検査エラー 0 を返します。

```
SEQUENCE ON Date Amount D
```

検査するフィールドと同じ優先順位および方向を使用しています。

無効な結果

順番検査エラー 2 を返します。

```
SEQUENCE ON Amount D Date
```

レコード 番号	Amount	Date
4	900	2011/01/16
6	4,700	2011/01/17

検査するフィールドと異なる優先順位を使用しています。**Amount** フィールドをネストされていないフィールドとして扱っています。

無効な結果

順番検査エラー 5 を返します。

SEQUENCE ON Date Amount

レコード番号	Date	Amount
2	2011/01/15	1,200
3	2011/01/15	600
5	2011/01/16	100
7	2011/01/17	900
8	2011/01/17	500

検査するフィールドのうちの1つが異なる方向を使用しています。Amount フィールドが昇順にソートされているものとして扱っています

手順

順番検査オプションを使用すると、アクティブなテーブル内の1つまたは複数のフィールドが順番どおりに並んでいるかどうかを確認したり、順序が正しくない項目を識別したりすることができます。

メモ

現在アクティブなテーブルにクイックソートが適用されていないことを確認してください。順番検査で有効な結果を提供するためには、元となる Analytics テーブルの実際の物理的な順序でビューが表示されている必要があります。

方法の説明

1. **分析 > 順番検査**]の順に選択します。
2. **メイン**]タブで、次のいずれかを行います。
 - **順番検査の対象**]リストから、検査の対象となるフィールドを選択します。
 - **順番検査の対象**]をクリックして、フィールドを選択するか、または式を作成します。

複数のフィールドを選択する場合、フィールドを選択する順序が検査する優先度を示します。最初に選択したフィールドについてレコードが検査されます。その最初のフィールドに、同じ値が連続して複数存在している場合は、そのグループ内のレコードが2番目に選択したフィールドで検査される、というようになります。複数のフィールドを選択していない場合は、グループ内のレコードは、どの二次検査の対象にもなりません。

メモ

複数のフィールドで並べ替えまたはインデックス付けされている(入れ子の並べ替えまたはインデックス付け)テーブルを検査している場合、結果を有効なものにするには、検査する優先順位が、並べ替えやインデックスの優先順位(主キーフィールド、副キーフィールドなど)と一致している必要があります。

フィールドを選択する順序が、結果に表示する列の順序になります。

3. **順番検査の対象]**をクリックした場合は、任意で並べ替えの矢印  (デフォルトは昇順) をクリックすることにより、1 つまたは複数の選択したフィールドを降順にするよう指定することができます。

メモ

以前に並べ替えまたはインデックス付けされているフィールドを検査する場合、結果を有効なものにするには、指定する並べ替え順の方向、つまり昇順または降順が、検査するフィールドの方向と一致している必要があります。

4. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

5. **出力]**タブをクリックします。
6. **出力先]**パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果をテキストファイルに保存または追加するには、このオプションを選択します。ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

7. 出力の種類として **[ファイル]**を選択した場合は、**出力指定]**パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは、使用している Analytics のエディションに応じて、"**ASCII テキストファイル**" または "**Unicode テキストファイル**" のいずれかのみです。結果を新しいテキストファイルに保存するか、または既存のテキストファイルに追加します。
 - **名前** - **名前]**テキストボックスにファイル名を入力します。もしくは、**名前]**ボタンをクリックして、**保存]**または **[ファイル]を保存する名前]**ダイアログボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル

名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、`C:\Results\Output.txt` または `Results\Output.txt` のように指定します。

- ローカル- 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。
8. 出力の種類によっては、任意で **[ヘッダー]** および **[フッター]** をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキスト やフッターテキスト を左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**[ヘッダー]** または **[フッター]** ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミicolon (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

9. **詳細]** タブをクリックします。
10. **適用範囲]** パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダー を利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>

メモ

"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。

ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。

11. **[エラー制限]**テキスト ボックスで、列挙する順序どおりでない項目の最大数を指定するか、またはデフォルトの10のままにしておきます。

制限に達すると、Analytics によって処理が停止され、その時点までの順序どおりでない項目が出力されます。**[エラー制限]**の数は、検査されるすべてのフィールドでのエラーの合計数に適用されます。フィールドごとの制限ではありません。

12. 出力の種類として **[ファイル]**を選択した場合、出力結果を既存のテキスト ファイルの末尾に追加するときは、**[既存のファイルに追加する]**を選択します。
13. **[OK]**をクリックします。
14. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

ギャップの検査

順番に並んでいる数値フィールドや日付時刻フィールドにおける欠落は、データファイルが不完全であることを示す可能性があります。フィールドの順番に並んでいる値にギャップがないかを調べ、1つまたは複数のギャップや欠落項目がある場合は、それを識別することができます。

有効な結果を得るために、検査されるフィールドは、事前に順番どおりに整列されている必要があります。フィールドの並べ替えは、前もって実行しておくか、ギャップの検査時に **あらかじめ並べ替える**]オプションを使用して行うことができます。

数値フィールドや日付時刻フィールド、あるいは文字フィールド内の数字を検査することができます。一度に1つのフィールドしか検査できません。

ギャップの幅の列挙または欠落している項目の列挙

ギャップの検査結果を出力する際のオプションは2つあります。

- ギャップ幅の列挙
- 欠落項目の列挙

ギャップ幅の列挙

ギャップの開始点と終了点、およびギャップ内の欠落項目の総数を割り出します。

ギャップの開始点と終了点を特定する値は、その値自体は欠落項目ではありません。開始点と終了点は連続した値で、ギャップの直前および直後に現れる値です。これらには、「**ギャップ開始(除く)**」と「**ギャップ終了(除く)**」というラベルが付けられます。たとえば、小切手番号 12345 と 12350 がある場合、それら2つの番号の間に4つの欠落項目を含むギャップが割り出されます。

欠落項目の列挙

ギャップ内の個々の欠落項目を一覧表示します。たとえば、小切手番号 12346、12347、12348、および 12349 となります。

これらの欠落した項目は計算された値であり、実際は検査するデータに現れません。このオプションを使用する場合は、**欠落項目の最大数**]フィールドで、各ギャップについて個別に列挙する欠落項目の最大数を指定できます。これは、ギャップが大きい場合に有用です。最大数を超えると、Analytics は代わりに範囲の方法を用いてギャップを特定します。1つの差異について、ギャップの開始点と終了点を識別する値が、連続する欠落項目の最初と最後になります。これらには、「**ギャップ開始(含める)**」と「**ギャップ終了(含める)**」というラベルが付けられます。

メモ

欠落項目の最大数]に指定した数は、ギャップ単位で適用されます。これは、データセット全体での結果となる欠落項目の総数を制限するものではありません。個別に列挙されるか、それともグループで列挙されるかどうかを制限します。

欠落項目の方法を使用する場合は、**欠落項目の最大数**]フィールドの値とそれぞれ異なるギャップのサイズに応じて、結果には個々の欠落項目と範囲が混在して含まれます。

数値データ内のギャップ検査

数値データのギャップを検査する場合、次のようにデータの小数点以下の桁数によってデータ内の許容間隔が制御されます。

- **数値データに整数のみ(小数部なし)が含まれる場合**、-許容間隔は1
です。1より大きい間隔が1つのギャップです。範囲として報告されるギャップの場合、欠落項目の数は欠落している整数の数です。
- **数値データには小数点が含まれます**-許容される間隔は最小の小数点間隔と等しくなります
たとえば、数値フィールドが小数点2桁の場合、許容される間隔は0.01です。最も小さい小数間隔よりも大きい間隔が1つのギャップです。範囲として報告されるギャップの場合、欠落項目の数は欠落している小数間隔の数です。

数値データ内のギャップ検査の例

最初の例では、数値データに整数のみが含まれます。許容される間隔は1です。

検査値	欠落項目	欠落項目の数
-2	2	1(整数)
-1	3	1(整数)
0	6 ~ 14 (6と14も含む)	9(整数)
1		
4		
5		
15		

2番目の例では、数値データに小数点2桁が含まれます。許容される間隔は0.01です。

検査値	欠落項目	欠落項目の数
4.24	4.27	1(0.01間隔)

検査値	欠落項目	欠落項目の数
4.25	4.28	1(0.01 間隔)
4.26	4.31 ~ 4.99(4.31 と4.99 も含む)	69(0.01 間隔)
4.29		
4.30		
5.00		

日付時刻データ内のギャップ検査

日付、日付時刻または時刻データでギャップを検査することができます。

- 日付フィールドにおける許容間隔は1日です**
 1日以上の間隔が1つのギャップです。範囲として報告されるギャップの場合、欠落項目の数は欠落している日数です。
- 日付時刻または時刻フィールドにおける許容間隔は1秒です**
 1秒以上の間隔が1つのギャップです。範囲として報告されるギャップの場合、欠落項目の数は欠落している秒数です。したがって、1時間のギャップは3,600個の欠落項目がある範囲として報告され、1日のギャップは86,400個の欠落項目がある範囲として報告されます。

日付および日付時刻データ内のギャップ検査の例

最初の例では、データに日付のみが含まれます。許容間隔は1日です。

検査値	欠落項目	欠落項目の数
2014/12/27	2014/12/29	1(日)
2014/12/28	2014/12/30	1(日)
2014/12/31	2015/01/03 ~ 2015/01/11	9(日)
2015/01/01		
2015/01/02		
2015/01/12		
2015/01/13		

2番目の例では、データに日付時刻が含まれます。許容間隔は1秒です。

検査値	欠落項目	欠落項目の数
2014/12/31 23:59:54	2014/12/31 23:59:56	1(秒)
2014/12/31 23:59:55	2014/12/31 23:59:57	1(秒)
2014/12/31 23:59:58	2015/01/01 00:00:00 ~ 2015/01/01 00:59:59	3,600(秒)
2014/12/31 23:59:59	2015/01/01 01:00:02 ~ 2015/01/02 01:00:01	86,400(秒)
2015/01/01 01:00:00		
2015/01/01 01:00:01		
2015/01/02 01:00:02		

文字フィールド内の数値データのギャップ検査

文字フィールドに存在する数値データ(通常は文字データとして書式設定される小切手番号など)に対してギャップ検査を行うことができます。

文字フィールドに文字と数字が続けて現れる場合は、数字のみが検査され、文字は無視されます。

文字フィールド内の数字のギャップ検査の例

英字のプレフィックスは無視されます。数値のみが考慮されます。

検査値	欠落項目	欠落項目の数
A123 C124		0(文字数字)
A123 B125	124	1(文字数字)

文字フィールドの並べ替えがギャップの検査に影響することがある

文字フィールドの値の文字と数字の配列によっては、ギャップ検査の結果間に矛盾が存在する可能性があります。たとえば、一部の数字は文字の前にあり、一部は前でない場合、あるいは Analytics の非 Unicode 版で、前にある一部の文字は小文字で、一部は大文字である場合は、結果が正確でないかもしれません。

不正確な結果になるのは、アルファベット文字の存在が一貫していない、あるいはアルファベット文字の大文字/小文字の表記が統一されていないため、**あらかじめ並べ替える**] オプションで適切な順序で並

べ替えられないことが原因です。次の表の [欠落項目] 欄で示す 126 と 127、および 124 は実際には欠落している項目ではありませんが、検査値が英数文字として並べ替えられているため、欠落項目として挙げられています。

検査値に矛盾があるのではないかと疑わしい場合は、そのフィールドで個別に並べ替えて、ギャップが検査される文字フィールドの値の順序を明らかにします。連続する数値の順序が文字の存在によって不適切になる場合に、有効な結果を確実に得るには、ギャップを検査する前に、INCLUDE() などの Analytics 関数を使用して文字を取り除いておくことができます。

ギャップ検査の不正な結果の例

一貫性がない英字の存在、または一貫性がない英字の大文字と小文字により、項目が欠落していると誤って報告されます。

検査値	欠落項目	欠落項目の数
123	126	1(文字数字)
124	127	1(文字数字)
125		
128		
129		
A-126		
A-127		
A-123	124	1(文字数字)
a-124		
A-125		
A-128		
A-129		
A-126		
A-127		

手順

順番に並んでいる数値や日付時刻値に欠落(ギャップ)がないかどうかを検出するために、アクティブなテーブルについて一度に1つのフィールドを検査することができます。

方法の説明

1. **分析] > ギャップ]** を選択します。
2. **メイン]** タブで、次のいずれかを行います。
 - **ギャップの対象]** リストから、検査の対象となるフィールドを選択します。

- **ギャップの対象**]をクリックして、フィールドを選択するか、または式を作成します。一度に1つのフィールドしか検査できません。
3. **ギャップの対象**]をクリックした場合は、任意で並べ替えの矢印  (デフォルトは昇順) をクリックすることにより、選択したフィールドの出力結果を降順にするよう指定することができます。
 4. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用してIF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

5. 対象フィールドが前の操作で既にソート済みであれば、任意で **あらかじめ並べ替える**]をオフにすることにより、大きなテーブルの欠落を検査する場合の時間を節約できます。フィールド内のデータがソートされていない場合は、すべての欠落が確実に検出されるようにするために、**あらかじめ並べ替える**]はオンのままにしておいてください。

メモ

あらかじめ並べ替える]をオフにする場合、結果を有効なものにするには、ギャップ検査の対象として選択するフィールドが事前にソートされている必要があります。ソートされていないフィールドに対してギャップ検査を行うと、結果に**"警告:ファイルの順序が正しくありません"**というメッセージが提示されます。結果を Analytics テーブルに出力する場合、警告メッセージはコマンド ログに記載されます。

6. 次のいずれかを実行します。
 - ギャップの開始点と終了点、およびギャップ内の個々の欠落項目の総数を割り出すには、**ギャップの幅を列挙する**]を選択します。
 - ギャップ内の個々の欠落項目を一覧表示するには、**欠落している項目を列挙する**]を選択します。**欠落項目の最大数**]テキスト ボックスで、各ギャップについて個別に列挙する欠落項目の最大数を指定するか、またはデフォルトの5のままにしておきます。
7. **出力**]タブをクリックします。
8. **出力先**]パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果を Analytics テーブルまたはテキスト ファイルに保存または追加するには、このオプションを選択します。Analytics テーブルに保存または追加する場合、そのテーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。テキスト ファイルに保存または追加する場合、ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

9. 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - 結果を新しい **Analytics テーブル**に保存する、または既存の Analytics テーブルに追加する場合は、"Analytics テーブル"を選択します。結果をテキスト ファイルに保存または追加する場合は、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**を選択します。
 - **名前** - **名前**]テキスト ボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- **ローカル** - サーバー テーブルに接続されており、Analytics テーブルに結果を保存または追加する場合にのみ有効になります。プロジェクトと同じ場所にファイルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル**]を選択します。サーバーの Prefix フォルダーにファイルを保存する場合は、**ローカル**]を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**ローカル**]を選択します。結果テーブルを AX Server にインポートするために **ローカル**]設定を選択解除することはできません。

10. 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ(<)を入力します。複数行のヘッダーやフッターを入力する場合は、**[ヘッダー]**または**[フッター]**ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキストボックスで、改行文字としてセミコロン(;)を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

11. **詳細]**タブをクリックします。
12. **適用範囲]**パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキストボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキストボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。[While]テキストボックスに条件を入力するか、または[While]ボタンをクリックし、式ビルダーを利用してWHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または"次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

13. 出力の種類として **[ファイル]**を選択した場合、出力結果を既存ファイルの末尾に追加するときは、次のいずれかを行います。
 - テキストファイル、または出力結果とまったく同じ構造であることがわかっている Analytics テーブルに追加する場合は、**既存のファイルに追加する]**を選択します。
 - Analytics テーブルに追加するときに、出力結果と既存のテーブルのレコード長を ACL に比較させたい場合は、**既存のファイルに追加する]**を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する**]を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、「既存テーブルへの出力結果の追加」ページ 207を参照してください。

14. 出力の種類として**ファイル(Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く**]を選択します。
15. **OK**]をクリックします。
16. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。
最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**いいえ**]をクリックして操作をキャンセルし、「既存テーブルへの出力結果の追加」ページ 207を参照してください。

重複の検査

1つ以上のフィールドの値が重複している、すなわち重複レコードは、データの入力ミスの結果であるか、もしくは調査を回避するためにクレジットカードの決済を分けている、などの不正行為の結果である可能性があります。

一意の値の要件

重複を含んではいけないフィールドとは、その値によってレコードを一意に識別するフィールドです。たとえば、従業員テーブルの場合、従業員番号は各従業員を一意に識別するものなので重複してはいけません。

有効な重複

重複値が有効な場合もあります。たとえば、取引テーブルには、重複する顧客番号が含まれているかもしれませんが、これは同じ顧客によって複数の取引が行われたためである可能性があります。

さまざまな種類の重複検査

Analytics では、次のような方法で重複を検査することができます。

テスト範囲	次の場合にはこのテストを使用します。
1つのフィールド	特定フィールドのすべての値が一意である必要がある、たとえば、従業員番号や小切手番号など。
2つ以上のフィールドの組み合わせ	<p>各フィールドが分離した状態では一意性が要求されないが、フィールドを組み合わせた状態では一意性が要求される。</p> <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;"> <p>例</p> <p>年間の給与支払ファイルでは、従業員番号フィールドと給料日フィールドはそれぞれ多くの重複を含んでいるでしょう。従業員は2週間ごとに給与が支払われ、複数の従業員が同じ日に給与を受け取ります。</p> <p>ただし、個々の従業員は特定の日付に一度しか現れてはいけません。従業員番号と支払日を組み合わせると全体で重複が存在する場合、従業員が同じ給与期間に2回支払われている可能性があります。</p> </div>

テスト範囲	次の場合にはこのテストを使用します。
レコードのすべてのフィールド	レコード内のすべてのフィールドが複製されている、全体が重複しているレコードを調べる。全体が重複しているレコードは、データの入力ミスの結果であるか、もしくは他の不正な取引行為の結果である可能性があります。

並べ替えと重複

一般に、重複がないかどうかをテストするには、1つまたは複数のキーフィールドを並べ替える必要があります。キーフィールド内の重複値が、それら同士がすぐ隣に並んでいる場合にのみ検出できるためです。

キーフィールドを並べ替えずに重複値がないかどうかをテストしても、隣り合っていない重複値は重複値として報告されません。同じ重複値の塊が2つ以上ある場合でも、それらの塊は別々のグループの重複値として報告されます。

分析の目的によっては、キーフィールドを並べ替えずに重複値がないかどうかをテストすることにも意味がある場合があります。たとえば、ソーステーブル内ですぐ隣に並んでいる重複値のみを検出して、隣り合っていない重複値は無視したい場合が、その例です。

出力テーブルにグループ番号フィールドを含める

重複値の出力テーブルには、グループ番号フィールドを含めることができます。このフィールドは、一意の各重複値グループに対し、1ずつ増える連番を割り当てたものです。重複値グループを番号で参照するこの機能は、出力テーブル内のデータを分析する場合に使用できます。

重複値の出力テーブルをグループ番号でフィルター処理する

複数のキーフィールドを組み合わせることで、買掛金テーブルに重複レコードがないかどうかをテストすることができます。

- Vendor_No(ベンダー番号)
- Invoice_No(請求書番号)
- Invoice_Date(請求書番号)
- Invoice_Amount(請求金額)

結果の重複値出力テーブルをフィルター処理することで、いくつかの重複値グループのみを追加処理することができます。

キーフィールドの組み合わせを使ってフィルターを作成することは、次のように骨が折れます。例：

```
SET FILTER TO ((Vendor_No = "11475") AND (Invoice_No = "8752512") AND  
(Invoice_Date = `20191021`) AND (Invoice_Amount = 7125.80)) OR  
((Vendor_No = "12130") AND (Invoice_No = "589134") AND (Invoice_Date =  
`20191117`) AND (Invoice_Amount = 10531.71)) OR ((Vendor_No = "13440")  
AND (Invoice_No = "5518912") AND (Invoice_Date = `20191015`) AND  
(Invoice_Amount = 11068.20))
```

このようにする代わりに、次のようにグループ番号を基準にしたフィルターを作成することで、同じ結果が得られます。

```
SET FILTER TO MATCH(GROUP_NUM, 3 , 8, 11)
```

手順

重複する値や、全体が重複しているレコードが存在するかどうかを検出するために、アクティブなテーブル内の1つ以上のフィールドを検査することができます。

方法の説明

文字、数値、および日付時刻フィールドの重複を検査することができます。文字フィールドに文字と数字が続けて現れる場合、すべての英数文字が検査されます。

メモ

有効な結果を得るために、検査されるフィールドは、事前に順番どおりに整列されている必要があります。フィールドの並べ替えは、前もって実行しておくか、重複の検査時に**あらかじめ並べ替える**オプションを使用して行うことができます。

フィールドの選択

1. 重複値がないかどうかをテストするテーブルを開きます。
2. **分析 > 重複**を選択します。
3. 1つまたは複数のフィールド内の重複値を検出するには
 - a. **メイン**タブで、**重複の対象**リストから検査するフィールドを選択するか、または**重複の対象**をクリックして、フィールドを選択したり式を作成したりします。

フィールドを選択する順序が、結果に表示する列の順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

- b. **重複の対象**をクリックした場合は、任意で並べ替えの矢印  (デフォルトは昇順) をクリックすることにより、1つまたは複数の選択したフィールドの出力結果を降順にするよう指定することができます。

- c. 出力結果に任意の追加フィールドを含めるには、**フィールドの一覧**リストから1つ以上のフィールドを選択します。または **フィールドの一覧**をクリックして、フィールドを選択したり、フィールドを **すべて追加**したり、式を作成したりすることもできます。

追加フィールドによって、結果に有益なコンテキストを提供することができます。重複の検査対象に選択したフィールドは、結果レコードの先頭に自動的に表示されるので、**フィールドの一覧**で特に選択する必要はありません。

- d. 省略可能。出力テーブルに **グループ番号**フィールドを含めたい場合は、**グループの追加**を選択します。

グループ番号フィールドは、一意の各重複値グループに対し、1ずつ増える連番を割り当てたものです。

4. すべての重複値レコードを検出するには

- メイン**タブの **重複の対象**をクリックします。
- すべて追加**をクリックして、すべてのフィールドを **選択済みのフィールド**に追加します。
- 任意で、並べ替えの矢印  (デフォルトは昇順) をクリックすることにより、1つまたは複数のフィールドの出力結果を降順にするよう指定することができます。
- OK**をクリックします。

結果レコードには、テーブル内のすべてのフィールドが表示されるので、**フィールドの一覧**リストからフィールドを選択する必要はありません。

- e. 省略可能。出力テーブルに **グループ番号**フィールドを含めたい場合は、**グループの追加**を選択します。

グループ番号フィールドは、一意の各重複値グループに対し、1ずつ増える連番を割り当てたものです。

レコードを処理対象から除外する(省略可能)

現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または **[f]**ボタンをクリックし、**式ビルダー**を利用してIFステートメントを作成します。

メモ

If条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IFステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

事前並べ替え]をオフにする(省略可能)

検査対象とするフィールドが前の操作で既にソート済みであれば、任意で **あらかじめ並べ替える**]をオフにすることにより、大きなテーブルの重複を検査する場合の時間を節約できます。

メモ

あらかじめ並べ替える]をオフにする場合、結果を有効なものにするには、重複検査の対象として選択するフィールドは、前にソートされたフィールドと同じものにする必要があります。

選択したフィールドとソート済みフィールドの間に不一致がある場合は、結果に**"警告：ファイルの順序が正しくありません"**というメッセージが提示されます。結果を Analytics テーブルに出力する場合、警告メッセージはコマンド ログに記載されます。

フィールド内のデータがソートされていない場合は、すべての重複が確実に検出されるようにするために、**あらかじめ並べ替える**]はオンのままにしておいてください。

出力を構成する

1. **出力**]タブをクリックします。
2. **出力先**]パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果を Analytics テーブルまたはテキスト ファイルに保存または追加するには、このオプションを選択します。Analytics テーブルに保存または追加する場合、そのテーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。テキスト ファイルに保存または追加する場合、ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

3. 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - 結果を新しい **Analytics テーブル**に保存する、または既存の Analytics テーブルに追加する場合は、**"Analytics テーブル"**を選択します。結果をテキスト ファイルに保存または追加する場合は、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**を選択します。
 - **名前** - **名前**]テキスト ボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、`C:\Results\Output.fil` または `Results\Output.fil` のように指定します。

メモ

Analytics テーブル名は、拡張子 `.FIL` を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (`_`) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- ローカル- サーバー テーブルに接続されており、Analytics テーブルに結果を保存または追加する場合にのみ有効になります。プロジェクトと同じ場所にファイルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**[ローカル]** を選択します。サーバーの Prefix フォルダーにファイルを保存する場合は、**[ローカル]** を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**[ローカル]** を選択します。結果テーブルを AX Server にインポートするために **[ローカル]** 設定を選択解除することはできません。

- 出力の種類によっては、任意で **[ヘッダー]** および **[フッター]** をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ (`<`) を入力します。複数行のヘッダーやフッターを入力する場合は、**[ヘッダー]** または **[フッター]** ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (`;`) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

処理の範囲を指定する

- [詳細]** タブをクリックします。
- [適用範囲]** パネルで、適切な出力オプションを選択します。

- すべて
- 先頭
- 次へ
- While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端

	の列の実際のリコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>

設定の最終処理を行う

- 出力タイプとして **[ファイル]** を選択し、出力テーブルを既存のファイルの最後に追加する場合は、次のいずれかを実行します。
 - 出力結果と構造が同じであることが確実なテキストファイルまたは Analytics テーブルに追加する場合は、**[既存のファイルに追加]** を選択します。
 - Analytics テーブルに追加するときに、出力結果と既存のテーブルのレコード長を ACL に比較させたい場合は、**[既存のファイルに追加する]** を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**[既存のファイルに追加する]** を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207 を参照してください。

- 出力の種類として **ファイル (Analytics テーブル)** を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**[出力テーブルを開く]** を選択します。

3. **OK]**をクリックします。

メモ

次の両方を行う場合は、重複する値またはレコードのみが表示され、値またはレコードの初回の発生は表示されません。

- 結果を画面またはテキスト ファイルに出力する
 - 出力結果にテストされたフィールドのみを含め、追加のフィールドを選択しない
- 画面に出力する場合は、値をクリックすると、重複に加えて値またはレコードの最初の出現を見ることができます。

4. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**[いい]**をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

重複の削除

要約操作を使用して、データセットから重複値またはレコードを削除し、残った一意の値またはレコードを新しい Analytics テーブルに保存することができます。

方法の説明

フィールドの選択

1. 重複値を削除するテーブルを開く
 2. **分析 > 要約]**の順に選択します。
 3. **メイン]**タブで、次のいずれかを行います。
 - **要約の対象]**リストから、重複値を含んでいる可能性のあるフィールドを選択します。
 - **要約の対象]**をクリックして、フィールドを選択するか、または式を作成します。
- フィールドを選択する順序が、結果に表示する列の順序になります。

メモ

該当するフィールドを選択し、必要な一意性の度合いを実現します。

たとえば、重複する従業員レコードを削除し、姓フィールドのみを選択する場合、同じ姓で名前が異なる従業員のすべてのレコードを削除するリスクがあります。姓フィールドと名フィールドの両方を選択すると、一意性の度合いが上がります。

完全に重複するレコードのみを削除するには、**要約の対象]**と**すべて追加]**をクリックします。

4. **小計フィールド]**は何も選択しないでください。

5. 省略可能。次のいずれかを実行します。

- **その他のフィールド**]リストから、出力結果に含めるその他のフィールドを選択します。
- **その他のフィールド**]をクリックして、フィールドを選択するか、または式を作成します。

メモ

各要約グループ内のすべてのレコードが同じ値を含んでいるフィールドに対してのみを選択してください。詳細については、「その他のフィールド オプション」ページ 1330を参照してください。

事前並べ替え]をオフにする(省略可能)

重複値を含んでいる可能性のあるフィールドが既にソート済みであれば、任意で **あらかじめ並べ替える]**をオフにすることもできます。フィールド内のデータがソートされていない場合は、有効な結果を得るために、**あらかじめ並べ替える]**はオンのままにしておいてください。

レコードを処理対象から除外する(省略可能)

現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

出力を構成する

1. **出力]**タブをクリックします。
2. **出力先]**パネルで、**[ファイル]**を選択します。
3. **出力指定]**パネルで次の情報を指定します。
 - **ファイルの種類** - **Analytics テーブル**が唯一のオプションです。結果を新しい Analytics テーブルに保存するか、または既存の Analytics テーブルに追加します。
 - **名前** - **名前]**テキスト ボックスにテーブル名を入力します。もしくは、**名前]**ボタンをクリックして、**保存]**または **[ファイルを保存する名前]**ダイアログ ボックスでテーブル名を入力するか、既存のテーブルに上書きまたは追加する場合はそのテーブルを選択します。Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にテーブルを保存したり、その場所にあるテーブルに追加したりすることもできます。たとえば、`C:\Results\No_duplicates.fil` または `Results\No_duplicates.fil` と指定します。

- ローカル- サーバー テーブルに接続している場合のみ有効になります。プロジェクトと同じ場所に出カテーブルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、`[ローカル]` を選択します。Analytics サーバーの Prefix フォルダーに出カテーブルを保存する場合は、`[ローカル]` を選択解除されたままにしておきます。

メモ

出力結果が Analytics Exchange のサーバー テーブルの分析や処理から生成される場合は、`[ローカル]` を選択します。`[ローカル]` 設定は、結果テーブルを AX Server へインポートするために使用することはできません。

処理の範囲を指定する

- 詳細] タブをクリックします。
- 適用範囲] パネルで、適切な出力オプションを選択します。

- すべて
- 先頭
- 次へ
- While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 <code>[While]</code> テキスト ボックスに条件を入力するか、または <code>[While]</code> ボタンをクリックし、<code>式ビルダー</code> を利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または"次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>

メモ

"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。

ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。

設定の最終処理を行う

1. 処理の完了時に、出力テーブルを自動的に開くようにするには、**出力テーブルの使用]**をオンにします。
2. 出力テーブルを既存の Analytics テーブルの最後に追加する場合は、次のいずれかを実行します。
 - 出力結果と既存のテーブルの構造が同じであることが確実な場合は、**既存のファイルに追加]**を選択します。
 - 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、**既存のファイルに追加する]**を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する]**を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

3. **OK]**をクリックします。
4. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**[いいえ]**をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

あいまい重複分析

あいまい重複は、実体としては同一の可能性があるほぼ同一の文字値です。たとえば、以下の4つの値はすべて同じ会社である可能性があります。

- Intercity Couriers
- Inter-city Couriers
- Intercity Couriers Inc.
- Intrecity Couriers

あいまい重複の原因としてよく挙げられるのは、タイプミスやスペルミス、データの書式設定の相違、および異なるデータ入力規則などのデータ入力エラーです。ほぼ同一の値を意図的に作成することは、不正を意味する可能性があります。あいまい重複は、一貫性のある実体を基準とするデータに依存するデータ分析の妨げになります。

曖昧結合と曖昧重複

曖昧重複機能は、単一の Analytics テーブルの単一のフィールドの値を分析します。曖昧一致を使用するには、2つの Analytics テーブルのフィールドを組み合わせ、新しい3番目のテーブルにします。"曖昧結合" ページ 985を参照してください。

機能の仕組み

Analytics のあいまい重複では、テーブルの特定の文字フィールドをテストし、フィールドに含まれるあいまい重複を特定できます。出力結果は、指定した相違の度合いに基づいてあいまい重複をグループ化します。相違の度合いを調整すると、出力グループの数とサイズ、グループメンバー間の相違の量を制御できます。

あいまい重複グループメンバーが実際に同一のエントティティを参照しているかどうかを確認するには、テストフィールド以外のフィールドの重複テストといった追加の分析を実行しなければならない場合があります。

メモ

あいまい重複の検査は、厳密な重複の特定よりも複雑です。あいまい重複間の相違の度合いを制御する設定や、あいまい重複を出力結果でどのようにグループ化するかを制御する設定を理解しておけば、この機能を最大限に利用することができます。

あいまい重複の出力結果

以下は、あるテーブルの Last Name フィールドに対してあいまい重複の検査を実行した出力結果の例です。

	グループ	姓	グループ番号	オリジナルレコード番号
1	グループ 2	Hansen	2	2
2		Hanssen	2	4
3		Hanson	2	5
4		Jansen	2	6
5	グループ 3	Janson	3	3
6		Hanson	3	5
7		Jansen	3	6
8		Jansan	3	7
9		Jansn	3	9
10	グループ 6	Jansen	6	6
11		Janszen	6	8
<< ファイルの終わり >>				

出力結果は、2、3、6として特定されたグループに配置されます。各グループの先頭にあるあいまい重複の元のレコード番号は、グループの識別に使用されます。たとえば、"Janson" は元のテーブルでレコード番号 3 の名前です。この "Janson" は元のテーブルのレコード順序に従い、グループ内で 1 番目の値となるため、そのグループは "グループ 3" として分類されます。詳細については、"あいまい重複をグループ化する方法" ページ 1297 を参照してください。

あいまい重複機能は文字に基づく比較を使用します。

2つの値を比較する場合、あいまい重複機能は単語ベースではなく文字ベースでの比較を実行します。この機能では、単語間の空白やスペースを文字として扱い、個々の単語間を区別しません。値内の単語数に関係なく、その値を切れ目のない単独文字列として扱います。

この方法の実装は、データの性質や **あいまい重複** ダイアログボックスで指定した設定の違いによって、人間の目にはあいまい重複と思われる値でも出力結果に含まれない可能性があります。

例

次の名前を考えます。

- "JW Smith" と "John William Smith"
- "Diamond Tire" と "Diamond Tire & Auto"

1 番目の例は、同じ名前の 2 つのバージョンで、ファーストネームとミドルネームを頭文字で略したものと、略せずに記述したものです。2 番目の例は、会社名を短いバージョンと長いバージョンで記述しています。

これらのペアはいずれも、相違の設定をかなりゆるくしないとあいまい重複として出力されません。設定をゆるくすることは、多くの誤検知も出力されてしまう弊害も生じます。

あいまい重複機能は、例に挙げたそれぞれのペアを単純に2つの文字列として処理します。それぞれの例で、対となる2つの文字列の長さは大きく異なるので、文字レベルで判断すると、文字列は互いに大きく異なります。

詳細については、「設定による動作の違い」ページ 1293を参照してください。

あいまい重複分析の効果の改善

目的の結果を得るには、あいまい重複の主機能に加え、テスト データセットのサイズの制限、あいまい重複のヘルパー関数の使用、検査フィールドの連結が必要となる場合もあります。

次の表は、あいまい重複分析の効果を改善するためのさまざまな手法についてまとめています。

ヘルパー関数の詳細については、「あいまい重複のヘルパー関数」ページ 1284を参照してください。

手法	Analytics 機能	詳細
検査するデータセットのサイズを制限する	フィルター データのサブセットを抽出	分析対象として有効なレコードのみを処理することで実行時間を短縮する
テスト フィールド値の個別の要素を並べ替える	SORTWORDS() 関数	テスト値の個別の要素の物理的な配置の重要度を最小化することで、結果のサイズを減らし、精度を高める メモ あいまい重複機能は文字に基づく比較を使用しますが、テスト値の単語または要素を並べ替えると、比較対象の文字列の間により関連性を持たせながら文字を配置することができます。
テスト フィールド値から汎用要素を削除する	OMIT() 関数	意味のある相違が生じる可能性があるテスト値の部分のみを分析の対象とすることで、サイズを減らし、結果の精度を高める
テスト値の一意性を高めるためにフィールドを連結する	加算演算子(+)を使用した Analytics 式	テストする値の一意性を高めることで、サイズを減らし、結果の精度を高める(一意性を高める値は、2つ以上のフィールドを連結することで生成できません)
あいまい重複出力結果で、特定の値のあいまい重複の単一かつ網羅的なリストを生成する	ISFUZZYDUP() 関数	分析の目的に特に関連の深い出力値を対象に、あいまい重複の便利で完全な一覧を生成する

テスト フィールドの並べ替えが必要かどうか

あいまい重複のフィールドをテストするために、フィールドを並べ替える必要はありません。テストの前にテストフィールドでテーブルを並べ替えても、あいまい重複処理の効果が上がることはありません。ただし、出力結果を見やすくするため、また **あいまい重複** ダイアログボックスには **あらかじめ並べ替える** オプションがないことから、事前に検査フィールドを並べ替えてもよいでしょう。

メモ

テスト フィールド値を並べ替えても効果は上がりませんが、住所などの複数の要素を含むフィールド値の個別の要素を並べ替えると、効果が大幅に向上することがあります。詳細については、"あいまい重複のヘルパー関数" ページ 1284を参照してください。

完全な重複を含める

あいまい重複を検査する場合、出力結果に完全な重複も含めることができます。完全な重複のみを検索することが目的であれば、"重複の検索" 機能を使用してください。詳細については、"重複の検査" ページ 1265を参照してください。

あいまい重複の検査

アクティブなテーブル内の1つの文字フィールドを検査し、ほぼ同一の値(あいまい重複)が存在するかどうかを検出することができます。任意で、ほぼ同一の値だけでなく、厳密に同一(完全な重複)の値も出力結果に含めることができます。

出力結果に含まれる1つ以上のあいまい重複グループのサイズが最大サイズに達した場合には、ログにメッセージが表示されます。詳細については、「あいまい重複の結果のサイズ制御」ページ 1290を参照してください。

あいまい重複テストの効果の改善

次の手法を1つ以上取り入れると、あいまい重複テストの効果を大幅に高めることができます。

- テストフィールド値の個別の要素を並べ替える
- テストフィールド値から汎用要素を削除する
- テストフィールドを連結する

詳細については、「あいまい重複のヘルパー関数」ページ 1284と「フィールドの連結」ページ 233を参照してください。

実行時間と出力結果のサイズを減らす

あいまい重複機能は、検査フィールドの値を、そのフィールドの後続の各値と比較する必要があるため、プロセッサを消費します。

分析に支障がなければ、レコードのフィルターリングやサブセット抽出などの方法を利用して、検査対象のデータセットのサイズを制限してください。データセットのサイズを抑えることで、全体的な実行速度も上がり、また出力結果のサイズの制御にも役立ちます。

手順

メモ

手順の後に詳細情報が表示されます。「あいまい重複」ダイアログボックスオプション" ページ 1282を参照してください。

1. **分析 > あいまい重複**の順にクリックします。
2. **メイン**タブで、次のいずれかを行います。
 - **あいまい重複の対象**ドロップダウンリストから、検査の対象となるフィールドを選択します。

- **あいまい重複の対象]**をクリックして、フィールドを選択するか、または式を作成します。

ヒント

式の作成は、テストフィールドを連結したり、テストフィールド値から汎用要素を削除したり、テストフィールド値の個別の要素を並べ替えたりする方法です。詳細については、"あいまい重複のヘルパー関数" ページ 1284と"フィールドの連結" ページ 233を参照してください。

3. 省略可能。結果に任意の追加フィールドを含めるには、**フィールドの一覧]**リストから1つ以上のフィールドを選択します。または **フィールドの一覧]**をクリックして、フィールドを選択したり、フィールドを **すべて追加]**したり、式を作成したりすることもできます。

フィールドを追加することで、結果に有用な前後関係を提供でき、また、あいまい重複が実体としては同一のものを指しているかどうかの検証にも役立ちます。

メモ

あいまい重複の検査対象に選択したフィールドは、結果レコードの先頭に自動的に表示されるので、**フィールドの一覧]**で特に選択する必要はありません。

4. あいまい重複間の相違の量を制御するには **相違のしきい値]**を指定します。
設定は次の説明のとおりです。
5. 次のいずれかを実行します。
 - **相違のパーセント]**を指定すると、各あいまい重複における相違の割合を制御することができます。
 - **相違のパーセント]**オプションのチェックをオフにすると、このオプション指定は無効になります。
設定は次の説明のとおりです。
6. 次のいずれかを実行します。
 - **結果サイズ (%)]**で、検査フィールドのサイズに対する結果の最大サイズを指定します。
 - **結果サイズ (%)]**オプションのチェックをオフにすると、このオプション指定は無効になります。
設定は次の説明のとおりです。
7. 結果にあいまい重複だけでなく厳密な重複も含めようになりたい場合は、**完全な重複を含める]**オプションを選択します。
詳細については、"あいまい重複をグループ化する方法" ページ 1297を参照してください。
8. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー]**を利用して IF ステートメントを作成します。
IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。
9. サーバーテーブルに接続されている場合は、次のいずれかを実行します。
 - プロジェクトと同じ場所に出カテーブルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル]**を選択します。

- サーバーのPrefix フォルダーに出力テーブルを保存する場合は、**[ローカル]**を選択解除されたままにしておきます。

メモ

出力結果がAnalytics Exchange のサーバー テーブルの分析や処理から生成される場合は、**[ローカル]**を選択します。**[ローカル]**の設定は、結果テーブルをAnalytics Exchange へインポートする場合に選択解除することはできません。

10. 次のいずれかを実行します。

- **保存先]**テキスト ボックスで、出力結果を含める Analytics テーブルの名前を指定します。
- **保存先]**ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスで既存のテーブルに上書きまたは追加する場合はそのテーブルを選択します。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にテーブルを保存したり、その場所にあるテーブルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

保存または追加するテーブルの場所に関係なく、そのテーブルが開いているプロジェクトにまだ存在しないときは、プロジェクトに追加されます。

Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

11. 処理の完了時に、出力テーブルを自動的に開くようにするには、**出力テーブルの使用]**をオンにします。
12. **OK]**をクリックします。
13. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

あいまい重複]ダイアログ ボックス オプション

次の表は、**あいまい重複]**ダイアログ ボックスのオプションの詳細を示します。

オプション- あいまい重複] ダイアログ ボックス	説明
相違のしきい値	あいまい重複間で許可される相違の量。 1 から 10 までの数値を指定します。 相違のしきい値] の値を大きくすると、あいまい重複の各値間で許容される相違の文字数が多くなり、結果のサイズが大きくなります。 詳細については、「設定による動作の違い」ページ 1293を参照してください。
相違のパーセント	各あいまい重複における相違の割合。 1 から 99 までのパーセントを指定します。 相違のパーセント] の値を大きくすると、あいまい

オプション- [あいまい重複] ダイアログボックス	説明
	<p>重複の値に対して許容される相違の割合が高くなり、結果のサイズが大きくなります。</p> <p>相違のパーセント]オプションをオフにすると、あいまい重複における相違の割合は結果に考慮しません。この結果のサイズは、相違のパーセント]オプションを使用して値を設定したときよりも大きくなります。</p> <p>詳細については、"設定による動作の違い" ページ 1293を参照してください。</p>
結果サイズ (%)	<p>検査フィールドのサイズに対する結果の最大サイズ。</p> <p>1 から 1000 までのパーセントを指定します。このオプションを使用すると、結果のサイズが、有用と考えられるサイズを超えたときにあいまい重複操作を自動的に終了させることができます。</p> <p>たとえば、50,000 個の値を持つ検査フィールドの場合、結果サイズ (%)]に 1 を設定すると、結果のあいまい重複数が 500 を超えた場合に処理が終了します。処理が終了した場合、出力テーブルは作成されません。</p> <p>結果サイズ (%)]オプションをオフにすると、結果のサイズに制限がなくなります。</p> <p>注意</p> <p>結果サイズ (%)]オプションをオフにすると、過度に大きな結果セットが生成され、処理時間が非常に長くなったり、使用可能なメモリの超過を引き起こして、処理が終了してしまう可能性があります。結果が、対処できる妥当なサイズであると確信している場合のみ、このオプションをオフにしてください。</p> <p>詳細については、"あいまい重複の結果のサイズ制御" ページ 1290を参照してください。</p>

あいまい重複のヘルパー関数

Analytics の次の2つの関数を使用すると、あいまい重複機能がより効果を発揮します。

- SORTWORDS()
- OMIT()

2つの関数は別々に使用するか組み合わせて使用できます。

3つ目の関数 ISFUZZYDUP() では、フィールド全体ではなく、特定の値のあいまい重複を特定できません。

SORTWORDS 関数

あいまい重複機能を使用するときには、SORTWORDS() 関数を使用して、テスト フィールド値の個別の要素を連続する順序に並べ替える式または演算フィールドを作成します。

住所の構成要素などの要素を並べ替えると、あいまい重複比較における要素の物理位置の重要度が低下します。結果として、効果が改善され、大幅に低い相違のしきい値を使用し、誤検出が少なく、絞り込まれた小さい結果セットを生成できます。

詳細については、「SORTWORDS() 関数」ページ 2498を参照してください。相違のしきい値]の詳細については、「設定による動作の違い」ページ 1293を参照してください。

SORTWORDS() の概要を説明する動画については、「[Fuzzy Matching Using SORTWORDS\(\)](#)」(英語のみ)をご覧ください。

例

次の2つの値では、22以上の相違のしきい値をあいまい重複出力結果に含める必要があります。

- 125 SW 39TH ST, Suite 100
- Suite 100, 125 SW 39TH ST

許可される最大の相違のしきい値は10であるため、あいまい重複機能は2つの値を相互のあいまい重複として特定しません。それらは明確に同じ住所です。

対照的に、SORTWORDS() を使用して、個別の住所の要素を並べ替える式または演算フィールドを作成する場合、相違のしきい値 2のみが2つの住所を相互のあいまい重複として返します。

- 100 125 39TH ST, SW Suite
- 100, 125 39TH ST SW Suite

OMIT 関数

あいまい重複機能を使用するときには、OMIT() 関数を使用して、テスト フィールド値から汎用要素を削除する式または演算フィールドを作成します。

ハイフン、カンマ、数値記号、"Inc."、"Street"、"St." などの単語や略語といった要素を削除すると、あいまい重複比較が、有意な相違が発生する可能性があるテスト値の部分にのみ集中します。結果として、効果が改善され、大幅に低い相違のしきい値を使用し、誤検出が少なく、絞り込まれた小さい結果セットを生成できます。

詳細については、「OMIT() 関数」ページ 2390を参照してください。相違のしきい値]の詳細については、「設定による動作の違い」ページ 1293を参照してください。

例

次の2つの値では、8以上の相違のしきい値をあいまい重複出力結果に含める必要がありません。

- Intercity Couriers Corporation
- Inter-city Couriers Corp.

相違のしきい値 8 では、焦点が定まらない大きな結果のセットが生成され、その結果のほとんどが誤検出となるでしょう。ただし、相違のしきい値では、2つの値が相互のあいまい重複として検出されないようにすることができます。

対照的に、OMIT() を使用して、「Corporation」や「Corp.」などの汎用要素を削除する式または演算フィールドを作成する場合、相違のしきい値 1 のみが2つの名前を相互のあいまい重複として返します。

- Intercity Couriers
- Inter-city Couriers

ISFUZZYDUP 関数

あいまい重複機能を使用して結果を確認後、ISFUZZYDUP() 関数を使用し、結果の特定の値に対するあいまい重複の単一かつ完全な一覧を出力することができます。分析目標に特に関連していると思われる値に対して、この追加のステップを実施できます。

「網羅的」とは、検査値から指定された相違の度合いの範囲内であれば、検査値に関連する検査フィールド内の位置に関係なく、すべての値が返されることを意味します。

設計では、あいまい重複機能は、網羅的ではないグループの出力結果を整理します。結果は全体として網羅的ですが、個々のグループはそうでない場合もあります。このアプローチでは、出力結果が大きくなりすぎて管理できなくなる状況を防止できます。

完全でないグループでも、分析の目的を果たしている可能性があります。そうでない場合は、ISFUZZYDUP() を使用して、個々の値に対する完全な結果を生成することができます。

詳細については、"ISFUZZYDUP() 関数" ページ 2319を参照してください。完全でないグループの詳細については、"あいまい重複をグループ化する方法" ページ 1297を参照してください。

あいまい重複の出力結果を使った作業

あいまい重複の出力テーブルにおけるグループ所有者と各グループメンバー間の実際の相違のしきい値(レーベンシュタイン距離)、また所有者とメンバー1組ごとに適用される実際の相違のパーセントを見る場合は、演算フィールドを追加してそれらの値を表示することができます。演算フィールドを追加したら、あいまいさの度合いによって出力結果を順序付けるために、入れ子の並べ替えを作成できます。

3つの演算フィールドを次の順序で作成する必要があります。

- グループ所有者用の演算フィールド
- レーベンシュタイン距離用の演算フィールド
- 相違のパーセント用の演算フィールド

相違のしきい値用フィールドおよび相違のパーセント用フィールドを追加するには

1. あいまい重複の出力結果が含まれるテーブルを開き、**編集 > テーブルレイアウト**]の順に選択します。
2. 次の手順で、グループ所有者用の演算フィールドを作成します。
 - a. **フィールド/式の編集**]タブで、**新しい式の追加** ]をクリックします。
 - b. **名前**]フィールドに、「**Group_Owner**」と入力します。
 - c. **デフォルト値**]フィールドで、出力結果にあるあいまい重複検査フィールドの物理名(「vendor_name」など)を入力するか、または **f(x)** ]をクリックして**式ビルダー**で一覧からそのフィールドを選択します。
 - d. **[f]**フィールドに式 **NOT ISBLANK(GROUP_FL)** を入力します。
 - e. **静的**]オプションを選択します。
 - f. **入力を受け入れる** ]をクリックします。
3. 次の手順で、レーベンシュタイン距離用の演算フィールドを作成します。
 - a. **新しい式の追加** ]をクリックします。
 - b. **名前**]フィールドに、「**Lev_Dist**」と入力します。
 - c. **デフォルト値**]フィールドに次の式を入力するか、**f(x)** ]をクリックして、**式ビルダー**で式を作成します。
`LEVDIST(ALLTRIM(Group_Owner),ALLTRIM(fuzzy_dup_test_field),F)`
`fuzzy_dup_test_field`を実際のあいまい重複テストの名前に置き換えます。
 - d. **入力を受け入れる** ]をクリックします。

4. 次の手順で、相違のパーセント用の演算フィールドを作成します。

- a. **新しい式の追加**  をクリックします。
- b. **名前**フィールドに、「Diff_Pct」と入力します。
- c. **デフォルト値**フィールドに次の式を入力するか、**f(x)**  をクリックして、**式ビルダー**で式を作成します:

```
100*DEC(Lev_Dist,2)/MINIMUM(LENGTH(ALLTRIM(Group_Owner)), LENGTH
(ALLTRIM(fuzzy_dup_test_field))
```

*fuzzy_dup_test_field*をあいまい重複テストフィールドの実際の名前に置き換えます。

- d. **入力を受け入れる**  をクリックし、**閉じる**  ボタンをクリックして **テーブルレイアウト**ダイアログボックスを終了します。
5. **Lev_Dist** および **Diff_Pct** 演算フィールドをビューに追加します。

グループ所有者と各グループメンバー間の実際の相違のしきい値(レーベンシュタイン距離)、および所有者とメンバー1組ごとに適用される実際の相違のパーセントが表示されるようになります。

フィールドをビューに追加する方法については、「ビューに列を追加する」ページ 849を参照してください。

6. 出力結果をあいまいの度合いによって順序付けする場合は、以下の作業を行います。
 - a. グループフィールド以外のフィールドはすべて新しいテーブルに抽出し、そこからグループフィールドが空でないレコードを取り除きます。

この抽出操作に対応するACLScript構文は次のようになります。

```
EXTRACT FIELDS Lev_Dist Diff_Pct GROUP_NUM Group_Owner ORIG_REC_NUM
fuzzy_dup_test_field IF ISBLANK(GROUP_FL) TO "Ranked_Fuzzy_Dupes_1"
OPEN
```

*fuzzy_dup_test_field*をあいまい重複テストフィールドの実際の名前に置き換えます。

- b. 抽出したテーブルで入れ子の並べ替えを実行します。1番目(主)の並べ替えフィールドとして **Lev_Dist** を使用し、2番目(副)の並べ替えフィールドとして **Diff_Pct** を使用します。

この並べ替え操作に対応するACLScript構文は次のようになります。

```
SORT ON Lev_Dist Diff_Pct TO "Ranked_Fuzzy_Dupes_2" OPEN
```

出力結果のあいまいさはテーブルの下方に行くにつれて大きくなります。**グループ番号** フィールドは、あいまい重複グループにおけるグループ所有者のオリジナルのレコード番号であり、**オリジナ**

ルレコード番号]フィールドは、あいまい重複グループにおけるグループメンバーの元のレコード番号です。

入れ子の並べ替えを作成する方法の詳細については、「複数のキーフィールドでの並べ替え」ページ 1189を参照してください。

あいまい重複の結果のサイズ制御

あいまい重複機能では検査フィールドの値に対し多対多比較を実行するアルゴリズムを使用するため、あいまい重複の結果のサイズが非常に大きくなる可能性があります。また、設計上、この比較では、完全一致が必要な比較よりも簡単に一致が返されます。

データの性質と指定する差異設定によっては、結果がテスト対象のテーブルの何倍も大きくなる可能性があります。結果が検査テーブルに対して大きくなり過ぎる場合は、検査の意味がなく、また結果のほとんどが誤検出である可能性があります。

あいまい重複の結果のサイズ制御方法

以下の方法を使用すれば、あいまい重複の結果のサイズを制御し、返される誤検出数を減らすことができます。

- **複数のテストフィールドを使用する**-検査値の一意性の度合いを高めるために検査フィールドを連結します。
- **テストフィールド値の要素を並べ替える**-SORTWORDS(関数を使用して、テストフィールド値の個別の要素を連続する順序に並べ替えます。これにより、小さい**相違のしきい値**を使用できます。
- **テストフィールド値から汎用要素を削除する**-OMIT()関数を使用してテストフィールド値から総称要素を除去します。これにより、**相違のしきい値**]に指定する値を小さくすることができます。
- **相違のしきい値** - **相違のしきい値**]に最初は小さな値(たとえば3以下)を指定して実行し、結果への制限が強すぎると感じる場合には値を大きくします。
- **相違のパーセント** - **相違のパーセント**]に最初は既定の値(50)を指定して実行し、結果への制限が強すぎると感じる場合には値を大きくします。特に理由がない限り、**相違のパーセント**]オプションをオフにしないでください。
- **結果サイズ(%)**-結果のサイズが大きくなり過ぎて扱いにくくならないよう、検査フィールドの値の数に基づき**結果サイズ(%)**]を指定します。**結果サイズ(%)**]で、検査フィールドのサイズに対する結果の最大サイズを設定します。特に理由がない限り、**結果サイズ(%)**]オプションをオフにしないでください。

メモ

この設定は、誤検出を含めるか除外するかには影響しません。

- **あいまい重複グループサイズを制限する**-SETコマンドを使用して、あいまい重複グループの最大サイズに既定のサイズ20より小さい値を指定します。たとえば、`SET FUZZYGROUPSIZE TO 10`。

メモ

この設定は、誤検出を含めるか除外するかには影響しません。

注意

上記の一部の方法では、制限が強すぎると有効なあいまい重複が除外される可能性があります。設定の組み合わせをいろいろ試してみて、特定のデータセットでどれが最も良い状態で機能するかを調べる必要があるかもしれません。

有効なあいまい重複を除外する可能性が最も低い方法は、`SORTWORDS()` 関数と `OMIT()` 関数を使用して連結されます。

最大結果サイズの指定

結果サイズ (%) オプションに結果の最大サイズを指定すると、結果のサイズが、有意と考えられるサイズを超えたときにあいまい重複操作を自動的に終了させることができます。操作が終了した場合、出力テーブルは作成されません。

結果サイズ (%) オプションは、処理時間が非常に長くなることを防止するための安全なメカニズムです。返される結果の有効性には関係しません。大きい結果サイズ制限を指定すると、結果の誤検出数が増える場合があります。逆に、小さい結果サイズを指定すると、すべての有効なあいまい重複が取り込まれる前に、処理が終了する可能性があります。

適切な制限を選択する

結果サイズに適切な制限を選択することは判断の問題であり、ある程度の実験が必要になることもあります。低めの制限から始めてください。制限を超え、処理が終了した場合、制限を上げることができます。処理を完了できる制限になったら、結果を検査します。誤検出の比率が多い場合は、1つ以上の「あいまい重複の結果のサイズ制御方法」前のページを使用することをお勧めします。

最適な結果セットでは、テスト フィールドにすべての有効なあいまい重複 (真の検出) が含まれますが、誤検出数も最小限に抑えられます。一般的に、最適な結果セットを達成するには、すべてのあいまい重複設定と使用可能なヘルパー方法でバランスを取る必要があります。

100 パーセントを超える結果サイズ制限を指定できる理由

結果セットのサイズは、デフォルトで検査フィールドのサイズの 10 % です。このサイズは変更可能で、1 から 1000 までのパーセントを指定できます。1000% の制限は多対多一致の性質に対応し、処理の暴走を回避します。多対多一致では、元の検査データセットより大きい結果が生成されるかもしれません。ただし、元の検査データセットのサイズを超える結果は、主に誤検出である可能性があります。

結果サイズ計算の端数処理

結果サイズの計算では、正の整数になるよう丸め (四捨五入) を使用します。また、2 未満の数字については最小結果サイズである 2 (グループ所有者 1 とメンバー 1) に切り上げます。

結果サイズ制限を無効にする

一般的に、結果が、対処できる妥当なサイズであると確信できない限り、**結果サイズ(%)**オプションをオフにしないでください。結果の数に制限を設けることなくあいまい重複操作を実行すると、処理時間が非常に長くなる、または使用可能なメモリを超過し、その結果、処理が終了します。

あいまい重複グループの最大サイズの設定

SET コマンドを使用してあいまい重複グループの最大サイズを指定することは、グループのサイズを制限する1つの方法です。サイズが制限されなければ、多くの誤検出が含まれるでしょう。この機能は、出力結果のいくつかのグループのみを対象にサイズを制限する設定を見つける場合には非常に有効です。全グループまたはグループの大半がその最大サイズに達するような場合は、設定が小さすぎるかもしれません。また、これによって有効なあいまい重複が除外されてしまう可能性があります。このほか、相違の設定が十分に制限されていない可能性もあります。これにより、グループのサイズが大きくなってしまいます。

最大グループサイズの既定の値は20です。これにグループの所有者は含みません。既定以外の値にする場合は、2から100までの数値を指定できます。指定された最大サイズは、Analytics セッションの間は有効となります。

グループが最大サイズに達した場合の動作

1つのあいまい重複グループが最大サイズに達すると、それ以降、そのグループ所有者に対するあいまい重複は検出されず、そのグループには現れません。その除外されたあいまい重複が、後続のグループに現れるかどうかはわかりません。これは、そのあいまい重複が後続のあいまい重複一致の一部であるかどうかで決まります。

分析において、最大サイズに達したグループの所有者に対するあいまい重複の完全な一覧を生成することが重要な場合は、この目的のためにISFUZZYDUP()関数を使用することができます。詳細については、"あいまい重複のヘルパー関数" ページ 1284を参照してください。

1つ以上のグループがその最大サイズに達した場合、メッセージがログに表示されます。最大サイズに達したグループ数が10以下の場合、そのグループはグループ番号によって個別に識別されます。

完全な重複はグループサイズ計算に含まれます

完全な重複は結果に含めないようにした(**完全な重複を含める**]オプションがオフ)場合でも、それはグループサイズの計算に含まれます。たとえば、あるグループが、最大グループサイズ20(グループ所有者1とグループメンバー20)に達したとしてログに表示されたが、結果にはグループメンバーが18個しか現れなかった場合、検査フィールドにグループ所有者の完全重複が最低でも2つ存在しています。

すべてが完全重複で構成されるグループも、最大グループサイズに達した場合はログに表示されます。しかし、完全な重複を結果に含めないようにしていた(**完全な重複を含める**]オプションがオフ)場合は、そのグループは出力結果に現れません。

詳細については、"SET コマンド" ページ 2081を参照してください。

設定による動作の違い

あいまい重複は、指定した相違の度合いに基づいて選択され、出力結果でグループ化されます。相違の度合いは、**あいまい重複** ダイアログボックスにある以下の2つの設定の組み合わせで決まります。

- **相違のしきい値** -では、2つのあいまい重複間で許容される相違数を制御します。
- **相違のパーセント** -では、個々の値に対する相違の割合を制御します。

この2つの設定は、それぞれ独立したしきい値として動作します。検査対象のフィールドの値が結果のあいまい重複のグループに含まれるようにするには、それらの値がこの両方のしきい値の範囲内にある必要があります。2つの設定を調整することで、結果の精度や有用性を最大限に高めることができます。

相違のパーセントの指定はオフにすることができます。その場合、**相違のしきい値**の範囲内にのみ値がある必要があります。**相違のしきい値**の指定をオフにすることはできません。

相違のしきい値の詳細

相違のしきい値は、2つの値があいまい重複と認定されるために、それらの値間で許容されるレーベンシュタイン距離の最大値です。

レーベンシュタイン距離の概要

レーベンシュタイン距離は、ある値を別の値と同一にするために、1つの文字を編集する必要がある最小回数です。必要な編集回数は、コンピューティングサイエンスアルゴリズムによって計算されます。

レーベンシュタイン距離の例

“Smith”と“Smythe”の間のレーベンシュタイン距離は2です。

- **編集 1** - ‘y’ は ‘i’ に代入される必要があります
- **編集 2** - ‘e’ は挿入される必要があります

レーベンシュタイン距離がより大きければ、2つの値間の相違も大きくなります。距離が0(ゼロ)ということは、2つの値がまったく同じということです。

以下の表では、レーベンシュタイン距離のさまざまな例を示しています。レーベンシュタイン距離の詳細については、[LEVDIST\(\)](#)を参照してください。

メモ

レーベンシュタインアルゴリズムでは、単語間の空白やスペースを文字として扱います。

値 1	値 2	レーベンシュタイン距離	相違のしきい値]が3の場合、結果に含まれる
Smith	Smith	0	はい (完全な重複を含める]オプションが選択された場合)
Smith	Smithe	1	はい
Smith	Smythe	2	はい
Hanssen	Jansn	3	はい
Smith	Brown	5	いいえ
Intercity Couriers	Intercity Couriers Inc.	5	いいえ
Diamond Tire	Diamond Tire & Auto	7	いいえ
JW Smith	John William Smith	10	いいえ

相違のしきい値の変更

相違のしきい値を大きくすると、許容されるレーベンシュタイン距離の最大値も大きくなります。これにより、相違の度合いがより大きい値も含まれることになり、結果のサイズが大きくなります。相違のしきい値には1から10までの数値を指定できます。

この数値に上限を設けている理由は、最大レーベンシュタイン距離が一定の限度を超える大きな値になると、多くの誤検出を含む非常に大きい結果セットが生成されてしまうからです。

この数値に下限を設けている理由は、0(ゼロ)を指定すると完全な重複のみが対象となってしまうからです。完全な重複のみを検索することが目的であれば、「重複の検索」機能を使用してください。

相違のパーセントの詳細

相違のパーセントは、2つの値があいまい重複と認定されるために、比較する2つ値のうち短い方の値に対して許容される相違の割合の最大値です。

相違のパーセントの計算方法

検査フィールドで比較する値のペアごとのレーベンシュタイン距離を使用して、Analytics は次の内部的な計算を実行します。

レーベンシュタイン距離 / 短い方の値の文字数 × 100 = 相違のパーセント

相違のパーセントの例

"Smith"と"Smythe"の間のレーベンシュタイン距離は2で、2つの値のうち短い方の値は5文字なので、相違のパーセントは40(2/5)になります。

この相違のパーセントが **相違のパーセント**] オプションで指定した割合以下の場合、2つの値は結果に含まれます。これは、その値がレーベンシュタイン距離の許容される最大値 (**相違のしきい値**) の範囲内であることを前提とします。

以下の表では、相違のパーセントのさまざまな例を示しています。

値 1(長さ)	値 2(長さ)	レーベンシュタイン距離、相違のパーセント	相違のパーセント]に50が設定された場合、結果に含まれる
Smith (5)	Smith (5)	0, 0% (0/5)	はい (完全な重複を含める] オプションが選択された場合)
Smith (5)	Smithe (6)	1, 20% (1/5)	はい
Smith (5)	Smythe (6)	2, 40% (2/5)	はい
Hanssen (7)	Jansn (5)	3, 60% (3/5)	いいえ
Smith (5)	Brown (5)	5, 100% (5/5)	いいえ
Intercity Couriers (18)	Intercity Couriers Inc.(23)	5, 27.77% (5/18)	はい
Diamond Tire (12)	Diamond Tire & Auto (19)	7, 58.33% (7/12)	いいえ
JW Smith (8)	John William Smith (18)	10, 125% (10/8)	いいえ

相違のパーセントの変更

相違のパーセントを大きくすると、相違の割合がより大きい値も含まれることになり、結果のサイズが大きくなります。**相違のパーセント**には1から99までの数値を指定できます。

この数値に上限を設けている理由は、相違のパーセントが100以上になると、同じあいまい重複グループ内で完全に異なる値のペアが含まれるようになるからです。たとえば、"ABC"と"XYZ"のレーベンシュタイン距離は3、短い方の値の長さは3なので、相違のパーセントは100になります。

この数値に下限を設けている理由は、0(ゼロ)を指定すると完全な重複のみが対象となってしまうからです。完全な重複のみを検索することが目的であれば、"重複の検索"機能を使用してください。

相違のパーセントをオフにする

相違のパーセントの指定は、状況に応じてオフにすることができます。相違のパーセント]オプションをオフにすると、値における相違の割合は結果に考慮しません。これにより、"JW Smith"と"John William Smith"などの有効なあいまい重複も出力される可能性があります。ただし、その場合のあいまい重複グループには、"Smith"と"Brown"などの完全に異なる値も含まれることとなります。この結果のサイズは、相違のパーセント]オプションを使用して値を設定したときよりも大きくなります。

相違のしきい値]および 相違のパーセント] オプションを併用した場合の動作

以下の表は、相違のしきい値]および 相違のパーセント]オプションを併用した場合の動作を示します。"相違のしきい値の詳細" ページ 1293 と"相違のパーセントの詳細" ページ 1294 で挙げた比較値が出力結果に含まれるようにするには、両オプションのしきい値の範囲内にある必要があります。

相違のしきい値と相違のパーセントが個別に判断される場合に、"Hanssen/Jansn"と"Intercity Couriers/Intercity Couriers Inc."が含まれます。ただし、2つの設定がまとめて考慮されるときには、含まれません。両方のしきい値の境界に当てはまらないためです。

値 1(長さ)	値 2(長さ)	レーベンシュタイン距離、相違のパーセント	相違のしきい値]に3、相違のパーセント]に50が設定された場合、結果に含まれる
Smith (5)	Smith (5)	0, 0% (0/5)	はい (完全な重複を含める]オプションが選択された場合)
Smith (5)	Smithe (6)	1, 20% (1/5)	はい
Smith (5)	Smythe (6)	2, 40% (2/5)	はい
Hanssen (7)	Jansn (5)	3, 60% (3/5)	いいえ
Smith (5)	Brown (5)	5, 100% (5/5)	いいえ
Intercity Couriers (18)	Intercity Couriers Inc.(23)	5, 27.77% (5/18)	いいえ
Diamond Tire (12)	Diamond Tire & Auto (19)	7, 58.33% (7/12)	いいえ
JW Smith (8)	John William Smith (18)	10, 125% (10/8)	いいえ

あいまい重複をグループ化する方法

あいまい重複操作では、データを処理するときに検査フィールドを下方(後方)へ順に移動します。この操作では、フィールドの1番目の値を後続の各値と比較し、次にフィールドの2番目の値を後続の各値と比較するというように、すべての値がその後続の各値と比較されるまで、そのフィールドをループします。前(上方)の値と比較されることはありません。

レコード番号	姓
1	Ronson
2	Hansen
3	Janson
4	Hanssen
5	Hanson
6	Jansen
7	Jansan
8	Janszen
9	Jansn

あいまい重複操作では、指定されている相違の設定に基づいて、2つの比較値があいまい重複かどうかを判断します(相違の設定に関する詳細については、「設定による動作の違い」ページ 1293を参照してください)。2つの値があいまい重複の場合、これらの値は共に同じグループに入れられます。重複するあいまい重複は表示されません(これについては後で説明します)。あいまい重複操作の結果には、複数のグループが含まれる可能性があります。

グループ所有者とグループメンバー

グループ内の1番目のあいまい重複が、グループの制御値であり所有者です。これは単に、検査対象のフィールドにおけるグループメンバーの中で最初に現れる値を基準にしているだけです。検査フィールドに同じデータが複数存在しているが、それらが別々に格納されている場合は、グループ所有者が異なる別々のグループが作成されます。

グループはグループ所有者のレコード番号によって識別されます。以下の例は、**姓**フィールドのテスト結果を示します。"Janson"は(相違の設定に基づいて)グループを形成します。この"Janson"は元のテーブルでレコード番号が3なので、グループは「グループ3」となります。

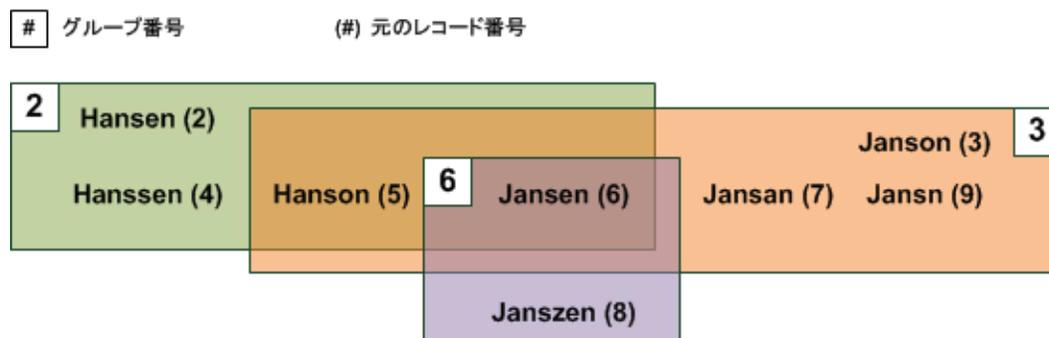
	グループ	姓	グループ番号	オリジナルレコード番号
1	グループ 2	Hansen	2	2
2		Hanssen	2	4
3		Hanson	2	5
4		Jansen	2	6
5	グループ 3	Janson	3	3
6		Hanson	3	5
7		Jansen	3	6
8		Jansan	3	7
9		Jansn	3	9
10	グループ 6	Jansen	6	6
11		Janszen	6	8

<< ファイルの終わり >>

グループ所有者は、正確な値である必要はありません。

グループ所有者は、正確な値または基準となる値である必要はありません。これは単に、指定された相違の度合いに基づく値が、グループの形成の処理で測定または計算されるということです。グループの全メンバーはそのグループ所有者の、指定された相違の度合いの範囲内にあります。メンバーが互いに、指定された相違の度合いの範囲内にあるかどうかはわかりません。

以下の図は、上記の出カテーブルの結果を視覚的に表したものです。相違のしきい値が1ということは、グループのメンバーはグループ所有者と最大1文字まで異なってもよいことを意味します。あいまい重複のいくつかは、複数のグループに現れていることに注意してください。



完全な結果と完全でない結果

結果のサイズが大きくなり過ぎないようにするため、あいまい重複機能は**完全でない**グループを生成するよう設計されています。「完全でない」とは、あいまい重複の個々のグループには、そのグループ所有者の指定された相違の度合いの範囲内にある、検査フィールドのすべてのあいまい重複が含まれない可能

性があることを意味します。ただし、グループ所有者が検査フィールド内の別の値のあいまい重複である場合は、2つの値が結果の(グループ所有者と関連があるグループに限らず)どこかのグループと一緒に現れます。このため、グループは完全ではない可能性があります、全体の結果は完全です。

分析において、検査フィールド内の特定の値に対するあいまい重複の、単一かつ完全な一覧を生成することが重要な場合は、この目的のためにISFUZZYDUP()関数を使用することができます。詳細については、「あいまい重複のヘルパー関数」ページ 1284を参照してください。

グループ形成の詳細

あいまい重複機能では、既に前のグループのグループ所有者に伴って出現した値は、以降のグループから除外することによって、完全でないグループを作成します。この方法によって、グループ形成では余分なあいまい重複のペアの数が減るので、結果全体のサイズを制御するのに役立ちます。

グループ形成を管理する規則については、後で例を挙げて説明します。

ルール	説明
所有者とメンバーの関係は非相互的	<p>あいまい重複操作では、検査フィールドを下方(後方)へ順に移動するので、グループ所有者は、フィールド内でそのグループ所有者の下(後続)に現れるあいまい重複のみと関連付けられます。グループ所有者の前に現れるあいまい重複とは関連付けられません。</p> <p>グループ所有者は、既にその上で現れる1つ以上のグループのメンバーである可能性があります。ただし、その逆はありません。上方(前方)にあるグループの所有者は、後続グループのメンバーにはなりません。値がグループの所有者になった場合、その値はそれ以降のグループに現れることはありません。</p> <p>上記の例では、グループ6の所有者である"Jansen"は、前の2つのグループのメンバーです。しかし、それら2つのグループの所有者("Hansen"と"Janson")は、"Jansen"のあいまい重複であるにもかかわらず、グループ6のメンバーではありません。</p>
2つの値が前のグループのメンバーであり、値の1つがその後続グループの所有者となる場合は、その2つの値が共に後続グループに入ることはありません。	<p>上記の例では、"Jansen"、"Jansan"および"Jansn"はすべてグループ3のメンバーです。"Jansen"がグループ6の所有者になった場合、"Jansan"と"Jansn"は、共にあいまい重複であり、それらが検査フィールドで"Jansen"の後方に現れるとしても、そのグループには入りません。</p>
2つの値が前のグループのメンバーであり、どちらの値もその後続グループの所有者とならない場合は、その2つの値は共に後続グループに入ることができます。	<p>上記の例では、"Hanson"と"Jansen"はグループ2とグループ3の両方に出現します。この場合、相違の度合いが、2つの値間ではなく、それぞれのグループの所有者から測定されるので、複数のグループへ一緒に現れることになります。</p>
<p>メモ</p> <p>2番目および3番目の規則には例外があります。実行時に、あいまい重複操作では値を一時的に格納します。これらの一時的な値のために割り当てられる容量がいっぱいになった場合、1つ以上の重複するグループメンバーを持つグループ所有者が出力されることがあります(所有者とメンバーは前のグループと一緒に現れています)。あいまい重複グループ用に指定する最大サイズを小さくすると、この余分な重複が起こりやすくなります。</p>	

あいまい重複によるデータ処理とグループ形成

以下の表は、上記の例のレコード単位の処理を示します。

入力データ		降順によるデータ処理	出力結果		
レコード番号	姓	あいまい重複が見つかりました	グループ番号	グループ所有者	グループメンバー(前のグループのグループ所有者と一緒に出現したメンバーは除外)
1	Ronson				
2	Hansen	Hanssen, Hanson, Jansen	2	Hansen	Hanssen, Hanson, Jansen
3	Janson	Hanson, Jansen, Jansan, Jansn	3	Janson	Hanson, Jansen, Jansan, Jansn
4	Hanssen				
5	Hanson				
6	Jansen	Jansan, Janszen, Jansn	6	Jansen	Janszen
7	Jansan	Jansn			
8	Janszen				
9	Jansn				

(相違の設定: 相違のしきい値 = 1, 相違のパーセント = 99)

結果に完全な重複を含める

あいまい重複操作では、データを処理するときに必ず完全な重複を含めますが、**あいまい重複**ダイアログボックスの**完全な重複を含める**オプションを選択していなければ、その完全な重複は結果から除去されます。

正確な重複には、あいまい重複と同じグループ形式ルールが適用されます。前のグループのグループ所有者と一緒に出現したメンバーはグループから除外されます。グループの所有者と除外された値が完全重複の場合、その除外された値は所有者のグループに入れる必要があるように思えるかもしれませんが。しかし、2つの値は共に前のグループに入っているため、この除外はグループの形成規則に合致していません。

以下の表は、正確な重複の処理を示します。

- "Ronson(3)" は "Ronson(4)" とグループを形成しません。これら2つの値は既にグループ1に入っているからです。

データの分析

- "Jansen(9)" は "Jansen(8)" で形成されたグループからは除外されます。これら 2 つの値は既にグループ 2 とグループ 5 に入っているからです。

入力データ		降順によるデータ処理	出力結果		
レコード番号	姓	検出されたあいまい重複と完全重複	グループ番号	グループ所有者	グループメンバー(前のグループのグループ所有者と一緒に出現したメンバーは除外)
1	Ronson	Ronson (3), Ronson (4)	1	Ronson	Ronson (3), Ronson (4)
2	Hansen	Hanssen, Hanson, Jansen (8), Jansen (9)	2	Hansen	Hanssen, Hanson, Jansen (8), Jansen (9)
3	Ronson	Ronson (4)			
4	Ronson				
5	Janson	Hanson, Jansen (8), Jansen (9), Jansan, Jansn	5	Janson	Hanson, Jansen (8), Jansen (9), Jansan, Jansn
6	Hanssen				
7	Hanson				
8	Jansen	Jansen (9), Jansan, Janszen, Jansn	8	Jansen	Janszen
9	Jansen	Jansan, Janszen, Jansn	9	Jansen	Janszen
10	Jansan	Jansn			
11	Janszen				
12	Jansn				

(相違の設定: 相違のしきい値 = 1, 相違のパーセント = 99, 完全な重複を含める = Y)

データのグループ化

データをグループ化すると、パターン、傾向、不規則性、または異常値を識別するのに役立つ概要が作成されます。フィールドの値、または複数のフィールドの値の組み合わせに基づいて、データをグループ化します。

グループ化により、選択した評価基準または識別子によって、集中させるレコード数、および値や数量の量を決めることができます。

グループ化で使用できる値

グループ化の1つの強力な要素は、さまざまな異なる型の値でグループ化できることです。

- 数値または値の範囲
- 特定の数
- 類似した数値
- 期間
- 特定の日付
- 次のようなレコード識別子
 - 業者または顧客番号
 - 決済コード
 - 製品識別子
 - ロケーションコード

グループ化処理

操作	データ型	機能	出力
階層化	数値	<ul style="list-style-type: none"> ◦ 数値データをグループ化します。 ◦ ある範囲内の数値間隔でレコードをグループ化し、指定した数値フィールドの小計を計算します。範囲のサイズと、その範囲内における間隔数を指定できます。 	<ul style="list-style-type: none"> ◦ 画面 ◦ グラフ ◦ 印刷 ◦ ファイル(テキストまたは Analytics テーブル)
年齢調べ	日付時刻	<ul style="list-style-type: none"> ◦ 日付時刻データをグループ化します。 ◦ 日付範囲または年齢調べ間隔でレコードをグループ化し、指定した数値フィールドの小計を計算します。 	<ul style="list-style-type: none"> ◦ 画面 ◦ グラフ ◦ 印刷 ◦ ファイル(テキスト)
分類化	文字 数値	<ul style="list-style-type: none"> ◦ 文字または数値データをグループ化します。 ◦ 文字または数値フィールドの値に基づいてレコードをグループ化し、指定した数値フィールドの小計を計算します。 	<ul style="list-style-type: none"> ◦ 画面 ◦ グラフ ◦ 印刷 ◦ ファイル(Analytics テーブル)

操作	データ型	機能	出力
要約	文字 数値 日付時刻	<ul style="list-style-type: none"> 文字、数値、または日付時刻データをグループ化します。 1つ以上の文字フィールド、数値フィールド、または日付時刻フィールドの値に基づいてレコードをグループ化し、指定した数値フィールドの小計を計算します。 要約は分類化と似ていますが、複数のフィールド別にレコードをグループ化できる点が異なります。 	<ul style="list-style-type: none"> 画面 印刷 ファイル(Analyticsテーブル)
クロス集計	文字 数値	<ul style="list-style-type: none"> 文字または数値データをグループ化します。 2つ以上の文字または数値フィールドの値に基づいてレコードをグループ化し、その結果として生じるグループを行と列のグリッドに表示します。また、数値フィールドの小計を計算します。 クロス集計は、2つの文字または数値フィールドを使用する要約と似ています。 	<ul style="list-style-type: none"> 画面 グラフ 印刷 ファイル(テキストまたはAnalyticsテーブル)
ヒストグラム	文字 数値	<ul style="list-style-type: none"> 文字または数値データをグループ化します。 文字フィールドまたは数値フィールドの値に基づいてレコードをグループ化し、その結果として生じるグループを棒グラフで表示します。数値フィールドの小計をサポートしていません。 文字フィールドを使用するヒストグラムは、分類化に似ています。数値フィールドを使用するヒストグラムは、階層化に似ています。 	<ul style="list-style-type: none"> 画面 グラフ 印刷 ファイル(テキスト)
クラスター	数値	<ul style="list-style-type: none"> 数値データをグループ化します。 1つ以上の数値フィールドの類似または付近の値に基づいて、レコードをグループ化します。 	<ul style="list-style-type: none"> ファイル(Analyticsテーブル)

データの階層化

階層化は、数値フィールドの値に基づいて、テーブル内のレコードを数値間隔(値範囲)でグループ化し、各間隔のレコード数をカウントします。

たとえば、売掛金テーブルを請求金額フィールドに基づいて5000ドル間隔、つまり、0 ~ 4,999.99ドルの請求書、5,000 ~ 9,999.99ドルの請求書というようにグループ化し、各間隔の取引の総数と取引金額の総額を調べることができます。

関連付けられた数値フィールドの小計

階層化すると、任意で1つ以上の関連付けられた数値フィールドを小計できます。上記の例では、割引金額フィールドの小計を計算して、各間隔の割引金額の総額を調べることができます。

メモ

小計フィールドを指定しないと、階層化の対象とするフィールドの小計が自動的に計算されます。

数値間隔の動作

数値間隔とは値の範囲です。数値間隔を作成するオプションは2つあります。

- 均等サイズの間隔
- カスタムサイズの間隔

均等サイズの間隔

Analytics は、キーフィールドの値を特定の間隔の数にグループ化して均等サイズの間隔を計算します。

均等サイズの間隔を作成するには、第1間隔の最小値と最終間隔の最大値、および必要な間隔の数を指定します。

ヒント

フィールドの実際の最小値と最大値を使用する場合は、間隔のサイズは一般に丸め金額ではありません。間隔のサイズを丸め金額とする場合は、たとえば、0と5000のように、丸め金額の最小値と最大値を指定することができます。

カスタム サイズの間隔

Analytics は、キーフィールドの値を、指定する開始値を持つ間隔にグループ化してカスタム サイズの間隔を計算します。

カスタム サイズの間隔を作成するには、各間隔の開始値、および最後の間隔の終了値を指定します。均等 サイズの間隔、またはサイズがさまざまな間隔を作成できます。

均等 サイズの間隔とカスタム サイズの間隔 の例

以下のテーブルは、48.19ドルから4,792.83までの範囲にある値一式で作成しうる間隔のタイプの例を示したものです。

均等 サイズの間隔 (実際の最小値と最大値を使用)	均等 サイズの間隔 (丸めの数として指定された最小値 と最大値)	カスタム サイズの間隔
48.19 - 997.11	0.00 - 999.99	0.00 - 99.99
997.12 - 1,946.04	1,000.00 - 1,999.99	100.00 - 999.99
1,946.05 - 2,894.97	2,000.00 - 2,999.99	1,000.00 - 5,000.00
2,894.98 - 3,843.90	3,000.00 - 3,999.99	
3,843.91 - 4,792.83	4,000.00 - 5,000.00	

階層化と並べ替え

テーブルが並べ替えられているかどうかに関係なく階層化できます。並べ替えられていないテーブルを階層化する場合、Analytics は階層化操作の一環として出力結果を自動的に並べ替えます。

統計オプション

[小計フィールドの統計を含める] オプションを使用すると、小計を出す各数値フィールドの平均値、最小値、および最大値を計算することができます。上の例で、統計オプションを使用すると、間隔ごとに請求金額の平均値、最小値、および最大値が計算されます。さらに、割引金額フィールドの小計を計算している場合は、間隔ごとに割引金額の平均値、最小値、および最大値も計算されます。

階層化の詳細

階層化では、次の操作を実行します。

操作	"階層化の結果" 下での該当箇所
数値フィールドに基づいてレコードを間隔ごとにグループ化する	Trans Amount フィールド, 1番目
各間隔に該当するレコード数をカウント(小計)し、その小計が全体に占める割合(カウントの割合)を計算する	カウント フィールド カウントの割合フィールド
階層化される数値フィールドの最小値と最大値を提供する	表示されません
任意で、間隔ごとに1つ以上の数値フィールドの値の小計を計算し、最初に選択したフィールドを対象にその小計がフィールド合計に占める割合を計算する	Trans Amount フィールド, 2番目 フィールドの割合フィールド
任意で、小計を計算する各数値フィールドの平均値、最小値、および最大値を計算する	表示されません
出力結果に含まれる全数値フィールドの合計を提供する	合計行
任意で、出力結果を文字フィールド(顧客IDやトランザクションタイプなど)の値に基づいてブレイクダウンする(文字フィールドは、階層化を実行する前にソートされている必要があります)。	表示されません

階層化の結果

次によって生成される出力結果:

- 売掛金テーブルの取引金額の階層化
(Ar テーブルは [ACL DATA\Sample Data Files\Sample Project.ACL](#) にあります)
- \$1000 間隔を使用
- 結果を画面に出力する

Trans Amount	カウント	カウントの割合	フィールドの割合	Trans Amount
-4,000.00 - -3,000.01	1	0.13%	-0.76%	-3,582.98
-3,000.00 - -2,000.01	3	0.39%	-1.36%	-6,371.13
-2,000.00 - -1,000.01	16	2.07%	-4.69%	-21,974.43
-1,000.00 - -0.01	141	18.26%	-5.65%	-26,468.32
0.00 - 999.99	421	54.53%	44.55%	208,898.46
1,000.00 - 1,999.99	151	19.56%	43.96%	206,110.67
2,000.00 - 2,999.99	24	3.11%	12.15%	56,980.77
3,000.00 - 3,999.99	12	1.55%	8.61%	40,357.68
4,000.00 - 4,999.99	2	0.26%	2%	9,380.78
5,000.00 - 6,000.00	1	0.13%	1.18%	5,549.19
合計	772	100%	100%	468,880.69

手順

テーブル内のレコードを均等な数値間隔またはカスタム サイズの数値間隔でグループ化することによって、データを階層化することができます。

各間隔について、任意で、関連する数値フィールドの小計、平均値、最小値、最大値の計算を含めることができます。

方法の説明

1. **分析 > 階層化]**の順に選択します。
2. **メイン]**タブで、次のいずれかを行います。
 - **階層化の対象]**ドロップダウンリストから、階層化の対象となるフィールドを選択します。
 - **階層化の対象]**をクリックして、フィールドを選択するか、または式を作成します。
3. 省略可能。 **小計フィールド]**を1つ以上選択するか、または **小計フィールド]** ボタンをクリックして、小計フィールドを選択するか式を作成します。

小計フィールドを選択しないと、階層化の対象とするフィールドの小計が自動的に計算されます。階層化するフィールドと、それ以外の1つ以上のフィールドの小計を出したい場合、または小計された階層化対象フィールドの統計を含めたい場合は、階層化するフィールドを明示的に選択する必要があります。

小計フィールドを選択する順序は、列が結果に表示される順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

4. **最小値]**で、最初の間隔の最小値を入力します。

階層化するフィールドに対して事前にプロファイルまたは統計を実行している場合は、階層化するフィールドの最小値がデフォルトで入力されています。必要に応じて、デフォルトを変更できます。
5. **最大値]**で、最後の間隔の最大値を入力します。

階層化するフィールドに対して事前にプロファイルまたは統計を実行している場合は、階層化するフィールドの最大値がデフォルトで入力されています。必要に応じて、デフォルトを変更できます。
6. 次のいずれかを実行します。
 - **間隔]**を選択し、 **最小値]**と **最大値]**によって指定された範囲内に必要な、均等な間隔の数を入力します。デフォルトの間隔数は10です。

ヒント

デフォルトの間隔数は、 **ツール > オプション]**の順にクリックして **コマンド]**タブを選択し、 **間隔の数]**を更新することによって変更できます。

- カスタム サイズの間隔を作成するには **フリー]**を選択し、各間隔の開始値および最後の間隔の終了値を入力します。各値は改行で区切って入力する必要があります。

フリー]を使用する場合、 **最小値]**と **最大値]**の指定は任意です。 **最小値]**と **最大値]**を指定した場合は、これらの値がそれぞれ最初の間隔の開始点と最後の間隔の終了点とな

り、入力した値が範囲内に追加の間隔を生成します。入力する値は、**最小値**]で指定された値より大きく、かつ **最大値**]で指定された値以下である必要があります。

- 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

- 省略可能。小計された各数値フィールドの平均値、最小値、および最大値を計算する場合は、**小計フィールドの統計を含める**]を選択します。
統計を含めるには、少なくとも 1 つの小計フィールドを選択する必要があります。
- 出力**]タブをクリックします。
- 出力先**]パネルで、適切な出力オプションを選択します。
 - 画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- 印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- ファイル** - 結果を Analytics テーブルまたはテキスト ファイルに保存または追加するには、このオプションを選択します。Analytics テーブルに保存または追加する場合、そのテーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。テキスト ファイルに保存または追加する場合、ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

- 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - ファイルの種類** - 結果を新しい **Analytics テーブル**に保存する、または既存の Analytics テーブルに追加する場合は、"Analytics テーブル"を選択します。結果をテキスト ファイルに保存または追加する場合は、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル**または **Unicode テキスト ファイル**を選択します。
 - 名前** - **名前**]テキスト ボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでファイル名を入力するか、既存

のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- ローカル-サーバーテーブルに接続されており、Analytics テーブルに結果を保存または追加する場合にのみ有効になります。プロジェクトと同じ場所にファイルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**[ローカル]** を選択します。サーバーの Prefix フォルダーにファイルを保存する場合は、**[ローカル]** を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**[ローカル]** を選択します。結果テーブルを AX Server にインポートするために **[ローカル]** 設定を選択解除することはできません。

- 出力の種類によっては、任意で **[ヘッダー]** および **[フッター]** をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキスト やフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**[ヘッダー]** または **[フッター]** ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

- [詳細]** タブをクリックします。
- [適用範囲]** パネルで、適切な出力オプションを選択します。

- すべて
- 先頭
- 次へ
- While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端

	の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>

15. **最小値と最大値**で指定された値を超える値を含めないようにする場合は、**範囲外を出力しない**]を選択します。
 16. 出力結果を文字フィールドの値に基づいてブレイクダウンさせる場合は、**[ブレイク]** テキスト ボックスにフィールド名を入力するか、**[ブレイク]** をクリックしてフィールドを選択または式を作成します。
- たとえば、売掛金テーブルを決済額で階層化した結果を、さらに顧客別に分類する可能性があるでしょう。**[ブレイク]** で使用できるのは1つの文字フィールドのみであるため、入れ子のブレイクダウンには対応していません。

メモ

[ブレイク] オプションによって有効な結果を得るためには、階層化を実行する前にブレイクダウン対象の文字フィールドを並べ替えておく必要があります。

17. 出力の種類として **[ファイル]** を選択した場合、出力結果を既存ファイルの末尾に追加するときは、次のいずれかを行います。
 - テキスト ファイル、または出力結果とまったく同じ構造であることがわかっている Analytics テーブルに追加する場合は、**既存のファイルに追加する**]を選択します。
 - Analytics テーブルに追加するときに、出力結果と既存のテーブルのレコード長を ACL に比較させたい場合は、**既存のファイルに追加する**]を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する**]を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、**"既存テーブルへの出力結果の追加"** ページ 207を参照してください。

18. 出力の種類として**ファイル(Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く**]を選択します。
19. **OK**]をクリックします。
20. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。
最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**いいえ**]をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

データの年齢調べ

年齢調べは、日付または日付時刻フィールドの値に基づいてテーブル内のレコードを年齢調べ間隔でグループ化し、各期間のレコードの数をカウントします。

年齢調べは、販売傾向の評価、取扱量の調査、未払い日数による請求書のグループ化などによく使用されます。

たとえば、売掛金テーブルの年齢調べを請求日フィールドに基づいて行うことができます。レコードを30日の期間、つまり、締切日から29日前まで、30日前から59日前まで、というようにグループ化し、各期間の未払い請求書の総数を調べます。

メモ

日付時刻の値を対象として年齢調べを行うことはできますが、処理の対象となるのはその値の日付部分のみです。時刻部分は無視されます。時刻データだけで年齢調べを行うことはできません。

数値フィールドの小計

年齢調べを行う場合、任意で1つ以上の数値フィールドを小計できます。上記の例では、請求金額フィールドの小計を計算して、各年齢調べ間隔の未払い請求金額の合計を調べることができます。

年齢調べ間隔の動作

年齢調べ間隔は、次のいずれかの時から遡って測定される日付の間隔(つまり、日数)に基づきます。

- 現在のシステム日付
- 会計期間末日など、指定する締切日

30の単一の日付間隔を指定すると、締切日より30日前、またはそれ以前の日付を含む年齢調べ間隔が作られます。

複数日付の間隔を指定すると、複数の年齢調べ間隔が作られます。年齢調べ間隔の開始点を0、90、120などの日数間隔で指定できますが、デフォルトの設定の0、30、60、90、120、10,000を使用することもできます。

10,000日の間隔、または指定した適切な最後の間隔は、無効と思われる日付が入っているレコードの抽出に使用されます。

以下のテーブルは、5つの年齢調べ間隔を作るために締切日と日付間隔がいかに結合されているか、および各期間に含まれる日付を示したものです。

締切日	日付の間隔				
2016年12月 31日	0	30	60	90	120
	含む: 2016年12月 31日 ~ 2016年12月 02日	含む: 2016年12月 01日 ~ 2016年11月 02日	含む: 2016年11月 01日 ~ 2016年10月 03日	含む: 2016年10月 02日 ~ 2016年9月02 日	含む: 2016年09月 01日 ~ 最も古い日付

統計オプション

[小計フィールドの統計を含める]オプションを使用すると、小計を出す各数値フィールドの平均値、最小値、および最大値を計算することができます。上の例で、統計オプションを使用すると、各年齢調べ間隔の請求金額の平均値、最小値、および最大値が計算されます。

年齢調べの詳細

年齢調べでは、次の操作を実行します。

操作	"年齢調べの結果" 見開きページでの該当箇所
締切日および間隔(日数)に基づいてレコードを年齢調べ間隔ごとにグループ化する	年齢調べ間隔フィールド
年齢調べ間隔ごとにレコード数のカウント(小計)を計算し、その小計が全体に占める割合(カウントの割合)を計算する	カウント フィールド カウントの割合フィールド
レコードの年齢の最低値および最高値(最も新しいおよび最も古い)を提供する	表示されません
任意で、年齢調べ間隔ごとに1つ以上の数値フィールドの値の小計を計算し、最初に選択したフィールドを対象にその小計が全体に占める割合を計算する	Trans Amount フィールド フィールドの割合フィールド
任意で、小計を計算する各数値フィールドの平均値、最小値、および最大値を計算する	表示されません
出力結果に含まれる全数値フィールドの合計を提供する	合計行
任意で、出力結果を文字フィールド(顧客IDやトランザクションタイプなど)の値に基づいてブレイクダウンする(文字フィールドは、年齢調べを実行する前にソートされている必要があります)。	表示されません

年齢調べの結果

次によって生成される出力結果：

- 売掛金テーブルの請求データの年齢調べ
(Ar テーブルは [ACL DATA\Sample Data Files\Sample Project.ACL](#) にあります)
- 取引金額の小計
- 30 日の年齢調べ期間を使用する
- 結果を画面に出力する

経過期間	カウント	カウントの割合	フィールドの割合	Trans Amount
0 - 29	212	27.46%	6.06%	28,422.47
30 - 59	240	31.09%	36.16%	169,527.02
60 - 89	179	23.19%	27.54%	129,133.34
90 - 119	107	13.86%	25.63%	120,153.91
120 - 10,000	34	4.4%	4.62%	21,643.95
合計	772	100%	100%	468,880.69

メモ

結果を画面またはグラフに出力する場合、グラフは、年齢調べ間隔ごとの小計を表示します。あるいは年齢調べ操作で1つ以上の数値の小計フィールドを含める場合は数値の小計を表示します。

手順

テーブル内のレコードを年齢調べ間隔でグループ化することによって、データの年齢を調べることができます。

各期間について、任意で、関連する数値フィールドの小計、平均値、最小値、最大値の計算を含めることができます。

方法の説明

1. **分析 > 年齢調べ**の順に選択します。
2. **メイン**タブで、次のいずれかを行います。
 - **年齢調べの対象**ドロップダウンリストから、年齢調べの対象とするフィールドを選択します。
 - **年齢調べの対象**をクリックして、フィールドを選択するか、または式を作成します。
3. **締切日**フィールドは、デフォルトの現在の日付のままにするか、以下のいずれかの手順を使用して締切日を指定します。
 - **締切日**フィールド内に日付を直接入力します。
 - 下矢印をクリックして、カレンダーから日付を選択します。カレンダーの上部左右にある左矢印または右矢印をクリックすると、一ヶ月単位で過去または未来へ移動します。上部中央で年月、年または10年期間と順にクリックしていく程、長期間の単位で過去または未来へ移動することができます。

締切日の指定が可能なので、年齢調べの最初の期間の期首を会計年度末の日付などに合わせることができます。デフォルトの日付のままにしておくと、年齢調べの最初の期間は現在の日付から開始します。これは、目的とする分析に適切でない場合があります。

4. 使用する年齢調べ間隔を **年齢調べ間隔** テキスト ボックスに入力するか、デフォルトの値をそのまま使用します。

年齢調べ間隔の値は日数で入力する必要があります。各値は、昇順(最近から最も古い順)で1行に1つずつ入力する必要があります。値 '0' は、最初の年齢調べ間隔が指定した締切日から始まることを示します。最後の値は最も古い年齢調べ間隔の期末を示します。

メモ

デフォルトの年齢調べ間隔として設定される値は、**ツール > オプション** の順にクリックし、**日付と時刻** タブの **年齢調べ間隔** の数値を更新することによって変更できます。

5. 省略可能。 **小計フィールド** を1つ以上選択するか、または **小計フィールド** ボタンをクリックして、小計フィールドを選択するか式を作成します。

小計フィールドを選択する順序は、列が結果に表示される順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

6. 現在のビューの中に処理から除外したいレコードがある場合は、 **[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

7. 省略可能。小計された各数値フィールドの平均値、最小値、および最大値を計算する場合は、 **小計フィールドの統計を含める** を選択します。

統計を含めるには、少なくとも1つの小計フィールドを選択する必要があります。

8. **出力** タブをクリックします。
9. **出力先** パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- 印刷 - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- グラフ - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- ファイル - 結果をテキスト ファイルに保存または追加するには、このオプションを選択します。ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

- 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは、使用している Analytics のエディションに応じて、"ASCII テキスト ファイル" または "Unicode テキスト ファイル" のいずれかのみです。結果を新しいテキスト ファイルに保存するか、または既存のテキスト ファイルに追加します。
 - **名前** - **名前**]テキスト ボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

- **ローカル** - 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。
- 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー**]または **フッター**]ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

- 詳細**]タブをクリックします。
- 適用範囲**]パネルで、適切な出力オプションを選択します。

- **すべて**
- **先頭**
- **次へ**
- **While**

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードか

	ら処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
	<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>

- 指定した年齢調べ間隔に該当しない値を出力結果から除外したい場合は、**範囲外を出力しない**] チェック ボックスをオンにします。
- 出力結果を文字フィールドの値に基づいてブレイクダウンさせる場合は、**[ブレイク]** テキスト ボックスにフィールド名を入力するか、**[ブレイク]** をクリックしてフィールドを選択または式を作成します。

たとえば、売掛金テーブルに対し一定期間の集計を行う場合、顧客別に分類する可能性があります。 **[ブレイク]** で使用できるのは 1 つの文字フィールドのみであるため、入れ子のブレイクダウンには対応していません。

メモ

[ブレイク] オプションによって有効な結果を得るためには、年齢調べを実行する前に文字フィールドを並べ替えておく必要があります。

- 出力の種類として **[ファイル]** を選択した場合、出力結果を既存のテキスト ファイルの末尾に追加するときは、**既存のファイルに追加する**] を選択します。
- [OK]** をクリックします。

結果を画面またはグラフに出力する場合、表示領域の下部にある **[テキスト]** ボタンと **[グラフ]** ボタンを使用して出力タイプを切り替えることができます。

- 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

分類化と要約の比較

分類化と要約は、データのグループ化方法は類似していますが、それらは異なるオプションを持ち、データの処理方法も異なります。テーブルがソートされているかどうかに関係なく分類または要約できます。

次のいずれかを実行する場合は、分類化ではなく要約を行う必要があります。

- 日付時刻キーフィールドの操作
- 複数のキーフィールドの使用
- キーフィールド以外のフィールドを出力テーブルに含める

要件	分類化	要約
テーブルにおけるキーフィールド値の出現回数を計算して報告する	はい	はい
選択した数値フィールドの小計を計算して表示する	はい	はい
小計を計算する数値フィールドの平均値、最小値、および最大値を計算して表示する	はい	はい
パーセンテージを計算して表示する	はい	はい
小計された数値フィールドの追加の統計値を計算し、表示する (標準偏差、中央値、最頻値、第1および第3四分位数)	いいえ	はい
キーフィールドは文字でも可	はい	はい
キーフィールドは数値でも可	はい	はい
キーフィールドは日付時刻でも可	いいえ	はい
複数のキーフィールドが使用可能	いいえ	はい
キーフィールド以外のフィールドを出力に含める	いいえ	はい
処理を行う主な場所	RAM(ランダムアクセスメモリ)	ハードディスク
キーフィールドの長さの制限	はい (最大 2048 文字)	いいえ
出力結果のソート	はい	はい(あらかじめ並べ替える]がオン) いいえ(あらかじめ並べ替える]がオフ)
テーブル、画面、または印刷への出力	はい	はい
グラフへの出力	はい	いいえ

データの分類化

分類化は、等しいキーフィールドの値に基づいてテーブル内のレコードをグループ化し、各グループのレコード数をカウントします。キーフィールドには文字または数値が可能です。

たとえば、取引テーブルを顧客番号フィールドに基づいて分類化し、各顧客の総取引数を調べることができます。

以下の例では、入力テーブルの顧客番号]フィールドに10個の値があります。一意の値もあれば、等しい値もあります。要約後、値は4つの一意のグループにグループ化されます。カウントにより、レコードまたは取引がいくつ各顧客番号グループにあるかが分かります。

入力テーブル	出力結果	
キーフィールド: 顧客番号	分類されたグループ	カウント
795401	230575	1
518008	518008	5
518008	795401	3
925007	925007	1
518008		
795401		
518008		
230575		
795401		
518008		

関連付けられた数値フィールドの小計

分類すると、任意で1つ以上の関連付けられた数値フィールドを小計できます。上記の例では、取引金額フィールドの小計を計算して、各顧客の総取引額を調べることができます。

分類化および並べ替え

テーブルがソートされているかどうかに関係なく分類できます。ソートされていないテーブルを分類する場合、Analytics は分類化作業の一環で出力結果を自動的に並べ替えます。

並べ替えられていないテーブルを分類化する場合、Analytics では、キーフィールドの等しい値のセットごとに変数を作成し、テーブル全体が読み込まれるまでこれらの変数をメモリに格納する必要があります。大きいテーブルを使用して作業する場合、必要な変数の並べ替えを行うには多くの RAM が必要なので、実行速度が低下する可能性があります。

統計オプション

[小計フィールドの統計を含める] オプションでは、指定する小計フィールドで平均値、最小値、および最大値を計算することができます。計算結果は、分類された出力テーブルでグループ別に内訳が示されます。

上の例では、統計オプションで、各顧客の取引金額の平均値、最小値、および最大値が計算されます。

分類化の詳細

分類化では、次の操作を実行します。

操作	"分類化の結果" 次のページでの該当箇所
文字または数値フィールドの等しい値に基づいてレコードをグループ化する	Product Class フィールド
グループごとにレコード数のカウント(小計)を行い、その小計が全体に占める割合(カウントの割合)を計算する	カウント フィールド カウントの割合 フィールド
任意で、グループごとに1つ以上の数値フィールドの値の小計を計算し、最初に選択した数値フィールドを対象にその小計が全体に占める割合を計算する	原価の在庫金額フィールド フィールドの割合 フィールド
任意で、小計を計算する各数値フィールドの平均値、最小値、および最大値を計算する	表示されません
出力結果に含まれる全数値フィールドの合計を提供する	合計行
任意で、出力結果を文字フィールド(顧客 ID やトランザクションタイプなど)の値に基づいてブレイクダウンする(文字フィールドは、分類化を実行する前にソートされている必要があります)。	表示されません

分類化の結果

次によって生成される出力結果：

- 棚卸資産テーブルの製品クラス分類
(Inventory テーブルは [ACL DATA\Sample Data Files\Sample Project.ACL](#) にあります)
- 棚卸資産価格の小計
- 結果を画面に出力する

結果は、03、04、08、09 の 4 製品クラスに在庫金額が集中していることを示しています。

Product Class	カウント	カウントの割合	フィールドの割合	原価の在庫金額
01	17	11.18%	5.14%	34,954.68
02	19	12.5%	3.02%	20,544.20
03	20	13.16%	15.09%	102,702.76
04	17	11.18%	13.08%	89,018.95
05	13	8.55%	6.24%	42,479.36
06	17	11.18%	8.59%	58,479.60
07	7	4.61%	7%	47,609.10
08	19	12.5%	27.66%	188,230.86
09	21	13.82%	11.85%	80,646.05
13	1	0.66%	1.67%	11,352.48
18	1	0.66%	0.66%	4,461.90
合計	152	100%	100%	680,479.94

手順

文字または数値フィールドの等しい値に基づいてテーブル内のレコードをグループ化することによって、データを分類化することができます。

各グループについて、任意で、関連する数値フィールドの小計、平均値、最小値、最大値の計算を含めることができます。

方法の説明

メモ

分類化は、最大 2048 文字のキーフィールド長をサポートしています。

2048 文字より長いキーフィールドを使用してテーブルを分類化したい場合は、長さの制限がない要約を使用することができます。詳細については、「データの要約」ページ 1326 を参照してください。

大きいデータセットを分類し、結果を画面またはグラフに出力する場合は、使用可能なメモリ量を超える場合があります。画面に出力するときにメモリ使用量を減らすには、**コマンド結果の XML 出力を生成しない**(**ツール > オプション > コマンド**) を選択します。

1. **分析 > 分類化**] の順に選択します。
2. **メイン**] タブで、次のいずれかを行います。
 - **分類化の対象**] ドロップダウンリストから、分類の対象となるフィールドを選択します。
 - **分類化の対象**] をクリックして、フィールドを選択するか、または式を作成します。
3. 省略可能。 **小計フィールド**] を 1 つ以上選択するか、または **小計フィールド**] ボタンをクリックして、小計フィールドを選択するか式を作成します。

小計フィールドを選択する順序は、列が結果に表示される順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。
4. 現在のビューの中に処理から除外したいレコードがある場合は、 **[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、 **式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

5. 省略可能。小計された各数値フィールドの平均値、最小値、および最大値を計算する場合は、 **小計フィールドの統計を含める**] を選択します。

統計を含めるには、少なくとも 1 つの小計フィールドを選択する必要があります。
6. **出力**] タブをクリックします。
7. **出力先**] パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果を Analytics テーブルに保存または追加するには、このオプションを選択します。テーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

8. 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは "Analytics テーブル" のみです。結果を新しい Analytics テーブルに保存するか、または既存の Analytics テーブルに追加します。
 - **名前** - **名前**]テキスト ボックスにテーブル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでテーブル名を入力するか、既存のテーブルに上書きまたは追加する場合はそのテーブルを選択します。Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にテーブルを保存したり、その場所にあるテーブルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- **ローカル** - サーバーテーブルに接続している場合のみ有効になります。プロジェクトと同じ場所に出力テーブルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル**]を選択します。サーバーの Prefix フォルダーに出力テーブルを保存する場合は、**ローカル**]を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**ローカル**]を選択します。結果テーブルを AX Server にインポートするために **ローカル**]設定を選択解除することはできません。

9. 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー**]または **フッター**]ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

10. **詳細**]タブをクリックします。

11. **適用範囲]** パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While] テキスト ボックスに条件を入力するか、または While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、"すべて"、"先頭"、または"次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

12. 出力結果を文字フィールドの値に基づいてブレイクダウンさせる場合は、**ブレイク]** テキスト ボックスにフィールド名を入力するか、**ブレイク]** をクリックしてフィールドを選択または式を作成します。

たとえば、売掛金テーブルを決済タイプで分類化した結果を、さらに顧客別に分類する可能性があるでしょう。**ブレイク]** で使用できるのは1つの文字フィールドのみであるため、入れ子のブレイクダウンには対応していません。

メモ

ブレイク] オプションによって有効な結果を得るためには、分類化を実行する前にブレイクダウン対象の文字フィールドを並べ替えておく必要があります。

13. 出力の種類として**ファイル (Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く]**を選択します。

14. 出力の種類として**ファイル]**を選択した場合、出力結果を既存の Analytics テーブルの末尾に追加するときは、次のいずれかを行います。

- 出力結果と既存のテーブルの構造がまったく同じであることがわかっている場合は、**既存のファイルに追加する]**を選択します。
- 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、**既存のファイルに追加する]**を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する]**を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、「既存テーブルへの出力結果の追加」ページ 207を参照してください。

15. **OK]**をクリックします。
16. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。
最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**[いいえ]**をクリックして操作をキャンセルし、「既存テーブルへの出力結果の追加」ページ 207を参照してください。

データの要約

要約は、1つまたは複数のキーフィールドの等しい値に基づきテーブルのレコードをグループ化し、各グループのレコード数をカウントします。また、各グループに各種の統計計算を実行するオプションもあります。

複数のキーフィールド別に要約する場合は(入れ子の要約)、グループはキーフィールド全体の値の等しい組み合わせに基づきます。

キーフィールドには、文字、数値、または日付時刻があります。

1つのキーフィールドによる要約

1つのキーフィールドによる要約は、最も容易な形の要約です。

たとえば、取引テーブルを顧客番号フィールドに基づいて要約し、顧客別の総取引数を調べることができます。

以下の例では、入力テーブルの「顧客番号」フィールドに10個の値があります。一意の値もあれば、等しい値もあります。要約後、値は4つの一意のグループにグループ化されます。カウントにより、レコードまたは取引がいくつ各顧客番号グループにあるかが分かります。

入力テーブル	出力結果	
キーフィールド: 顧客番号	要約されたグループ	カウント
795401	230575	1
518008	518008	5
518008	795401	3
925007	925007	1
518008		
795401		
518008		
230575		
795401		
518008		

複数のキーフィールドによる要約

複数のキーフィールドによる要約、または入れ子の要約により、データをさらに詳細に分析することができます。

たとえば、取引テーブルを顧客番号フィールドと取引日フィールドに基づいて要約し、顧客別に、その顧客が取引を行った日付ごとの総取引数を調べることができます。

以下の例では、入力テーブルの「顧客番号」フィールドに10個の値があり、「請求日付」フィールドに付随する日付があります。顧客番号と日付の組み合わせは、一意であるものもあれば、等しいものもあります。要約後、顧客番号と日付の組み合わせは7つの一意のグループに分けられます。カウントにより、レコードまたは取引がいくつ各グループにあるかが分かります。

入力テーブル		出力結果		
キーフィールド 1: 顧客番号	キーフィールド 2: 請求日付	入れ子の要約されたグループ		カウント
795401	08/20/2016	230575	06/13/2016	1
518008	10/15/2016	518008	04/30/2016	1
518008	07/17/2016	518008	07/17/2016	3
925007	05/21/2016	518008	10/15/2016	1
518008	04/30/2016	795401	06/30/2016	1
795401	08/20/2016	795401	08/20/2016	2
518008	07/17/2016	925007	05/21/2016	1
230575	06/13/2016			
795401	06/30/2016			
518008	07/17/2016			

詳細な入れ子の要約

複数のキーフィールドで要約する場合、出力結果で入れ子になった要約されたグループを作成します。

入れ子の階層

キーフィールドを選択する順序が、入れ子の階層を決定します。レコードは、まず、1番目に選択したフィールド別に要約され、次にこれら一次グループのそれぞれで、2番目に選択したフィールド別に要約さ

れます。

メモ

この2つの要約キーフィールドの選択順序を逆にすると、まったく異なる結果が得られる場合があります。

出力結果のフィールドの順序

キーフィールドを選択する順序が、出力結果に表示される列の順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

要約と並べ替え

要約では、データは並べ替えられていなくても処理できます。**あらかじめ並べ替える**] オプションにより、要約時にデータの初回の並べ替えを含めることができます。

あらかじめ並べ替える]を使用する場合

あらかじめ並べ替える]を使用する場合、出力結果は並べ替えられており、キーフィールド内の同一値セットごと、または値の等しい組み合わせごとに1つの一意のグループが含まれます。

ヒント

入力テーブルがすでに並べ替えられている場合は、**あらかじめ並べ替える**]オプションを選択解除して処理時間を節約することができます。

あらかじめ並べ替える]を使用しない場合

あらかじめ並べ替える]を使用しない場合、出力結果は、入力テーブルの並べ替え順序を使用します。

キーフィールドに入っている同一の値が連続して並んでいない場合、出力結果には、同一値セットごと、または値の等しいの組み合わせごとに2つ以上のグループが含まれることとなります。

メモ

コンテキストによっては、同一値セットごと、または値の等しい組み合わせに2つ以上のグループが含まれることが、要約の目的よりも優先されることがあります。

数値フィールドの小計

要約すると、任意で1つ以上の数値フィールドを小計できます。上記の例では、計算する取引金額フィールドを小計できます。

- 顧客ごとの総取引額
- 顧客が取引した各日付の顧客ごとの総取引額

統計オプション

指定する小計フィールドで統計演算を実行するオプションがあります。統計計算は、出力テーブルでグループ別に内訳が示されます。

上記の例では、取引金額フィールドを小計する場合、計算する統計オプションの1つを使用することもできます。

- 顧客ごとの平均、最小、および最大の取引金額
- 顧客が取引した各日付の顧客ごとの平均、最小、および最大の取引金額

詳細な小計と統計オプション

以下のテーブルは、小計と統計のオプションと計算に関する詳細を示したものです。

詳しく表示

オプション	出力テーブルの列見出し (表示名)	出力テーブルのフィールド名	小計フィールドで実行された計算
小計フィールド	Total + 小計対象となる代替列見出し	小計対象となるフィールド名	各グループの小計された値
平均、最小、最大	Average + 小計対象となる代替列見出し	a_ 小計対象となるフィールド名	各グループの平均値
	Minimum + 小計対象となる代替列見出し	m_ 小計対象となるフィールド名	各グループの最小値
	Maximum + 小計対象となる代替列見出し	x_ 小計対象となるフィールド名	各グループの最大値
標準偏差、フィールドの%	STDDEV + 小計対象となる代替列見出し	d_ 小計対象となるフィールド名	各グループの標準偏差
	% Field + 小計対象となる代替列見出し	f_ 小計対象となるフィールド名	フィールドの合計の割合として表現される各グループの小計
中央値、最頻値、Q25、	Median + 小計対象となる	c_ 小計対象となるフィールド	各グループの中央値

オプション	出力テーブルの列見出し (表示名)	出力テーブルのフィールド名	小計フィールドで実行された計算
Q75	代替列見出し	ド名	<ul style="list-style-type: none"> 奇数の値セット: 中央値 偶数の値セット: 中央にある2つの値の平均
	Mode + 小計対象となる代替列見出し	o_小計対象となるフィールド名	各グループの最も頻繁に発生する値 <ul style="list-style-type: none"> 2回以上出現する値がない場合は、"N/A"が表示される 関連付けの場合は、最も低い値が表示される
	Q25 + 小計対象となる代替列見出し	q_小計対象となるフィールド名	各グループの最初の四分位数(下四分位数値) <ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
	Q75 + 小計対象となる代替列見出し	p_小計対象となるフィールド名	各グループの3番目の四分位数(下四分位数) <ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
カウントの%	パーセントカウント	COUNT_PERCENTAGE	各グループに属するソーステーブルレコードの割合 <p>メモ 小計フィールドは必要ありません。</p>

その他のフィールド オプション

「その他のフィールド」では、出力に含めたい追加の文字フィールド、数値フィールド、または日付時刻フィールドを選択できます。選択したフィールドに含まれる値が、要約される各グループの全レコードに対応するものであれば、このオプションによって有用な情報を提供できます。

たとえば、テーブルを顧客番号で要約する場合、該当する「その他のフィールド」は顧客名となります。顧客名は、同じ顧客番号を持つすべてのレコードで同じです。

要約されるグループに対し異なった値を含んでいるフィールドを「その他のフィールド」として指定すると、グループの最初のレコードの値のみが表示され意味がありません。

たとえば、業者テーブルを”州”別に要約し、「その他のフィールド」として”市”を選択すると、出力で、各州に対して表示される市が最初の1つのみとなります。このような場合、州と市の両方をキーフィールドとして(州、市の順番で)要約を行うことをお勧めします。

要約の対象

以下の例は、顧客番号と取引の種類に関して売掛金テーブルを要約した結果を示したものです。取引金額が一部の関連付けられた統計と共に小計されています。結果が画面に出力されています。

この例では、ACL DATA\Sample Data Files\Sample Project.ACLにおけるArテーブルからの顧客番号のサブセットを使用しています。

Cust Number	Trans Type	合計 Trans Amount	平均 Trans Amount	最小 Trans Amount	最大 Trans Amount	パーセントカウント	カウント	Name
051593	CN	-73.40	-73.40	-73.40	-73.40	0.80	1	CONNECTICUT CORP.
051593	IN	1,189.11	1,189.11	1,189.11	1,189.11	0.80	1	CONNECTICUT CORP.
056016	IN	1,807.66	903.83	736.74	1,070.92	1.60	2	CITIZENS INTERNATIONAL
056016	PM	-1,807.66	-903.83	-1,070.92	-736.74	1.60	2	CITIZENS INTERNATIONAL
065003	CN	-685.59	-52.74	-146.83	-9.17	10.40	13	UNIVERSITY ELECTRONICS
065003	IN	105,020.57	1,207.13	73.40	4,954.64	69.60	87	UNIVERSITY ELECTRONICS
065003	PM	-8,443.97	-562.93	-1,954.88	116.72	12.00	15	UNIVERSITY ELECTRONICS
081559	IN	1,779.07	1,779.07	1,779.07	1,779.07	0.80	1	KIDDER ENTERPRISES
090398	IN	634.38	317.19	55.02	579.36	1.60	2	AMSER SYSTEMS
097627	IN	1,301.83	1,301.83	1,301.83	1,301.83	0.80	1	STEPPING ELECTRONICS
合計		100,722.00	5,105.26	1,889.14	10,172.34	100.00	125	

要約の詳細

要約では、次の操作を実行します。

操作	上記の"要約の対象"上の場所
1つ以上の文字、数値、または日付時刻のキーフィールドで、等しい値、または値の等しい組み合わせに基づきレコードをグループ化する	顧客番号 フィールド 取引タイプ フィールド

操作	上記の"要約の対象"上の場所
任意で、グループごとに1つ以上の数値フィールドの小計を計算する	総取引額 フィールド
任意で、各小計された数値フィールドで統計計算を実行します	平均、最小、および最大取引金額フィールド <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> メモ 追加統計計算が表示されません </div>
各グループに属するソーステーブルレコードの割合を任意で計算する	カウントの割合フィールド
グループごとにレコード数をカウントする(小計する)	カウント フィールド
任意で、補足的な情報として文字フィールド、数値フィールド、または日付時刻フィールドを追加表示する	Name フィールド
出力結果に含まれる全数値フィールドの合計を提供する <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> メモ 合計]行は、結果が画面に出力される場合のみ提供されます。 </div>	合計行

手順

1つ以上の文字フィールド、数値フィールド、または日付時刻フィールドの等しい値、または値の等しい組み合わせに基づいてテーブル内のレコードをグループ化することによって、データを要約することができます。

任意で関連付けられた数値フィールドを小計できます。指定する小計フィールドで統計演算を実行することもできます。統計演算の結果は、集計された出力テーブルでグループ別に内訳が示されます。

方法の説明

1. **分析 > 要約**]の順に選択します。
2. **メイン**]タブで次のいずれかを実行します。
 - **要約の対象**]リストから、要約するフィールドを選択します。
 - **要約の対象**]をクリックして、フィールドを選択するか、式を作成します。

メモ

複数のフィールドを選択した場合、入れ子になった要約グループが出力結果に作成されます。詳細については、「詳細な入れ子の要約」ページ 1327を参照してください。

3. 省略可能。 **小計フィールド**]を1つ以上選択するか、または **小計フィールド**]ボタンをクリックして、小計フィールドを選択するか式を作成します。

小計フィールドを選択する順序は、列が結果に表示される順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

4. 省略可能。次のいずれかを実行します。
 - **その他のフィールド**] リストから、出力結果に含めるその他のフィールドを選択します。
 - **その他のフィールド**] をクリックして、フィールドを選択するか、または式を作成します。

メモ

各要約グループ内のすべてのレコードが同じ値を含んでいるフィールドに対してのみを選択してください。詳細については、"**その他のフィールド オプション**" ページ 1330を参照してください。

5. 要約対象のフィールドが既にソート済みであれば、**あらかじめ並べ替える**] を選択解除することもできます。

メモ

並べ替えられていないフィールドの要約も可能ですが、同じ値に対して要約されたグループが複数生じる結果となり、要約する意味がなくなってしまう可能性があります。

分析の性質によっては、ソートされていないフィールドを要約することが適切な場合もあります。

6. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

7. 省略可能。小計フィールドの統計計算を行うために、統計オプションを1つ以上選択します：
 - **平均、最小、最大**
 - **標準偏差、フィールドの%**
 - **中央値、最頻値、Q25、Q75**
 - **カウントの%**

詳細については、"**統計オプション**" ページ 1329を参照してください。

メモ

統計情報を含めるには、少なくとも 1 つの小計フィールドを選択する必要があります。

カウントの % では、小計フィールドは必要ありません。

これらの統計を計算するには、追加のコンピューターメモリが必要です。非常に大きいデータセットの統計を計算すると、コンピューターのメモリを超過し、エラーメッセージが表示される場合があります。

8. **出力]** タブをクリックします。
9. **出力先]** パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果を Analytics テーブルに保存または追加するには、このオプションを選択します。テーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

10. 出力の種類として **ファイル]** を選択した場合は、**出力指定]** パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは "Analytics テーブル" のみです。結果を新しい Analytics テーブルに保存するか、または既存の Analytics テーブルに追加します。
 - **名前** - **名前]** テキスト ボックスにテーブル名を入力します。もしくは、**名前]** ボタンをクリックして、**保存]** または **ファイルを保存する名前]** ダイアログ ボックスでテーブル名を入力するか、既存のテーブルに上書きまたは追加する場合はそのテーブルを選択します。Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にテーブルを保存したり、その場所にあるテーブルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- ローカル-サーバーテーブルに接続している場合のみ有効になります。プロジェクトと同じ場所に出力テーブルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル]**を選択します。サーバーの Prefix フォルダーに出力テーブルを保存する場合は、**ローカル]**を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**ローカル]**を選択します。結果テーブルを AX Server にインポートするために **ローカル]**設定を選択解除することはできません。

- 出力の種類によっては、任意で **ヘッダー]** および **フッター]** をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキスト やフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー]** または **フッター]** ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミicolon (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

- 詳細]** タブをクリックします。
- 適用範囲]** パネルで、適切な出力オプションを選択します。

- すべて
 - 先頭
 - 次へ
 - While
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While] テキスト ボックスに条件を入力するか、または While] ボタンをクリックし、式ビルダー を利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。While オプションは、「すべて」、「先頭」、または「次」オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>

メモ

"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。

ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。

14. 出力の種類として**ファイル (Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く**]を選択します。
15. 出力タイプとして **ファイル**]を選択し、出力テーブルを既存の Analytics テーブルの最後に追加する場合は、次のいずれかを実行します。
 - 出力結果と既存のテーブルの構造が同じであることが確実な場合は、**既存のファイルに追加**]を選択します。
 - 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、**既存のファイルに追加する**]を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する**]を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

16. **OK**]をクリックします。
17. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**いいえ**]をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

データのクロス集計

クロス集計は、2つ以上のキーフィールドの値の、等しい組み合わせに基づいてテーブル内のレコードをグループ分けし、各グループのレコード数をカウントします。キーフィールドには文字または数値が可能です。

結果として生じるグループが、ピボットテーブルに似た行と列のグリッドで表示されることにより、データ内の関係やパターンが視覚化されます。

たとえば、在庫テーブルを**製品の場所**と**製品クラス**のフィールドに基づいてクロス集計し、各場所におけるクラス別のレコードの数を調べることができます。

キーフィールド 1 (製品の場所)	キーフィールド 2 (製品クラス)	クロス集計グループ (製品の場所と製品クラス)		レコード数
A-01	17	A-01	16	1
F-19	22	A-01	17	3
F-19	08	B-03	17	2
A-01	16	F-19	22	2
B-03	17	F-19	08	1
Q-28	03	Q-28	03	1
A-01	17			
F-19	22			
A-01	17			
B-03	17			

数値フィールドの小計

クロス集計すると、任意で1つ以上の数値フィールドを小計できます。上記の例では、在庫金額フィールドを小計して、各場所の製品クラスごとに合計在庫金額を見つけることができます。

クロス集計は要約と似ています。

クロス集計は、2つのフィールドを使用する要約と似ています。両方の操作で、出力結果のカウントや小計は同じですが、その情報の配置は異なります。

クロス集計では、カウントや小計がゼロ(0)でも表示しますが、要約では表示しません。実施しようとする分析の種類によっては、カウントや小計のゼロ表示が有用になることがあります。

クロス集計と並べ替え

テーブルは、並べ替えられているかどうかに関係なくクロス集計できます。並べ替えられていないテーブルをクロス集計すると、Analytics はクロス集計操作の一環として出力結果を自動的に並べ替えます。

クロス集計の詳細

クロス集計では、次の操作を実行します。

操作	"クロス集計の結果" 下での該当箇所
2つ以上の文字または数値フィールドの値の等しい組み合わせに基づいてレコードをグループ化し、そのグループを行と列のグリッドに表示する	Cust Number フィールド (行) と Type フィールド (列) の交差位置
任意で、グループごとに1つ以上の数値フィールドの小計を計算する	Amount フィールド
任意で、グループごとにレコード数の小計を計算する	カウント フィールド
<p>メモ</p> <p>小計フィールドが何も選択されていない場合は、自動的にカウントが含まれます。</p>	
出力結果に含まれる全列の合計を提供する	合計行

クロス集計の結果

次によって生成される出力結果:

- 売掛金テーブル
(ACL DATA\Sample Data Files\Sample Project.ACL の Ar テーブル) の顧客番号と決済タイプを対象にクロス集計を実行した結果の例です。
- 取引金額の小計
- 結果を画面に出力する

Cust Number	Amount Type CN	カウント CN	Amount Type IN	カウント IN	Amount Type PM	カウント PM
051593	-73.40	1	1,189.11	1	0.00	0
056016	0.00	0	1,807.66	2	-1,807.66	2
065003	-685.59	13	105,020.57	87	-8,443.97	15
081559	0.00	0	1,779.07	1	0.00	0
090398	0.00	0	634.38	2	0.00	0
097627	0.00	0	1,301.83	1	0.00	0
113236	0.00	0	681.93	1	0.00	0
176437	-241.49	3	14,825.62	13	-1,779.01	2
202028	-26.60	1	1,767.74	3	0.00	0
207275	0.00	0	3,678.68	3	0.00	0
合計	-1,027.08	18	132,686.59	114	-12,030.64	19

手順

2つ以上の文字または数値フィールドの値の等しい組み合わせに基づいてテーブル内のレコードをグループ化することによって、データをクロス集計することができます。

結果として生じるグループが、ピボットテーブルに似た行と列のグリッドで表示されることにより、データ内の関係やパターンが視覚化されます。

方法の説明

1. **分析 > クロス集計**]の順に選択します。
2. **メイン**]タブで、次のいずれかを行います。
 - **行**]リストから、行として表示するフィールドを選択します。
 - **行**]をクリックして、フィールドを選択するか、または式を作成します。

複数のフィールドを選択すると、出力結果に入れ子の追加レベルが作成されます(1行と1列を使用するクロス集計は、既に入れ子の一形態です)。フィールドを選択する順序が、入れ子の階層を決定します。レコードは、まず、1番目に選択したフィールド別にクロス集計され、次にこれら一次グループのそれぞれで、2番目に選択したフィールド別にクロス集計されます。この2つのフィールドの選択順序を逆にすると、まったく異なる結果が得られます。

フィールドを選択する順序は、フィールドが結果に表示される順序でもあります。既存のAnalyticsテーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

3. 次のいずれかを実行します。
 - **列**]のドロップダウンリストから、列として表示するフィールドを選択します。
 - **列**]をクリックして、フィールドを選択するか、または式を作成します。
4. 省略可能。 **小計フィールド**]を1つ以上選択するか、または **小計フィールド**]ボタンをクリックして、小計フィールドを選択するか式を作成します。

小計フィールドを選択する順序は、列が結果に表示される順序になります。既存の Analytics テーブルに結果を追加する場合は、結果と既存テーブル内の列の選択と順序は同一である必要があります。

- 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

- 行と列の交差ごとのレコード数のカウントを含める場合は、**カウントを含める]**を選択します。小計フィールドが何も選択されていない場合は、自動的にカウントが含まれます。
- 出力]**タブをクリックします。
- 出力先]**パネルで、適切な出力オプションを選択します。
 - 画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソース テーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- 印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- ファイル** - 結果を Analytics テーブルまたはテキスト ファイルに保存または追加するには、このオプションを選択します。Analytics テーブルに保存または追加する場合、そのテーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。テキスト ファイルに保存または追加する場合、ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

- 出力の種類として **ファイル]**を選択した場合は、**出力指定]**パネルで次の情報を指定します。
 - ファイルの種類** - 結果を新しい **Analytics テーブル]**に保存する、または既存の Analytics テーブルに追加する場合は、"Analytics テーブル"を選択します。結果をテキスト ファイルに保存または追加する場合は、使用している Analytics のエディションに応じて、**ASCII テキスト ファイル]**または **Unicode テキスト ファイル]**を選択します。
 - 名前** - **名前]**テキスト ボックスにファイル名を入力します。もしくは、**名前]**ボタンをクリックして、**保存]**または **ファイルを保存する名前]**ダイアログ ボックスでファイル名を入力するか、既存

のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- ローカル-サーバーテーブルに接続されており、Analytics テーブルに結果を保存または追加する場合にのみ有効になります。プロジェクトと同じ場所にファイルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル**]を選択します。サーバーの Prefix フォルダーにファイルを保存する場合は、**ローカル**]を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**ローカル**]を選択します。結果テーブルを AX Server にインポートするために **ローカル**]設定を選択解除することはできません。

- 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキスト やフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー**]または **フッター**]ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

- 詳細**]タブをクリックします。
- 適用範囲**]パネルで、適切な出力オプションを選択します。

- すべて
- 先頭
- 次へ
- While

詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端

	の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 [While] テキスト ボックスに条件を入力するか、または [While] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
	<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>

13. 出力の種類として**ファイル(Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く]**を選択します。
14. 出力の種類として **[ファイル]**を選択した場合、出力結果を既存ファイルの末尾に追加するときは、次のいずれかを行います。
 - テキスト ファイル、または出力結果とまったく同じ構造であることがわかっている Analytics テーブルに追加する場合は、**既存のファイルに追加する]**を選択します。
 - Analytics テーブルに追加するときに、出力結果と既存のテーブルのレコード長を ACL に比較させたい場合は、**既存のファイルに追加する]**を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する]**を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、**"既存テーブルへの出力結果の追加"** ページ 207を参照してください。

15. **[OK]**をクリックします。
16. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

最後に追加] オプションが表示されることを期待していたのに表示されない場合は、**[いいえ]**をクリックして操作をキャンセルし、**"既存テーブルへの出力結果の追加"** ページ 207を参照してください。

ヒストグラムの作成

ヒストグラムを作成すると、テーブル内のレコードをグループ化し、各グループ内のレコード数をカウントして、そのグループとカウントを垂直棒グラフで表示します。

レコードをグループ化することができます。

- 文字フィールドの等しい値に基づく([分類化](#)と類似)
- 等しいサイズ、またはカスタムのサイズの数値間隔に([階層化](#)と類似)

一部の出力書式に利用できるブレイクフィールド

棒グラフのほかに、画面、テキストファイル、または印刷に結果を出力することもできます。これらの形式に結果を出力する場合は、任意で、顧客 ID や取引タイプなどの文字フィールドの値に基づいて、結果をブレイクダウンすることができます。このオプションによって有効な結果を得るためには、ヒストグラムを作成する前に、ブレイクダウン対象の文字フィールドを並べ替えておく必要があります。

数値フィールドの小計

Analytics 内の他のグループ化操作とは異なり、ヒストグラムは数値フィールドの小計をサポートしていません。

ヒストグラムと並べ替え

ヒストグラムは、テーブルが並べ替えられていなくても作成できます。並べ替えられていないテーブルを使用した場合でも、Analytics により出力結果がヒストグラム作成の一環として自動的に昇順に並べ替えられます。

手順

テーブル内のレコードをグループ化し、そのグループを棒グラフで表示する、ヒストグラムを作成することができます。

レコードをグループ化することができます。

- 文字フィールドの等しい値に基づく
- 等しいサイズ、またはカスタムのサイズの数値間隔に

方法の説明

1. **分析 > ヒストグラム]**の順に選択します。
2. **メイン]**タブで、次のいずれかを行います。
 - **ヒストグラムの対象]**ドロップダウンリストから、ヒストグラムの対象とするフィールドを選択します。
 - **ヒストグラムの対象]**をクリックしてフィールドを選択するか、式を作成します。
3. **ヒストグラムの対象]**で数値フィールドまたは式を選択した場合は、以下のことを行います。
 - a. **最小値]**で、最初の間隔の最小値を入力します。

数値フィールドに対して事前にプロファイルまたは統計を実行している場合は、フィールドの最小値がデフォルトで入力されています。必要に応じて、デフォルトを変更できます。
 - b. **最大値]**で、最後の間隔の最大値を入力します。

数値フィールドに対して事前にプロファイルまたは統計を実行している場合は、フィールドの最大値がデフォルトで入力されています。必要に応じて、デフォルトを変更できます。
4. **ヒストグラムの対象]**で数値フィールドまたは式を選択した場合は、以下のいずれかを行います。
 - **間隔]**を選択し、**最小値]**と**最大値]**によって指定された範囲内に必要な、均等な間隔の数を入力します。デフォルトの間隔数は10です。

ヒント

デフォルトの間隔数は、**ツール > オプション]**の順にクリックして **コマンド]**タブを選択し、**間隔の数]**を更新することによって変更できます。

- カスタムサイズの間隔を作成するには **フリー]**を選択し、各間隔の開始値および最後の間隔の終了値を入力します。各値は改行で区切って入力する必要があります。

フリー]を使用する場合、**最小値]**と**最大値]**の指定は任意です。**最小値]**と**最大値]**を指定した場合は、これらの値がそれぞれ最初の間隔の開始点と最後の間隔の終了点となり、入力した値が範囲内に追加の間隔を生成します。入力する値は、**最小値]**で指定された値より大きく、かつ**最大値]**で指定された値以下である必要があります。
5. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキストボックスに条件を指定します。直接入力するか、または **[f]**ボタンをクリックし、**式ビルダー]**を利用してIFステートメントを作成します。

メモ

If条件は、任意の範囲オプション(**先頭**、**次へ**、**While**)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IFステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

6. **出力]**タブをクリックします。

7. **出力先**]パネルで、適切な出力オプションを選択します。

- **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。
- **ファイル** - 結果をテキストファイルに保存または追加するには、このオプションを選択します。ファイルは Analytics の外部に保存されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

8. 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。

- **ファイルの種類** - オプションは、使用している Analytics のエディションに応じて、"ASCII テキストファイル" または "Unicode テキストファイル" のいずれかのみです。結果を新しいテキストファイルに保存するか、または既存のテキストファイルに追加します。
- **名前** - **名前**]テキストボックスにファイル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログボックスでファイル名を入力するか、既存のファイルに上書きまたは追加する場合はそのファイルを選択します。Analytics によってファイル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にファイルを保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、**C:\Results\Output.txt** または **Results\Output.txt** のように指定します。

- **ローカル** - 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。

9. 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキストボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ(<)を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー**]または **フッター**]ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキストボックスで、改行文字としてセミコロン(;)を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。

10. **詳細**]タブをクリックします。

11. **適用範囲**]パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While テキスト ボックスに条件を入力するか、または While ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭"または"次"オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次"は"先頭"のように動作します。</p>	

12. **最小値と最大値**で指定された値を超える値を含めないようにする場合は、**範囲外を出力しない**を選択します。

13. 省略可能。ヒストグラムの結果をテキスト ファイルに出力する場合は、棒グラフをテキストで表現するときの x 軸の長さを **列数**]に入力して指定します。

入力する数値は、x 軸 (および y 軸のラベル) に使用する文字スペース (テキスト列) の数です。ほとんどの場合、**列数**]を空白のままにして、デフォルトの文字スペース数の 78 を使用できます。

14. 省略可能。画面、ファイル、またはプリンターにヒストグラムの結果を出力している場合、**ブレーク**]テキスト ボックスにブレークフィールド名を入力するか、**ブレーク**]をクリックして、フィールドを選択するか、式を作成します。

たとえば、売掛金テーブルのヒストグラムは顧客別に分類する可能性があるでしょう。**ブレーク**]で使用できるのは 1 つの文字フィールドのみであるため、入れ子のブレークダウンには対応していません。

メモ

ブレイク] オプションによって有効な結果を得るためには、ヒストグラムを作成する前に文字フィールドを並べ替えておく必要があります。

15. 出力の種類として **ファイル]** を選択した場合、出力結果を既存のテキストファイルの末尾に追加するときは、**既存のファイルに追加する]** を選択します。
16. **OK]** をクリックします。
17. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。

機械学習分析

Analytics における機械学習 (AutoML) は自動化されています。データ前処理、アルゴリズム選択、ハイパーパラメーター調整、モデル検証などの複雑な計算作業が、Analytics によって自動化されます。この自動化によって、専門的なデータサイエンスのスキルを必要とせず、比較的少ない労力で、自社のデータに対する機械学習を実行できます。

管理ありなしの機械学習

管理ありなしの機械学習が Analytics によりサポートされています。

管理ありの機械学習では、カテゴリや数値のラベルが付いたデータを、ラベルが付いていない類似のデータにおけるカテゴリや数値を予測するためのベースにします。

管理なしの機械学習では、カテゴリ分けやラベル付けが行われていないデータから、カテゴリを発見します。

機械学習は 32 ビット コンピューターではサポート外

32 ビット コンピューターに Analytics をインストールする場合には、機械学習処理はサポートされないため、**機械学習**メニューは表示されません。機械学習に必要な演算は、プロセッサの負荷が高いため、64 ビット コンピューターの方が適しています。

機械学習処理

操作	ML(機械学習)タイプ	サポートされているデータ型	機能	出力
学習	監視あり	文字 数値 日付時刻 論理	<ul style="list-style-type: none"> 自動機械学習を使用して、予測モデルを作成します。 	<ul style="list-style-type: none"> 予測モデルファイル (* .model) モデル評価ファイル (Analytics テーブル)
予測	監視あり	文字 数値 日付時刻	<ul style="list-style-type: none"> 予測モデルをラベルがないデータセットに適用し、クラスまたは数値を予測します。 	<ul style="list-style-type: none"> 結果ファイル (Analytics テーブル)

データの分析

操作	ML(機械学習)タイプ	サポートされているデータ型	機能	出力
		論理		
クラスター	監視なし	数値	<ul style="list-style-type: none"> 数値データをグループ化します。 1つ以上の数値フィールドの類似または付近の値に基づいて、レコードをグループ化します。 	<ul style="list-style-type: none"> 結果ファイル (Analytics テーブル)

クラスと数値の予測

Analytics の自動機械学習を使用して、ラベルがないデータに関連付けられたクラスまたは数値を予測します。関心があるクラスまたは数値がデータに存在しない場合、データはラベルがありません。たとえば、機械学習を使用して、融資の債務不履行または将来の住宅価格を予測できます。

予測問題	予測タイプ	説明
融資の債務不履行	分類	年齢、職業、信用評価などの申請者情報に基づいて、融資を実行した場合に債務不履行になる申請者を予測します。 つまり、申請者がクラス Default = Yes または Default = No に該当するかどうかということです。
将来の住宅価格	再帰	年齢、平方メートル、郵便番号、間取り、浴室・トイレの数などの特長に基づいて、将来の住宅の販売価格を予測します。

自動機械学習

Analytics の機械学習は「自動」です。2つの関連するコマンドである学習と予測は、予測モデルの学習と、予測モデルをラベルがないデータセットに適用することに関連するすべての演算作業を実行します。Analytics の自動化によって、専門的なデータ科学技術を必要とせずに、自社データに対する機械学習を実行できます。

学習および予測ワークフロー

学習および予測ワークフローは2つの関連するプロセス、および2つの関連するデータセットから成ります。

- **学習プロセス**-は学習データセット(ラベル付き)を使用します
- **予測プロセス**-は新しいデータセット(ラベルなし)を使用します

学習プロセス

学習プロセスが先に実行され、ラベル付きフィールド(対象フィールド)を含む学習データセットを使用します。

ラベル付きフィールドには、学習データセットの各レコードに関連付けられる、既知のクラス、既知の数値が含まれます。たとえば、借り手が融資で債務不履行になったかどうか(Y/N)、や住宅の販売価格です。

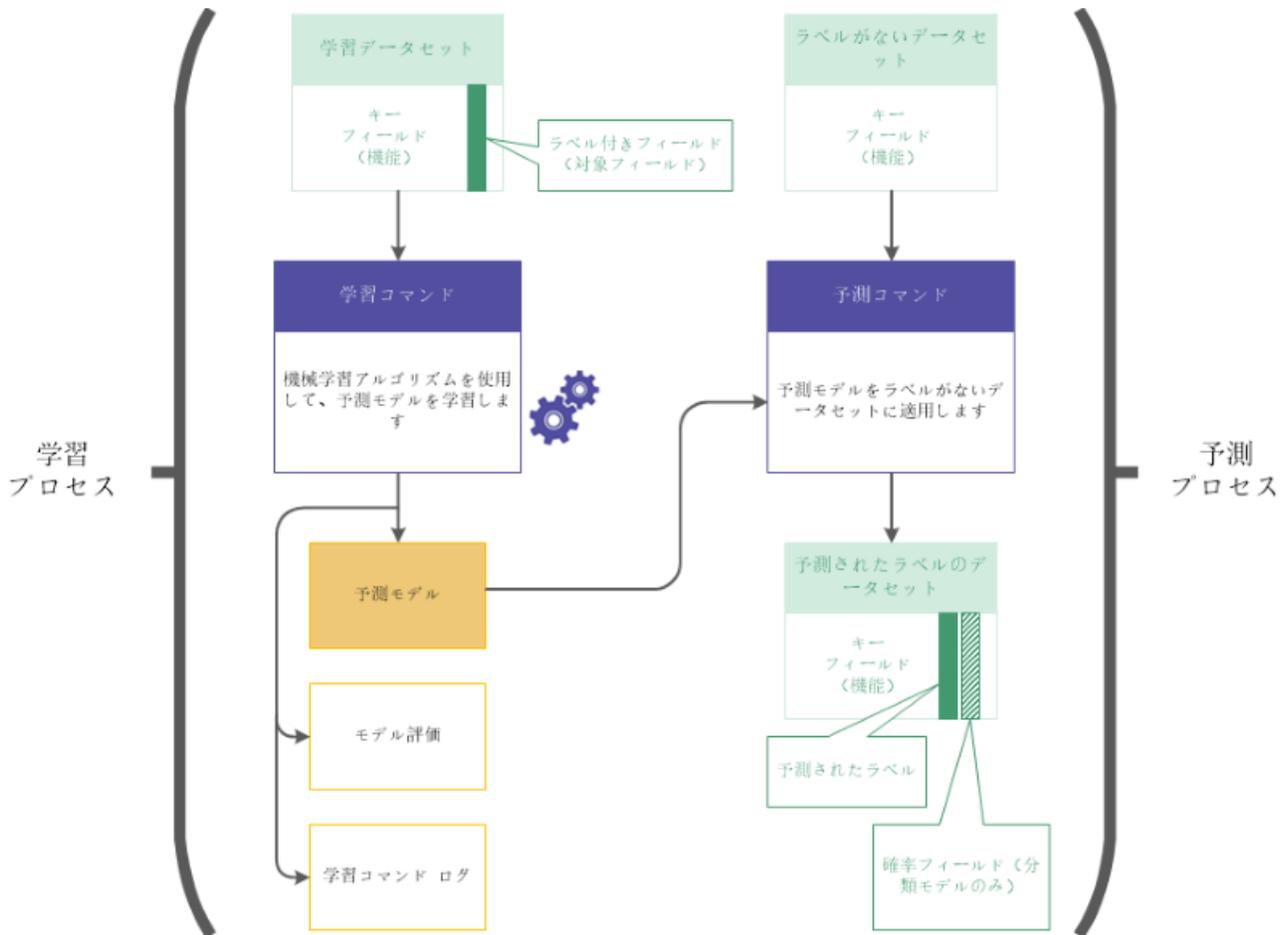
機械学習アルゴリズムを使用して、学習プロセスが**予測モデル**を生成します。学習プロセスは、実行している予測タスクに最適なモデルを検出するために、さまざまな異なるモデル順列を生成します。

予測 プロセス

予測プロセスは2番目に実行されます。学習プロセスによって生成される予測モデルが、学習データセットのデータに似ているデータを含む新しいラベルがないデータセットに適用されます。

融資の債務不履行情報や住宅販売価格などのラベル値は、将来のイベントであるため、新しいデータセットに存在します。

予測モデルを使用して、予測プロセスは、新しいデータセットの各ラベルがないレコードに関連付けられたクラスまたは数値を予測します。



学習および予測ワークフローの詳細

	プロセス	説明	データセットの例
1	トレーニング (学習コマン)	<ul style="list-style-type: none"> 学習コマンド - 学習データセットに対して学習コマンドを実行し、予測モデルを学習します。 	<ul style="list-style-type: none"> 融資データ - 融資債務不履行情報 (Y/N) を

	ド)	<p>コマンドは、予測タスクに最適な単一のモデルを選択する前に、複数の異なる機械学習アルゴリズムを使用して、さまざまなモデルを生成します(「ウィニングモデル」)。</p> <ul style="list-style-type: none"> ○ 学習データセット - データセットにはキーフィールド(機能)とラベル付きフィールド(対象フィールド)があります。 ○ 学習 - 学習プロセスは、キーフィールドとラベル付きフィールドとの関係を表す数学モデルを作成するものです。 ○ 例 - たとえば、すべての他の機能が同等の場合、学習プロセスは、4部屋あると、住宅の販売価格が\$35,000増加したことを検出するとします。 <p>"Number of bedrooms" はキーフィールドで "sale price" はラベル付きフィールドです。</p> <ul style="list-style-type: none"> ○ 予測モデル - 学習プロセスは、予測モデルを出力ファイルに保存します。 	<p>含む履歴融資データ</p> <p>"Default" ラベル付きフィールドです。</p> <ul style="list-style-type: none"> ○ 住宅データ - 販売価格を含む最近の住宅販売データ <p>"Sale price" はラベル付きフィールドです。</p>
2	予測 (予測コマンド)	<ul style="list-style-type: none"> ○ 予測コマンド - 予測コマンドを使用して、学習コマンドで生成された予測モデルを適用します。 ○ 新しいデータ - 学習データセットと同じキーフィールド(機能)があり、ラベル付きフィールドがない、新しいデータセットにモデルを適用します。 ○ 予測 - 予測プロセスは、予測モデルに保存された数学的関係を使用して、新しいデータセット内にある同様の関係を持つキーフィールド同士の関係に対するラベル値を予測します。 ○ 例 - たとえば、他のすべての機能が等しい場合、予測プロセスは3ベッドルームの部屋の販売価格を\$400,000と予測し、4ベッドルームの場合\$435,000と予測します。 ○ 確率 - (分類のみ) 各予測された値に対して、予測出力には、予測が正確である確率または信頼度が含まれます。 	<ul style="list-style-type: none"> ○ 融資データ - 現在の融資申請者データ <p>融資債務不履行情報は存在しません。融資は申請段階だからです。</p> <ul style="list-style-type: none"> ○ 住宅データ - 住宅価格評価データ <p>最近の販売データは存在しません。住宅はまだ販売されていないためです。</p>

処理時間

機械学習に必要な演算は時間がかかり、プロセッサの負荷が高くなります。多数のフィールドを含む大きいデータセットを使用して予測モデルを学習するには数時間かかることがあり、通常は夜間に行うタスクです。

学習プロセスに日付時刻キーフィールドを含めると、各日付時刻フィールドを使用して自動的に10個の合成特徴量を抽出するため、特にプロセッサの負荷が高くなります。日付時刻合成特徴量は予測データの範囲を大幅に拡大できますが、関連性が考えられる場合にのみ日付時刻フィールドを含めてください。

ヒント

Analyticsの機械学習に慣れる場合は、小さいデータセットを使用し、処理時間を管理可能に保つと、結果が比較的速く表示されます。

学習データセットのサイズを削減するためのストラテジー

さまざまなストラテジーを使用して、結果の予測モデルの精度に大幅な影響を与えずに、学習データセットのサイズ、関連付けられた処理時間を減らすことができます。

- 予測精度に影響せずに学習プロセスからフィールドを除外する関連のないフィールドと冗長なフィールドを除外します。
- 予測精度に影響しない場合には、学習プロセスからフィールドを除外します。ただし、日付時刻フィールドが関連していないことを想定するには注意してください。詳細については、「日付時刻キーフィールド」下を参照してください。
- 学習データセットを抽出し、学習プロセスの入力として、抽出されたデータを使用します。考えられるサンプリングアプローチ：
 - 大半のクラスをサンプリングしてデータクラスのサイズのバランスを保ち、平均的な少数のクラスサイズを近似する
 - 学習データセット全体のランダムサンプリング
 - 機能に基づく階層化サンプリング
 - クラスタに基づく階層化サンプリング

日付時刻キーフィールド

予測モデルを学習するときには、1つ以上の日付時刻フィールドをキーフィールドとして使用できます。一般的に、日付時刻フィールドに存在する一意の値が多すぎると、フィールドは学習プロセスで適切なカテゴリのソースまたは特定可能な特徴量になりません。未加工の日付時刻データは、該当する対象フィールドに関連していないように見える場合もあります。

しかし、分類されると、日付時刻データが関連性をもつように見える場合があります。たとえば、調査しているイベントが特定の曜日または特定の時間に発生しているというパターンが見られる場合があります。

学習プロセスは、未加工の日付時刻データを分類することで、各日付時刻フィールドから多数の**合成特徴量**を自動的に抽出します。これらの合成特徴量は、**予測モデル**を生成するアルゴリズムに含まれません。

日付時刻フィールドから抽出された合成特徴量

日付、時刻、日付時刻フィールドから自動的に抽出される合成特徴量の一覧は次のとおりです。

合成特徴量の説明	特徴量の種類	合成特徴量名
曜日	数値(1 ~ 7)	フィールド名_DOW
月	数値(1 ~ 12)	フィールド名_MONTH
四半期	数値(1 ~ 4)	フィールド名_QTR
月初からの日数	数値(1 ~ 31)	フィールド名_DAY

合成特徴量の説明	特徴量の種類	合成特徴量名
年初からの日数	数値(1 ~ 366)	フィールド名_DOY
秒	数値(00 ~ 59)	フィールド名_SECOND
時間	数値(1 ~ 24)	フィールド名_HOUR
1日の開始からの秒数	数値(1 ~ 86400)	フィールド名_SOD
日の四分位	カテゴリー: <ul style="list-style-type: none"> ○ 00:00-06:00 ○ 06:00-12:00 ○ 12:00-18:00 ○ 18:00-24:00 	フィールド名_QOD
日の八分位	カテゴリー: <ul style="list-style-type: none"> ○ 00:00-03:00 ○ 03:00-06:00 ○ 06:00-09:00 ○ 09:00-12:00 ○ 12:00-15:00 ○ 15:00-18:00 ○ 18:00-21:00 ○ 21:00-24:00 	フィールド名_OOD

予測モデルの学習

メモ

学習プロセスで使用されるデータセットのサポートされる最大サイズは1GBです。
機械学習メニューオプションが無効な場合は、Python エンジンがインストールされていない可能性があります。詳細については、「ACL for Windows をインストールする」ページ 2679を参照してください。

手順

学習プロセスの基本設定を指定する

1. 学習データセットがある Analytics テーブルを開きます。
2. Analytics メインメニューで、**機械学習 > 学習**]を選択します。
3. 学習プロセスに割り当てられた時間を指定します。

最適なモデルを検索する時間	予測モデルの生成とテスト、およびウィニングモデルの選択にかかる合計時間(分)。 モデルごとに、最大評価時間の10倍の検索時間を指定します。
----------------------	--------------------------------------------------------------------------

モデル評価の最大時間	モデル評価ごとの最大実行時間(分)。 100 MB の学習データごとに45分を割り当てます。
------------	---------------------------------------------------

メモ

学習プロセスの合計実行時間は、検索時間と最大モデル評価時間を最大で2倍した値の合計です。

これらの提案された時間割り当ては、処理時間と多様なモデルタイプの評価を可能にすることの間で、合理的なバランスを取っています。

4. 使用する予測タイプを指定する:

- **分類** - 分類アルゴリズムを使用して、モデルを学習します
レコードが属するクラスまたはカテゴリを予測する場合は分類を使用します。
- **回帰** - 回帰アルゴリズムを使用して、モデルを学習します
ラベルがないデータセットのレコードに関連付けられた数値を予測する場合は回帰を使用します。

分類と回帰で使用される特定のアルゴリズムについては、「学習アルゴリズム」ページ 1359を参照してください。

5. **モデルスコアラー**]ドロップダウンリストで、学習プロセス中に生成されたモデルのスコアを決定するときに使用するメトリクスを選択します。

生成されたモデルのうち、このメトリクスの最善値を有するモデルが保持され、そうでないモデルは破棄されます。

使用している予測タイプによって、異なるサブセットのメトリクスを使用できます。

分類	ログ損失 AUC 正確度 F1 適合率 再現率
再帰	平均二乗誤差 平均絶対誤差 R2

メモ

分類メトリクス **AUC** は、バイナリデータ(Yes/No、True/Falseなどの2つのクラス)が含まれるターゲット フィールドで使用されるときにのみ有効です。

フィールドの選択

1. **学習対象**]リストで、モデルを学習するときに入力として使用する1つ以上のキーフィールドを選択します。

キーフィールドは、ラベルがないデータセットで対象フィールド値を予測する基礎となる機能です。キーフィールドは、文字、数値、日付時刻、または論理型を使用できます。合成特徴量は日付時刻キーフィールドから自動的に抽出されます。

メモ

文字フィールドは「分類的」である必要があります。つまり、カテゴリまたはクラスを特定し、一意の値の最大数以下でなければなりません。

最大値は **最大カテゴリ** オプション(ツール > オプション > コマンド) で指定されません。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

2. **対象フィールド** リストから、対象フィールドを選択します。

対象フィールドは、入力キーフィールドに基づいて予測するようにモデルが学習されているフィールドです。

分類と回帰はさまざまな対象フィールド データ型で機能します。

- **分類** - 文字または論理対象フィールド
- **回帰** - 数値対象フィールド

モデルファイルと出力 Analytics テーブルの名前の指定

1. **[モデル名]** テキストボックスで、学習プロセスで出力されたモデルファイル出力の名前を指定します。

モデルファイルには、学習データセットに最適なモデルが含まれます。モデルファイルを予測プロセスに入力し、新しい未確認のデータセットに関する予測を生成します。

2. **保存先** テキストボックスで、学習プロセスで出力されたモデル評価テーブル出力の名前を指定します。

モデル評価テーブルには、以下の2つの異なるタイプの情報が格納されています。

- **スコアラー/メトリクス** - これらは、学習プロセスによって生成されたモデルファイルの予測パフォーマンスの定量的な推定、分類メトリクスまたは回帰メトリクスを意味します。
- **重要性/係数** -(降順) : モデルによって生成された予測に対する各機能(予測印子)の寄与度を示す値。

メモ

Analytics のテーブル名は、最長で64文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

3. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用してIFステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(先頭、次へ、While)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

If ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

学習データセットのサブセットのみが使用されることを指定する(任意)

詳細]タブで、**範囲]**パネルのオプションのいずれかを選択します。

すべて(デフォルト)	テーブルのすべてのレコードが処理されます。
先頭	このオプションを選択してテキストボックスに数を入力すると、テーブル内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキストボックスに数を入力すると、テーブルビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。 行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	WHILE ステートメントを使用して、条件に基づいてテーブル内のレコードの処理を制限するには、このオプションを選択します。

学習プロセスの詳細設定を指定する

1. **詳細]**タブ **交差検証分割数]**を指定します。

既定の数値の5を使用するか、別の数値を指定します。有効な数値は2 ~ 10です。

分割は学習データセットの下位分割であり、モデル評価と最適化中に、交差検証プロセスで使用されます。

一般的に、モデルの学習時に、5 ~ 10 分割を使用すると、適切な結果が得られます。

ヒント

分割数が増えると、モデルの予測性能の推定値が改善されることがありますが、全体的な実行時間も長くなります。

2. 省略可能。 **シード]**を選択し、数値を入力します。

Analytics の乱数ジェネレーターを初期化するために使用するシード値。

シードを省略した場合は、シード値がランダムに選択されます。

明示的にシード値を指定し、将来に学習プロセスを同じデータセットで複製したい場合は、それを記録します。

3. 省略可能。線形モデルのみを学習し、スコアを決定するには、 **線形モデルのみを評価する]**を選択します。

このオプションを未選択にする場合、分類または再帰に関連するすべてのモデルタイプが評価されます。

メモ

データセットが大きくなると、一般的に、線形モデルのみが含まれている場合に、学習プロセスがより短い時間で完了します。

線形モデルのみを含めると、出力の係数が保証されます。

4. 省略可能。学習プロセスからこれらの下位プロセスを除外するには、**機能選択と前処理を無効にする**]を選択します。

機能選択は、予測モデルを最適化する際に最も有用な学習データセットで、自動化されたフィールドの選択です。自動化された選択は予測性能を改善することがありますが、モデル最適化に関連するデータ量が減ります。

データ前処理は、学習データセットでの調整や標準化などの変換を実行し、学習アルゴリズムにより適したものにします。

注意

理由がある場合にかぎり、機能選択とデータ前処理を無効にしてください。

5. **OK**]をクリックします。

学習プロセスが開始し、指定した入力設定と経過した処理時間を示すダイアログボックスが表示されます。

予測モデルをラベルがないデータセットに適用する

メモ

機械学習メニューオプションが無効な場合は、Python エンジンがインストールされていない可能性があります。詳細については、「ACL for Windows をインストールする」ページ 2679を参照してください。

手順

1. ラベルがないデータセットがある Analytics テーブルを開きます。
2. Analytics メインメニューで、**機械学習 > 予測**]を選択します。
3. **モデル**]をクリックし、**ファイルを選択**]ダイアログボックスで、前の学習プロセスによって生成されたモデルファイルを選択し、**開く**]をクリックします。

モデルファイルのファイル拡張子は ***.model** です。

メモ

モデルファイルは、ラベルのないデータセットと同じフィールドまたはほぼ同じフィールドのデータセットで学習されている必要があります。

バージョン 14.1 の Analytics で学習されたモデルファイルは使用できません。バージョン 14.1 のモデルファイルは、それ以降のバージョンの Analytics と互換性はありません。予測プロセスと使用するには、新しい予測モデルを学習してください。

4. **保存先]**テキストボックスで、予測プロセスで出力された Analytics テーブル出力の名前を指定します。

この出力テーブルには、学習プロセス中に指定したキーフィールドと、予測プロセスで生成された、以下の1つまたは2つのフィールドが含まれます。

- **予測** - ラベルがないデータセットの各レコードに関連付けられた予測されたクラスまたは数値
- **確率** - (分類のみ) 予測されたクラスが正しい確率

メモ

Analytics のテーブル名は、最長で 64 文字の英数字 (拡張子 .FIL を含まない) に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

5. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]** テキストボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー** を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション (**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

6. 省略可能。ラベルがないデータセットのサブセットのみを処理するには、**詳細]** タブで、**範囲]** パネルのオプションのいずれかを選択します。
7. **OK]** をクリックします。

学習アルゴリズム

3つの学習コマンド オプションは、予測モデルを学習するために使用される機械学習アルゴリズムを指示します。

オプション	学習ダイアログボックスタブ
分類または再帰	メイン] タブ
線形モデルのみを評価	詳細] タブ

オプション	学習ダイアログボックス タブ
機能選択と前処理を無効にする	詳細] タブ

次のセクションでは、使用されるアルゴリズムを制御するオプションを要約します。

アルゴリズム名は、Analytics ユーザー インターフェイスに表示されません。学習コマンドによって最終的に選択されたモデルを生成するために使用されるアルゴリズムの生がログに表示されます。

メモ

アルゴリズムの詳細については、[scikit-learn ドキュメント](#)を参照してください。scikit-learn は、Analytics が使用する Python 用の機械学習ライブラリです。

分類アルゴリズム

詳しく表示

✔ 使用されるアルゴリズム ✖ 使用されないアルゴリズム

アルゴリズム名	常に含まれる	線形モデルのみを評価		機能選択と前処理を無効にする	
		選択されていないオプション(デフォルト)	選択されたオプション	選択されていないオプション(デフォルト)	選択されたオプション
アルゴリズム タイプ: 分類器					
ロジスティック回帰	✔				
線形サポート ベクター マシン	✔				
ランダム フォレスト		✔	✖		
Extremely Randomized Trees		✔	✖		
勾配ブースティング マシン		✔	✖		
アルゴリズム タイプ: 特徴量プリプロセッサ					
ワンホット エンコーディング (カテゴリ特徴量)	✔				
Fast 独立成分分析				✔	✖
Feature Agglomeration				✔	✖

データの分析

アルゴリズム名	常に含まれる	線形モデルのみを評価		機能選択と前処理を無効にする	
		選択されていないオプション(デフォルト)	選択されたオプション	選択されていないオプション(デフォルト)	選択されたオプション
主成分分析(特異値分解)				✔	✘
Second Degree Polynomial Features				✔	✘
Binarizer				✔	✘
ロバスト スカラー				✔	✘
標準スカラー				✔	✘
最大絶対スカラー				✔	✘
最小最大スカラー				✔	✘
Normalizer				✔	✘
ニューストレムカーネル近似				✔	✘
RBFカーネル近似				✔	✘
ゼロカウンター				✔	✘
アルゴリズムタイプ: 特徴量選択					
ファミリーワイズエラー率				✔	✘
最高スコア百分位				✔	✘
差異しきい値				✔	✘
再帰的特徴消去				✔	✘
重要度重み				✔	✘

再帰アルゴリズム

詳しく表示

✔ 使用されるアルゴリズム ✖ 使用されないアルゴリズム

アルゴリズム名	常に含まれる	線形モデルのみを評価		機能選択と前処理を無効にする	
		選択されていないオプション(デフォルト)	選択されたオプション	選択されていないオプション(デフォルト)	選択されたオプション
アルゴリズムタイプ: 回帰					
Elastic Net	✔				
Lasso	✔				
Ridge	✔				
線形サポートベクターマシン	✔				
ランダムフォレスト		✔	✖		
Extremely Randomized Trees		✔	✖		
勾配ブースティングマシン		✔	✖		
アルゴリズムタイプ: 特徴量プリプロセッサ					
ワンホットエンコーディング(カテゴリ特徴量)	✔				
Fast 独立成分分析				✔	✖
Feature Agglomeration				✔	✖
主成分分析(特異値分解)				✔	✖
Second Degree Polynomial Features				✔	✖
Binarizer				✔	✖
ロバストスカラー				✔	✖
標準スカラー				✔	✖
最大絶対スカラー				✔	✖

データの分析

アルゴリズム名	常に含まれる	線形モデルのみを評価		機能選択と前処理を無効にする	
		選択されていないオプション(デフォルト)	選択されたオプション	選択されていないオプション(デフォルト)	選択されたオプション
最小最大スカラー				✔	✘
Normalizer				✔	✘
ニューストレムカーネル近似				✔	✘
RBFカーネル近似				✔	✘
ゼロカウンター				✔	✘
アルゴリズムタイプ: 特徴量選択					
ファミリーワイズエラー率				✔	✘
最高スコア百分位				✔	✘
差異しきい値				✔	✘
重要度重み				✔	✘

データのクラスター

クラスターは、1つ以上の数値キーフィールドの類似した値に基づいて、テーブルのレコードをグループ化します。類似した値は、データセット全体のコンテキストで相互に近い値です。これらの類似した値は、特定されると、データのパターンを明らかにするクラスターを表します。

メモ

分析プログラムでクラスターを定期的に行う場合は、Diligent アカデミー コースの「[Analytics で CLUSTER コマンドを使用してデータグループを検索する\(ACL 361\)](#)」を受講することをお勧めします(お客様のログインが必要です)。

他の Analytics グループ化コマンドとクラスターの違い

クラスターは他の Analytics グループ化コマンドとは異なります。

- クラスターでは、取引タイプや加盟店カテゴリコードといった既存のデータカテゴリや、数値の境界がハードコーディングされた定義済みの層でグループ化する必要がありません。代わりに、クラスターはデータ内の類似した数値(つまり、相互に近い値)に基づいてデータをグループ化します。
- 複数のフィールドに基づくクラスターは、ネストされない(非階層)結果を出力します。

クラスター対象のフィールドの選択

クラスターにより、存在することを知らないデータの有機的なグループを検出することができます。特に、機械学習を利用せずに、複数の数値フィールドに基づくクラスター(多次元クラスター)を特定することは困難です。この意味で、クラスターは説明的であり、監視されない機械学習の例です。

ただし、出力クラスターの有意性を保証するには、クラスターで選択するフィールド間に有意な関係が存在する必要があります。

単一フィールドのクラスター

単一フィールドでのクラスターは比較的簡単です。単一の値のセットに集中し、クラスターは値間の近さ(近似)に基づいて値をグループ化します。たとえば、金額フィールドをクラスター化し、値の範囲で金額が集中する場所を見つけることができます。

階層化などの従来のアプローチに対するクラスターの利点は、前もって、集中が存在する可能性がある場所を想定したり、任意の数値境界を作成したりする必要がないことです。クラスターは、特定のクラスター数の境界がある場所を検出します。

例

単一の数値フィールドでのクラスターの例

Ap_Trans テーブルを **Invoice Amount** でクラスター化し、値の範囲で金額が集中する場所を見つけることができます。想定される結果は、金額のほとんどが範囲の下限でクラスター化されることです。クラスターは、想定されたパターン通りになるかどうかを確認します。

Invoice Amount フィールドを 5 つのクラスターにグループ化することを決め、クラスターを要約して、各クラスターのレコード数を検出します。

出力結果

次の出力結果では、最初の 5 つのレコードがシステムで生成され、指定した任意のクラスター数と同じになります。**Invoice Amount** フィールドでは、5 つのレコードが**中心**を示します。これは、請求金額の 5 つのクラスターそれぞれに対してクラスターアルゴリズムが計算する中心点です。たとえば、クラスター 3 (C3) の中心は 2,969.04 です。詳細については、"クラスターアルゴリズムの仕組み" ページ 1368 を参照してください。

システムで生成されたフィールドの下では、ソースデータフィールドがクラスターにグループ化されません。これはクラスター 0 から始まります。**Distance** フィールドの値は、実際の請求金額から、そのクラスターの計算された中心値までの距離です。このため、たとえば、レコード 6 では、請求金額 618.30 から距離 64.935317 を差し引くと、中心値 553.36 と等しくなります。

メモ

実際の値が中心値より大きい小さいかによって、距離値を加算または減算します。

	Invoice Amount	Cluster	Distance	Invoice Date	Invoice Number	Vendor Number	Vendor Name
1	553.36	C0	0.000000				
2	56767.20	C1	0.000000				
3	18010.28	C2	0.000000				
4	2969.04	C3	0.000000				
5	8061.46	C4	0.000000				
6	618.30	0	64.935317	2000/11/17	5981807	11663	More Power Industries
7	49.68	0	503.684683	2000/01/31	517506	13136	Muller Corp.
8	783.99	0	230.625317	2000/01/31	122088	10721	Witz & Partners
9	187.60	0	365.764683	2000/01/31	2653864	10448	PacRim Engineered Products
10	538.47	0	14.894683	2000/01/31	54324133	11435	Group Services
11	1151.15	0	597.785317	2000/11/14	239388	12701	Harris Projects
12	965.77	0	412.405317	2000/11/14	232195	10025	Mitchell Ent.
13	760.77	0	207.405317	2000/08/14	294698	14438	Bloom County Construction
14	92.16	0	461.204683	2000/02/14	26530	11009	Waterson Services
15	1744.40	0	1191.035317	2000/02/14	8752383	11475	Triathalon Group
16	540.80	0	12.564683	2000/02/14	2650620	10448	PacRim Engineered Products
17	561.20	0	7.835317	2000/02/14	70936	10134	Stars Trading
18	397.80	0	155.564683	2000/12/01	54326778	11435	Group Services
19	287.00	0	266.364683	2000/02/13	121053	10721	Witz & Partners
20	1271.00	0	717.635317	2000/02/13	8757170	11475	Triathalon Group
21	537.74	0	15.624683	2000/09/30	293732	14438	Bloom County Construction

クラスタの要約

Cluster フィールドを要約し、要約された出力をカウントで並べ替えると、次の結果が出力され、値の分布が想定通りであることを確認します。全体的に、請求金額が低い値に向かって大きく歪みます。(比較しやすいように中心値がテーブルに追加されています。)

クラスタの単一の大きい値は単独で異常値のように出現し、一般的に調査が必要です。

クラスタ	カウント	中心値
0	73	553.36
3	16	2,969.04
4	8	8,061.46
2	4	18,010.28
1	1	56,767.20

複数のフィールドのクラスター

2つ以上のフィールドでクラスター化するときには、フィールドを関連付ける方法を確認する必要があります。クラスターを使用して、仮定をテストできます。たとえば、会社が従業員の退職率に関心があるとしません。管理者は、より若い給与が低い従業員に集中していると考えています。

クラスターを使用して、次の間に強い関係があるかどうかを検出できます。

- 勤続期間と従業員の年齢(2次元クラスター)
- 勤続期間、従業員の年齢、給与(3次元クラスター)

メモ

この分析では、取得された傷病休暇日数など、仮定とは明確に関係しないフィールドを含めないようにする必要があります。

出力クラスターの評価

クラスターアルゴリズムは、常に、指定されたクラスター数のテーブルを出力します。出力テーブルのすべてのレコードはクラスターにあります。

この時点では、クラスターに分析上の有意性または意味があるかどうかを評価する必要があります。アルゴリズムがクラスターのレコードをグループ化するからといって、必ずしも、グループ化が有意であるわけではありません。クラスターが有意なパターンを形成しているかどうかを確認する必要があります。クラスターがストーリーを伝えているかどうか

ヒント

クラスター出力テーブルをレポート ツールの散布図としてグラフ化し、各クラスターに別の色を割り当てると、出力クラスターの全体的な特性をすばやく、簡単に評価できます。

次の特性は、出力されたクラスターの有意性を評価する際に役立ちます。

- **クラスターの一貫性** - クラスターの個別の値すべてが比較的中心に近い位置にあるか、クラスターが分散しているか。クラスターの一貫性が高いほど、クラスターを構成する値の関係が強くなります。
- **クラスターのサイズ** - 1つまたは2つの大きいクラスターにほとんどの値が含まれているか。この場合、値が多数のクラスター間で比較的均等に分散しているデータセットに比べ、データセットが大きく偏っています。
- **異常値** - 重要なクラスターのいずれにも含まれない値を検討します。このような異常値は、追加で精査する必要がある項目を示している可能性があります。また、「内部異常値」を検討します。つまり、重要なクラスターに含まれていて、クラスターの外側の端にある値です。

メモ

上記の特性はすべて、クラスター評価でも、人間が行う主観的な方法です。クラスター評価ではさまざまな機械的な方法が存在しますが、そのような方法は本 Analytics ヘルプの対象範囲外です。

クラスタ アルゴリズムの仕組み

Analytics のクラスタは、k 平均法 アルゴリズムを使用します。これは、一般的な機械学習アルゴリズムです。k 平均法の詳細については、インターネットの説明をご覧ください。

アルゴリズムの概要は以下のとおりです。

詳しく表示

k 平均法 アルゴリズムは、反復プロセスを使用して、クラスタを最適化します。

1	クラスタ数の指定	<ul style="list-style-type: none"> データセットをグループ化するために使用するクラスタまたはグループ数を決定します。"K" は指定するクラスタ数を表します。 データセットのデータ点は、単一数値フィールドの値、または複数の数値フィールドに基づいてアルゴリズムが演算する複合値にすることができます。
2	クラスタ中心の初期化	<ul style="list-style-type: none"> ランダム データ点のセットを生成し、クラスタ計算の初期中心または中央点として使用します。 生成された中心数は、指定したクラスタ数と同じです。
3	各データ点を最も近い中心に割り当てる	<ul style="list-style-type: none"> 各データ点から中心への最短距離を見つけます。距離の比較は平方ユークリッド距離を使用します。 各データ点を最も近い中心に割り当てます。特定の中心に割り当てられなかったすべてのデータ点がクラスタになります。
4	中心の再計算	<ul style="list-style-type: none"> クラスタのすべてのデータ点の平均または中央値を計算します。中央値がクラスタの新しい中心になります。
5	反復	<ul style="list-style-type: none"> 手順 3 および 4 を繰り返します。 <ul style="list-style-type: none"> 各データ点から中心への最短距離を再計算します。 各データ点を最も近い中心に割り当てます。これにより、一部のデータ点がもう一度別のクラスタに割り当てられます。 中心を再計算します。 データ点が割り当てられなくなるまで、または指定された反復の最大数に達するまで、反復を繰り返します。 各反復で、クラスタの構成がより一貫します。つまり、クラスタのデータ点はより近くなります。

クラスタ数 (K 値) の選択

データのクラスタ化で使用する最適なクラスタ数を決定するには、テストと実験が必要になることがあります。特定のデータセットに対する、正確な答えはありません。

詳しく表示

最適なクラスタ数を決定するためのガイドライン

- データを理解する - 前もってデータセットを理解し、データのプロフィール、および値の明確な集中について概要を把握します。

- **最初は高めにする** - 最初は 8 ~ 10 の比較的多いクラスター数を選択します。
- **別のクラスター数を試す** - 複数回クラスターを実行し、毎回別の K 値を指定します。出力結果をレビューすると、クラスターを増やすのか減らすのかを判断できます。
- **エルボー法** - エルボー法を使用すると、クラスターの最適な数をプログラミングで特定します。最適な数とは、クラスターの一貫性が最も高く、追加クラスターの収穫逡減(既に一貫しているクラスターを分割しても限界的にしか一貫性を実現できないこと)を回避する点です。

エルボー法の結果を折れ線グラフでプロットし、「エルボー」または変曲点を視覚的に特定します。ここでは、クラスター数を増やしても一貫性は大きく改善されません。

Analytics で使用可能なエルボー法のスクリプトは ScriptHub からダウンロードできます。[エルボー法 - K の残差平方和 \(SSE\)](#) (お客様のログインが必要です)

文字または日付時刻フィールドでクラスター化できますか。

一般的に、文字または日付時刻フィールドでクラスター化できません。クラスターアルゴリズムは数値だけを許可し、数値で計算を実行します(ユークリッド距離、中央値)

詳しく表示

カテゴリ文字データ

ロケーション ID などのカテゴリ文字データが数値の形式になっていることがあります。あるいは、演算フィールドを使用して、文字カテゴリを作成する数値コードのセットにマッピングできます。このデータを数値データ型に変換し、クラスターで使用できます。ただし、結果のクラスターは有効になりません。数値ではない対象を表す数値に対して数学的演算を実行するためです。

たとえば、ロケーション ID のリストの平均に基づいて中心位置を計算すると、意味のない数値になります。計算は、ロケーション番号間の数学的な距離が何らかの実際の世界の測定可能な距離と等しいという無効な想定に基づいています。

物理的な距離を考慮する場合、ロケーション 1 とロケーション 9 の間の距離がロケーション 1 とロケーション 5 の間の距離の 2 倍であるということは無意味です。ロケーション 1 と 9 は隣で、ロケーション 5 が数マイル離れている可能性があります。

ロケーションや物理的な距離に関するクラスター分析では、使用する有効なデータは地理座標です。

スケールを表すカテゴリデータ

スケールを表すカテゴリデータでクラスターできます。たとえば、不可から優までの評価スケールと、対応する 1 ~ 5 のコードです。この場合、数値コードの平均には意味があります。

日付時刻データ

Analytics 関数を使用して、日付時刻データを数値データに変換できます。ただし、結果の数値データは連続していません。これは、数値の連続セットを想定するクラスター分析では問題となります。

たとえば、次の3つの数値は、日付としては、1日違いです。ただし、数値としては、最初と2番目の数値にかなりのギャップまたは距離があります。

- 20181130
- 20181201
- 20181202

クラスター分析では、シリアル日付値を使用できます。シリアル日付は、1900年1月1日からの経過日数を表す、整数の連続セットです。

手順

メモ

機械学習メニューオプションが無効な場合は、Python エンジンがインストールされていない可能性があります。詳細については、「ACL for Windows をインストールする」ページ 2679を参照してください。

クラスターアルゴリズムの設定の指定

1. クラスター化するデータがあるテーブルを開きます。
2. Analytics メインメニューで、**機械学習 > クラスター**]を選択します。
3. **クラスター数(k 値)**]で、データをグループ化するために使用するクラスター数を指定します。
4. **最大繰り返し数**]で、クラスターアルゴリズムで実行される反復数の上限を指定します。
5. **初期化数**]で、ランダム中心の初期セットを生成する回数を指定します。
6. 省略可能。[**シード**]を選択し、数値を入力します。

データ処理方法の指定

複数のキーフィールドでクラスター化する場合は、クラスターで使用する前に、**前処理**機能を使用して、フィールドのスケールを標準化してください。

一般的に、異なる数値フィールドのスケールと単位は異なります。たとえば、年間のドル金額を含む給料フィールドの範囲は20,000 ~ 100,000で、年を含む年齢フィールドの範囲は18 ~ 70であるという可能性があります。給料フィールドと年齢フィールドを使用してクラスター化し、調整を行わない場合、出力されたクラスターは、給料/年齢クラスターではなく、基本的に給料クラスターとなります。その場合、年齢数と比較して、給料の数値のサイズにクラスターが偏ります。

前処理を実行すると、次の方法によって、すべてのクラスターのキーフィールドのすべての値を調整するため、クラスター処理中に均等に重み付けすることができます。

前処理オプション	説明
標準化	<p>キーフィールド値の中央を平均ゼロ(0)にし、調整します。これは、値をz得点相当(標準得点)に変換するプロセスです。</p> <p>z得点は、未加工の値を各フィールドの未加工の平均値から分離する標準偏差数の測定です。調整されたフィールドでは、平均値がゼロ(0)で表されます。表される未加工の値がフィールドの未加工の平均値より大きい小さいかによって、z得点が正または負になります。</p> <p>メモ キーフィールドにほとんどゼロ以外の値が含まれている場合はこのオプションを使用します(密行列)。</p> <p>例</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <p>z得点の計算の例</p> <p>調整された年齢フィールドでは、元の年齢値 55 は、z得点では 1.038189 になります。</p> <ul style="list-style-type: none"> • 年齢フィールドの元の平均値 : 42.04054054 • 年齢フィールドの標準偏差 : 12.48276021 • 元の平均値を減算して元の値を中心にする : $55 - 42.04054054 = 12.95945946$ • 標準偏差で除算して中心の元の値を調整する : $12.95945946 / 12.48276021 = 1.038189$ • 55 は元の平均値から離れた標準偏差 1.038189 です。 </div>
単位分散になるようにスケール	<p>標準偏差で除算してキーフィールド値を調整しますが、中央を平均ゼロ(0)にしません。</p> <p>メモ 1つ以上のキーフィールドに多数のゼロ(0)値が含まれる場合にはこのオプションを使用します(疎行列)。</p> <p>例</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <p>中心化せずに調整する例</p> <p>調整された年齢フィールドでは、元の年齢値 55 は、調整された値 4.406077 になります。</p> <ul style="list-style-type: none"> • 年齢フィールドの標準偏差 : 12.48276021 • 標準偏差で除算して元の値を調整する : $55 / 12.48276021 = 4.406077$ </div>
なし	<p>キーフィールドは中心化も調整もされません。クラスターを計算するときに、クラスターは中心化も調整もされない、元の値を使用します。</p>

フィールドの選択

1. **対象クラスター]**リストから、テーブルのレコードをクラスター化するために使用する1つ以上のキーフィールドを選択します。
キーフィールドは数値である必要があります。
2. 省略可能。 **他のフィールド]**リストで、出力テーブルに含める1つ以上の追加フィールドを選択します。

ヒント

隣接する複数のフィールドを選択するには、**Shift キー + クリック**を、隣接していない複数のフィールドを選択するには、**Ctrl キー + クリック**を利用できます。

コマンド入力の確定

1. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用してIF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭**、**次へ**、**While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

2. **保存先]**テキスト ボックスに、出力テーブルの名前を指定します。

メモ

Analytics のテーブル名は、最長で64文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(`_`)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

3. 省略可能。 **詳細]**タブ
 - a. レコードのサブセットのみが処理されることを指定するには、**範囲]**パネルでオプションのいずれかを選択します。
 - b. 出力テーブルを自動的に開くようにするには、**出力テーブルの使用]**をオンにします。
4. **OK]**をクリックします。

ベンフォード分析の実行

ベンフォード分析は、フィールドに表示される最初の桁(1 ~ 9)や最初の桁の組み合わせの数をカウントし、実数と予測数を比較します。

ベンフォード式で計算された、想定されたカウントは、ベンフォード分布を提供します。自然に発生する数字のセットにおいて、先頭の数の実際のカウントの頻度分布は、ベンフォード分布に近似します。

テストされるデータの1つ以上の先頭の数または数の組み合わせが、ベンフォード分布から大幅に逸脱している場合は、数が操作された可能性があることを示すことがあります。逸脱はシンプルかつ合理的な説明である場合もあり、必ずしも操作を示すものではありません。

ベンフォード分析を使用してテストできるデータ

ベンフォード分析は、会計金額、取引金額、費用、またはアドレス番号などの「自然発生数」から構成される数値データをテストする目的でのみ、使用してください。ベンフォード分析は、いかなる方法でも制約を受けた数値データには適していません。

ベンフォード分析に適した数値データを特定するには、次のガイドラインに従ってください。

- **データセットのサイズ** - 有効な分布をサポートするには、データセットが十分に大きいサイズである必要があります。ベンフォード分析では、499レコード以下の場合には、信頼できる結果が得られない場合があります。
- **先頭の数の要件** - 1 ~ 9のすべての数値が、先頭の数として発生する可能性がなければなりません。
- **先頭の数の組み合わせの要件** - 0 ~ 9のすべての数値が、先頭から2番目の数および分析対象の追加の数として発生する可能性がなければなりません。
- **制約されたデータ** - あらかじめ定義されたパターンに従って割り当てまたは生成された数値データは、ベンフォード分析に適していません。たとえば、次の分析では、ベンフォード分析を使用しないでください。
 - 連番の小切手または請求書番号
 - 特定のパターンにマッピングされる社会保障番号または電話番号
 - 特定の数字が出現しない範囲がある番号体系
- **乱数** - 乱数生成器で生成された数値は、ベンフォード分析に適していません。

使用の詳細

以下の表は、Analyticsでのベンフォード分析の使用に関する詳細を示します。

分析する先頭桁数	先頭桁数を最高6桁まで分析できます。4桁以上を分析する場合、ベンフォード分析の結果は、画面に表示したり、プリンターへ送信したりするのではなく、ファイルに出力する必要があります。
----------	------------------------------------------------------------------------------------------

処理時間	作業対象のレコード数にもよりますが、5桁以上の先頭桁数の分析には数分かかる可能性があります。分析している桁数にかかわらず、 Esc キーを押せばいつでもこのコマンドを終了させることができます。
データセットのサイズ	効果的にベンフォード分析を実行するには、大きなデータセットが必要になります。Analyticsでは、指定した桁数に対してデータセットが小さすぎると、結果出力で警告メッセージが表示されます。
正の値と負の値	正の値と負の値を別々に分析すると、異常なデータはより明白になります。分析を始める前にフィルターを使用して、正と負の値を分けます。
ゼロおよび数値以外の文字	値がゼロのレコードは無視されますが、その除外されたレコード数は報告されます。 また、先頭にあるゼロ、小数点やドル記号などの数値書式、およびその他の数値以外の桁、さらにテスト基準を満たさないレコードも無視されます。結果として出された数字の桁数が指定した数より少ない場合は、Analyticsは結果の右側にゼロを追加します。

ベンフォード分析出力結果

ベンフォード分析では以下の出力結果が生成されます。

先頭桁	検証する先頭桁が表示されます。たとえば、1桁の先頭桁を指定した場合、1から9の数字が表示されます。2桁の先頭桁を指定した場合は、10から99の数字が表示されます。
実数	各先頭桁の総数か、フィールドに含まれる先頭桁の組み合わせの実際の総数が表示されます。
予測数	ベンフォードの法則に従って計算された、各先頭桁の予測数か、先頭桁の組み合わせの予測数が表示されます。
Z統計量	実際のカウントと予測されるカウントの間の距離の標準偏差の測定である、各数または数の組み合わせのZ統計量を示します。たとえば、Z統計量 0.500は標準偏差の半分を表します。
下限範囲 上限範囲 (省略可能)	各先頭の数または数の組み合わせのカウントの計算された上限値と下限値を示します。 出力結果の複数の桁または桁の組み合わせの実際のカウントが境界のいずれかを超える場合、データは操作された可能性があり、調査が必要になります。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ 下限と上限の値が含まれるのは、[ベンフォード]ダイアログボックスの [制限および下限を含める] チェックボックスをオンにした場合のみです。</p> </div>

手順

フィールドでベンフォード分析を実施し、1つ以上の先頭の数または数の組み合わせがベンフォード分布から大幅に逸脱するかどうかを検査します。

方法の説明

1. 分析するフィールドを含むテーブルを開きます。
2. **分析 > ベンフォード]**の順に選択します。
3. **メイン]**タブで、次のいずれかを実行します。
 - **ベンフォードの対象]**ドロップダウンリストから、分析の対象となるフィールドを選択します。
 - **ベンフォードの対象]**をクリックしてフィールドを選択するか、式を作成します。

メモ

取引金額などの「自然発生数」を含むフィールドを選択します。ベンフォード分析は、いかなる方法でも制約を受けた数値データには適していません。詳細については以下を参照 "ベンフォード分析を使用してテストできるデータ" ページ 1373

4. **分析する先頭桁数]**に、分析する桁数を 1 から 6 までの数値で入力します。

メモ

4 桁以上の先頭桁を分析する場合、結果はファイルへ出力する必要があります。4 桁以上の分析の結果は、画面へ表示、プリンターへ送信、またはグラフで表示することはできません。

5. 現在のビューの中に処理から除外したいレコードがある場合は、**[f]**テキスト ボックスに条件を指定します。直接入力するか、または **[f]** ボタンをクリックし、**式ビルダー**を利用して IF ステートメントを作成します。

メモ

If 条件は、任意の範囲オプション(**先頭、次へ、While**) が適用された後に、テーブルに残るレコードに対してのみ評価されます。

IF ステートメントは、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。

6. (省略可能) 各数または数の組み合わせの出力結果に計算された境界値を含める場合は、**[制限および下限を含める]**を選択します。
7. **出力]**タブをクリックします。
8. **出力先]**パネルで、適切な出力オプションを選択します。
 - **画面** - Analytics の表示領域に結果を表示するには、このオプションを選択します。

ヒント

表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。

出力レコードに多数のレコードが含まれている場合は、結果を画面に表示するよりファイルに保存した方が、処理が速く便利です。

- **印刷** - デフォルトのプリンターに結果を送信するには、このオプションを選択します。
- **グラフ** - 結果を基にグラフを作成し、それを Analytics 表示領域に表示するには、このオプションを選択します。

- **ファイル**- 結果を Analytics テーブルに保存または追加するには、このオプションを選択します。テーブルがまだプロジェクトに存在しないときは、開いているプロジェクトに追加されます。

メモ

当該の分析操作に対して適用されない出力オプションは無効になります。

- 出力の種類として **ファイル**]を選択した場合は、**出力指定**]パネルで次の情報を指定します。
 - **ファイルの種類** - オプションは "Analytics テーブル" のみです。結果を新しい Analytics テーブルに保存するか、または既存の Analytics テーブルに追加します。
 - **名前 - 名前**]テキスト ボックスにテーブル名を入力します。もしくは、**名前**]ボタンをクリックして、**保存**]または **ファイルを保存する名前**]ダイアログ ボックスでテーブル名を入力するか、既存のテーブルに上書きまたは追加する場合はそのテーブルを選択します。Analytics によってテーブル名があらかじめ設定されている場合は、その設定されている名前を受け入れることも、あるいは変更することもできます。

また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所にテーブルを保存したり、その場所にあるテーブルに追加したりすることもできます。たとえば、**C:\Results\Output.fil** または **Results\Output.fil** のように指定します。

メモ

Analytics テーブル名は、拡張子 .FIL を含まない 64 文字の英数字に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- **ローカル**- サーバー テーブルに接続している場合のみ有効になります。プロジェクトと同じ場所に出力テーブルを保存する、パスを指定する、または別のローカルフォルダーへ移動する場合は、**ローカル**]を選択します。サーバーの Prefix フォルダーに出力テーブルを保存する場合は、**ローカル**]を選択解除されたままにしておきます。

メモ

出力結果が AX Server テーブルの分析や処理から生成される場合は、**ローカル**]を選択します。結果テーブルを AX Server にインポートするために **ローカル**]設定を選択解除することはできません。

- 出力の種類によっては、任意で **ヘッダー**]および **フッター**]をそれぞれのテキスト ボックスに指定できます。

デフォルトでは、ヘッダーとフッターは中央揃えで出力されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行のヘッダーやフッターを入力する場合は、**ヘッダー**]または **フッター**]ボタンをクリックします。あるいは、ヘッダーまたはフッターのテキスト ボックスで、改行文字としてセミコロン (;) を入力してもかまいません。複数行を左揃えにするには、各行の先頭に始め山かっこが必要です。
- 詳細**]タブをクリックします。
- 適用範囲**]パネルで、適切な出力オプションを選択します。

- **すべて**
 - **先頭**
 - **次へ**
 - **While**
- 詳しく表示

すべて	デフォルトで、このオプションが選択されます。ビュー内のすべてのレコードを処理するように指定するには、選択されたままにしておきます。
先頭	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内の先頭レコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。
次へ	このオプションを選択してテキスト ボックスに数を入力すると、ビュー内で現在選択されているレコードから処理が開始され、指定した件数のレコードのみが処理対象とされます。行内のデータではなく、左端の列の実際のレコード番号が選択されている必要があります。
While	<p>WHILE ステートメントを使用して、特定の条件または条件のセットに基づいてビュー内のレコードの処理を制限するには、このオプションを選択します。 While テキスト ボックスに条件を入力するか、または While ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>WHILE ステートメントでは、指定した条件が true と評価される間のみ、ビュー内のレコードを処理することができます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。 While オプションは、"すべて"、"先頭"、または "次" オプションと組み合わせて使用することができます。1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p>	

13. 出力の種類として**ファイル(Analytics テーブル)**を選択した場合、操作の完了時に、自動的に出力テーブルを開く場合は、**出力テーブルを開く**]を選択します。
14. 出力タイプとして **ファイル**]を選択し、出力テーブルを既存の Analytics テーブルの最後に追加する場合は、次のいずれかを実行します。
 - 出力結果と既存のテーブルの構造が同じであることが確実な場合は、**既存のファイルに追加**]を選択します。
 - 出力結果と既存のテーブルのレコード長を Analytics に比較させたい場合は、**既存のファイルに追加する**]を選択解除されたままにしておきます。レコード長が同一でなければ、データ構造は同じではなく、追加は正しく動作しません。

メモ

出力結果と既存のテーブルの構造が同一であるかどうか不確かな場合は、**既存のファイルに追加する**]を選択解除されたままにしておくことをお勧めします。追加およびデータ構造の詳細については、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

15. **OK]**をクリックします。
16. 上書きを確認するメッセージが表示されたら、適切なオプションを選択します。
最後に追加]オプションが表示されることを期待していたのに表示されない場合は、**[いい]**をクリックして操作をキャンセルし、"既存テーブルへの出力結果の追加" ページ 207を参照してください。

R スクリプトの実行

Analytics テーブルを外部 R スクリプトで分析して返されたデータを使って、新しいテーブルを Analytics プロジェクト内に作成します。ソースデータはデータフレームとして R スクリプトに渡されます。このデータフレームは同梱の関数を使って参照できます。

R での Analytics データの操作

Analytics 内から実行する R スクリプトを作成する場合は、ACL と R スクリプトの間でデータをやり取りする方法に習熟してください。Analytics に同梱されている R 関数や、作成する R スクリプト内で使用することで、RCOMMAND を正常に実行できます。

詳しく表示

R スクリプトでの Analytics データの参照

Analytics テーブルは R データフレームとしてスクリプトに渡されます。データフレームはタブ形式のデータオブジェクトであり、データの異なるモードまたは型の列を含むことがあります。

R スクリプトで Analytics によって作成されたデータ型を操作するには、`acl.readData()` 関数を呼び出し、返されたデータフレームを変数に格納します。

```
# Analytics テーブルをデータ フレーム myTable に格納します。これはスクリプトを使用して参照  
# できます  
myTable<-acl.readData()
```

データフレームのセルからデータを取得するには、次のアプローチのうちの 1 つを使用できます。

- 行と列の座標の使用:

```
# データ フレームの 1 行 2 列の値を取得します。  
myTable[1,2]
```

メモ

座標はコマンドで指定されたフィールドの順序に基づいており、テーブルレイアウトや、現在開いているビューに基づいておりません。

- 行と列の名前の使用:

```
# データ フレームの 1 行 "myColumnTitle" 列の値を取得します。
myTable["1", "myColumnTitle"]
```

列の名前を使用するには、コマンドの **KEEPTITLE** オプションを指定する必要があります。

行は、"1"、"2"、"3" と命名されており、そのように増加します。名前と座標の組み合わせを使用することもできます。

Analytics へのデータの返送

データ フレームまたはマトリクスを Analytics に戻し、新しいテーブルを作成するには、次の構文を使用します。

```
# myNewTable データ フレームを Analytics に戻し、新しいテーブルを作成します
aql.output<-myNewTable
```

メモ

R スクリプトが終了するときに、データ フレームまたはマトリクスを Analytics に戻す必要があります。データ フレームまたはマトリクスの列には原子値のみが含まれ、リスト、マトリクス、配列、非原子オブジェクトがないことを確認します。値を Analytics データ型に変換できない場合は、コマンドが失敗します。

R データのマッピング

Analytics データ型は、Analytics プロジェクトと R スクリプトの間の翻訳プロセスを使用して、R データ型に変換されます。

Analytics データ型	R データ型
論理	論理
数値	数値
文字	文字
日付時刻	日付、POSIXct、POSIXlt

パフォーマンスとファイル サイズ制限

R スクリプトの実行と返されるデータの処理にかかる時間は、入力データが 1 GB を超える場合には長くなります。R では、2 GB 以上の入力ファイルはサポートされていません。

また、R に送信されるレコード数もパフォーマンスに影響します。ファイル サイズは同じだがレコード数が異なる 2 つのテーブルでは、レコード数が少ないテーブルの方が処理が高速になります。

複数バイト文字データの処理

中国語のような、複数バイトの文字セットでデータを R に送信する場合、R スクリプトにシステム ロケールを適切に設定する必要があります。複数バイトのデータのテーブルを R に正常に送信するには、R スクリプトの第 1 行に、次の関数が含まれている必要があります。

```
# ロケールを中国語に設定する例
Sys.setlocale("LC_ALL", "Chinese")
```

Sys.setlocale() の詳細については、R ドキュメントを参照してください。

「Hello world」の例

Analytics コマンド

```
RCOMMAND FIELDS "Hello", ", world!" TO "r_result" RSCRIPT "C:\scripts\r_scripts\analysis.r"
```

R スクリプト(analysis.r)

```
srcTable<-acl.readData()

# ACL に返送するテーブルを作成
output<-data.frame(
  c(srcTable[1,1]),
  c(srcTable[1,2])
)

# 列名を追加して ACL にテーブルを返送
colnames(output) <- c("Greeting", "Subject")
acl.output<-output
```

R スクリプトの実行

1. メニューから、**分析 > R**]を選択します。
[RCOMMAND] ダイアログ ボックスが表示されます。
2. **R スクリプト**]フィールドの横にある **参照**]をクリックして、コンピューターにある実行対象の R スクリプトに移動します。

3. **フィールドの選択**]をクリックし、Analytics によって R スクリプトからアクセスできるようにされるデータフレームに、1 つ以上のフィールドを追加します。

ヒント

データフレームには、フィールドとして式も追加することができます。式を作成するには、**式**]をクリックし、表示されるダイアログボックスで選択できる関数、フィールド、演算子を使用します。詳細については、「式ビルダーの概要」ページ 870を参照してください。

4. 省略可能。**RCommand オプション**のセクションでは、Analytics データを R スクリプトに送信する方法を定義します。

詳細については、「RCOMMAND のオプション」下を参照してください。

5. 省略可能。R スクリプトに送信されるレコードをフィルタリングするには、**[f]**]をクリックし、**式ビルダー**]ダイアログボックスを使って、フィルターとして使用する条件式を作成します。

式ビルダーを使用した式の作成方法の詳細については、「式ビルダーによる式の作成」ページ 872を参照してください。

6. 出力テーブルを指定するには、**転送先**]をクリックして、**ファイル名**]フィールドにテーブルの名前および関連する **.FIL** ファイルを入力します。

フォルダー エクスプローラーを使って、ソース データ ファイルの保存に使用したいフォルダーに移動します。

メモ

Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

7. 省略可能。ダイアログボックスの **詳細**]タブで、RCOMMAND の範囲関連オプションを指定します。

詳細については、「詳細]タブ」次のページを参照してください。

8. RCOMMAND を実行するには、**OK**]をクリックします。

RCOMMAND] ダイアログ ボックスのオプション

RCOMMAND のオプション

オプション	説明
フィールド名付きでエクスポートする	R スクリプトのデータフレームのヘッダー値として、ソースの Analytics テーブルの列見出しを使用します。このオプションは、RCOMMAND に対して KEEPTITLE オプションを設定します。また、このオプションは、R スクリプト内で列名を使ってデータを取得したい場合に必要です。

オプション	説明
列区切り	R スクリプトにデータを送信する際に、フィールド間の区切り文字として使用する文字。
テキスト修飾子	R スクリプトにデータを送信する際にフィールド値を識別するテキスト修飾子として使用する文字。

詳細]タブ

オプション	説明
すべて	ビュー内のすべてのレコード(デフォルトの選択対象)を処理します。
先頭	テーブル内の冒頭のレコードと、そのレコードから数えて指定された件数のレコードまでを処理します。
次へ	<p>テーブル内の現在選択されているレコードと、そのレコードから数えて指定された件数のレコードまでを処理します。</p> <p>メモ</p> <p>"先頭" または "次" オプションで指定されたレコード数は、テーブル内の物理的な順番またはインデックス順のレコードを参照するもので、ビューに適用されたフィルターやクイックソートは一切無視します。ただし、分析操作の結果ではすべてのフィルターを考慮します。</p> <p>ビューでクイックソートが実行されている場合、"次" は "先頭" のように動作します。</p> <p>注意</p> <p>現在のバージョンでは、RCOMMAND を実行する際、"次" の動作に問題があることが知られています。このオプションは使用しないでください。選択しているレコードに関係なく、レコード参照で最初のレコードが参照されるようにリセットされてしまう可能性があるためです。</p>
While	<p>WHILE ステートメントを使用すると、条件に基づいて主テーブル内のレコードの処理を制限できます。</p> <p>指定した条件が true と評価される間のみ、ビュー内のレコードが処理されます。条件が false と評価されるとすぐに処理が終了し、それ以上レコードは判断されません。詳細については、"式ビルダーによる式の作成" ページ 872を参照してください。</p>

調査結果のレポート作成

通常、Analytics で分析を完了した後には、調査結果を報告する必要があります。Analytics 標準のレポート機能を使用するか、Analytics データを Tableau などのサードパーティのレポートアプリケーションにインポートできます。

作成するレポートの種類は、分析の結果の表し方やレポートレイアウトの複雑さによって決まります。

Analytics を使って調査結果を報告

Analytics には、調査結果と分析を表すために利用できるレポートが2種類あります。

- Analytics レポート
- Analytics グラフ

Analytics レポート

Analytics で定義したビューに基づいて、テキスト レポートを作成できます。さまざまな設定可能なオプションを使用して、レポートの内容やレイアウトを決定できます。

詳細については、「Analytics レポートの書式設定および生成」 ページ 1387を参照してください。

Analytics グラフ

Analytics の一部の操作の出力としてグラフを生成できます。また、ビューで選択したデータをグラフ化することもできます。グラフツールバーには、グラフの書式を設定するためのさまざまなオプションがあります。

詳細については、「Analytics グラフを使った作業」 ページ 1393を参照してください。

サードパーティレポート アプリケーションを使用した調査結果の報告

ODBC 準拠のレポート アプリケーションを使用して、Analytics データに接続し、調査結果を報告できます。使用できるアプリケーション

- Tableau
- Microsoft Power BI Desktop
- Excel
- Crystal Reports
- Qlik
- MicroStrategy

詳細については、"サードパーティレポート アプリケーションからの Analytics への接続" ページ 1405を参照してください。

Analytics レポートの書式設定および生成

データの形式とビューの列に基づいて、Analytics で従来の表形式のレポートを作成できます。レポートにデータを表示する方法を制御するには、ビューのさまざまな設定を構成します。これらの設定はビューと一緒に保存されます。

ヒント

個別のビューを作成して、レポート設定を構成して保存できます。これは、データを表示するためのビューとは異なります。

初めてレポートを生成するとき、行間、ヘッダー/フッターテキスト、およびレポートの出力タイプなど、さらに多くのプロパティを指定することができます。これらのプロパティは Analytics プロジェクトの保存時に一緒に保存されます。

ビューを設定してレポートの書式を設定する

レポートの書式を設定するためにビューを設定するときには、画面上でデータを表示するビューをそのまま使用することもできますが、レポート用に新しいビューを作成することもできます。必要に応じて、同じデータセットに基づいて異なるレポートで異なる設定の複数のビューを作成することができます。

より見やすく、わかりやすいレポートを作成するために、多くの方法でビューを設定することができます。

データのフィルター	フィルターを作成して、関連のないレコードをビューから除去します。除外されたレコードはレポートに含まれません。たとえば、売上データのテーブルをフィルタリングし、関心のある店舗のみを含めることができます。
レコードのインデックス付け	インデックスを作成し、ビューの1つ以上の列を基準にレコードを並び替えます。
特定の列の選択	ビューへ列を追加する、またはビューから列を削除して、必要なデータのみを表示させます。物理データフィールドまたは演算フィールドをテーブルレイアウトに含めることができます。
列の整列	ビューの列を並び替え、必要な情報のシーケンスのみを表示することができます。
サブセクションの作成	ブレイク列を指定し、レポートを小計のセクションに分割します。 たとえば、請求書テーブルで Customer_Name(顧客名) フィールドをブレイク列として指定すると、顧客名でグループ化され、そのグループ単位で請求金額の小計を計算します。また、レポートの各グループの後にページを挿入することもできます。
データのカスタマイズ	繰り返される名前などの重複する識別子値を抑制し、意味のない数値の合計を抑制して、ゼロを空欄として表示します。

レポート幅の制御	ビューの各行を複数行に拡張します。
メモ	ビューの設定の詳細については、"ビュー内の列のカスタマイズ" ページ 849を参照してください。

レポート フォントの指定

印刷またはHTMLファイルへの個別のレポート出力のフォントを指定できます。

方法の説明

1. レポートに設定されたビューを開きます。
2. 表示領域の上で、**フォントの選択** **[Ff]** を選択します。
3. **ビューフォントの選択** ダイアログボックスで、書式設定するレポートの部分のボタンをクリックします。
4. **フォント** ダイアログボックスで、フォント情報を指定し、**OK** をクリックします。
5. フォントの指定が完了したら、**ビューフォントの選択** ダイアログボックスで、**OK** をクリックします。

指定されたフォントは、ビューから生成されたレポートで使用されます。

レポートの生成

レポートの基本として使用するビューを設定すると、レポートを生成できます。

方法の説明

手順

メモ

手順の後に詳細情報が表示されます。" [レポート] ダイアログボックスのオプション" 次のページを参照してください。

1. レポートに設定されたビューを開きます。
2. **データ > レポート** の順にクリックします。
3. 以下のテーブルをガイドとして使用し、**レポート** ダイアログボックスでオプションを指定してから、**OK** をクリックします。

レポートが生成されます。

4. (省略可能) 指定したレポート オプションを完全に保存する場合は、**ファイル > プロジェクトの保存** を選択します。

レポート オプションはビューとともに保存され、次回同じビューからレポートを生成するときにあらかじめ選択されます。

ヒント

同じレポート オプションを使用して、同じビューから後続のレポートをすばやく生成するには、**[ファイル > 印刷]**を選択します。

[レポート] ダイアログ ボックスのオプション

次の表は、[レポート] ダイアログ ボックスのオプションの詳細を示します。

[メイン] タブ

オプション - [レポート] ダイアログ ボックス	説明
ヘッダー フッター 省略可能	<p>レポート ヘッダーおよびフッターを作成します。</p> <p>ヘッダーおよびフッターは複数行の入力が可能で、デフォルトで中央に配置されます。ヘッダーテキストやフッターテキストを左揃えにするには、テキストの前に始め山かっこ (<) を入力します。複数行を左揃えにするには、各行の先頭に山かっこが必要です。</p> <p>Analytics のセッション中に生成されるすべてのレポートに適用される標準のヘッダーとフッターを指定することもできます。詳細については、「標準のヘッダーおよびフッターを指定する」ページ 1392を参照してください。</p>
もし次の 省略可能	<p>レポートに表示されるレコードを指定する条件を作成します。</p> <p>[F] テキスト ボックスに条件を入力するか、または [F] ボタンをクリックし、式ビルダーを利用して WHILE ステートメントを作成することができます。</p> <p>IF 条件は、ビュー内のすべてのレコードを判断し、指定された条件を満たさないレコードを除外します。</p>
事前並べ替え 省略可能	<p>レポートの並べ替え条件：</p> <ul style="list-style-type: none"> ブレーク列 - 列プロパティで ブレーク列 が選択されたビューの任意の列 列を並べ替える - 列プロパティで キー列を並べ替える が選択されたビューの任意の列 <p>ネストされた並べ替えは、左端の列から、ビューのブレーク列または並べ替え列の順序を使用します。</p> <p>メモ この あらかじめ並べ替える オプションは、ビューの1つ以上の列で キー列を並べ替える オプションまたは ブレーク列 オプションが選択されている場合にのみ利用できます。 詳細については、「列プロパティの変更」ページ 854を参照してください。</p>
要約 省略可能	<p>キーブレークフィールドの小計と合計のみを含んで詳細行を除外するレポートが生成されません。</p>
詳細の空白行を出力しない 省略可能	<p>レポートから空白の明細行を自動的に削除します。</p>

オプション - [レポート]ダイアログボックス	説明
シングルスペース ダブルスペース トリプルスペース	レポートの間隔を指定します。
設定	さまざまな印刷のオプションを指定します。
プレビュー 省略可能	レポートのプレビューを表示します。
ページ幅に合わせる 省略可能	<p>レポートを調整し、ビューのすべての列を含めます。</p> <p>ページ幅に合わせる オプションを選択しない場合は、ビュー内のページ幅インジケータ (垂直点線) より左側にある列のみが印刷レポートに表示されます。</p> <p>メモ ファイルとして保存されるレポート、または画面に表示されるレポートには、ビュー内のすべての列が自動的に含まれます。</p>

出力]タブ

オプション - [レポート]ダイアログボックス	説明
パネルへ	<ul style="list-style-type: none"> 画面 -- Analytics の表示領域にレポートを表示します。レポートに多数のレコードが含まれている場合は、レポートを画面に表示するよりファイルに保存した方が、処理が速く便利です。 印刷 - 通常使うプリンターにレポートを送信します ファイル - レポートをテキスト ファイルまたは HTML ファイルに保存または追加します。ファイルは Analytics の外部に保存されます。
パネルとして	<p>ファイル出力のみ。</p> <ul style="list-style-type: none"> ファイルタイプ - テキスト ファイルまたは HTML ファイルへの出力を指定します。 名前 - 出力ファイルの名前を指定します。 <p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> 名前] テキスト ボックスにファイル名を入力します 名前] をクリックしてファイル名を入力します 名前] をクリックして、保存] または 名前を付けて保存] ダイアログボックスで既存のファイルを選択し、ファイルを上書きするか、ファイルの最後に追加します <p>また、絶対ファイルパスや相対ファイルパスを指定したり、別のフォルダーへ移動したり、プロジェクトの場所以外の場所に保存したり、その場所にあるファイルに追加したりすることもできます。たとえば、C:\Results\Report.txt または Results\Report.htm のように指定します。</p> <ul style="list-style-type: none"> ローカル - 選択された状態で無効になっています。ファイルをローカルに保存するのが唯一のオプションです。

オプション - [レポート]ダイアログボックス	説明
ヘッダー フッター 省略可能	<p>[メイン]タブからヘッダーまたはフッターテキストを複製します。</p> <p>必要に応じて、[出力]タブでヘッダーまたはフッターテキストを追加するか、既存のヘッダーまたはフッターテキストを更新できます。既存のテキストの変更は、自動的に、[メイン]タブで更新されます。[ヘッダー]または[フッター]テキストボックスでセミicolon(;)を入力し、改行を作成します。</p>
OK	<p>レポートを生成します。</p> <p>レポートをファイルに保存する場合に、上書きを確認するメッセージが表示されたら、適切なオプションを選択します。</p>

グローバルレポート オプション

複数のレポート オプションをグローバルで指定できます。グローバルオプションのほとんどは、印刷するレポート出力にのみ適用されます。

追加情報の追加

Analytics テーブル名 やフィールド定義情報などの追加情報を、印刷されたレポートの最後に追加することができます。[オプション]ダイアログボックスで追加情報のオプションを指定します。

方法の説明

- [ツール > オプション > 印刷]の順に選択します。
- 以下の任意のオプションを指定し、[OK]をクリックします。

指定された情報は、生成するすべての印刷レポートの最後に表示されます。

オプション	説明
レポートにレポート履歴を含める	<p>次の情報を印刷レポートの最後に追加します。</p> <ul style="list-style-type: none"> Analytics プロジェクト、テーブル、データファイル名 レポートを生成するためのREPORTコマンド テーブル履歴情報 任意のテーブルレイアウト ノート
テーブル履歴にフィールド定義を含める	<p>次の情報を印刷レポートの最後に追加します。</p> <ul style="list-style-type: none"> テーブルレイアウト内の各物理データフィールドおよび演算フィールドのフィールド定義がレポートに追加されます。 任意のフィールド ノート <p>このオプションは、[レポートにレポート履歴を含める]オプションも選択する必要があります。</p>
レポート履歴にビューノート	<p>次の情報を印刷レポートの最後に追加します。</p>

オプション	説明
トを含める	<ul style="list-style-type: none"> ビュー名 任意のビューノート <p>このオプションは、[レポートにレポート履歴を含める]オプションも選択する必要があります。</p>
<p>メモ</p> <p>レコード ノートをレポートに含める場合は、まず、ecordNote 列をビューに追加する必要があります。レコード ノートは、グローバルレポート オプションの影響は受けません。</p>	

余白の設定

印刷されたレポートで使用される余白を指定できます。

方法の説明

- [ツール > オプション > 印刷] の順に選択します。
- 余白** テキストボックスを使用して、印刷されたレポートの余白を指定します。

指定された余白は、生成するすべての印刷レポートで使用されます。

標準のヘッダーおよびフッターを指定する

Analytics のセッション中に生成されるすべてのレポートに適用される標準のヘッダーとフッターを指定できます。指定するヘッダーとフッターは、HEADER および FOOTER システム変数に格納されます。

方法の説明

- Analytics コマンドラインで、`HEADER = "ヘッダー テキスト"`、または `FOOTER = "フッター テキスト"` を入力します。

複数行のヘッダーまたはフッターを作成するには、行をセミコロンで区切ります。例：

```
HEADER = "ヘッダー行 1;ヘッダー行 2"
```

- Enter キーを押します。

ここで指定した値は、Analytics 操作の実行中に指定されたヘッダーやフッターによって上書きされない限り、自動的にすべての Analytics レポートのヘッダーまたはフッターとして使用されます。

HEADER および FOOTER の値は、変数が更新または削除されるまで、もしくは現在の Analytics セッションが終了するまで有効です。

Analytics グラフを使った作業

Analytics データは、さまざまな種類のグラフを用いて表示することができます。テーブルデータからグラフを生成するには、次の2つの方法があります。

- **コマンドの結果からグラフを生成する** - [分析]メニューにある以下のコマンド、およびこれらに相当する ACLScript コマンドでは、結果をグラフとして出力するオプションが使用できます。
 - 年齢調べ
 - 分類化
 - クロス集計
 - ヒストグラム
 - ベンフォード解析
 - 階層化

適切なコマンドを含むスクリプトでは、そのコマンドの結果をグラフで表示することができます。これにより、時間の経過に伴い変化するデータの分析に、グラフベースのレポート生成を自動化することができます。

コマンドの結果からグラフを生成した場合、グラフ化されたデータのサブセットを分析するためにドリルダウン オプションを使用することができます。分析対象のグラフデータをドリルダウンすると、Analytics のフィルターが作成され、選択されたレコードのサブセットが Analytics ビューに表示されます。

- **ビューで選択したデータを基にグラフを生成する** - Analytics ビューでデータを選択し、そのデータに対応するグラフを作成することができます。Analytics ビューでデータを選択して作成したグラフの場合、ドリルダウン オプションは利用できません。これは、そのデータのサブセットが既にビューで選択されているからです。

グラフの書式設定の変更

グラフツールバーには、グラフレイアウトのある特定の側面を変更するために使用できるさまざまなコマンドがあります。また、以下の一般的な Windows 操作を実行すれば、**グラフ** タブまたはグラフ ウィンドウ (**選択データのグラフ化**) 内でグラフを直接編集することもできます。

- グラフまたはグラフの凡例をクリックして(サイズ変更) ハンドルを表示し、1 つのハンドルをドラッグしてその項目のサイズを変更する
- 項目をクリックして選択し、その項目をドラッグして移動する
- グラフまたはグラフの凡例をダブルクリックして、**プロパティ** ダイアログ ボックスを表示する

グラフの表示領域

分析 メニューの分析用コマンドを使用してグラフを生成する場合は、Analytics 表示領域の新しいタブでグラフが表示されます。グラフのヘッダーには、そのグラフを生成するために実行した ACLScript コマンド名が表示され、フッターにはそのコマンドの完全な構文が表示されます。

ビューからグラフを直接生成する場合は、別 ウィンドウが開き、そのウィンドウでグラフが表示されます。グラフのヘッダーには "ビューの列をグラフ化" というテキストが表示されます。

コマンドおよびビューのどちらから生成されたグラフでも、その表示領域には同じ機能が含まれています。ただし、ビューから生成したグラフではツールバーのドリルダウン オプションが無効になっています。

グラフ ツールバー

グラフのヘッダーの上部にあるグラフ ツールバーには、以下のことを行うことができるボタンがあります。

- グラフのさまざまな表示設定
- グラフの保存、コピーまたは印刷
- グラフ データの特定のサブセットを分析するため、該当するビューにドリルダウン

グラフの生成に使用する方法および表示されるグラフの種類によっては、利用できないツールバー機能もあります。たとえば、ビューでデータを選択してグラフを生成する場合はドリルダウン コマンドが使用できません。

アイコン	名前	説明
	グラフの種類	表示するグラフの種類を選択します。
	グラフプロパティ	フォント、背景、フレームなどのグラフプロパティや、グリッド線、軸または凡例をグラフ表示に含めるかどうかを指定するオプションを編集します。
	凡例プロパティ	グラフに表示される各フィールドのフォント、境界線および配色などの凡例プロパティを編集します。

アイコン	名前	説明
	軸プロパティ	フォント、スタイルおよび目盛りなど、軸のプロパティを編集します。
	データの書式設定	フォント、グラフに含めるフィールド、データ系列の文字の方向、ラベルおよび配色など、データ書式のプロパティを編集します。
	ラベルプロパティ	フォント、境界線および方向など、ラベルのプロパティを編集します。
	凡例を表示する/しない	グラフの凡例の表示/非表示を切り替えます。
	軸を表示する/しない	グラフ軸の表示/非表示を切り替えます。
	グラフの印刷	インストールされている Windows 印刷 デバイスにグラフを出力します。
	グラフをビットマップとして保存	グラフをビットマップ イメージ ファイルとして保存します。
	グラフをクリップボードにコピー	グラフを別のアプリケーションに貼り付けるために、クリップボードにコピーします。
	コマンドの編集	グラフを生成するために実行したコマンドを編集します。
	グラフを左に回転	円グラフを左へ回転します。
	グラフを右に回転	円グラフを右へ回転します。
	ドリルダウン	分析のために、選択したグラフ セグメントをテーブルビューで開きます。

グラフの種類の変更

デフォルトのグラフタイプは3D棒グラフです。作業しているデータの種類によって、次に挙げるフォーマットでグラフを表示することができます。グラフのデータに適用されるオプションのみが使用できます。適用しないオプションは使用できません。

メモ

Analytics で生成されるデフォルトのグラフタイプを変更する場合は、コマンドラインで適切な SET GRAPH コマンドを入力します。

アイコン	グラフの種類	アイコン	グラフの種類	アイコン	グラフの種類
	2D 棒グラフ		3D 積み上げグラフ		3D 階層グラフ
	3D 棒グラフ		2D 円グラフ		折れ線グラフ
	2D 積み上げグラフ		3D 円グラフ		ベンフォード グラフ

方法の説明

1. **グラフの種類**  をクリックします。

メモ

グラフ出力の結果に表示される項目が1つしかない場合は、積み上げと3D階層のオプションが淡色表示になります。複数の項目がグラフに表示される場合は、積み上げと3D階層のオプションは使用できますが、円グラフのオプションは使用不可になります。

2. 利用可能なグラフフォーマットの中から1つをクリックします。
3. **OK** をクリックします。

ヒント

円グラフ内の個々のデータ区分を切り離して表示するには、データ区分を右クリックして **円グラフの切り出し** を選択します。

グラフプロパティの変更

グラフプロパティ ダイアログボックスでは、グラフの背景色、軸のプロパティ、および枠線を変更することができます。また、グラフ上のすべてのテキスト要素に使用されているフォントを変更することもできます。

方法の説明

1. **グラフプロパティ**  をクリックします。
2. 変更するグラフプロパティがあるタブをクリックします。
 - **グローバルフォント** -- グラフ上のすべてのテキスト要素のフォント、スタイル、サイズ、色、または背景色の設定を変更します。
 - **背景** -- グラフ表示領域の境界線や背景色の設定を変更します。

- フレーム -- グラフ化されたデータを含むフレームの境界線や背景色の設定を変更します。
 - オプション -- グリッド線、軸および凡例の表示オプションを選択します。
3. [OK] をクリックします。

グラフの凡例プロパティの変更

[凡例プロパティ] ダイアログボックスでは、凡例のフォントや色を変更したり、凡例ボックスをカスタマイズすることができます。

方法の説明

1. [凡例プロパティ]  をクリックします。
2. 変更する凡例プロパティがあるタブをクリックします。
 - 凡例の属性 -- 境界線のスタイル、色、太さなどの属性を変更します。
 - フォント -- 凡例テキストのフォント、スタイル、サイズ、色または背景色の設定を変更します。
 - データ系列 -- フィールドのグラフ表示の配色、および表示オプション(透明または塗りつぶし)の設定を変更します。
3. [OK] をクリックします。

メモ

凡例を右クリックして、凡例に表示するフィールドを選択する、あるいはその凡例を非表示にすることができます。ツールバーの [凡例を表示する/しない] をクリックすると、凡例の表示/非表示を切り替えることもできます。

グラフの軸プロパティの変更

[軸プロパティ] ダイアログボックスを使用して、グラフ軸のスタイル、目盛り、およびフォントを変更することができます。

方法の説明

1. [軸プロパティ]  をクリックします。
2. 変更する軸プロパティがあるタブをクリックします。
 - 軸スタイル -- スタイル、色、太さ、および軸上を一定の間隔で線引きする目盛りオプションなどのプロパティを変更します。
 - 目盛り軸 -- 軸上に表示する最小値と最大値、および目盛り間隔を指定します。[自動]を選択すると、これらの値が自動的に割り当てられます。表示するよう選択したテキスト値の表示方向を指定することもできます。
 - フォント -- 軸テキストのフォント、スタイル、サイズ、色または背景色の設定を変更します。

3. [OK]をクリックします。

ヒント

軸を右クリックして、軸の表示を横方向または縦方向に切り替えることができます。ツールバーの **軸を表示する/しない** をクリックすると、軸の表示/非表示を切り替えることもできます。

グラフデータ表示プロパティの変更

データの書式設定 ダイアログボックスを使用すると、グラフに表示されるフィールドを選択したり、各フィールドに割り当てられる色を変更したりすることができます。たとえば、表示するフィールドを減らす、またはデフォルトで非表示になっていたフィールドを表示させることができます。

また、データラベルのフォント サイズ、スタイル、色、方向などを変更することもできます。あるいはデータラベルをまとめて非表示にすることも可能です。

方法の説明

1. **データの書式設定**  をクリックします。
2. 変更するデータ表示プロパティがあるタブをクリックします。
 - **データ** -- グラフに表示するフィールドを **使用可能なフィールド** リストから選択して、**選択済みのフィールド** リストへ移動します。[上へ] ボタンおよび [下へ] ボタンを使用すると、選択したフィールドがグラフや凡例で表示される順序を変更することができます。

ヒント

グラフ表示領域で右クリックし、**フィールドの選択** を選択して、グラフに表示するフィールドを選択することもできます。

- **オプション** -- ラベルテキストの表示方向オプションを変更します。
 - **フォント** -- x 軸上のテキストのフォント、スタイル、サイズまたは色を変更します。
 - **データ系列** -- フィールドのグラフ表示の配色、および表示オプション(透明または塗りつぶし)の設定を変更します。
3. [OK]をクリックします。

グラフラベルの追加

グラフラベルは、必要に応じて追加および移動させることができます。

方法の説明

1. グラフ領域内のラベルを置く位置で右クリックし、**ラベルの追加** を選択します。
2. テキスト ボックスにラベルテキストを入力します。
3. [OK]をクリックします。

4. ラベルを移動させる場合は、以下の手順で行います。
 - a. ラベルをクリックして、そのラベルボックスの境界線が表示されるようにします。
 - b. ラベルボックスの境界線上にマウスポインターを置くと、そのポインターが4方向を指す矢印に変わります。
 - c. そのラベルボックスをグラフ上の目的の移動先までドラッグします。

グラフラベルプロパティの変更

[ラベルプロパティ]ダイアログボックスを使用すると、ラベルプロパティを変更することができます。ラベルは、コマンドまたはビューから生成されたグラフの要素についての説明を表すために使用できます。グラフ表示領域で右クリックして [ラベルの追加] を選択すると、ラベルを新規作成することができます。

方法の説明

1. 変更するラベルをクリックして、[ラベルプロパティ - 2. 変更するラベルプロパティがあるタブをクリックします。
 - **方向** -- ラベルの表示方向を角度で指定、あるいはラベルテキストの横方向または縦方向の配置を変更します。
 - **フォント** -- 選択したラベルのテキストのフォント、スタイル、サイズ、色または背景色の設定を変更します。
 - **属性** -- ラベルの境界線のスタイル、色、太さおよび影付きなどの属性を変更します。
- 3. [OK] をクリックします。

ヒント

ラベルを右クリックすると、そのラベルを切り取り/コピー/貼り付け、または削除することができます。

円グラフの回転

円グラフは左方向または右方向へ回転させることができます。回転オプションを使用できるのは、円グラフの場合のみです。

方法の説明

1. 円グラフを回転させる方向によって、以下のいずれかの手順を実行します。
 - 左へ回転させるには、**グラフを左に回転**  - 右へ回転させるには、**グラフを右に回転** 
2. 必要に応じて上記の動作を繰り返し、円グラフを 22.5 度ずつ回転させます。

グラフの元データへのドリルダウン

ドリルダウン オプションを使用すると、グラフで選択したデータ区分に対応するデータをビューに表示することができます。Analytics はグラフを使って、選択したグラフデータのみを表示するフィルターを定義し、フィルターリングされたデータを開いているビューに表示します。

メモ

ドリルダウン機能は、ビュー内でデータを選択して作成したグラフには利用できません。この制限があるのは、そのグラフの作成に使用されたデータ自体が、既にビューで選択されたものであるからです。

グラフのドリルダウン機能を使用するには

1. 次のいずれかを実行します。
 - 棒グラフまたは円グラフで、1つのデータ区分をクリックして選択します。
 - 棒グラフまたは円グラフで、隣接している複数のデータ区分をまとめて選択するには、まず最初のデータ区分をクリックし、次に **Shift** キーを押したまま、最後のデータ区分をクリックします。
 - 棒グラフまたは円グラフで、隣接していない複数のデータ区分を選択するには、**Ctrl** キーを押したまま、それら選択対象のデータ区分をクリックします。
2. **ドリルダウン**  をクリックします。

ヒント

1つのデータ区分に対するテーブルデータを分析する場合は、そのデータ区分をダブルクリックするだけでドリルダウンできます。

グラフで選択したデータ区分に対応するフィルターリングされたデータのビューが表示されます。このデータに対し、さらにほかの Analytics コマンド や関数を使用して分析を続けることができます。元のテーブルビューに戻す場合は、**フィルターの削除** をクリックします。

グラフコマンドの編集

コマンドの編集 ボタンを使用すると、グラフを生成したコマンドを編集することができます。

メモ

この **コマンドの編集** ボタンは、グラフが Analytics コマンドによって生成された場合のみ有効になります。

グラフの生成に使用されたコマンドを編集するには

1. **コマンドの編集**  をクリックします。
2. 変更するコマンド パラメーターがあるタブをクリックします。たとえば、分類化 (Classify) コマンドを使用してグラフが生成されていた場合は、**分類化** ダイアログボックスの **メイン** タブと **詳細** タブで設定を変更し、変更後のパラメーターでグラフを新たに生成することができます。
3. **OK** をクリックします。

グラフをクリップボードへコピー

グラフを別のドキュメントに貼り付けるために、クリップボードにコピーすることができます。グラフのクリップボードへのコピーが失われる前に、ビットマップファイルを読み込むことができるアプリケーションを開き、そのファイルにグラフを貼り付けます。

グラフをクリップボードへコピーするには

- **グラフをクリップボードにコピー**  をクリックします。

グラフをイメージとして保存

グラフをビットマップファイルとして保存することで、別のドキュメントに挿入したり、グラフ化したデータのスクリーンショットとして保持したりすることができます。

グラフを保存するには

1. **グラフをビットマップとして保存**  をクリックします。
2. **[グラフを保存する名前]** ダイアログボックスで、保存先のフォルダーを指定し、保存するグラフの名前を入力します。
3. **保存** をクリックします。

グラフの印刷

グラフの印刷]コマンドを使用すると、インストールされているプリンター デバイスでグラフを印刷することができます。グラフを最大解像度で印刷するには、印刷前にグラフ ウィンドウを最大化してください。

グラフを印刷するには

1. **グラフの印刷**  をクリックします。
2. **印刷**]ダイアログ ボックスで、適切なプリンター デバイスを選択し、印刷のプロパティを変更し、必要であれば印刷部数を指定します。
3. **OK**]をクリックします。

サードパーティレポート アプリケーションからの Analytics への接続

ODBC 準拠のレポート アプリケーションを使用して、Analytics データに接続し、調査結果を報告できます。使用できるアプリケーション

- Tableau
- Microsoft Power BI Desktop
- Excel
- Crystal Reports
- Qlik
- MicroStrategy

動作

選択したレポート アプリケーションで、ODBC 機能を使用して、任意の Analytics プロジェクトへの接続を ACL Connector for Analytics を使用して作成します。

接続されると、Analytics プロジェクトからテーブルとフィールドを選択し、結合やフィルタリングなどのサードパーティアプリケーションで使用できる追加の ODBC 機能を使用できます。

メモ

サードパーティレポート アプリケーションのデータ接続機能は異なります。たとえば、一部のアプリケーションでは、結合する複数のテーブルに接続する必要がありますが、別のアプリケーションでは複数のテーブルへの個別の接続をサポートします。

ACL Connector for Analytics はローカル Analytics テーブルへの接続のみをサポートします。Analytics プロジェクトのサーバー テーブルへの接続はサポートされません。

最適なパフォーマンス

ACL Connector for Analytics から最適なパフォーマンスを得るには、接続する Analytics データセットのサイズを制限します。コネクタはさまざまなレポート ツールをサポートするように設計されていますが、ソース データ テーブル全体ではなく、結果に共通のより小さいデータセットでのみ動作することを意図しています。接続処理でテーブルを結合すると、特にリソースを消費します。

Analytics データはフラットファイルに格納されるため、データベースよりも ODBC でのアクセス速度が低下します。

データ接続の作成 (DSN)

使用するレポート アプリケーションによっては、最初に Windows でデータ接続を作成してから、Analytics にレポート アプリケーションから接続する必要があります。

例：

- Microsoft Power BI Desktop では、手動で接続文字列を入力せずすむように、最初にデータ接続を作成します。
- Tableau と Excel では、自動的に接続が作成されるため、データ接続を作成する必要がありません。

データ接続は DSN で、データソース名を表します。

ヒント

異なる Analytics プロジェクトへの複数の接続を保存する方法として、Tableau または Excel 用のデータ接続を手動で作成することができます。

1. Windows オペレーティングシステムの管理ツールフォルダーから、接続する第三者のアプリケーションのビット (32 ビットまたは 64 ビット) に対応する ODBC データソース管理者のバージョンを開きます。

たとえば、32 ビット版の Excel から接続する場合は、32 ビットの ODBC Data Source Administrator を開きます。

注意

間違ったバージョンの ODBC Data Source Administrator でデータ接続を作成すると、サードパーティアプリケーションで接続が表示されないか、アクセスできません。あるいは、アクセスできても、接続エラーが発生します。

2. ODBC Data Source Administrator で次のいずれかを実行します。

- コンピューターを使用する全員がデータ接続を使用できるようにする場合は、**[システム DSN]** を選択します。
- 自分だけがデータ接続を使用する場合に、**[ユーザー DSN]** を選択します。

メモ

デフォルト Analytics データ接続 **ACL ODBC** または **ACL ODBC 64** は既に **[システム DSN]** タブに存在します。このデフォルト データ接続を修正しないでください。詳細については、"デフォルト Analytics データ接続" 次のページを参照してください。

3. **[追加]** をクリックし、**[ACL Connector for Analytics]** を選択し、**[完了]** をクリックします。
4. **[ACL データストア インターフェイス DSN]** ダイアログ ボックスで、次の情報を入力します。
 - **データソース名** - "Analytics 勘定元帳プロジェクト" などのわかりやすい名前を入力します。

- **説明** - "勘定元帳監査 2017" などの Analytics プロジェクトのわかりやすい説明を入力します。
 - **ACL プロジェクト ファイル- 参照]**をクリックして、**プロジェクト ファイルを開く]**ダイアログ ボックスで Analytics プロジェクトを選択します。
5. **OK]**をクリックします。
- 指定した Analytics プロジェクトへの新しいデータ接続が作成され、サードパーティレポート アプリケーションで選択できます。
- 必要に応じて、他の Analytics プロジェクトへの追加のデータ接続を作成できます。
6. **OK]**をクリックし、**ODBC Data Source Administrator**を終了します。

デフォルト Analytics データ接続

Analytics をインストールするには、次の名前の 32 ビットまたは 64 ビット接続 (DSN) がコンピューターで作成されます。

- **ACL ODBC (32 ビット)**
- **ACL ODBC 64 (64 ビット)**

これらは指定されていない Analytics プロジェクトがある Analytics 接続です。これらを使用して、異なる Analytics プロジェクトにその場で接続できます。つまり、Analytics プロジェクトを選択して、接続する時点で接続できます。一部のレポート アプリケーションではこの使用をサポートしていない場合があります。特定の Analytics プロジェクトを事前に指定したデータ接続が必要な場合があります。

メモ

別の Analytics プロジェクトにその場で接続する機能を維持する場合は、デフォルトのデータ接続のいずれかに接続情報を追加しないでください。

Analytics プロジェクトへの接続

以下の手順は、サードパーティアプリケーションを使用して Analytics プロジェクトに接続する 3 つの例を示します。

- "Tableau Desktop 10.1 から接続" 見開きページ
- "Microsoft Power BI Desktop 2.42 から接続" ページ 1409
- "Excel から接続" ページ 1410

メモ

これらの手順は一般的なガイダンスのみを提供し、指定されているサードパーティアプリケーションのバージョン固有です。

サードパーティアプリケーションでの ODBC 接続の作成の詳細については、アプリケーションのヘルプを参照してください。

Tableau Desktop 10.1 から接続

1. Tableau の **Connect**] パネルの **To a Server**] の下で **More**] をクリックします。
2. **他のデータベース(ODBC)**] をクリックします。
3. **他のデータベース(ODBC)**] ダイアログボックスで、**DSN**] を選択し、**DSN**] ドロップダウンリストで次のいずれかを選択します。

- Analytics プロジェクトへの既存のデータ接続を使用するには、接続名を選択し、**接続**] をクリックします。

メモ

既にデータ接続を作成している必要があります。詳細については、"データ接続の作成 (DSN)" ページ 1406を参照してください。

- Analytics プロジェクトへのデータ接続をその場で作成するには、**ACL ODBC** または **ACL ODBC 64** を選択し、**接続**] をクリックします。

メモ

ACL ODBC と **ACL ODBC 64** の両方が表示される場合は、Tableau のバージョンのビット数 (32ビットまたは64ビット) と一致する方を選択します。詳細については、"デフォルト Analytics データ接続" 前のページを参照してください。

4. **ACL ODBC** または **ACL ODBC 64** を選択した場合は、**プロジェクト ファイルを開く**] ダイアログボックスで、Analytics プロジェクト (.acl) に移動して、ファイルを選択し、**開く**] をクリックします。
5. **他のデータベース(ODBC)**] ダイアログボックスで **サインイン**] をクリックします。

Tableau は Analytics プロジェクトに接続します。

6. 省略可能。同時に複数の Analytics プロジェクトに接続する場合は、**データソース**] タブの **接続**] パネルで **追加**] をクリックし、手順 2~5 を繰り返します。
7. **データソース**] タブの **データベース**] ドロップダウンリストで、接続する Analytics プロジェクトを選択します。

複数の Analytics プロジェクトに接続する場合は、最初に **接続**] パネルで該当するプロジェクトを選択します。

8. **テーブル**] パネルで、次の一方を実行します。

- Analytics プロジェクトのテーブルをすべて一覧表示するには: **検索** ] をクリックします。
- 特定のテーブルを検索するには: テーブルの名前を入力し、**Enter** を押します。

ヒント

テーブル名は大文字と小文字を区別します。

完全一致検索ではどのテーブルも返されない場合は、**含むまたはで始まる**を試してください。

9. 返された Analytics テーブルを作業エリアにドラッグします。
10. データプレビュー エリアで **今すぐ更新**] をクリックし、Analytics テーブルのデータを表示します。
11. 必要な追加タスクを実行します。

- 別のテーブルを作業領域に追加し、結合します
必要に応じて、別の Analytics プロジェクトからテーブルを結合します。

- データのフィルター
- フィールド名を更新

テーブルまたは他のデータ準備タスクの詳細については、Tableau Desktop ヘルプを参照してください。

メモ

1 つまたは両方のテーブルが大きい場合、結合の速度が低下する場合があります。

12. **ファイル > 保存**] をクリックし、Tableau ワークブックを保存します。

Microsoft Power BI Desktop 2.42 から接続

1. Power BI ホーム タブで **データの取得**] ドロップダウン リストをクリックし、**その他**] を選択します。
2. **データの取得**] ダイアログ ボックスで、**その他**] > **ODBC**] を選択し、**接続**] をクリックします。
3. **ODBC から**] ダイアログ ボックスで、次のいずれかを実行します。

- Analytics プロジェクト への既存のデータ接続を使用するには、**データソース名 (DSN)**] ドロップダウン リストから接続名を選択し、**OK**] をクリックします。

メモ

既にデータ接続を作成している必要があります。詳細については、「データ接続の作成 (DSN)」 ページ 1406 を参照してください。

- Analytics プロジェクト へのその場でのデータ接続を作成するには、**ACL ODBC** または **ACL ODBC 64** を **データソース名 (DSN)**] ドロップダウン リストから選択し、**詳細オプション**] をクリックして、該当する **接続文字列** を入力し、**OK**] をクリックします。

接続文字列は次の形式を使用する必要があります。DBF=;DBQ=<アナリティクス プロジェクトパスと filename.acl>

例: DBF=;DBQ=C:\Users\john_smith\Documents\ACL Data\Sample Data Files\Sample Project.acl

メモ

ACL ODBCとACL ODBC 64の両方が表示される場合は、Power BIのバージョンのビット数(32ビットまたは64ビット)と一致する方を選択します。詳細については、「デフォルト Analytics データ接続」ページ 1407を参照してください。

4. **[ODBCドライバー]**ダイアログボックスが表示される場合は、次を実行します。
 - a. **[Windows]**を選択します。
 - b. **[現在の資格情報を使用する]**を選択します。
 - c. **[接続]**をクリックします。
Power BIはAnalyticsプロジェクトに接続します。
5. **[ナビゲーター]**ダイアログボックスで、Analyticsプロジェクトテーブルを含むノードを展開し、接続先の1つ以上のテーブルを選択します。
テーブルをハイライトすると、テーブルデータのプレビューが右側のプレビューペインに表示されます。
6. 次のいずれかを実行します。
 - **[読み込み]**をクリックして、選択したテーブルをPower BIに読み込みます。
 - **[編集]**をクリックして、ODBCクエリを編集します。編集が完了したら、**[閉じて適用]**をクリックします。
複数のテーブルが個別にPower BIに読み込まれます。必要に応じてPower BIでテーブルを関連付けることができます。場合によっては、テーブルの関連付けが自動的に生成されます。
テーブルの関連付けまたはODBCクエリの編集については、Power BI Desktopヘルプを参照してください。
7. 省略可能。同時に複数のAnalyticsプロジェクトに接続する場合は、手順1～6を繰り返します。
必要に応じてPower BIで異なるAnalyticsプロジェクトのテーブルを関連付けることができます。
8. Power BIファイルを保存します。

Excel から接続

メモ

Excelからの接続手順は、Excelのバージョンによっては、以下の手順とは少し異なる場合があります。

1. Excel **データ** タブで **他のソースから** ドロップダウンリストをクリックし、**[Microsoft Query]** からを選択します。
2. **[データソースの選択]** ダイアログボックスで次の手順を実行します。**[クエリウィザードを使用してクエリを作成/編集する]** が選択されていることを確認します。

3. **データベース]**タブで以下のいずれかを実行します。

- Analytics プロジェクト への既存のデータ接続を使用するには、接続名を選択し、**[OK]**をクリックします。

メモ

既にデータ接続を作成している必要があります。詳細については、"データ接続の作成 (DSN)" ページ 1406を参照してください。

- Analytics プロジェクト へのデータ接続をその場で作成するには、**ACL ODBC** または **ACL ODBC 64** を選択し、**[OK]**をクリックします。

メモ

ACL ODBC と **ACL ODBC 64** の両方が表示される場合は、Excel のバージョンのビット数 (32ビットまたは64ビット) と一致する方を選択します。詳細については、"デフォルト Analytics データ接続" ページ 1407を参照してください。

4. **ACL ODBC** または **ACL ODBC 64** を選択した場合は、**プロジェクト ファイルを開く]**ダイアログボックスで、Analytics プロジェクト (.acl) に移動して、ファイルを選択し、**開く]**をクリックします。
5. **クエリウィザード**で、画面の手順に従い、次を実行します。

- Analytics プロジェクト からインポートするテーブルまたは列を選択します。
- 複数の Analytics テーブルを選択する場合は、テーブルを結合します。
- 省略可能。インポートされるデータをフィルタリングします。
- 省略可能。インポートされたデータの並べ順を指定します。

クエリウィザードの詳細については、Excel ヘルプを参照してください。

メモ

ODBC 接続処理中にテーブルを結合せずに複数の Analytics テーブルに接続する場合は、個別の接続処理を実行する必要があります。

6. **データのインポート]**ダイアログボックスで、必要なオプションを指定し、**[OK]**をクリックします。Excel は ODBC クエリを実行し、Excel ワークシートに Analytics データを入力します。
7. **ファイル > 保存]**をクリックし、Excel ワークブックを保存します。

参照情報

このセクションでは、Analytics 全体で適用される情報について説明します。

"Analytics における文字およびサイズの制限" 見開きページ	Analytics では、ある特定の入力範囲とユーザー定義のパラメーターに、アプリケーションで強制的に適用される制限があります。
"予約キーワード" ページ 1421	Analytics の内部プロセス用に予約され、フィールドまたは変数名で使用できないキーワード
"キーボード ショートカット" ページ 1423	Analytics で使用できるキーボード ショートカット

Analytics における文字およびサイズの制限

Analytics では、ある特定の入力範囲とユーザー定義のパラメーターに制限があります。以下のセクションは、超過するときの上限と結果を説明します。

備考

これらの詳細は、Analytics における文字とサイズの制限に適用されます。

- **非 Unicode と Unicode** -記載されている場合を除き、指定された制限は、非 Unicode 版 Analytics と Unicode 版 Analytics の両方に適用されます。
- **文字とバイト** -文字制限は非 Unicode 版で使用されるシングルバイト文字エンコーディング、と Unicode 版で使用されるダブルバイト文字エンコーディングに適用されます。つまり、256 文字の制限は、いずれの版では 256 文字であることを意味します。
- **ソースデータの場所** -Analytics における制限は、ソースデータが Analytics データファイル(.fil)にある場合でも、外部ファイルまたはデータベースにある場合でも同じです。外部データソースは Analytics における制限とは異なる固有の制限を適用する場合があります。

Analytics プロジェクトにおける制限

パラメーター	制限	制限を超過した場合の結果
Analytics プロジェクトの最大ファイルサイズ	1 GB(テスト成功済み)	
Analytics プロジェクト内のテーブルの最大数	5,150(テスト成功済み)	
Analytics プロジェクトのパスと名前の合計の最大長	259 文字 ファイルパス、プロジェクト名、およびファイル拡張子(.acl)を含む	259 文字に切り詰め
Analytics プロジェクトのパスとコマンドログ名の合計の最大長	259 文字 ファイルパス、コマンドログ名、およびファイル拡張子(.log)を含む	エラーメッセージ
Analytics プロジェクトフォルダー名の最大長	64 文字	64 文字に切り詰め
Analytics プロジェクトフォルダー名における有効な文字	英数文字とアンダースコア文字(_) 名前に特殊文字や空白を使用した	無効な文字をアンダースコア文字(_)に置き換え

パラメーター	制限	制限を超過した場合の結果
	り、名前を数字で始めることはできません。	
ノートの最大長(テーブル、レイアウト、ビュー、プロジェクト)	32,765 文字	32,765 文字に切り詰め
フィールド ノートの最大長	4,996 文字	4,996 文字に切り詰め

テーブルの制限

パラメーター	制限	制限を超過した場合の結果
テーブルのレコードの最大数	2,147,483,647(テスト成功済み)	
最大レコード長	32,767 文字(非 Unicode 版) 16,383 文字(Unicode 版)	Analytics は 32,767 文字または 16,383 文字にリセットされます
テーブルのフィールドの最大数	1,498(テスト成功済み)	
テーブルレイアウト名の最大長	64 文字	64 文字に切り詰め
テーブルレイアウト名で有効な文字	英数文字とアンダースコア文字(_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字(_)に置き換え
ソースデータファイルのパスと名前の最大長	255 文字	Analytics は動作を停止します

ビューの制限

パラメーター	制限	制限を超過した場合の結果
ビューにおける列の最大数	列数には特定の制限はありません 列は最大レコード長の 32,767 文字(非 Unicode)/16,383 文字(Unicode)を超えることはできません	エラー メッセージ

パラメーター	制限	制限を超過した場合の結果
	<p>メモ</p> <p>データの最初のインポートの後は、それよりも多い列がインポートされた場合でも、デフォルト ビューには最大 256 列が表示されます。必要に応じて、追加の列を手動でビューに追加できます。詳細については、「ビューに列を追加する」 ページ 849を参照してください。</p>	
ビュー名の最大長	64 文字	64 文字に切り詰め
ビュー名で有効な文字	英数文字とアンダースコア文字 (_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字 (_) に置き換え

フィールドと列の制限

パラメーター	制限	制限を超過した場合の結果
最大フィールド長	32,767 文字 (非 Unicode 版) 16,383 文字 (Unicode 版)	エラー メッセージ
フィールド名の最大長	256 文字 ODBC インポートの場合は 64 文字 Microsoft Excel および Access にエクスポートする場合は 64 文字	256 文字に切り詰め 64 文字に切り詰め エラー メッセージ
フィールド名で有効な文字	英数文字とアンダースコア文字 (_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字 (_) に置き換え
代替列見出しの最大文字数 (表示名)	256 文字	256 文字に切り詰め
演算フィールドの最大幅	255 文字	入力値が 255 文字を超える場合はエラーメッセージと値が 1 にリセットされる
ビューにおける列の最大幅	255 文字	入力値が 255 文字を超える場合はエラーメッセージと値が 1 にリセットされる

パラメーター	制限	制限を超過した場合の結果
		ビュー内で列の境界をドラッグして、255文字を一時的に超えることは可能だが、[列の変更]ダイアログボックスでリセットされる
数値フィールドの最大桁数 (小数点以下を含む)	22	切り詰めまたはエラーメッセージ

フィルター制限

パラメーター	制限	制限を超過した場合の結果
フィルター名の最大長	256文字	256文字に切り詰め
フィルター名で有効な文字	英数文字とアンダースコア文字(_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字(_)に置き換え

インデックスの制限

パラメーター	制限	制限を超過した場合の結果
インデックス名の最大長	64文字	64文字に切り詰め
インデックス名で有効な文字	英数文字とアンダースコア文字(_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字(_)に置き換え
インデックスキーフィールドの最大長	247文字	エラーメッセージ
インデックスキーフィールドの最大数	246文字の合計長	エラーメッセージ

クイックソート上限

パラメーター	制限	制限を超過した場合の結果
クイックソートフィールドの最大長	247文字	クイックソートメニューオプションが無

パラメーター	制限	制限を超過した場合の結果
		効です

コマンドの制限

パラメーター	制限	制限を超過した場合の結果
コマンドの文字の最大数	32,765 文字 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ 場合によっては、規定された制限を超えるコマンド構文が動作することがあります。ただし、Analytics の後続のバージョンが制限の超過を引き続きサポートするという保証はありません。</p> </div>	32,765 文字に切り捨てられ、関連するエラーメッセージが表示される

パスワードの制限

パラメーター	制限	制限を超過した場合の結果
指定または入力するパスワードの最大文字数	<ul style="list-style-type: none"> ○ パスワード要件があるほとんどのコマンド - 256 文字 ○ DEFINE TABLE DB、NOTIFY - 30 文字 	最大長に切り詰めて接続エラー

式の制限

パラメーター	制限	制限を超過した場合の結果
式内の最大文字数	8188 文字	エラーメッセージ
IF または WHILE 条件での最大文字数	4094 文字	エラーメッセージ
個々の条件や値での最大文字数	4094 文字	エラーメッセージを表示して、2 KB にリセット
条件演算フィールドに含まれる条件の	32,000 文字	エラーメッセージ

パラメーター	制限	制限を超過した場合の結果
文字の最大数		

変数の制限

パラメーター	制限	制限を超過した場合の結果
変数名の最大長	31 文字	エラー メッセージ
変数名で有効な文字	英数文字とアンダースコア文字 (_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字 (_) に置き換えるか、またはエラー メッセージを表示

スクリプトの制限

パラメーター	制限	制限を超過した場合の結果
スクリプト名の最大長	64 文字	64 文字に切り詰め
スクリプト名で有効な文字	英数文字とアンダースコア文字 (_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字 (_) に置き換え

ワークスペースの制限

パラメーター	制限	制限を超過した場合の結果
ワークスペース名の最大長	64 文字	64 文字に切り詰め
ワークスペース名で有効な文字	英数文字とアンダースコア文字 (_) 名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。	無効な文字をアンダースコア文字 (_) に置き換え

日付の制限

パラメーター	制限	制限を超過した場合の結果
サポートされる日付の範囲	西暦 1900 年 1 月 1 日 ~ 9999 年 12 月 31 日	エラー メッセージ

ダイアログビルダーの制限

パラメーター	制限	制限を超過した場合の結果
ドロップダウン リストのオプションの最大数	32 リスト項目	ラベル一覧で、33 番目の項目のテキスト入力は受け入れ拒否
各ドロップダウン項目のラベルの最大サイズ	256 文字	256 文字に切り詰め

オプション ダイアログ ボックスの制限

パラメーター	制限	制限を超過した場合の結果
最大バッファ サイズ	255 KB	エラー メッセージ

予約キーワード

Analytics では、特定の目的のために特定のキーワードが予約されています。これら予約キーワードの値をフィールド名や変数名に使用することはできません。

予約キーワードに接尾辞を付ければ、それをフィールド名や変数名として使用することができます。たとえば、"Field" という名前は許可されませんが、"Field_1" や "Field_2" は許可されます。

メモ

また一部の例では、"Can"(CANCEL)、"Form"(FORMAT)、"Rec"(RECORD)などの、予約キーワードの省略形を使用することもできません。

予約キーワード	Analytics の目的
ALL	既に定義されているすべてのフィールドを表します。
AND	論理演算子のANDを表します。
AS	表示名を出力フィールドまたは式に割り当てます。
CANCEL	現在のコマンドをキャンセルします。
D	直前に指定した式またはフィールド名のソート順が降順であることを表します。
END	一連の入力を終了し、空白行のように振る舞います。
EXPR	デフォルトの出力フィールド名の前に付けます。
F	論理式の <i>False</i> 値を表します。
FIELD/FIELDS	EXPORT コマンド、EXTRACT コマンド、JOIN コマンド、SAMPLE コマンドの一部として使用します。
FORMAT	Analytics テーブルレイアウトの古い名前。 Analytics テーブルの名前としては使用できません。
IF	条件を指定します。
LINE	DEFINE COLUMN コマンドで、フィールドを指定された行数に分断するかどうかを指定します。
NODUPS	Analytics レポートで、空のフィールドの重複表示値を表示しません。
NOT	論理演算子 NOT を表します。
NOZEROS	数値フィールドまたはレポートで、ゼロ値を空白として印刷または表示します。
ON	フィールドリスト名の前に置きます。

予約キーワード	Analytics の目的
OR	論理演算子 OR を表します。
OTHER	SUMMARIZE コマンドの出力に含めるが、小計しないフィールドまたは式を示します。
PAGE	REPORT コマンドを使用して、改ページを作成します。
PICTURE	数値フィールドの形式を表します。
PRIMARY	特定のタイプの結合を指定します。
RECORD	入力レコード全体を現状のままで表します。
RECORD_LENGTH	レコード長を表します。この値はレコード処理が必要な操作で使用します。
SECONDARY	特定のタイプの結合を指定します。
SUPPRESS	数値の合計を出力しないようにします。
T	論理式の <i>True</i> 値を表します。
TAPE	Analytics でデータにアクセスするための古い方法を参照します。 Analytics テーブルの名前としては使用できません。
TO	コマンドで使用する出力先ファイルを指定します。
WIDTH	特定のフィールドまたは式に対してデフォルトで設定されている印刷幅を変更します。

キーボード ショートカット

次の表は、Analytics で使用できるキーボード ショートカットを列挙しています。

ショートカット	Analytics における使用場面	アクション
Alt+F4	グローバル	Analytics を終了する
Ctrl+Tab	グローバル	次のタブを表示する
Ctrl+Shift+Tab	グローバル	前のタブを表示する
Ctrl+PgDn	ナビゲーター	総覧]タブ、 ログ]タブ、および 変数]タブ間で表示を切り替える
	ビュー	次のビューを表示する(テーブルに複数のビューがある場合)
	コマンド結果	結果をグラフとして表示する(結果をグラフ化できるコマンドの場合)
Ctrl+PgUp	ナビゲーター	総覧]タブ、 ログ]タブ、および 変数]タブ間で表示を切り替える
	ビュー	前のビューを表示する(テーブルに複数のビューがある場合)
	コマンド結果	結果をテキストとして表示する
Ctrl+1	グローバル	スクリプトを実行する
Ctrl+2	グローバル	テーブル履歴からスクリプトを作成する
Ctrl+3	グローバル	カウント]ダイアログ ボックスを開く
Ctrl+4	グローバル	合計]ダイアログ ボックスを開く
Ctrl+5	グローバル	階層化]ダイアログ ボックスを開く
Ctrl+6	グローバル	順番検査]ダイアログ ボックスを開く
Ctrl+7	グローバル	結合]ダイアログ ボックスを開く
Ctrl+8	グローバル	ヒストグラム]ダイアログ ボックスを開く
Ctrl+9	グローバル	サンプル]ダイアログ ボックスを開く
Ctrl+0	グローバル	要約]ダイアログ ボックスを開く
Ctrl+B	グローバル	最後に使った Analytics プロジェクトを開く
Ctrl+I	グローバル	テーブルレイアウト]ダイアログ ボックスを開く

ショートカット	Analytics における使用場面	アクション
Ctrl+O	グローバル	Analytics プロジェクトを開く
Ctrl+R	グローバル	[レポート] ダイアログ ボックスを開く
Ctrl+S	グローバル	現在の Analytics プロジェクトを上書き保存する
Ctrl+Y	スクリプト エディター	操作をやり直す
Ctrl+Z	スクリプト エディター	操作を元に戻す
F1	グローバル	状況依存のオンライン ヘルプを表示する
F2	コマンド ライン [フィルター] テキスト ボックス スクリプト エディター 試 テキスト ボックス	フィールドの挿入 ダイアログ ボックスを表示する
F4	コマンド ライン [フィルター] テキスト ボックス スクリプト エディター 試 テキスト ボックス	プロジェクト項目の挿入 ダイアログ ボックスを開く
F5	スクリプト エディター	スクリプトを実行する
F8	コマンド ライン [フィルター] テキスト ボックス スクリプト エディター 試 テキスト ボックス	日付と時刻セレクター ダイアログ ボックスを開く
F9	スクリプト エディター	ブレークポイントを挿入/削除する
F10	グローバル スクリプト エディター	ステータス バーの表示/非表示を切り替える 1度に1つのスクリプト行をステップ実行する
Enter	総覧 タブ	選択した項目を開く
	コマンド ダイアログ ボックス コマンド ライン	コマンドを実行する

このページは意図的に空白のままにしています

Analytics のスクリプト

Analytics のスクリプト

スクリプト作成は、Analytics での作業を自動化する方法です。スクリプトを作成せずに、アドホックまたは手動によって、Analytics を使用して、さまざまな価値を得ることができます。ただし、Analytics の価値、能力、および効率を最大化するには、スクリプトを作成する必要があります。

ですが、幸いにも、Analytics には、初心者でもスクリプトを比較的容易に作成できるツールが用意されています。

Analytics スクリプトの機能

Analytics スクリプトとは、特定のタスク、またはいくつかの関連するタスクを実行する、一連の ACLScript コマンドです。スクリプトは、Analytics の能力を活用する鍵です。

1 度に複数のコマンドを実行する	スクリプトは、複数の Analytics コマンドを組み立てて、それらを連続したシーケンスで実行できるようにします。
繰り返しの自動化	スクリプトは繰り返し実行することができます。同じ分析テストや、同じ分析関連のタスク、たとえば、特定のデータファイルのインポートや準備などを定期的に行う場合には、労力が大幅に軽減されます。
一貫性の確保	スクリプトは、実行されるたびに、同じコマンドを同じ順序で実行することによって、一貫性を確保します。
分析の共有	スクリプトは移動可能かつ共有可能です。他のユーザーに送信したり、ネットワークの場所で開催可能にしたり、Analytics プロジェクト間でコピーしたりすることができます。
専門知識の拡充	スクリプトは、熟練の Analytics ユーザーが分析テストを開発し、スクリプトを記述することがないユーザーとそのテストを共有できるようにします。
ユーザー操作の許可	スクリプトで、ユーザーに入力を求めることができます。つまり、ユーザーが独自に名前付きのテーブルやフィールドに対して、独自の入力条件を使用してスクリプトを実行できるように設計することができます。
自動実行のスケジュール	スクリプトをスケジュールしたり、無人で実行したりすることができます。これにより各社では、時間のかかるデータ処理を夜間に行ったり、自動化された分析または継続的な分析のプログラムをセットアップしたりすることが可能になります。

コアスクリプト リソース

Analytics スクリプトを作成するユーザー向けのコアリソースは ACLScript 言語リファレンスです。このリファレンスでは、すべての Analytics コマンド、関数、アナリティクス タグについて詳細に説明し、さまざまな例やコード スニペットを提供しています。

- "コマンドの概要" ページ 1611 -Analytics コマンド
- "関数の概要" ページ 2139 -Analytics 関数
- "アナリティクス スクリプトの概要" ページ 2579 -ロボットで実行できるアナリティクス スクリプト 開発の指針

スクリプト チュートリアル

さまざまなチュートリアルが用意され、Analytics でのスクリプト作成の基本について説明します。"スクリプトの基本" ページ 1431を参照してください。

スクリプトの操作

Analytics ユーザー インターフェイスでのスクリプト作成の基本事項と、各種スクリプト作成ツールについては、"スクリプトの操作" ページ 1533を参照してください。

このページは意図的に空白のままにしています

スクリプトの基本

Analytics でスクリプトを開始する方法がわからない場合、このセクションでは、レベルに応じて役立つ情報を提供します。新しいユーザーは初心者向けチュートリアルを使用できます。上級ユーザーは ACLScript の仕組みの概要について基本セクションを参照できます。

適切な対象者向けの適切なコンテンツ

スクリプトの背景と Analytics の信頼度によっては、このセクションの一部のトピックが他よりもニーズに合っています。

初心者

スクリプトと Analytics についてゆっくりと学習したいユーザーの場合は、インタラクティブな初級者レベルのチュートリアルについて次のセクションを参照してください。

- "初心者向けのスクリプト" 見開きページ
- "関数を使用する方法" ページ 1477
- "Analytics でのスクリプト作成の基本事項" ページ 1450

中級から上級ユーザー

Analytics またはスクリプト全般に関する知識がある場合は、クイックリファレンスとより詳細なチュートリアルとして次のセクションが提供されています。

- "Analytics でのスクリプト作成の基本事項" ページ 1450
- "高度な関数の使用" ページ 1499
- "最もよく使用される 30 個の Analytics 関数" ページ 1517
- "Analytics 関数を使って検索とフィルタリングを行う" ページ 1235

初心者向けのスクリプト

ACLScript では多数の処理を実行できます。また、さまざまなコマンドと関数を使用できるため、少しずつ始め、Analytics の実践的な知識を学ぶことをお勧めします。

Analytics のインストール

スクリプトを開始する前に、Analytics をインストールし、コンピューターで実行する必要があります。詳細については、"ACL for Windows インストールおよびアクティベーションガイド" ページ 2661を参照してください。

サンプルプロジェクトを開く

このセクションでは、Analytics をインストールしてアクティブ化していること、また、同アプリケーションに同梱されているサンプルデータにアクセスできることを前提にしています。

サンプルプロジェクトを開くには、コンピューターでサンプルプロジェクトを含むフォルダーに移動し、**Sample Project.ACL** をダブルクリックします。

メモ

デフォルトでは、サンプルデータプロジェクトはローカルファイルシステムの **C:\Users\username\Documents\ACL Data\Sample Data Files** にインストールされます。

スクリプトとは何か

スクリプトは、Analytics で実行するコマンドのリストです。スクリプトは一連のタスクを自動的に実行する際に有用です。ユーザー インターフェイスから各コマンドを手動で実行する必要がないためです。スクリプトの詳細についての参照先: "スクリプトとは何か" 次のページ

あなたにとって最初の Analytics スクリプト

この短いチュートリアルを完了し、ACLScript の基本を学習します。このチュートリアルでは、テーブルを開くことと、レコードのサブセットを抽出を行います。このチュートリアルを表示するには、"あなたにとって最初の Analytics スクリプト" ページ 1437を参照してください。

スクリプトとは何か

スクリプトは一連の Analytics コマンドであり、連続して実行され、Analytics 内で作業を自動化するために使用されます。すべての Analytics コマンドはスクリプトに含めることができます。

スクリプトを使用する理由

スクリプトの使用にはさまざまな利点があります。

処理の自動化

一連の繰り返しタスクまたはルーチンを定期的に行う必要がありますか。現在これらのタスクを手動で行っていますか。その場合、スクリプトを使用して、これらのタイプのプロセスを自動化できる可能性があります。スクリプトを使用して、複雑なルーチンに関連付けられた手動作業を回避できます。ルーチンが複雑になるほど、スクリプトの実行で削減できる時間が多くなります。

処理のスケジュール

一般的に、スクリプトのスケジュールは、大きいデータセットを処理するときに必須です。ロボットを使用する場合は、業務時間外でも、スケジュールでスクリプトを実行できます。単一のスクリプトまたは一連のスクリプトをスケジュールし、特定の日に実行することもできます。

精度の改善

手動で実行されると、複雑なデータ分析ルーチンは人的エラーを起こしやすくなります。スクリプトを使用すると、処理の一貫性と正確性を保証できます。また、同じスクリプトが実行されるたびに、同じ命令が同じ順序で実行されることを絶対に確実にすることもできます。

複雑さの低減

スクリプトは複雑なファイル構造を処理でき、データフィールドで複雑な演算を作成できます。時々、より複雑な分析はスクリプトでのみ実行できます。たとえば、通常、継続監視プログラムでは、スクリプトで処理を自動化する必要があります。

分析の共有

スクリプトは移動可能かつ共有可能です。他のユーザーに送信したり、ネットワークの場所で利用可能にしたり、Analytics プロジェクト間でコピーしたりすることができます。

ユーザー操作の許可

スクリプトで、ユーザーに入力を求めることができます。つまり、ユーザーが独自に名前付きのテーブルやフィールドに対して、独自の入力条件を使用してスクリプトを実行できるように設計することができます。

ドキュメントの取り込み

スクリプトは監査レビューのための優れたドキュメントのソースであり、監査証跡の一部として使用できます。スクリプトを作成すると、アナリティクステストの結果を作成する処理を文書化しています。これは、将来に簡単に参照できます。スクリプトにコメントを追加して、ドキュメントをさらに補足することもできます。

スクリプトで自動化できる一般的なプロセス

スクリプトには、単一フィールドに対して1つのコマンドを実行するだけのシンプルなものもあれば、分析目標を達成するための作業の大部分を実行するほど内容の充実したものもあります。

スクリプトは、次の1つ以上のプロセスを実行するために最も一般的に使用されます。

データのインポート

スクリプトを使用すると、固定幅、区切り、レポート/PDF、Excel、ODBC 経由でアクセスされたファイルなど、さまざまなソースファイルを Analytics にインポートできます。

```
COMMENT *** Microsoft Access データベース ファイルを Analytics テーブル employees_list にデータをインポートする。  
IMPORT ACCESS TO employees_list PASSWORD 1 "C:\ACL DATA\Sample Data Files\employees_list.fil" FROM "Employees_List.mdb" TABLE "[Employees_List]"  
CHARMAX 60 MEMOMAX 70
```

データの準備

スクリプトを使用して分析のためにデータを準備できます。スクリプトを使用して、テーブルの結合または関連付けの前にフィールドを標準化し、先頭または末尾のスペースを値から削除し、不要な文字を削除し、フィールドのデータ型を変換できます。

```
COMMENT *** PO_No 値を含む新しい演算フィールドを作成します。すべての先頭の空白スペースは削除され、値が適切に左寄せされます。  
DEFINE FIELD c_PO_No COMPUTED ALLTRIM(PO_No)
```

データの分析

スクリプトはデータ分析コマンドと関数を使用して、分析目的を達成します。スクリプトを使用して、レコードをグループ化し、比較して、問題、傾向、または異常値を特定できます。

```
COMMENT *** Sales2016Actual テーブルを開き、Customer Number で分類し、Sales Order  
Amount で小計し、結果を Sales2016ByCustomer に送信する  
OPEN Sales2016Actual  
CLASSIFY ON Customer_Number SUBTOTAL Sales_Order_Amount TO  
Sales2016ByCustomer
```

スクリプトの例

シナリオ

毎月、クライアントは業者、請求書、および発注書情報を提供します。発注書フィールドに空欄がないことを確認し、データの整合性を検証する必要があります。

タスクが繰り返されるため、スクリプトを作成する良い機会であると考えられます。すべてのフィールドを分析で使用し、発注書フィールドの空欄を検索できます。

プロセス

次のアクションを実行するスクリプトを作成します。

1. Invoice_Amts テーブルを開きます。
2. 発注書フィールド (PO_No) の空欄を検索します。
3. 空の発注書番号のレコードを新しいテーブル(r_Blank_Purchase_Orders)に抽出し、例外をフォローアップできます。

ヒント

テーブルを容易に特定するには、次の名前付け規則を使用します。

- **準備されたテーブル**- テーブル名にプレフィックス p_ を付ける
- **一時テーブル**- テーブル名にプレフィックス t_ を付ける
- **結果テーブル**- テーブル名にプレフィックス r_ を付ける

結果

```
COMMENT *** "Invoice_Amts" テーブルを開く。  
OPEN Invoice_Amts
```

```
COMMENT *** 発注書フィールドの空欄を検索する。  
SET FILTER TO ISBLANK(PO_No)  
  
COMMENT *** 新しいテーブル "r_Blank_Purchase_Orders" に結果を抽出します。  
EXTRACT FIELDS Vendor_Name Invoice_No Payment_Date Invoice_Date Invoice_Amt  
Vendor_Name Invoice_No PO_No TO r_Blank_Purchase_Orders
```

次の手順

短いチュートリアル of ""あなたにとって最初の Analytics スクリプト" 次のページ" を演習し、独自のスクリプトを作成してください。

あなたにとって最初の Analytics スクリプト

ACLScript の基礎を短くシンプルなチュートリアルとして示します。このチュートリアルでは、テーブルを開くことと、レコードのサブセットを抽出することだけを行います。ACLScript ではもっと多くのことを行うことができます。

何を行いたいですか？

このチュートリアルでは、Analytics をインストールしてアクティブ化していること、また、同アプリケーションに同梱されているサンプルデータにアクセスできることを前提にしています。

メモ

デフォルトでは、サンプルデータプロジェクトはローカルファイルシステムの `C:\Users\username\Documents\ACL Data\Sample Data Files` にインストールされます。

セットアップ

サンプルの Analytics プロジェクトを開く

1. ACL for Windows を開きます。
2. **アナリティクスプロジェクトを開く**]をクリックし、`ACL Data\Sample Data Files` フォルダーから `Sample Project.ACL` を選択します。

スクリプトを作成してみましょう

1. ナビゲーターで、**総覧**]タブ内の **[スクリプト]** フォルダーを右クリックし、**新規作成 > スクリプト**]を選択します。
新しいスクリプト がナビゲーターに追加され、スクリプト エディターで表示されます。
2. 新しいスクリプト を右クリックし、**名前の変更**]を選択して、「`extract_invoices`」と入力します。

スクリプトのロジック

このスクリプトでは ACLScript を使って次のことを行おうとしています。

1. **Ap_Trans** テーブルを開く。
2. 1000.00 を超える請求金額が含まれるすべてのレコードをテーブルからコピーし、**Ap_Trans_High** という新しいテーブルに格納する。
3. この新しいテーブルを開き、結果を確認する。

Ap_Trans テーブルを開く

データはテーブルに格納されているため、データを操作するには、まずテーブルを開く必要があります。**OPEN** コマンドは、ユーザーが指定したテーブルを操作しようとしていることを通知し、そのテーブルのデータをスクリプトのコマンドからアクセスできるようにします。

```
OPEN Ap_Trans
```

この行をコピーし、スクリプト エディターに貼り付けて、Editor のツールバーの **実行**  をクリックします。

Ap_Trans テーブルが開いたら、スクリプトが機能していることとなります。テーブルを閉じて先に進みます。

すべてのレコードを Ap_Trans_High に抽出し、Ap_Trans を閉じる

スクリプトで **Ap_Trans** テーブルを操作できるようになったため、**EXTRACT** コマンドを使って、**Ap_Trans** 内のレコードを **Ap_Trans_High** という新しいテーブルにコピーします。

```
EXTRACT RECORD TO 'Ap_Trans_High'  
CLOSE Ap_Trans
```

この行をコピーし、スクリプト エディターの **OPEN** コマンド後の行に貼り付けて、Editor のツールバーの **実行**  をクリックします。

Ap_Trans_High テーブルが、ナビゲーターで **テーブル > Accounts_Payable**] に表示されます。この新しいテーブルには **Ap_Trans** からコピーされたレコードが格納されています。

レコードのサブセットを Ap_Trans_High に抽出し、Ap_Trans を閉じる

ここで、1000.00 を超える金額を含む請求書レコードのみがコピーされるようにする条件分岐の **IF** パラメーターを **EXTRACT** コマンドに追加できます。

Invoice_Amount フィールドの値が 1000.00 より大きいかどうかをテストするために、**IF** パラメーターをどのように使用するかに注目してください。このテストが True として評価されなかった場合には、レコードは抽出されません。

スクリプトの基本

```
EXTRACT RECORD TO 'Ap_Trans_High' IF Invoice_Amount > 1000.00  
CLOSE Ap_Trans
```

この行をコピーし、スクリプト エディター内の既存の `EXTRACT` コマンドをその行に置き換えて、Editor のツールバーの **実行**  をクリックします。

`Ap_Trans_High` テーブルを上書きするよう求められたら、`[はい]` をクリックします。これにより、`Ap_Trans_High` テーブルには、1000.00 を超える金額を含むレコードが `Ap_Trans` からコピーされて格納されます。

Ap_Trans_High テーブルを開く

スクリプトの最後に新しいテーブル、`Ap_Trans_High` を開くことで、`EXTRACT` コマンドの結果を確認できるようにします。これがこのスクリプトの最後のアクションであるため、このテーブルが開いたら、レコードを確認することができます。

```
OPEN Ap_Trans_High
```

この行をコピーし、スクリプト エディターの `EXTRACT` コマンド後の行に貼り付けて、Editor のツールバーの **実行**  をクリックします。

これにより、スクリプトの完了時に `Ap_Trans_High` が開き、`Ap_Trans` から抽出されたレコードを確認できます。

スクリプト全体

```
OPEN AP_Trans  
  
EXTRACT RECORD TO 'Ap_Trans_High' IF Invoice_Amount > 1000.00  
CLOSE Ap_Trans  
  
OPEN Ap_Trans_High
```

次の学習ステップ

- Analytics でのスクリプト作成の基本事項の一覧についての参照先: "Analytics でのスクリプト作成の基本事項" ページ 1450
- 上級トレーニングについては、[アカデミー](#) のスクリプト作成コースを参照してください。

テキスト データの比較

テキストを操作するときには、一般的に、1つの値を別の値と比較します。比較は大文字と小文字を区別するため、大文字と小文字が異なる形式で保存されたテキストを比較することが困難です。Analyticsは、比較しているテキストを正規化された大文字または小文字の形式に変換することで、比較の信頼性を高める機能を提供しています。

テキスト比較演算子

Analytics のスクリプトでは、2つの演算子を使用してテキスト値を比較できます。

- **等号演算子 (=)** - は、等号演算子の左側の値が右側の値と正確に同じ場合に真と評価します。
- **不等号演算子 (<>)** - は、不等号演算子の左側の値が右側の値と正確に同じでない場合に真と評価します。

いずれの演算子も大文字と小文字を区別し、2つの値を比較します。 `valueOne <> valueTwo`

テキスト比較を使用したテーブルのフィルタリング

次のテーブルを操作し、財務部のレコードのみが表示されるようにフィルタリングする必要があります。

部門	Max_Hourly	Min_Hourly	位置
管理職	205.13	166.67	CEO 兼社長
管理職	141.03	89.74	財務部長
ファイナンス	24.62	20.51	会計士
ファイナンス	23.08	17.95	担当者、現金支払い
ファイナンス	18.46	14.67	担当者、買掛金
ファイナンス	18.46	14.67	担当者、購買
情報システム	23.08	14.36	テクニカルサポート
情報システム	30.77	23.08	Web 管理者

テーブルをフィルタリングするには、等号演算子 (=) を使用して簡単な式を作成します。

```
COMMENT 部署が "Finance" のレコードを表示するためにテーブルをフィルタリング
SET FILTER TO Department = "Finance"
```

最初のフィルター結果

このフィルターに基づき、フィルタリングされたテーブルには4つのレコードが表示されることが想定されますが、代わりに2つのみが表示されます。

部門	Max_Hourly	Min_Hourly	位置
ファイナンス	24.62	20.51	会計士
ファイナンス	18.46	14.67	担当者、購買

等号演算子は大文字と小文字を区別するため、Departmentフィールドが"finance"を含むレコードが結果から除外されます。これらのレコードも結果に含める必要があります。

LOWER() 関数を使用してテーブルをフィルタリングする

比較を実行するには、Analyticsは、比較しているテキストを既知の大文字または小文字の形式に変換することで、比較の信頼性を高める機能を提供しています。

テーブルをフィルタリングし、大文字と小文字の形式に関係なく、財務部のすべての従業員を結果に含めるようにするには、同じ式を使用しますが、`LOWER()` 関数を使用してすべての値を小文字に変換します。

```
COMMENT 部署が "finance" のレコードを表示するためにテーブルをフィルタリング
SET FILTER TO LOWER(Department) = "finance"
```

式が評価されるときに、`LOWER("Finance")` は `"finance"` になり、等号演算子の右側の文字列と比較されます。

2番目のフィルター結果

`LOWER()` 関数を式で使用するときには、フィルターは財務部のすべての従業員を含みます。

部門	Max_Hourly	Min_Hourly	位置
ファイナンス	24.62	20.51	会計士
ファイナンス	23.08	17.95	担当者、現金支払い

部門	Max_Hourly	Min_Hourly	位置
ファイナンス	18.46	14.67	担当者、買掛金
ファイナンス	18.46	14.67	担当者、購買

空のデータ値のフィルタリング

時々、データは不完全であるか任意であるため、Analytics テーブルのフィールドが~ または空白の値になる場合があります。Analytics の式が空の日付値を検出すると、空の日付は最小システム値の 1900-01-01 として処理されます。日付を比較する式を作成するときには、常に、この動作を考慮する必要があります。

受注テーブル

次の受注テーブルを操作し、2011 年より前に送信されたすべての注文を分析できるようにフィルタリングします。レコードに注文日がない場合、キャンセル済み注文を表し、フィルタリングされたレコードに含めません。

カテゴリ	Order_Date	Order_ID	Quantity
事務用品		3	6
事務用品		293	49
事務用品	07/23/2012	293	27
技術	10/15/2010	483	30
事務用品	08/28/2010	515	19
家具	08/28/2010	515	21
事務用品	06/17/2011	613	12
事務用品	06/17/2011	613	22
事務用品	03/24/2012	643	21
事務用品	02/26/2009	678	44

簡易フィルターの使用

テーブルを最初にフィルタリングしようとするときには、次の簡易式を使用して、2011 年以降のすべての注文を除外します。

ヒント

リテラル日付値を囲むバッククォート (`) に注意してください。リテラル日付時刻値は常にこの修飾子で囲む必要があります。詳細については、「データ型」ページ 1457を参照してください。

```
COMMENT 2011 年 1 月 1 日以降の注文日付を含むレコードを除外
SET FILTER TO Order_Date < `20110101`
```

最初のフィルター結果

Analytics は空の日付値を 1900-01-01 として処理し、1900 年 1 月 1 日は 2011 年 1 月 1 日よりも前であるため、結果は除外したい空の `Order_Date` 値があるレコードを含みません。

カテゴリ	Order_Date	Order_ID	Quantity
事務用品		3	6
事務用品		293	49
技術	10/15/2010	483	30
事務用品	08/28/2010	515	19
家具	08/28/2010	515	21
事務用品	02/26/2009	678	44

フィルタリング中に空欄を確認する

関数を使用するときには、2011 年以降のレコードを除外する前に空の日付値を除外できます。

`ISBLANK()` 関数はテキスト値が空の場合に真を返すため、`Order_Date` フィールドの操作によっては、空の値を除外できます。

```
COMMENT 空の日付と 2011 年以降の注文日を除外する
SET FILTER TO NOT ISBLANK(DATETIME(Order_Date)) AND Order_Date < `20110101`
```

この式が評価されるときには、関数が内部から実行され、次の処理が発生します。

1. `DATETIME()` 関数が `Order_Date` 日付値をテキスト値 ('20100828' から "20100828") に変換します。
2. `ISBLANK()` 関数は、テキスト値が空白であるかどうかを確認し、真または偽を評価します。

3. NOT 演算子は ISBLANK() から返される論理値を反転します。

- 注文日が空白(真)の場合、値は偽に反転され、このレコードがフィルターにより除外されます
- 注文日が空(偽)でない場合には、値は真に反転され、フィルターにより、注文日が2011よりも前かどうかチェックされて、注文日の値が2011年1月1日より前のすべてのレコードが含まれます

ヒント

下位式がAND演算子の両側で真と評価されるレコードのみが含まれます。下位式のいずれかが偽と評価される場合、レコードは除外されます。

2番目のフィルター結果

2番目のフィルターは、2011年より前に注文が行われたかどうかをテストする前に空の日付値を除外します。このため、結果には、最初のフィルターに含まれていたキャンセルされた注文は含まれません。

カテゴリ	Order_Date	Order_ID	Quantity
技術	10/15/2010	483	30
事務用品	08/28/2010	515	19
家具	08/28/2010	515	21
事務用品	02/26/2009	678	44

スクリプトでの決定

実行することにはすべて決定があり、スクリプトも同じです。場合によっては、一部の他の条件が真である場合にのみコマンドを実行したいことがあります。また、含まれるデータによってテーブルの一部のレコードを処理する場合があります。ACLScript はスクリプトでの複数の決定方法を提供し、すべては条件式を使用します。

条件式とは何か

条件式は真または偽に評価される式です。条件式はスクリプトで実行される処理を支持し、スクリプト作成者によって指定されます。

これはかなり技術的な定義ですが、簡単な実際の例を使用すると、その意味がわかりやすくなります。

例

通りを歩き、知っている人にあつたとします。この人に挨拶することは礼儀正しいのですが、会ったときに「おはようございます」というのか、「こんにちは」というのでしょうか。

答えは簡単な条件によって決まります。12:00 PM を過ぎているかどうかです。答えが**はい**の場合、「こんにちは」、そうでない場合は「おはようございます」と言います。

この例では、条件式は、真(はい)と評価されるかどうかによって、実行するアクション(使用する挨拶)を決定します。

前の例の式は ACLScript で次のように翻訳できます。

```
COMMENT 現在時刻が 12:00 PM を過ぎているかどうかを確認する  
NOW() >= `t120000`
```

次の行をコピーして Analytics のコマンドラインに貼り付けると、この例を実行できます。これを実行する時刻によって、式が真または偽に評価されます。

```
DISPLAY NOW() >= `t120000`
```

ヒント

コマンドラインが表示されていない場合は、**[ウィンドウ> コマンドライン]**の順にクリックします。

例を実行すると、式でリテラル時刻値の 12:00 PM を変更し、反対に評価されるようにすることができます。

コマンドが実行されるかどうかを決定する

Analytics は `IF` コマンドを提供し、コマンドを実行するかどうかを決定できるようにします。コマンドには2つの入力が必要です。

- 条件式
- 式が真の場合に実行するコマンド

条件式が偽と評価された場合、コマンドは実行されません。

"こんにちは" という

上記の例を続け、次のコードをコマンドラインに貼り付けます。

```
IF NOW() >= `t120000` DISPLAY "こんにちは"
```

時刻が 12:00 PM を過ぎている場合は、`DISPLAY` コマンドが出力タブに "こんにちは" を出力します。一方、ローカル時刻がまだ午前中の場合は、何も出力されません。スクリプトは `DISPLAY` コマンドを実行しません。

"おはようございます" という

式が偽と評価された場合に、コマンドに "おはようございます" と出力させる方法は何でしょうか。一部のスクリプト言語は "else" 構文を提供し、真と偽の両方を処理しますが、ACLScript では提供されていません。代わりに、2番目の `IF` コマンドを反対の式で使用します。

この式をコマンドラインに貼り付けます。

```
IF NOW() < `t120000` DISPLAY "おはようございます"
```

この例は前のように動作します。ただし、時刻が 12:00 PM より前の場合、`DISPLAY` コマンドが "おはようございます" を出力します。

スクリプトの内容

これまでは、例がコマンドラインからのみ使用できる `DISPLAY` コマンドに制限されています。しかし、スクリプトでは、同じ原理が適用されます。挨拶を表示タブに出力する代わりに、この例ではスクリプトが挨拶を変数 `v_greeting` に格納します。

```
COMMENT 時間によって正しい挨拶を保存する
IF NOW() >= `t120000` ASSIGN v_greeting = "こんにちは"
IF NOW() < `t120000` ASSIGN v_greeting = "おはようございます"
```

このスクリプトが 12:00 PM より前に実行される場合、変数に格納された値は "おはようございます" です。12:00 PM 以降に実行される場合は、変数に格納された値は "こんにちは" です。これをスクリプト エディターに貼り付け、実行します。実行した後に変数タブで変数の値を確認できます。

処理するレコードを決定する

上記のようにスクリプトがコマンドを実行するかどうかを決定するときがありますが、テーブルの特定のレコードにのみコマンドを実行するときもあります。これは別の決定シナリオですが、IF コマンドを使用する場合とは異なります。

選択したレコードを処理する状況では、ACLScript はさまざまなコマンドで IF パラメーターを提供します。このアプローチを使用するときには、コマンドは入力として条件式を指定する必要があります。式はテーブルの各レコードに対してテストされ、真と評価されるときには、レコードが処理されます。

```
COMMENT Quantity が 5 より大きいレコードの Amount フィールドを合計する
TOTAL Amount IF Quantity > 5
```

午後に発生する取引を計算する

同じ条件式 `NOW() >= `t120000`` を使用して、午後に発生したテーブルのすべての取引の合計を計算できます。取引データの次のテーブルを考慮します。

Transaction_Amount	Unit_Cost	Product_No	Transaction_Date	Quantity
618.3	6.87	070104397	2000-11-17 12:00	90
6705.12	6.87	070104677	2000-11-17 9:30	976
7955.46	6.87	070104657	2000-11-17 14:45	1158
4870.83	6.87	070104327	2000-11-17 15:00	709
10531.71	6.87	070104377	2000-11-17 9:57	1533
5734	47	030414313	2000-10-30 1:00	122
2196	18	030414283	2000-10-30 18:25	122

Transaction_Amount フィールドの合計を計算するには、TOTAL コマンドを使用します。

```
COMMENT Transaction_Amount フィールドを合計
TOTAL Transaction_Amount
```

このコマンドは、テーブルのすべてのレコードを処理し、すべての取引の合計である合計 38,611.42 を計算します。

スクリプトの基本

決定をコマンドに追加し、12:00 PM 以降に発生した取引のみを処理するには、IF パラメーターを TOTAL に追加します。最初に例の同じ条件式を使用しますが、NOW() を取引日の時刻部分で置換します。

```
COMMENT 午後に発生したすべての取引の Transaction_Amount フィールドを合計  
COMMENT 関数を使用して Transaction_Date フィールドの時刻部分を抽出  
TOTAL Transaction_Amount IF CTOT(TIME(Transaction_Date)) >= `t120000`
```

コマンドでは、関数を使用して、取引日の時刻部分を分離する必要がありますが、一度実行すると、最初の例と同じ条件式(12:00 PM 以降か?)が決定になります。答えがはいの場合、金額が合計に含まれます。

このコマンドは、テーブルのすべての午後の取引の合計である合計 15,640.59 を計算します。

Analytics でのスクリプト作成の基本事項

ACLScript は Analytics コマンドをプログラミングして自動的に実行できるようにするコマンド言語です。ACLScript の構造および要素はシンプルですが強力です。

メモ

まったく初めてスクリプトを作成する場合は、いきなりこのコンテンツを学習する前に、アカデミーを訪れて基本的なトレーニングを受けてください。スクリプトの作成と Analytics の使用に関するコースについては、www.highbond.com を参照してください。

コマンド

スクリプト内のどの行も ACLScript コマンドを実行するため、コマンド名で始まります。コマンドは、Analytics で操作を実行するための命令です。

コマンド名の後に `parameter_name parameter_value` の形を持つパラメーターが 1 つ以上続きます。

ヒント

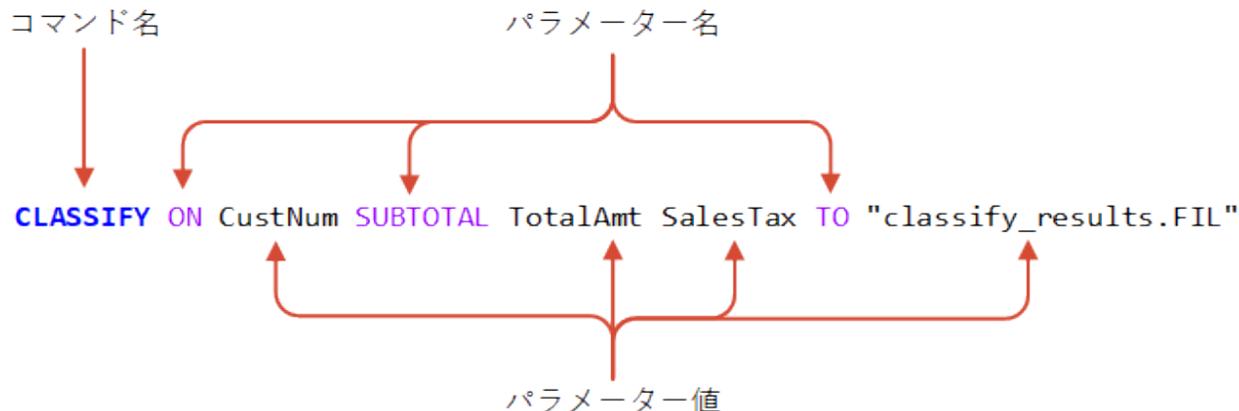
コマンドによって、必須になるパラメーターとオプションになるパラメーターがあります。オプションのパラメーターの指定は必須ではありません。省略すると、コマンドはそのパラメーターなしで実行するか、またはデフォルト値を使用します。

CLASSIFY コマンドを使用した例

次の例は、CLASSIFY コマンドとそれに続くパラメーターを示しています。

- ON - 分類の基準にする、対象テーブルのフィールドを指定します。
- SUBTOTAL - 出力における小計の対象にするオプションのフィールドを指定します。
- TO - CLASSIFY コマンドの結果を書き込む出力テーブルを指定します。

各パラメーターの後に 1 つまたは複数のパラメーター値が指定されていることに注目してください。



コマンド構文に関する重要なメモ

- スクリプトの行の最初の単語はコマンド名である必要がある
- ほとんどのコマンドにおいて、コマンド名の後に指定するパラメーターの順序は重要ではない
- 大半のコマンドにおいて、コマンドを実行する前にターゲット テーブルを開くこと、つまりコマンドの前に `OPEN テーブル名` を実行する必要があります。

コメント

ACLScript では、スクリプト 言語のように `COMMENT` キーワードを使ってコメントを追加できます。コメントを使うことで、作成するコードを理解しやすくとともに、スクリプトを参照、使用、理解、または更新しようとしているユーザーと対話することができます。ACLScript では、2 種類のコメントがサポートされています。

- **単一行のコメント** - `COMMENT` の後から行末までのすべてのテキストが無視されます。
- **複数行のコメントから成るブロック** - `COMMENT` から `END` キーワードまでのすべての行または空白行が無視されます。

詳細と例については、"コメント" ページ 1455を参照してください。

データ型

ACLScript では、4 つの基本的なデータ型がサポートされています。

- **論理型** - 最もシンプルなデータ型。論理型のデータは、真 (T) または偽 (F) という真理値を表します
- **数値型** - 0 ~ 9 の数字のほか、必要な場合に使用される負号や小数点がある
- **文字型** - 1 つまたは複数の英数字文字
- **日付時刻型** - サポートされる書式で表される日付、日付時刻、または時刻の値

各データ型は Analytics によって異なる方法で処理され、さまざまなコマンドや関数で使用できます。データ型の詳細については、"データ型" ページ 1457を参照してください。

式

式とは、値を持つ、または値を生成するすべてのステートメントのことです。最もシンプルな形の式は `2` や `"test"` などのリテラルですが、式は通常計算として使用され、演算子、条件、関数、および指定可能な値を有効に組み合わせることができる範囲で複雑にすることもできます。

```
((2 + (3 - 2)) * 2) > ROOT(9,0)
```

式は通常、Analytics 内で、演算フィールドに値を入力するために使用されたり、条件付きロジックの入力として使用されます。式の詳細については、「式」ページ 1458を参照してください。

関数

関数とは、一定数のパラメーターを入力とし、単一の値を返す組み込みルーチンです。関数を使用することで、コマンド内で使用する変数や、フィールド値を操作できます。

メモ

関数はフィールドデータを変更しません。入力としてフィールドデータまたは変数を使用するアルゴリズムまたは計算に基づいて、新しい値を生成して返します。関数が返す値をコマンドの入力として使用します。

関数は、関数名で始まり、そのすぐ後に左かっこ、関数に引数として渡される0個以上の値をカンマで区切ったリスト、右かっこ続きます。

例

`BETWEEN(値, 最小値, 最大値)` 関数は3つの引数を取り、値が最小値から最大値までの範囲内であれば True を返し、範囲外であれば False を返します。

- `値` - テストする式またはフィールド
- `最小値` - 範囲の最小値
- `最大値` - 範囲の最大値

```
BETWEEN(amount, 500, 5000)
```

関数の詳細については、「関数」ページ 1462を参照してください。

変数

変数とは、値を一時的に格納する場所です。変数には対応する ID があり、これを使ってコンピューターのメモリに格納されている値を参照、使用できます。

ACLScript は、`ASSIGN` コマンドを使って、変数を作成すると同時にそれに値を代入します。

```
ASSIGN v_age_in_years = 3
```

簡略化するために、`ASSIGN` キーワードを省略することもできますが、`ASSIGN` は暗黙に使用されて実行されます。

```
v_age_in_years = 3
```

メモ

ACLScript ではヌル値はサポートされていません。すべての変数には、サポートされているいずれかのデータ型を持つ、対応する値を設定する必要があります。スクリプトのインタープリターは、値の代入に使用されたデータ形式およびデータ修飾子を使ってデータ型を評価します。詳細については、「データ型」ページ 1457を参照してください。

変数の使用

変数は作成後、フィールド名や変数を参照する任意の場所で参照できます。また、`ASSIGN` コマンドを使って、それに新しい値を代入することもできます。

```
EXTRACT RECORD TO 'result.fil' IF age > v_age_in_years  
v_age_in_years = 5
```

また、文字列補間または変数代入を使用して、文字列リテラルを含めるには、`%` 文字に変数をラッピングできます。代入される変数が Analytics で検出されると、プレースホルダーが対応する値に置換されます。

```
ASSIGN v_table = "erp_data"  
OPEN %v_table%
```

変数の詳細については、「変数」ページ 1464を参照してください。

統制構造

制御構造は、指定したパラメーターに基づいて取るべき方向を決定する、スクリプトのコンポーネントです。ACLScriptには、条件分岐のロジックおよびループ構造が用意されています。

条件分岐ロジック

ACLScriptには、この言語のIFコマンドや、多数のコマンドのオプションパラメーターとして、条件付きロジックが実装されています。

ヒント

IFコマンドは他のコマンドを実行するかどうかを制御するのに使用するのに対し、IFパラメーターは他のコマンドの実行対象とするテーブル内レコードを決定するのに使用します。

IF コマンド

```
IF v_counter > 10 CLASSIFY ON customer_no
```

IF パラメーター

```
CLASSIFY ON customer_no IF state = 'NY'
```

ループ

LOOPコマンドは、ACLScriptにおけるループ制御構造を提供します。このコマンドは、統制テストの式がTrueと評価される限り、ループ内のステートメントを処理します。

制御構造の詳細についての参照先: "統制構造" ページ 1467

コメント

ACLScript では、スクリプト 言語のように `COMMENT` キーワードを使ってコメントを追加できます。コメントを使うことで、作成するコードを理解しやすくとともに、スクリプトを参照、使用、または理解しようとしているユーザーと対話することができます。

コメントの種類

ACLScript では、2 種類のコメントがサポートされています。

- **単一行のコメント** - `COMMENT` の後から行末までのすべてのテキストが無視されます。
- **複数行のコメントから成るブロック** - `COMMENT` から `END` キーワードまでのすべての行または空白行が無視されます。

単一行のコメント

単一行のコメントは、作成するスクリプト内の各ステップについて説明したり、変数について説明したりするために使用できます。

```
COMMENT *** 分析期間の開始日  
v_Start_Date = `20150101`
```

複数行のコメントから成るブロック

複数行のコメントから成るブロックは、スクリプトまたはスクリプトのセクションについて説明する場合に使用できます。

```
COMMENT  
*****  
** このスクリプトのセクションは、インポート対象となるデータを準備します  
*****  
END
```

ヘッダーコメント ブロック

各スクリプトの先頭にスクリプトの主な内容を記述したヘッダーコメント ブロックを追加するのは、有効な方法です。

```
COMMENT
*****
*** スクリプト名:   {App_ID}{Script name}
*** パラメーター:  {Detailed description}
*** 出力:          {Describe parameters}
*** 作成者:        {Name}, ABC Corporation, {Month YYYY}
*** 変更者:        {Name}, ABC Corporation、スクリプトの目的とロジック
*** バージョン:    1.1.1 {app_ver.script_ver.defect.fix}
*****
END
```

データ型

ACLScript では、論理、数値、文字、日付時刻という4つの基本的なデータ型がサポートされています。

型	説明	制限	修飾子あり	例
文字	1つまたは複数の文字です。	32,767 バイト	一重引用符または二重引用符	<ul style="list-style-type: none"> 'John Doe' "John Doe"
数値	数値型の値には、0 ~ 9 の数字のほか、必要な場合に使用される負号や小数点があります。	22 桁	修飾子なし	<ul style="list-style-type: none"> 100 -5 5.01 22222.1232
日付時刻	サポートされる書式で表される日付、日付時刻、または時刻の値です。	<ul style="list-style-type: none"> 最小値 = 1900-01-01 最大値 = 9999-12-31 	<ul style="list-style-type: none"> バッククォート 時刻値の場合：先頭の 't' または単一の空白 	<ul style="list-style-type: none"> `20160101` `141231` `t2359` `20141231T235959` `20141231 235959`
論理	最もシンプルなデータ型です。論理型のデータは、T (True、真) または F (False、偽) という真理値を表します。 '='、'>'、'<' などの比較演算子は論理型の値を返します。	<ul style="list-style-type: none"> T F 	修飾子なし	<pre>ASSIGN v_truth = 5 > 4</pre> は T と評価されます。

式

式とは、値を持つすべてのステートメントのことです。最もシンプルな形の式はリテラルですが、演算子、条件、関数、および指定可能な値を有効に組み合わせることができる範囲で複雑にすることもできます。

式の要素

リテラル値

リテラル値は、文字リテラル `'私の値'` など、文字どおりに解釈されるように記述された値です。リテラルの詳細については、「データ型」前のページを参照してください。

演算子

演算子は、算術を実行したり、指定された値の文字列評価、比較評価、または論理的评价を実行したりするようにスクリプトのインタプリターに指示する記号です。

演算子の種類(優先順位順に一覧表示)	演算子(優先順位順に一覧表示)	例
かっこ	<ul style="list-style-type: none"> <code>()</code> は優先順位を指定します <code>()</code> 関数の演算子 	<code>(5 + 3) * 2</code>
単項	<ul style="list-style-type: none"> 非論理 - 否定 	<code>v_truth = NOT (3 < 2)</code>
算術演算子	<ul style="list-style-type: none"> <code>^</code> 指数 <code>*</code> 乗算, / 除算 <code>+</code> 加算, - 減算 <p>メモ 乗算演算子同士の優先順位は等しいため、左から右に評価されます。 加算演算子同士の優先順位は等しいため、左から右に評価されます。</p>	<code>1 + 5 - 3 * 2</code>
文字列	+ 連結	<code>"これは私の" + "スクリプトです"</code>
比較演算子	<ul style="list-style-type: none"> <code><</code> より小さい <code>></code> より大きい <code>=</code> 等しい <code>>=</code> 以上 <code><=</code> 以下 	<code>IF amount <> 100</code>

演算子の種類(優先順位順に一覧表示)	演算子(優先順位順に一覧表示)	例
	<ul style="list-style-type: none"> <> 等しくない <p>メモ 比較演算子同士の優先順位は等しいため、左から右に評価されます。</p>	
二項論理演算子	<ul style="list-style-type: none"> AND または & OR または 	<code>IF amount > 5 AND amount < 10</code>

関数

式は、関数によって返された値を使って評価されます。関数は、式の全要素のうち、最も高い優先順位で実行されます。関数の詳細については、「関数」ページ 1462を参照してください。

式の例

6 として評価されます。

```
(2 + (3 - 2)) * 2
```

True として評価されます。

```
((2 + (3 - 2)) * 2) > ROOT(9,0)
```

'ACLScript チュートリアル' として評価されます。

```
'AC' + 'Lscri' + 'pt ' + 'チュートリアル'
```

式を使用した演算フィールドの定義

現在開いているテーブルに式を使って追加のデータフィールドを作成するには、**演算フィールド**を使用します。演算フィールドは、開いているテーブルに追加されるフィールドであり、指定された式の値が入力されます。

演算フィールドの構文

```
DEFINE FIELD 名前 COMPUTED 式
```

- **名前** - 生成する演算フィールドの名前
- **式** - フィールドの値を生成するための演算または計算

演算フィールドの例

```
DEFINE FIELD c_full_name COMPUTED first_name + ' ' + last_name
```

ヒント

演算フィールドの前には `c_` という接頭辞を付けることで、このフィールドが元のソースデータでなく演算データであることを示します。

条件付き演算フィールド値の定義

演算フィールドに条件を使うことで、様々な場合用に値を定義することもできます。

```
DEFINE FIELD c_total COMPUTED  
  
amount * ca_tax_rate IF state = 'CA'  
amount * ny_tax_rate IF state = 'NY' OR state = 'NJ'  
amount * general_rate
```

最初の条件式が True と評価された場合、その場合に指定されていた値が使用されます。この例では、`amount * general_rate` は、いずれの条件式も True と評価されなかった場合に使用されるデフォルト値です。

メモ

DEFINE FIELD コマンドで IF、WIDTH、PIC、または AS パラメーターを指定する場合を除いて、行のコマンドと条件の間には空白行を追加する必要があります。詳細については、"DEFINE FIELD ... COMPUTED コマンド" ページ 1716を参照してください。

関数

関数とは、一定数のパラメーターを入力し、単一の値を返す組み込みルーチンです。関数を使用することで、コマンド内で使用する変数や、フィールド値を操作できます。

メモ

関数はフィールドデータを変更しません。入力としてフィールドデータまたは変数を使用するアルゴリズムまたは計算に基づいて、新しい値を生成して返します。関数が返す値をコマンドの入力として使用します。

関数構文

関数は、関数名で始まり、そのすぐ後に左かっこ、関数に引数として渡される0個以上の値をカンマで区切ったリスト、右かっこと続きます。

例

`BETWEEN(値, 最小値, 最大値)` 関数は3つの引数を取り、値が最小値から最大値までの範囲内であればTrueを返し、範囲外であればFalseを返します。

- `値` - テストする式またはフィールド
- `最小値` - 範囲の最小値
- `最大値` - 範囲の最大値

```
BETWEEN(amount, 500, 5000)
```

関数の引数

関数の引数とは、その関数に渡される特定の入力値のことです。

関数の引数は引数リストを介して関数に渡されます。引数リストは、リテラル値、変数および、パラメーターのデータ型の値として評価される式をカンマで区切ったリストです。データ型の操作の詳細については、"データ型" ページ 1457を参照してください。

メモ

プロジェクトでヨーロッパの数値書式を使用する場合、または複数のリージョンで互換性のあるスクリプトを作成する場合、符号付き数値を渡すのであれば、カンマでなく空白文字で関数の引数を区切ってください。符号付き数値を受け入れる関数を使用するには、明示的な区切り文字が必要です。

関数とコマンド

コマンドと関数の違いは微妙ですが、この違いは ACLScript を使用する場合には重大な意味を持ちます。

関数	コマンド
入力としてフィールド、値、またはレコードを使用し、新しい値を生成して、それを返します。	入力としてテーブルを使用し、新しいレコードおよびテーブルを生成します。
式、演算フィールド、コマンド パラメーター値、変数、フィルターの中で使用され、コマンドの実行を支援、変更します。	データの分析およびインポート、ならびに結果の生成に使用されます。
スクリプト内で独立したステップになることはできません。	スクリプト内で独立したステップになることができます。

変数

変数とは、値を一時的に格納する場所です。変数には対応する ID があり、これを使ってコンピューターのメモリに格納されている値を参照、使用できます。

メモ

このトピックでは、ACLScript の変数の基本事項について説明します。包括的な情報については、"ACLScript での変数の操作" ページ 1554を参照してください。

ACLScript 内で変数を使用する方法

変数の作成とそれへの値の代入

ACLScript は、`ASSIGN` コマンドを使って、変数を作成すると同時にそれに値を代入します。

```
ASSIGN v_age_in_years = 3
```

簡略化するために、`ASSIGN` キーワードを省略することもできますが、`ASSIGN` は暗黙に使用されて実行されます。

```
v_age_in_years = 3
```

メモ

ACLScript ではヌル値はサポートされていません。すべての変数には、サポートされているいずれかのデータ型を持つ、対応する値を設定する必要があります。スクリプトのインタープリターは、値の代入に使用されたデータ形式およびデータ修飾子を使ってデータ型を評価します。詳細については、"データ型" ページ 1457を参照してください。

変数の使用

変数は作成後、フィールド名や変数を参照する任意の場所で参照できます。また、`ASSIGN` コマンドを使って、それに新しい値を代入することもできます。

```
EXTRACT RECORD TO 'result.fil' IF age > v_age_in_years  
v_age_in_years = 5
```

また、文字列補間または変数代入を使用して、文字列リテラルを含めるには、`%` 文字に変数をラッピングできます。代入される変数が Analytics で検出されると、プレースホルダーが対応する値に置換されません。

```
ASSIGN v_table = "erp_data"  
OPEN %v_table%
```

変数のタイプ

Analytics では次の種類の変数を使用されます。

- **システム生成変数** -: この変数は、コマンドを実行すると自動的に生成されます。
- **永久変数** -: この変数は、削除されるまで、コンピューターのメモリに残り、Analytics プロジェクトを閉じた後も継続します。

メモ

永久変数を定義するには、識別子の前に '_' を付けます: `_v_company_name = 'Acme'`.

- **セッション変数** -: この変数は、削除されるまで、また、Analytics プロジェクトを閉じるまで、コンピューターのメモリに残ります。

変数 ID

変数 ID は大文字と小文字を区別せず、変数の種類に関連した特定の規約に従います。

- システム生成変数 ID では `OUTPUTFOLDER` のようにすべて大文字が使用されます。
- 永続的変数 ID には、`_v_permanent` のように、 '_' を接頭辞として付ける必要があります。
- セッション変数 ID では規則により `v_varname` という書式が使用されますが、この命名規則に制限されることはありません。

変数値の表示

スクリプト開発中またはデバッグ中にスクリプトを実行する際、変数値を追跡すると有益な場合があります。スクリプトのログファイルに変数値を収集するには、`DISPLAY` コマンドを使用します。

```
DISPLAY v_age_in_years
```

ACLScript はこのコマンドに遭遇すると、そのコマンドをログファイルに記録します。スクリプト実行のこの段階で変数値を表示するには、ログに記録されたそのエントリをクリックします。

ヒント

変数を使って、デバッグを支援することもできます。それには、スクリプト内にブレークポイントを挿入したうえで、ナビゲーターの **変数** タブで変数値を調べます。

統制構造

制御構造は、指定したパラメーターに基づいて取るべき方向を決定する、スクリプトのコンポーネントです。ACLScriptには、条件分岐のIF ロジックおよびループ構造が用意されています。

IF を使用した条件分岐ロジック

ACLScriptには、この言語のIF コマンドや、多数のコマンドのオプション パラメーターとして、条件分岐ロジックが実装されています。

- コマンド -の指定により、そのコマンドを実行するかどうかを制御します。
- パラメーター-により、そのコマンドの実行対象となる、テーブル内のレコードを決定します。

IF コマンド

IF コマンドを使用する際には、条件式と、その式がTrue として評価された場合に実行するコマンドを指定します。

```
IF v_counter > 10 CLASSIFY ON customer_no
```

この条件付き構造は実行するコードを制御するので、テストの式に基づいてテーブル全体を処理したい場合には、IF コマンドを使用できます。式がTrue として評価された場合には、コマンドはテーブル内のすべてのレコードに対して実行されます。IF コマンドの詳細については、「IF コマンド」 ページ 1868を参照してください。

IF パラメーター

コマンドの実行対象にするレコードをフィルターで抽出するために使用できるオプションのIF パラメーターは、数多くのコマンドが受け付けています。

```
CLASSIFY ON customer_no IF state = 'NY'
```

このステートメントが実行されると、state フィールドの値が'NY' である、テーブル内のすべてのレコードが分類化されます。

ループ

LOOP コマンド

LOOP コマンドは、ACLScript におけるループ制御構造を提供します。

メモ

LOOP コマンドは、GROUP コマンド内で実行する必要があり、単独で実行することはできません。

以下の例の LOOP コマンドは、指定された WHILE 式が true である限り、ループ内のステートメントを処理します。

```
ASSIGN v_counter = 10
GROUP
  LOOP WHILE v_counter > 0
    v_total = v_total + amount
    v_counter = v_counter - 1
  END
END
```

この構造は amount フィールドの値を変数 v_total への加算を 10 回繰り返します。各繰り返しの終わりに v_counter 変数が 1 ずつ減算され、WHILE 式でテストされます。この式が False と評価されると、ループは終了し、スクリプトが進行します。

ループが完了すると、v_total には 10 レコードの amount フィールドの値の合計が保持されます。

ループの詳細については、「LOOP コマンド」ページ 1986 を参照してください。

LOOPING とサブスクリプト

時々、LOOP コマンドは必要な正確なループ機能を提供しません。このような場合、DO SCRIPT コマンドを使用して別の Analytics スクリプトを呼び出すこともできます。DO SCRIPT スクリプト名 WHILE 条件テスト。

次の一般的な方法のいずれかを使用して、ループが終了するのを制御できます。

- **フラグ** - 論理フラグ変数が偽に設定されるまでループが続きます。
- **カウンター** - 増分または減少変数が条件しきい値を超えるまでループが続きます。

サブスクリプトの呼び出しの詳細については、「DO SCRIPT コマンド」ページ 1757 を参照してください。

例

C:\data フォルダのすべての CSV ファイルをプロジェクトにインポートする必要があります。`DIRECTORY` コマンドを使用して、フォルダからファイルのリストを取得できます。ただし、`IMPORT` コマンドを `GROUP` 構造内で使用できません。`DIRECTORY` が作成するテーブルを使用した別のループ方法が必要です。

このためには、次のメイン スクリプトを作成します。

1. `DIRECTORY` コマンドを実行し、結果をテーブルに保存します。
2. テーブルのレコード数を取得し、カウンターとして使用します。
3. テーブルのレコードごとにサブスクリプトを 1 回呼び出し、現在のレコードに対して `IMPORT` コマンドを実行します。

メインのスクリプト

```
COMMENT メインのスクリプト

DIRECTORY "C:\data\*.csv" TO T_Table_To_Loop
OPEN T_Table_To_Loop
COUNT
v_Num_Records = COUNT1
v_Counter = 1
DO SCRIPT Import_Subscript WHILE v_Counter <= v_Num_Records
```

サブスクリプトのインポート

```
COMMENT Import_Subscript

OPEN T_Table_To_Loop
LOCATE RECORD v_Counter

COMMENT CSV ファイルをインポートするためのコード...

ASSIGN v_Counter = v_Counter + 1
```

変数はプロジェクトで実行されるすべてのスクリプトで共有されるため、メイン スクリプトは、`v_Counter` の値が `v_Num_Records` の値を超えるまでサブスクリプトを呼び出します。サブスクリプトが実行されるたびに、`v_Counter` が増えます。

この構造では、各レコードに対して `IMPORT` コマンドを呼び出しながら、テーブルをループできます。メインのスクリプトが完了すると、C:\data フォルダからすべての CSV ファイルがインポートされます。

グループ化とループ処理

GROUP および LOOP コマンドは、一連のコマンドを繰り返し実行するための2つの方法を提供します。GROUP は各レコードに対して1つ以上のコマンドを1回繰り返します。LOOP は単一レコードに対して一連のコマンドを複数回繰り返して実行し、GROUP ブロック内でのみ使用できます。

GROUP の簡単な例

請求書データのテーブル `Ap_Trans` があります。このデータを使用して、請求金額の合計を計算する必要があります。

Vendor_Number	Vendor_Name	Invoice_Number	日付	金額
11663	More Power Industries	5981807	2000-11-17	618.30
13808	NOVATECH Wholesale	2275301	2000-11-17	6705.12
12433	Koro International	6585673	2000-11-17	7955.46

この金額を計算するには、GROUP コマンドを使用します。GROUP の各繰り返し内で次のことを実行します。

1. 現在のレコードの時点で合計を計算します。
2. 請求書番号、金額、日付、合計を結果テーブルに抽出します。

```
OPEN AP_Trans

COMMENT 累計の初期値をゼロに設定 END
ASSIGN v_running_total = 0.00

COMMENT テーブルの各レコードを繰り返し、累計を計算して抽出 END
GROUP
  ASSIGN v_running_total = v_running_total + Amount
  EXTRACT Invoice_Number, Amount, Date, v_running_total AS "Running total"
TO results1
END
```

スクリプトが実行されると、GROUP ブロック内のコマンドはテーブルの各レコードに対して上から下に処理され、合計が計算され、抽出されます。実行される GROUP コマンドを見ると、次のような仕組みになっています。

GROUP の最初の繰り返し: 合計 = 0.00 + 618.30

GROUP は最初のレコードの請求金額を 0.00 の初期合計に追加し、フィールドを結果テーブルに抽出します。

Vendor_Number	Vendor_Name	Invoice_Number	日付	金額
11663	More Power Industries	5981807	2000-11-17	618.30
13808	NOVATECH Wholesale	2275301	2000-11-17	6705.12
12433	Koro International	6585673	2000-11-17	7955.46

GROUP の2番目の繰り返し: 合計 = 618.30 + 6705.12

GROUP は2番目のレコードの請求金額を新しい合計の618.30に追加し、フィールドを結果テーブルに抽出します。

Vendor_Number	Vendor_Name	Invoice_Number	日付	金額
11663	More Power Industries	5981807	2000-11-17	618.30
13808	NOVATECH Wholesale	2275301	2000-11-17	6705.12
12433	Koro International	6585673	2000-11-17	7955.46

GROUP の3番目の繰り返し: 合計 = 7323.42 + 7955.46

GROUP は3番目のレコードの請求金額を新しい合計 7323.42 に追加し、フィールドを毛かテーブルに抽出します。

Vendor_Number	Vendor_Name	Invoice_Number	日付	金額
11663	More Power Industries	5981807	2000-11-17	618.30
13808	NOVATECH Wholesale	2275301	2000-11-17	6705.12
12433	Koro International	6585673	2000-11-17	7955.46

最終結果テーブル

GROUP がテーブルの最終レコードを処理した後、次の結果テーブルが作成されます。

Invoice_Number	金額	日付	Running_total
5981807	618.30	2000-11-17	618.30
2275301	6705.12	2000-11-17	7323.42
6585673	7955.46	2000-11-17	15278.88

GROUP IF を使用したさまざまな場合の処理

上記と同じ Ap_Trans テーブルを使用して、3種類の請求書の累計を計算する必要があります。

- 高額(1000.00以上)
- 標準額(100.00~1000.00)
- 低額(100.00未満)

GROUP コマンドは異なる場合を処理するために IF/ELSE 構造を提供します。テストする条件式を提供し、レコードが真と評価される場合は、ブロック内のコマンドが実行されます。

場合のテスト方法

場合は上から下にテストされ、レコードは1つの IF/ELSE ブロックでのみ処理できます。レコードの真に評価される最初の場合は、レコードを処理するものです。

1. GROUP が最初のレコードを処理すると、最初の IF 条件に対してテストします(Amount >= 1000)。真と評価される場合、この場合のコードが実行され、他の場合はテストされません。
2. 最初の場合が偽と評価されると、次の ELSE IF 条件 (Amount >= 100) がテストされます。同時に真と評価される場合、この場合のコードが実行され、他の場合はテストされません。
3. いずれの IF または ELSE IF 場合が真と評価される場合、ELSE ブロックのデフォルトの場合がレコードを処理します。

メモ

レコードが複数の場合で真と評価される場合、レコードはテストする最初の IF/ELSE ブロックでのみ処理されます。レコードは GROUP コマンド内の複数の IF/ELSE ブロックでは処理されません。

```
OPEN AP_Trans
```

```
COMMENT 累計の初期値を設定 END
ASSIGN v_running_total_hi = 0.00
ASSIGN v_running_total_med = 0.00
ASSIGN v_running_total_low = 0.00
```

```
COMMENT GROUP IF を使用して、請求金額に応じて別の ASSIGN および EXTRACT コマンドを実
```

```

行 END
GROUP IF Amount >= 1000
  ASSIGN v_running_total_hi = v_running_total_hi + Amount
  EXTRACT Invoice_Number, Amount, Date, v_running_total_hi AS "Running
total" TO results_hi
ELSE IF Amount >= 100
  ASSIGN v_running_total_med = v_running_total_med + Amount
  EXTRACT Invoice_Number, Amount, Date, v_running_total_med AS "Running
total" TO results_med
ELSE
  ASSIGN v_running_total_low = v_running_total_low + Amount
  EXTRACT Invoice_Number, Amount, Date, v_running_total_low AS "Running
total" TO results_low
END

```

スクリプトの実行時には、GROUP コマンドは各レコードの請求金額をテストします。金額によっては、レコードは3つの合計(低、中、高)のいずれかを更新するために使用され、3つの結果テーブルが生成されます。

GROUP 内の LOOP

GROUP を使用してテーブルのレコードを処理するときには、LOOP コマンドを使用して、単一のレコードに対して一連のコマンドを複数回実行できます。LOOP は GROUP 内の繰り返し内で発生する2番目の繰り返しであり、指定するテスト条件が偽と評価されるまで実行されます。

LOOP を使用してフィールドを分割する

請求データを含む次のテーブルがあり、部署単位で請求金額の特定の情報を分離する必要があります。1つの請求書は複数の部署に関連付けることができ、部署コードがカンマ区切りでテーブルに保存されます。

Vendor_Number	Invoice_Number	日付	金額	Dept_Code
11663	5981807	2000-11-17	618.30	CCD,RDR
13808	2275301	2000-11-17	6705.12	CCD
12433	6585673	2000-11-17	7955.46	CCD,LMO,RDR

請求金額を部署ごとに抽出するには

1. GROUP コマンドを使用して、テーブルの各レコードを処理します。
2. 各レコードに関連付けられた部署 (n) 数を計算します。
3. LOOP コマンドを使用して、レコードに関連付けられた各部署のデータを抽出する操作を n 回繰り返します。

メモ

LOOP 内では v_counter 変数を増やす必要があります。そうしない場合、WHILE テストは常に真と評価され、スクリプトは無限ループに入ります。スクリプトに SET LOOP コマンドを含め、無限ループを防止できます。詳細については、[SET LOOP コマンド](#)を参照してください。

```
COMMENT
GROUP を使用すると、各部署コード フィールドのコメントをカウントし、レコードに関連付けられている
部署数を特定します
LOOP でフィールドの各コードをレコードごとに「ループ」します。ループを繰り返すたびに、単一のコード
のレコードが result1 テーブルに抽出されます
END

GROUP
v_department_count = OCCURS(Dept_Code, ',')
v_counter = 0
LOOP WHILE v_counter <= v_department_count
v_dept = SPLIT(Dept_Code, ',', (v_counter + 1))
EXTRACT FIELDS Invoice_Number, Amount, v_dept AS "Department" TO result1
v_counter = v_counter + 1
END
END
```

スクリプトが実行されると、GROUP ブロック内のコマンドは上から下にテーブルの各レコードに対して処理されます。各レコードでは、LOOP コマンドがカンマ区切りのリストの部署コード単位で1回レコードに対して繰り返され、レコードを抽出します。実行される GROUP および LOOP コマンドを見ると、次のような仕組みになっています。

GROUP の1番目の繰り返し: LOOP の2つの繰り返し

Vendor_Number	Invoice_Number	日付	金額	Dept_Code
11663	5981807	2000-11-17	618.30	CCD,RDR
13808	2275301	2000-11-17	6705.12	CCD
12433	6585673	2000-11-17	7955.46	CCD,LMO,RDR

テーブルの1番目のレコードでは、v_department_count の値が1になるため、LOOP は2回繰り返しません。

1. LOOP の最初の繰り返し:

- v_counter = 0
- v_depart = CCD

次のレコードが抽出され、v_counter の値が1になるため、LOOP はもう一度繰り返します。

5981807	618.30	CCD
---------	--------	-----

2. LOOP の 2 回目の繰り返し:

- v_counter = 1
- v_depart = RDR

次のレコードが抽出され、v_counter の値が 2 になるため、LOOP は繰り返さず、GROUP は次のレコードに進みます。

5981807	618.30	RDR
---------	--------	-----

GROUP の 2 番目の繰り返し: LOOP の 1 つの繰り返し

Vendor_Number	Invoice_Number	日付	金額	Dept_Code
11663	5981807	2000-11-17	618.30	CCD,RDR
13808	2275301	2000-11-17	6705.12	CCD
12433	6585673	2000-11-17	7955.46	CCD,LMO,RDR

テーブルの 2 番目のレコードでは、v_department_count の値が 0 になるため、LOOP は 1 回繰り返しません。

- v_counter = 0
- v_depart = CCD

次のレコードが抽出され、v_counter の値が 1 になるため、LOOP は繰り返さず、GROUP は次のレコードに進みます。

2275301	6705.12	CCD
---------	---------	-----

GROUP の 3 番目の繰り返し: LOOP の 3 つの繰り返し

Vendor_Number	Invoice_Number	日付	金額	Dept_Code
11663	5981807	2000-11-17	618.30	CCD,RDR
13808	2275301	2000-11-17	6705.12	CCD
12433	6585673	2000-11-17	7955.46	CCD,LMO,RDR

テーブルの 3 番目のレコードでは、v_department_count の値が 2 であるため、LOOP は 3 回繰り返しません。

1. LOOP の最初の繰り返し:

- v_counter = 0
- v_depart = CCD

次のレコードが抽出され、v_counter の値が1になるため、LOOP はもう一度繰り返します。

6585673	7955.46	CCD
---------	---------	-----

2. LOOP の2回目の繰り返し:

- v_counter = 1
- v_depart = LMO

次のレコードが抽出され、v_counter の値が2になるため、LOOP はもう一度繰り返します。

6585673	7955.46	LMO
---------	---------	-----

3. LOOP の3回目の繰り返し:

- v_counter = 2
- v_depart = RDR

次のレコードが抽出され、v_counter の値が3になるため、LOOP は繰り返さず、GROUP はテーブルの最後に達します。

6585673	7955.46	RDR
---------	---------	-----

最終結果テーブル

GROUP がテーブルの各レコードを処理し、LOOP がすべての部署コードを繰り返した後、次の結果テーブルが生成されます。

Invoice_Number	金額	部門
5981807	618.30	CCD
5981807	618.30	RDR
2275301	6705.12	CCD
6585673	7955.46	CCD
6585673	7955.46	LMO
6585673	7955.46	RDR

関数を使用する方法

スクリプトを作成しない Analytics の新しいユーザーまたは基本ユーザーの場合でも、関数を使用すると、簡単かつ効果的にデータを操作できます。

関数は、フィルター、演算フィールド、コマンド内の式、スクリプトなどの Analytics のさまざまな領域で使用できます。これらの領域について知識がない場合でも問題ありません。関数の使用方法の手順の一部として、概要を説明します。

短いチュートリアル

併せて、このセクションの短いチュートリアルでは、Analytics 関数の概要を説明し、有用なことを実現するための使用方法を示します。

順番にチュートリアルを行うか、即時のニーズに対応するチュートリアルのみを行うことができます。

行いたいこと...	チュートリアル
関数について理解する	"関数とは何か" ページ 1479 <ul style="list-style-type: none"> ○ 関数の概念 ○ 関数の3つの基本部分
Analytics 関数について理解する簡単な方法を学習する	"さまざまな関数の詳細" ページ 1483 <ul style="list-style-type: none"> ○ Analytics 関数をすばやく簡単に理解する方法 ○ 関数を使用するときの一般的なエラー
関数を使用してデータをフィルタリングまたは検索する方法を学習する	"関数を使用してフィルターを作成する" ページ 1487 <ul style="list-style-type: none"> ○ フィルターの概要 ○ 以下を行うための関数の使用： <ul style="list-style-type: none"> ● 日付でフィルター ● 複数の値でフィルター ● 曖昧値でフィルター
関数を使用してデータをクリーニングまたは準備する方法を学習する	"関数を使用してデータをクリーニングする" ページ 1491 <ul style="list-style-type: none"> ○ データのクリーニングの概要 ○ 以下を行うための関数の使用： <ul style="list-style-type: none"> ● 空白スペースの削除 ● 不要な文字の削除
関数を組み合わせて効率と能力を高める方法を学習する	"データを同時にクリーニングしてフィルタリングする" ページ 1495 <ul style="list-style-type: none"> ○ ネストされた関数の概要

高度な関数の使用

基本を理解した後は、追加のチュートリアルで、Analytics で使用する関数を配置するさまざまな方法を説明します。

"高度な関数の使用" ページ 1499を参照してください。

関数とは何か

Analytics の関数は、特定の有用なタスクを実行する小さいツールであると考えられます。たとえば、関数を使用して、一貫しない書式設定のテキストで、大文字と小文字を標準化できます。

関数と入力	出力
<pre>PROPER("john SMITH")</pre>	<pre>John Smith</pre>

関数に対するもう一つの考え方は「不透明ボックス」です。片側から入力され、内部で変換処理が実行され、反対側から出力されます。



関数の3つの基本部分

例の `PROPER("john SMITH")` は任意の関数の3つの基本部分を示します。

- この場合、**関数名** - は `PROPER` です
- **括弧のセット** - 開く括弧 (`(`) と閉じ括弧 (`)`)
- **関数入力** - 括弧内のすべて: この場合、`"john SMITH"` (引用符を含む)

メモ

Analytics のどのドキュメントでも、関数名は大文字で記載されていますが、これはあくまでも書式設定の表記規則上、そう記載されているにすぎません。Analytics 自体では、関数を大文字で入力する必要はありません。

入出力

関数をより技術的に説明すると、**入力と出力**を許可するコンピューターによって実行される演算または処理です。

上記の例では、入力は"john SMITH"、出力はJohn Smithです。PROPER()関数によって実行される処理は、すべての単語を適切な大文字と小文字に変換します(先頭が大文字、その後が小文字)。

関数はソースデータを修正しません

関数は入力として使用されたソースデータを修正しません。入力データを使用して、出力結果を計算し、結果はコンピューターのメモリに格納されるため、それらを利用できます。

上記の例では、物理データ john SMITH はコンピューターで変更されません。出力の John Smith は「仮想データ」と見なすことができます。これはメモリに存在し、後続の処理で使用できます。

この点については、作成された演算フィールドで関数を使用する際にもう一度説明します。

関数の範囲

特定の関数によって実行される演算または処理の範囲は狭いです。PROPER()関数はテキストの大文字と小文字の変換以外何もしません。ただし、チュートリアルでは、各関数の範囲が狭い場合でも関数は強力であり、Analytics のデータ分析において重要です。

関数とコマンドの違い

Analytics の関数とコマンドはいずれもデータに対して演算または処理を実行しますが、関数の範囲は狭く、コマンドの範囲は通常広いです。例：

- **狭い範囲** - PROPER()関数はテキストの大文字と小文字を変換します
- **広い範囲** - SUMMARIZE コマンドはテーブルのすべてのレコードをグループ化します

関数はコマンドの入力として使用できます。たとえば、PROPER()関数を使用して、Name フィールドの大文字と小文字を変換してから、SUMMARIZE コマンドを使用して、正規化された Name フィールドでレコードをグループ化できます。

逆は当てはまりません。コマンドは関数の入力として使用できません。

その他の例

関数を使用すると、テキストの先頭と末尾のスペースを削除したり、ID 番号からハイフンを削除したり、特定の日付範囲の日付のレコードを検索できます。

例

次の例では、3つの異なる関数を実行できます。

- ALLTRIM() 関数
- EXCLUDE() 関数
- BETWEEN() 関数

関数と入力	出力
<pre>ALLTRIM(" Chicago ")</pre>	<pre>Chicago</pre> <p>文字のみ、先頭と末尾のスペースなし。</p>
<pre>EXCLUDE("VT-123-45", "-")</pre>	<pre>VT12345</pre> <p>文字と数値のみ、ハイフンなし。</p>
<pre>BETWEEN(`20170701`, `20170101`, `20171231`)</pre> <p>関数入力:</p> <ul style="list-style-type: none"> ◦ 最初の日付がテストされています ◦ 2番目の日付が範囲の開始日です ◦ 3番目の日付が範囲の終了日です 	<pre>T</pre> <p>真の T を返します。2017年7月1日は、2017年1月1日と12月31日の間です。</p>

関数への入力としてのリテラル値とフィールド

上記の例は実際またはリテラル入力値を使用するため、各関数の処理を正確に示します。Analytics では、一般的に、関数の主な入力としてフィールドまたは変数を使用します。

関数入力としてのフィールドは以下のとおりです。関数入力としての変数は次のチュートリアルで説明します。

例

これは、上記からの BETWEEN() の例ですが、入力がリテラル値ではなく日付フィールドです

```
BETWEEN(Invoice_Date, `20170101`, `20171231`)
```

関数は真の T を、2017年に該当する Invoice_Date フィールドのすべての日付に対して返します。他の年の日付には偽の F を返します。

メモ

BETWEEN() 関数で指定する2つの境界値は両方含まれます。このような詳細は、各関数のヘルプトピックを参照してください。

次の学習ステップ

任意の Analytics の関数をすばやく簡単に理解する方法を学習する: "さまざまな関数の詳細" 次のページ

さまざまな関数の詳細

関数の動作を理解する最も簡単な方法は、Analytics コマンドラインで関数を使用してみることです。前のチュートリアル例を使用して試してみることができます。

<code>PROPER("john SMITH")</code>	<code>EXCLUDE("VT-123-45", "-")</code>
<code>ALLTRIM(" Chicago ")</code>	<code>BETWEEN(`20170701`, `20170101`, `20171231`)</code>

1. Analytics で Analytics プロジェクトを開きます。
2. コマンドラインが開いていることを確認します。



開いていない場合は、**[ウィンドウ > コマンドライン]**を選択します。

3. 上記の関数の例のいずれかをコピーして、コマンドラインに貼り付けます。
4. 貼り付けられた例の前に `DISPLAY` とスペースを入力し、Enter を押します。

戻り値という関数出力が Analytics 表示画面に表示されます。



メモ

表示画面の関数出力に対しては何の操作もできません。これは読み取り専用の出力であり、特定の入力に対して特定の関数が返す内容を表示できます。

- 関数出力を一時的に保存するには、表示画面を固定  します。

ヒント

表示画面でリンクされた関数をクリックすると、コマンドラインに簡単に読み込むことができます。

一部の入力値を変更してみます。...

関数をコマンドラインで入力するか読み込んで、1つ以上の入力値を変更して、出力の変化を確認します。

ヒント

コマンドラインで関数を多数実験する場合は、`DISPLAY` の代わりに `disp` と入力できます。

EXCLUDE() の例

EXCLUDE() の例では、`VT` を除外する文字に追加する場合、数値のみを含む出力にしてください。

```
EXCLUDE("VT-123-45", "VT-")
```

BETWEEN() の例

BETWEEN() の例では、リテラル請求日付を2016年7月1日に変更するとき何が起こるでしょうか。請求日は3つの入力値の最初の値です。

```
BETWEEN(`20160701`, `20170101`, `20171231`)
```

出力が `T` から `F` (真から偽) に変わったことがわかります。2016年7月1日は、2017年1月1日から12月31日の間ではないためです。

使用できる他の関数

コマンドラインで `DISPLAY` 方法を使用すると、Analytics 関数で実験できます。Analytics にはさまざまな目的に対応する130以上の関数があります。

メモ

コマンドラインで関数とともに `DISPLAY` を使用することは、テストまたは学習目的専用です。Analytics 使用時にこれら以外の目的で `DISPLAY` を関数と一緒に使用することはありません。

試してみる他の関数を見つける

1. このヘルプトピックのメニューで**関数**の上にカーソルを置き、関数カテゴリを選択します。
2. カテゴリページで、目的の関数名をクリックします。
3. 個別の関数ページの例をコピーし、コマンドラインに貼り付けます。
フィールドまたは汎用プレースホルダーではなく、リテラル値を使用する例を必ず選択します。
4. 貼り付けられたコマンドの前に `DISPLAY` とスペースを入力し、ヘルプトピックと同じ出力値を返すことを確認します。
5. 入力値を変更し、異なる出力値を作成して、関数の動作を詳細に学習します。

ヒント

一部の関す字入力を理解する必要がある場合は、関数ヘルプトピックを参照してください。

エラーメッセージが表示される場合

関数の実験中にエラーメッセージが表示される場合は、コマンドラインに関数を入力したときに細かいエラーがあった可能性が考えられます。

一部のエラーメッセージは重大に見える場合がありますが、一般的にエラーは微細であり、内容がわかれば簡単に修正できます。

関数を入力する方法を制御するルールは、Analytics のコマンドラインでもコマンドライン以外の場所でも、厳格に適用されます。

関数名	関数名は正しい綴りで入力する必要があります。
括弧	開くかっこは関数名の直後に置き、間にスペースを入れません。 <code>PROPER("john SMITH")</code> (<code>PROPER ("john SMITH")</code>)ではない 関数の括弧は開いて閉じる必要があります。 ネストされた関数の場合、開く括弧と閉じ括弧を追跡することが困難になってきます。
テキスト	リテラルテキスト値は、引用符で囲む必要があります。 疑問符は標準の疑問符である必要があります。一部のソースからコピーして貼り付けたときに発生する可能性がある曲がった疑問符や傾いた疑問符はエラーの原因になります。
日付	リテラル日付値は `バックオート` で囲み、YYYYMMDD 形式 (または YYMMDD) を使用する必要があります。
フィールドと数値	フィールド名と数値には句読点がありません。 ◦ Invoice_Date

	◦ 1000.00
DISPLAY	コマンドラインでは関数の前に <code>DISPLAY</code> を置く必要があります(他の場所では異なる)。
データ型	関数では、入力値が特定のデータ型である必要があります。 一部の関数は複数のデータ型を許可しますが、他の関数は文字データ型または数値データ型などのみを許可します。関数のヘルプトピックでは、各関数で有効なデータ型を説明します。 データ型の詳細については、"Analytics のデータ型" ページ 811を参照してください。

ヒント

関数構文の細かいエラーは見つけることが困難です。エラーが発生した場合は、構文を注意して確認してください。

関数ヘルプトピックでは、各関数に必要な構文の包括的な情報を提供します。

次の学習ステップ

関数を使用してさまざまな方法でデータをフィルターリングする方法を学習する: "関数を使用してフィルターを作成する" 次のページ

関数を使用してフィルターを作成する

Analytics の関数を使用してフィルターを作成できます。さまざまな関数により、特定のニーズに応じて、異なる種類のフィルターを作成できます。

フィルターの目的

フィルターはデータ分析の基本かつ重要なコンポーネントです。フィルターにより、現在関心がないレコードを除外し、検査したいレコードのみを含めることができます。

百万件のレコードがあるテーブルがあり、その一部のみを検査する場合、何らかのフィルターが必要です。

フィルターの仕組み

フィルターはテーブルのレコードを評価する式であり、値 "T" (True) または "F" (False) を各レコードに対して返します。

例

重大であると見なされる売掛金テーブルの金額のみを検査したいとします。重大性のしきい値は \$1000.00 であるため、次のフィルターを使用します。

```
Invoice_Amount >= 1000.00
```

このフィルターは \$1000.00 以上の金額に対して True、\$1000.00 未満の金額に False を返します。真と評価されたレコードはフィルターに含まれ、偽と評価されたレコードは除外されます。

除外されたレコードはフィルターの適用中はビューに表示されません。また、テーブルに対して実行する Analytics コマンドの実行対象から除外されます。

高度なフィルター

より大きい > や未満 < 記号などの演算子を使用して、簡易フィルターを作成できますが、関数を使用すると、より高度なフィルターを作成できます。

日付でフィルター

前のチュートリアルから BETWEEN() 例のバージョンを使用して、最初の四半期からの請求書のみを含むフィルターを作成します。

1. Analytics で **Sample Project.ACL** を開き、**Ap_Trans** テーブル (**Tables\Accounts_Payable\Ap_Trans**) を開きます。
Sample Project.ACL がない場合は、日付フィールドがある他のテーブルを開きます。この例を操作するときには、フィールドが日付時刻データ型である必要があります。
2. このバージョンの BETWEEN() 例を [ビュー] タブの上部にある [フィルター] テキストボックスにコピーして貼り付け、Enter を押します。

```
BETWEEN(Invoice_Date, `20000101`, `20000331`)
```



結果: テーブルがフィルタリングされ、年の最初の四半期からの請求書のみが表示されます。

Ap_Trans テーブルを使用しない場合、BETWEEN() 関数のフィールド名と境界日付を更新して、データと一致させます。

フィールド名は表示名 (代替列タイトル) ではなく、物理フィールド名である必要があります。日付フィールドのヘッダーを右クリックし、**プロパティ** を選択して、物理および表示フィールド名を確認します。

メモ

[フィルター] テキストボックスでは **DISPLAY** を使用しないでください。

3. いずれかまたは両方の境界日付を変更し、別の日付フィルターを作成してみてください。
 リテラル日付を入力するときには、**YYYYMMDD** 形式を使用する必要があります。**Ap_Trans** テーブルを使用する場合、すべての日付は 2000 です。

ヒント

BETWEEN() を使用して、数値またはテキスト データをフィルタリングします。テキスト入力は "引用符" で囲みます。フィールド名または数値入力は囲まないでください。

```
Invoice_Amount, 1000.00
```

複数の値でフィルター

MATCH() 関数を使用して、複数の値で同時にフィルタリングします。

1. 次の入力の MATCH() 関数をコピーし、[フィルター] テキストボックスに貼り付けて、Enter を押します。

```
MATCH(Vendor.Vendor_City, "Austin", "Chicago", "Salt Lake City")
```

結果: Ap_Trans テーブルのフィルターが更新され、指定された3つの都市の業者からの請求書のみを表示します。

メモ

Vendor_City フィールドは Vendor テーブルにあります。これは Ap_Trans テーブル (Sample Project.ACL) に関連付けられます。関数で関連するフィールドを参照するには、*table name.field name* 構文を使用します。

開いているテーブルでフィールドを参照するには、*field name* のみを使用します。

2. フィールドを変更し、照合する3つの用語を追加して、異なる種類のフィルターを作成してみてください。

メモ

MATCH() 関数の検索語は大文字と小文字を区別します。

曖昧値でフィルター

関数の能力と有用性を理解し始めているはずですが、この3番目のフィルターは ISFUZZYDUP() 関数を使用します。同じまたはほぼ同じ値でフィルタリングできます。

大きいテーブルで同様の処理を手動で実行することは、不可能ではないとしても、非常に時間がかかります。

1. 次の入力で ISFUZZYDUP() 関数をコピーして、[フィルター] テキストボックスに貼り付けて、Enter を押します。

```
ISFUZZYDUP(Vendor.Vendor_Name, "Miller Co", 4)
```

結果: Ap_Trans テーブルが更新され、"Miller Co" と同じかほぼ同じ名前の業者の請求書のみが表示されます。業者 "Muller Corp." の2つのレコードが表示されます。

2. から に曖昧度を上げ、Enter を押します。
"MGMT Mfg." のレコードがフィルターに含まれます。
3. **フィルターの削除**  をクリックし、業者名を確認して、"Miller Co" を完全に一致しない他の業者名に似たような名前に変更します。

さまざまな曖昧設定で実験する値設定は 1 ~ 10 です。

Analytics の簡易検索は実際にはフィルターです

Analytics の簡易検索機能は、実際には FIND() 関数を使用したフィルターが機能しているものです。

1. Ap_Trans テーブルで、**フィルター**の削除  をクリックします。
2. フィルターテキストボックスの検索語 **931** を入力し、Enter を押します。

結果: 文字 **931** がないかどうか、テーブルが検索され、フィルタリングされた結果に2つのレコードが含まれます。

- 1 つには末尾が 931 の請求書番号が含まれます
- もう1 つには末尾が 931 の製品番号が含まれます

フィルターテキストボックスでは、検索語が FIND() 関数に変換され、入力 "931" を使用します。 `FIND("931")`

ジョブの適切なツール

FIND() は、データをフィルタリングしたり特定の項目を検索したりするために使用できる Analytics 関数です。テーブルのすべてのデータを検索できます。

ただし、既に学習したとおり、データのフィルタリングと検索には、その他の強力で柔軟な方法を提供する他の関数があります。

すべての Analytics 関数に精通すると、現在実現しようとしている処理に応じて、選択すべき関数が決まることがわかるようになります。

要点

関数を使用すれば、Analytics のどの場所 (含む: スクリプト) でもフィルターを作成できます。関数で作成されたフィルターは、Analytics のデータ分析の基本要素です。

次の学習ステップ

関数を使用してデータのクレンジングまたはデータ準備タスクを実行する方法を学習する: "関数を使用してデータをクレンジングする" 次のページ

関数を使用してデータをクリーニングする

Analytics の関数を使用して、データをクリーニングできます。さまざまな関数により、特定のニーズに応じて、さまざまなタイプのデータクリーニングを実行できます。

データのクリーニングが必要な理由

頻繁に、Analytics にインポートされるデータはクリーンではありません。つまり、完璧な書式または正規化が行われていません。書式設定が正しくないか、標準データではない場合、Analytics のコマンドが機能しないか、不正確な結果を返します。

データクリーニングの仕組み

データをクリーニングするときには、実際のソースデータをクリーニングまたは修正していません。ソースデータは常に読み取り専用です。

代わりに、ソースデータを処理し、正しい書式と正規化の「仮想データ」に出力する関数にソースデータを入力します。次に、クリーニングされた仮想データを、元のソースデータではなく、Analytics のコマンドに入力します。

要点

1 つ以上の関数を使用すると、さまざまなデータクリーニングまたは**データ準備**タスクを実行し、ソースデータが一貫していない場合でも、効果的かつ正確に作業できます。データ準備はほとんどのデータ分析の基本的な事前準備タスクです。

空白スペースの削除

データの一貫しない空白スペースは不正確な結果の一般的な原因です。ALLTRIM() 関数を使用すると、先頭と末尾の空白スペースを削除し、正確な結果を保証します。

例：空白スペース

業者テーブルを都市で並べ替えますが、一部の都市名の先頭のスペースが不正確な並べ替えの原因です。

Vendor_City 並べ替え

[][][][] Salt Lake City

[][] Chicago

Ann Arbor

Austin

Englewood

[] = 空白

ALLTRIM() 関数を使用して、先頭のスペースを削除し、正確な並べ替えを保証します。

```
ALLTRIM(Vendor_City)
```

ALLTRIM(Vendor_City) 並べ替え

Ann Arbor

Austin

Chicago

Englewood

Salt Lake City

メモ

ALLTRIM() 関数を Vendor_City フィールドに適用するには、ALLTRIM() を使用する演算フィールドを作成します。演算フィールドは後続のチュートリアルで検討します。

不要な文字の削除

一貫しない文字または重要ではない文字は、一般的に、データ分析を阻害します。Analytics のコマンドでデータを処理する前に、異なる関数を使用して、特定の文字のみを含めるか、特定の文字または文字列を除外できます。

自分で試す

以下の関数の例をコピーして、コマンドラインに貼り付け、戻り値を検証するか、異なる入力で実験できます。

コマンドラインでは、`DISPLAY` とスペースを例の前に置く必要があります。コマンドラインでの実験は前のチュートリアルで説明しています。

例：不要な文字

テーブルで重複テストを実行しますが、データの書式が一貫していないため、不正確な結果になります。

たとえば、一致しない書式の Phone Number フィールドで重複コマンドを実行すると、明らかに重複している次の2つの電話番号が重複として報告されません。

- (604) 555-1212
- Tel.No: 604-555-1212

すべての重複が検出されるようにするには、重複処理を実行する前に、関数を使用してデータを標準化します。

タスク	関数の例
データ番号の正規化	<pre>INCLUDE("(604) 555-1212", "1234567890")</pre> <p>6045551212 を返します</p> <pre>INCLUDE("Tel.No: 604-555-1212", "1234567890")</pre> <p>6045551212 を返します</p> <p>INCLUDE() 関数には指定された文字のみが出力に含まれます。この場合は、0~9の数値のみです。</p> <p>ヒント 含める文字のセットが小さく、除外するセットが大きい場合は、INCLUDE() を使用します。</p>
住所の正規化	<pre>EXCLUDE("#1550-980 Howe St.", "#.")</pre> <p>1550-980 Howe St を返します</p>

タスク	関数の例
	<pre data-bbox="586 296 1304 359">EXCLUDE("1550-980 Howe St", "#.")</pre> <p data-bbox="537 401 841 426">1550-980 Howe St を返します</p> <p data-bbox="537 447 1357 501">EXCLUDE() 関数は出力から指定された文字を除外します。この場合ハッシュ記号 (#) とピリオド (.) です。</p> <p data-bbox="623 543 678 569">ヒント</p> <p data-bbox="623 579 1300 634">含める文字のセットが大きく、除外する文字のセットが小さい場合は、EXCLUDE() を使用します。</p>
住所を正規化し、番地の略語を削除する	<pre data-bbox="586 680 1304 743">OMIT("#1550-980 Howe St.", " Street, St.,#")</pre> <p data-bbox="537 785 813 810">1550-980 Howe を返します</p> <pre data-bbox="586 848 1304 911">OMIT("1550-980 Howe Street", " Street, St.,#")</pre> <p data-bbox="537 953 813 978">1550-980 Howe を返します</p> <p data-bbox="537 999 1369 1054">OMIT() 関数は指定された文字と文字列を出力から除外します。この場合、ハッシュ記号 (#) と書式が一致しない St. と Street です</p> <p data-bbox="623 1096 678 1121">ヒント</p> <p data-bbox="623 1131 1300 1186">特定の文字列を除外し、文字列を構成する個別の文字を除外しない場合は、OMIT() を使用します。</p> <p data-bbox="623 1197 1284 1251">たとえば、単位として出現する Street は除外しますが、別の単語の S、t、r、e、または t は除外しません。</p>

次の学習ステップ

関数を使用して複数のタスクを同時に実行する方法を学習する: "データを同時にクリーニングしてフィルタリングする" 次のページ

データを同時にクリーニングしてフィルタリングする

2つの前のチュートリアルでは、関数を使用してデータをフィルタリングし、データをクリーニングする方法を説明しました。ここでは、関数をネストして、両方のタスクを同時に実行する方法を説明します。

入れ子になった関数

別の関数内に関数をネストし、片方の関数だけでは得られない結果を得ることができます。

基本構造

1レベルのネストでネストされた関数の基本構造を示します。

```
FUNCTION_2( FUNCTION_1(function_1 <入力> ) , function_2 <入力> )
```

`FUNCTION_1()` は完全に `FUNCTION_2()` 内に含まれていることがわかります。

評価の順序

ネストされた関数は最も内側の関数から最も外側の関数に向かって評価されます。このため、上記の一般的な例では次のようになります。

1. `FUNCTION_1(function_1 <入力>)` が最初に評価されます。
2. `FUNCTION_1()` の出力は `FUNCTION_2()` の入力のいずれかになります。
3. `FUNCTION_2()` は2番目に評価されます。

入力と出力について考える

一般的に、別の関数内で任意の Analytics 関数をネストし、必要な場合は複数のネストレベルを作成できます。

ただし、関数の出力はそれを含む関数の入力要件を満たす必要があります。たとえば、関数にデータ入力が必要な場合、それを含む関数を日付時刻データ型の値を出力する必要があります。

要点

関数のネストは強力で柔軟な機能であり、有益なさまざまな結果を達成できます。データをコマンドに入力する準備で、ソースデータの複数の変換を同時に実行できます。

大文字と小文字の標準化、複数の値でのフィルタリング

前のチュートリアルでは、MATCH() を使用して、複数の値をフィルタリングしました。MATCH() は大文字と小文字を区別します。このため、入力値の大文字と小文字が異なる場合、フィルターは不正確な結果を生成します。

正確な結果を得るには、UPPER() 関数を MATCH() 関数内でネストします。

例

Vendor_City フィールドを使用して、テーブルのレコードをフィルタリングしますが、都市名の入力が一貫していません。一部は先頭のみが大文字 ("Austin") で、一部はすべて大文字 ("AUSTIN") です。

UPPER() 関数を MATCH() 関数内にネストして、次のことができます。

1. Vendor_City フィールドのすべての値を大文字に変換する
2. 都市でレコードをフィルタリングする

フィルター条件をすべて大文字に調整し、UPPER() 関数による大文字値の出力と一致させる必要があります。

```
MATCH( UPPER(Vendor_City) , "AUSTIN", "CHICAGO")
```

以下のテーブルは、MATCH() 関数のみの使用と、ネストされた関数の使用との違いを示します。

MATCH() だけでは、フィルターが厳しすぎ、含まれるべきレコードが除外されます。

関数:	関数:
MATCH(Vendor_City, "Austin", "Chicago")	MATCH(UPPER(Vendor_City), "AUSTIN", "CHICAGO")
Austin	Austin
Chicago	Chicago
	AUSTIN
	CHICAGO

ヒント

ネストされた関数を使用する代わりに、フィルター条件に変数を追加できます。
`MATCH(Vendor_City, "Austin", "AUSTIN", "Chicago", "CHICAGO")`。ただし、追加のフィルター条件では、このアプローチは作業が多くなり、"AUstin" などの入力ミスがある値を取り込めません。UPPER() のネストが適切な方法です。

メモ

MATCH() または UPPER() 関数を Vendor_City フィールドに適用するには、関数を使用する演算フィールドを作成します。演算フィールドは後続のチュートリアルで検討します。

大文字と小文字の標準化、先頭のスペースの削除、複数の値でのフィルタリング

1レベルのネストにのみ制限されていません。要件に応じて、複数レベルのネストを作成できます。

注意：

- ネストされた関数は、最も内側から最も外側の関数へと評価されます。
- 関数の出力は、それを含む関数の入力要件を満たす必要があります。

例

2番目の例では、Vendor_City フィールドのデータはさらに一貫性がありません。最初の文字が一致していないだけでなく、一部の値には1つ以上の先頭スペースがあり、スペースがないものもあります。

UPPER() 関数を ALLTRIM() 関数内で、ALLTRIM() 関数を MATCH() 関数内でネストすると、次のことができます。

1. Vendor_City フィールドのすべての値を大文字に変換する
2. すべての先頭空白スペースの削除
3. 都市でレコードをフィルタリングする

```
MATCH( ALLTRIM( UPPER(Vendor_City) ), "AUSTIN", "CHICAGO")
```

ヒント

ネストされた関数を作成するときには、先頭括弧と後の括弧の数が合わなくなるが多くなります。括弧がない場合や一致しないことは、関数エラーの一般的な原因です。

先頭括弧 (の数は常に後の括弧) の数と一致する必要があります。上記の例では、3つの先頭括弧と3つの閉じ括弧があります。

以下のテーブルは、MATCH() 関数のみの使用と、ネストされた関数の使用との違いを示します。MATCH() だけでは、フィルターが厳しすぎ、含まれるべきレコードが除外されます。

関数:	関数:
MATCH(Vendor_City, "Austin", "Chicago")	MATCH(ALLTRIM(UPPER(Vendor_City)), "AUSTIN", "CHICAGO")
Austin	Austin
Chicago	Chicago
	AUSTIN
	CHICAGO
	[] Austin
	[] [] [] Chicago
	[] [] AUSTIN
	[] CHICAGO
	[] = 空白

次の学習ステップ

"関数を使用する方法" ページ 1477のすべてのチュートリアルを修了したら、"高度な関数の使用" 次のページに進むことができます。

高度なチュートリアルでは、Analytics の主要機能での関数の使用方法を説明します。

高度な関数の使用

Analytics 関数の基本を理解した後は、Analytics で関数を使用する方法を学習できます。

このセクションのチュートリアルでは、演算フィールドの関数、コマンドに埋め込まれた関数、スクリプトの関数を使用する方法を説明します。関数の実際の有用性は、これらの中心的な Analytics 機能で使用するときに明らかになります。

演算フィールドまたはスクリプトに精通していない場合でも問題ありません。チュートリアルのコースで説明します。

メモ

このような他の状況で関数を使用することは、前のチュートリアルで説明した使用方法よりも少し複雑です。ただし、関数自体はまったく同じ方法で動作します。

Analytics コマンド ラインで関数をすばやく簡単にテストして機能を確認できます。"さまざまな関数の詳細" ページ 1483

短いチュートリアル

チュートリアルは順番に完了するように設計されています。

学習内容...	チュートリアル
フィールドのすべての値に関数を適用する	"関数を使用して月別にレコードをグループ化する" 見開きページ <ul style="list-style-type: none"> 以下を行うことによるフィールドのすべての値への関数の適用 <ul style="list-style-type: none"> 演算フィールドを作成する Analytics コマンドに関数を埋め込む
関数で変数を使用する	"関数で変数を使用してユーザー入力を許可する" ページ 1507 <ul style="list-style-type: none"> 変数の概要 関数の入力として変数を使用する
スクリプトで関数を使用する	"すべてまとめる: スクリプトで関数を使用する" ページ 1512 <ul style="list-style-type: none"> さまざまな関数を利用するインタラクティブ スクリプト

関数の使用の基本

関数の基本的な使用方法の概要については、"関数を使用する方法" ページ 1477を参照してください。

関数を使用して月別にレコードをグループ化する

前のチュートリアルでシンプルにするために、多数の Analytics 関数の例はリテラル入力値を使用しています。たとえば、"john SMITH"、"VT-123-45" などです。しかし、Analytics テーブルのフィールドの値の全体セットに関数を適用するにはどうするのでしょうか。

関数をフィールド全体に適用すると、月別のテーブルのレコードのグループ化などのタスクを実行できます。

演算フィールド

フィールドのすべての値に関数を適用する 1 つの方法は、**演算フィールド**を作成することです。演算フィールドは一般的に実際の物理フィールドに基づいて作成するフィールドですが、Analytics によって計算される値から構成されます。

関数の出力と同様に、メモリに存在する Analytics によって計算された仮想データと演算フィールドを見ることができます。計算すると、この仮想データは後続の処理で使用できます。

演算フィールドを作成し、月別にレコードをグループ化する

MONTH() 関数を使用して、請求日フィールドのすべての日付から月の部分を抽出する **Month** 演算フィールドを作成できます。月別にテーブルのレコードをグループ化できます。

演算フィールドを作成する

1. Analytics で **Sample Project.ACL** を開き、**Ap_Trans** テーブル (**Tables\Accounts_Payable\Ap_Trans**) を開きます。
Sample Project.ACL がない場合は、日付フィールドがある他のテーブルを開きます。この例を操作するときには、フィールドが日付時刻データ型である必要があります。
2. **Month** 演算フィールドを作成するには、次の手順を実行します。
 - a. テーブルビューの上部にある **テーブルレイアウトの編集**  を選択します。
 - b. **テーブルレイアウト**  ダイアログボックスで **新しい式の追加** をクリックします。
 - c. **Name** フィールドで **Month** と入力し、**Default Value** フィールドでこのバージョンの MONTH() 関数をコピーして貼り付けます。

MONTH(Invoice_Date)

Ap_Trans テーブルを使用していない場合は、フィールド名を更新して、データに合わせます。

- d. **入力を受け入れる**  をクリックし、**テーブルレイアウト** ダイアログボックスを終了します。
- e. テーブルビューで **Invoice Date** 列のヘッダーを右クリックし、**使用可能なフィールド** の下で **列の追加** を選択し、**月** をダブルクリックして、**OK** をクリックします。

結果: Month 演算フィールドがビューに追加されます。Invoice Date 列の各日付の月の部分が含まれ、1 ~ 12 の数値で表示されます。

- f. **開いているプロジェクトの保存**  をクリックして変更を保存します。

月別のレコードをグループ化する

Month 演算フィールドを作成したので、これを使用して、Ap_Trans テーブルのレコードをグループ化できます。

1. メインメニューから、**分析 > 要約** を選択します。
2. **要約日** リストから、**Month** フィールドを選択します。
3. **小計フィールド** リストで **Invoice_Amount** を選択します。
4. **出力** タブをクリックし、**ファイル** を選択して、**Name** フィールドに `Ap_Trans_grouped` と入力して、**OK** をクリックします。

結果: Analytics は新しいテーブルを出力します。これは Ap_Trans テーブルのレコードを月別にグループ化します。月ごとに請求金額小計があり、月に発生するレコード数のカウントがあります。

Month	Invoice Amount	Count
1	85,670.22	12
2	4,496.56	6
3	2,941.80	5
4	467.40	1
5	8,272.57	5
6	1,582.86	2
7	3,101.98	4
8	21,146.96	2
9	32,577.32	20

Month	Invoice Amount	Count
10	41,595.89	19
11	70,779.26	19
12	6,008.51	7

提案される活動：月名の表示

CMOY()関数を使用すると、月の名前を表示する場合に、2番目の演算フィールドを作成できます。CMOYは"Character Month of Year"の略語です。

1. **Ap_Trans** テーブルで、**Month** 演算フィールドを作成したのに使用したのと同じ手順に従い、テーブルビューに追加します。ただし、こちらの手順では次の点が異なります。
 - **Name** フィールドで「Month_2」と入力します。
 - **Default Value** フィールドでこのバージョンのCMOY()関数をコピーして貼り付けます。

```
CMOY(Invoice_Date, 9)
```

結果: Month_2 演算フィールドは各月の名前でビューに追加されます。

2. **Ap_Trans** テーブルから月別にレコードをグループ化するのに使用したのと同じ手順に従います。ただし、こちらの手順では次の点が異なります。
 - **その他のフィールド** リストで "Month_2" を選択します。
 - **出力** タブで出力ファイル名 `Ap_Trans_grouped_2` を指定します。

結果: Analytics は新しいテーブルを出力します。これは **Ap_Trans** テーブルのレコードを月別にグループ化します。ただし、月名が含まれます。

Month	Invoice Amount	Count	Month_2
1	85,670.22	12	January
2	4,496.56	6	February
3	2,941.80	5	March
4	467.40	1	April
5	8,272.57	5	May
6	1,582.86	2	June

Month	Invoice Amount	Count	Month_2
7	3,101.98	4	July
8	21,146.96	2	August
9	32,577.32	20	September
10	41,595.89	19	October
11	70,779.26	19	November
12	6,008.51	7	December

演算フィールドの作成をスキップする

多くの場合、関数を複数の値に適用する方法として演算フィールドを作成することが有効です。ただし、Analytics で同じ結果を実現し、作業を合理化するには、Analytics コマンド内で直接関数を埋め込みます。

関数を埋め込んで月別にレコードをグループ化する

上記と同じ例を使用しますが、演算フィールドを作成しません。代わりに、関数を直接要約コマンドに埋め込みます。

1. **Ap_Trans** テーブルを開く。
2. メインメニューから、**分析 > 要約**]を選択します。

MONTH() 関数を埋め込む

1. **要約オン**]をクリックし、**Expr**]をクリックします。
2. **式ビルダー**で **関数**]リストの "MONTH(date/datetime)" をダブルクリックします。

ヒント

MONTH() 関数を見つけやすくするには、**関数**リストの上部のドロップダウン フィルダから **日時**]を選択します。

3. **式**]テキストボックスで日付時刻を選択し、**使用可能なフィールド** リストで **Invoice_Date** をダブルクリックします。

式]テキストボックスには `MONTH(Invoice_Date)` があります。

メモ

式はよく知っている式のように見えます。前の例の演算フィールドと同じです。要約コマンドに埋め込まれているだけです。

4. **OK**]をクリックして、**式ビルダー**を終了し、**OK**]をクリックして、**フィールドの選択**]ダイアログボックスを終了します。

CMOY() 関数を埋め込む

1. **他のフィールド**]をクリックし、**Expr**]をクリックします。
2. **式ビルダー**で関数リストの**CMOY(date/datetime , length)**をダブルクリックします。
3. **式**]テキストボックスで、"日付/日付時刻"を"Invoice_Date"に、"長さ"を"9"に置き換えます。
式]テキストボックスに**CMOY(Invoice_Date , 9)**が表示されるはずです。
4. **OK**]をクリックして、**式ビルダー**を終了し、**OK**]をクリックして、**フィールドの選択**]ダイアログボックスを終了します。

要約処理の確定

1. **小計フィールド**]リストで**Invoice_Amount**を選択します。
2. **出力**]タブをクリックし、**ファイル**]を選択して、**Name**フィールドに**Ap_Trans_grouped_3**と入力して、**OK**]をクリックします。

結果: Analytics は新しいテーブルを出力します。これは**Ap_Trans**テーブルのレコードを月別にグループ化します。2つの埋め込まれた関数が表示されます。

MONTH(Invoice_Date)	Invoice Amount	Count	CMOY(Invoice_Date, 9)
1	85,670.22	12	January
2	4,496.56	6	February
3	2,941.80	5	March
4	467.40	1	April
5	8,272.57	5	May
6	1,582.86	2	June
7	3,101.98	4	July
8	21,146.96	2	August
9	32,577.32	20	September
10	41,595.89	19	October

MONTH(Invoice_Date)	Invoice Amount	Count	CMOY(Invoice_Date, 9)
11	70,779.26	19	November
12	6,008.51	7	December

要点

2つの異なる方法を使用して、まったく同じ結果を実現します。

- **演算フィールド** - コマンドでフィールドを使用する前に演算フィールドを作成することは、文字通り段階的なアプローチです。複数の目的で演算フィールドを使用する場合は、適切なアプローチであることがあります。
- **埋め込み関数** - 演算フィールドの作成を回避し、コマンドに関数を埋め込むことは、より合理化されたアプローチです。スクリプトのコンテキストでは適切なアプローチになることがあり、より効率的なスクリプトを生成します。

提案された活動：曜日別にレコードをグループ化する

関数の使用をさらに練習し、演算フィールドを作成するか、コマンドに関数を埋め込む場合は、上記の活動の一部またはすべてをやり直し、DOW() および CDOW() 関数を代替します。

DOW() および CDOW() は MONTH() および CMOY() に非常に似ています。ただし、年の月ではなく、日付の曜日を抽出します。

ヒント

DOW() および CDOW() を使用すると、さまざまな曜日で売上値がどのように変化するかを分析できます。

関数のテストのヒント

コマンドラインで `DISPLAY` 方法を使用して、演算フィールドを作成するときまたはコマンドに関数を埋め込むときの出力の概要を確認できます。

1. **Ap_Trans** テーブルが開いている状態で、このバージョンの UPPER() 関数をコマンドラインに貼り付け、貼り付けられた関数の前に `DISPLAY` とスペースを入力して、Enter を押します。

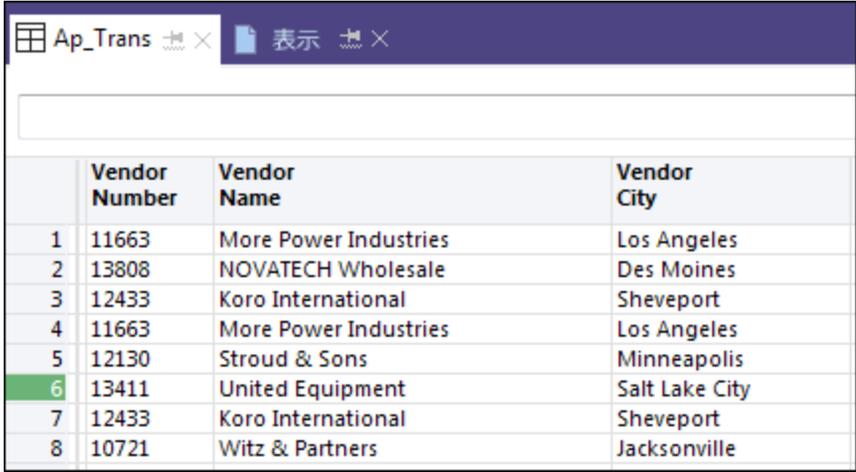
```
UPPER(Vendor.Vendor_Name)
```

UPPER() 関数はすべての入力テキストを大文字に変換します。

Ap_Trans テーブルの最初のレコードが選択される場合、関数出力は MORE POWER INDUSTRIES です。

2. テーブルで、レコード番号 6 を選択します。

レコードを番号をクリックし、レコードを選択します。選択されているレコードの番号が緑色で強調表示されます。



	Vendor Number	Vendor Name	Vendor City
1	11663	More Power Industries	Los Angeles
2	13808	NOVATECH Wholesale	Des Moines
3	12433	Koro International	Sheveport
4	11663	More Power Industries	Los Angeles
5	12130	Stroud & Sons	Minneapolis
6	13411	United Equipment	Salt Lake City
7	12433	Koro International	Sheveport
8	10721	Witz & Partners	Jacksonville

3. 関数をコマンドラインに入力するか再読み込みし、Enter を押します。

関数出力は UNITED EQUIPMENT です。

4. 1 つまたは 2 つの他のレコードを選択し、処理を繰り返します。

要点:レコード単位で、演算フィールドまたは埋め込み関数が Vendor.Vendor_Name フィールドのすべての値に対して実行する内容が表示されます。

入力用のフィールドを取得する Analytics 関数でこのテスト方法を使用できます。

次の学習ステップ

関数と変数を使用して対話機能を作成する方法を学習する: "関数で変数を使用してユーザー入力を許可する" 次のページ

関数で変数を使用してユーザー入力を許可する

これまで、関数の入力としてフィールドリテラル値を使用する方法を学習しました。このアプローチは Analytics で比較的簡単で、必要なすべてのことです。

Analytics スクリプトでは、フィールドやリテラル値ではなく、**変数**が一般的に関数の入力として使用されません。

このチュートリアルでは、関数で変数を使用します。このチュートリアルの最後には、変数を入力として使用する関数を含む簡単なスクリプトがあります。変数により、ユーザーは実際の入力値をインタラクティブに指定できます。

変数の概要と有用な理由

コンピューターのメモリの名前付きコンテナとして変数考えることができます。変数を作成するときには名前を指定します。変数は一時的またはより永久的に、ユーザーが選択または指定する値を格納できます。

変数には2つの非常に有用な質があります。

- **柔軟性** - 変数はスクリプトをより柔軟にします。
たとえば、特定のフィールド名または日付セットを要求する代わりに、スクリプトは変数を使用して、ユーザーに任意のフィールド名または日付を選択または指定させることができます。
- **明瞭性** - 変数は、確認または更新するときに、スクリプトをわかりやすくします。
未加工データよりも関数入力としてわかりやすい名前の変数を使用した方がわかりやすくなります。以下の2つの BETWEEN() 関数の例では、違いがわかります。

詳細については、"ACLScript での変数の操作" ページ 1554を参照してください。

BETWEEN() (変数を使用しない)

前のチュートリアルでフィルターを作成するために使用した BETWEEN() の例を考えます。

```
BETWEEN(Invoice_Date, `20000101`, `20000331`)
```

フィルターは Invoice_Date フィールドのレコードを2000年の最初の四半期に制限します。

別のコンテキストで使用するたびに、手動でフィールド名と開始日および終了日を変更する場合は、このバージョンの BETWEEN() 関数でも問題はありませぬ。

しかし、他のデータで他のユーザーによって実行されるスクリプトにこのフィルターを含め、他のユーザーが関数入力の更新方法がわからないとしたら、どうでしょうか。

また、視覚的な検査にのみ基づいて、関数入力を提供する未加工データの目的を知る方法はありません。

BETWEEN() (変数を使用)

実際のフィールドとリテラル値を BETWEEN() 関数の入力として使用する代わりに、変数を指定できます。

```
BETWEEN(v_date_field, v_start_date, v_end_date)
```

インタラクティブなスクリプトと組み合わせ、このバージョンの BETWEEN() 関数では、ユーザーは任意の日付を選択し、2つの境界日付を指定できます。

また、関数を見るだけで、各入力の目的が明確です。

メモ

規約として、スクリプト作成者は変数名の前に "v_" を付け、複雑なスクリプトでも何が変数で何が変数でないのかをわかりやすくします。

要点

変数を使用すると、幅広くより柔軟に関数を作成できます。

変数を使用する BETWEEN() 関数をテストする

BETWEEN() 関数を Analytics コマンドラインでテストし、変数の仕組みを正確に確認できます。

コマンドラインでの関数のテストの詳細については、前のチュートリアル("さまざまな関数の詳細" ページ 1483) を参照してください。

変数の作成

1. Analytics で、**ナビゲーター**の下部にある **変数** タブをクリックします。
変数 タブには、Analytics プロジェクトに現存するすべての変数と、各変数に現在設定されている値が表示されます。
2. コマンドラインで次の変数定義を1つずつ入力し、3つの変数例を作成します。
 - `v_date_field = `20170715``
この例では、実際のフィールドではなく、`v_date_field` 変数に単一のリテラル値を指定します。
 - `v_start_date = `20170701``

- `v_end_date = `20170731``

変数]タブでは、作成した3つの変数と割り当てられた値が表示されます。

がBETWEEN() のテスト

1. BETWEEN() の例をコピーし、コマンドラインに貼り付けます。

```
BETWEEN(v_date_field, v_start_date, v_end_date)
```

2. 例の前に `DISPLAY` とスペースを入力し、Enter を押します。

変数に含まれる値に基づき、結果は真の `T` です。

15 July 2017 は指定された開始日と終了日の間です。

変数値のいずれかを変更して結果を確認する

1. コマンドラインに次を入力して `v_start_date` の値を更新します。

```
v_start_date = `20170716`
```

変数]タブで、`v_start_date` の値が 16 July 2017 に更新されたことを確認できます。これは `v_date_field` 値の後です。

2. コマンドラインで BETWEEN() 関数を再実行します。

変数に含まれる値に基づき、結果は偽の `F` です。

15 July 2017 は指定された開始日と終了の間ではありません。

スクリプトで BETWEEN() 関数を試す

以下の簡易スクリプトでは、ユーザーは日付フィルターを日付フィールドがある Analytics テーブルに適用できます。

すべてのスクリプト構文を理解する必要はありません。要点は、スクリプトで実行中のさまざまな BETWEEN() 関数を確認することです。

スクリプト構文の前の `COMMENT` は構文の処理を簡単に説明しています。関数は茶色でハイライトされません。

Analytics でサンプルスクリプトを実行する方法

1. 日付フィールドがある1つ以上のテーブルを含む Analytics プロジェクトを開きます。

Sample Project.ACL には日付フィールドがある複数のテーブルがあります。

2. 新しい空のスクリプトを作成します。

- a. ナビゲーターでフォルダーまたは最上位のプロジェクト エントリをクリックし、**新規 > スクリプト**] を選択します。

- b. 以下のスクリプト全体をコピーして、スクリプト エディターの新しいスクリプトに貼り付けます。
- c. プロジェクトを保存します。
3. **実行**  をクリックして、スクリプトを実行します。
4. ダイアログボックス プロンプトに従い、テーブルと日付フィールドを選択し、開始日と終了日を指定します。

スクリプトが実行され、指定したフィールドと日付に基づいて、選択したテーブルがフィルタリングされます。

ヒント

空のテーブルまたはレコードが多いテーブルがある場合は、フィルタリングされていないテーブルで日付を確認し、レコード数が少ないことがわかっている境界日付でスクリプトを再実行します。

注記

- フィルタリングされたテーブルでは、BETWEEN() 関数は、指定した実際の入力値のフィルターテキストボックスに表示されます。
- **変数** タブを確認します。3つの例の変数の値は、スクリプトの実行時に選択および指定した値で更新されます。

スクリプト例：レコードを日付でフィルタリング

スクリプト例は、指定した日付を使用して、テーブルのレコードを日付でフィルタリングする

メモ

CTOD() 関数が BETWEEN() 関数内でネストされています。CTOD() 関数は文字値を日付値に変換します。これは、この状況が必要です。

詳細については、「ACCEPT コマンド」ページ 1625を参照してください。

```
COMMENT
この簡易スクリプトでは、日付フィールドがあるアナリティクス テーブルに日付フィルターを適用できません。
END

COMMENT Analytics プロジェクトでテーブルを選択する必要があります。
ACCEPT "日付フィールドがあるテーブルを選択してください:" FIELDS "xf" TO v_table_name

COMMENT 選択したテーブルを開きます。
OPEN %v_table_name%

COMMENT テーブルからフィールドを選択します。
ACCEPT "日付フィールドを選択してください:" FIELDS "D" TO v_date_field
```

```
COMMENT フィルターの開始日と終了日を指定します。  
ACCEPT "開始日を指定してください(YYYYMMDD):" TO v_start_date, "終了日を指定してくだ  
さい (YYYYMMDD):" TO v_end_date  
  
COMMENT 選択したテーブルとフィールドにフィルターを適用します。  
SET FILTER TO BETWEEN(%v_date_field%, CTOD(%v_start_date%), CTOD(%v_end_  
date%))
```

次の学習ステップ

複数の関数を使用して実際のタスクを実行するスクリプトをレビューして実行する: "すべてまとめる:スクリプトで関数を使用する" 見開きページ

すべてまとめる: スクリプトで関数を使用する

最後の Analytics 関数チュートリアルでは、変数と実際のタスクを実行するスクリプト内の多数の関数を使用して、すべてまとめます。

メモ

このチュートリアルを実行するには、スクリプトの知識は不要です。チュートリアルの下で作成済みスクリプトをコピーして、Analytics に貼り付けます。

スクリプトの処理

スクリプトの例では、スクリプトを実行して、日付フィールドの Analytics テーブルに日付フィルターを適用し、フィルターに含まれるレコードを月別にグループ化します。

スクリプトは、前の関数チュートリアルで説明した処理を組み合わせます。

関数がスクリプトに関連付けられる方法

1つのスクリプトでは、Analytics スクリプト作成者は、複数の関数を使用して、さまざまな小さく重要なヘルパータスクを実行し、スクリプトによる全体的なデータ分析を支援します。

スクリプトに関数を含めても、関数の動作は変わりません。スクリプトの関数は、Analytics コマンドラインで個別に関数をテストするときとまったく同じ方法で動作します。

提案された活動

- スクリプトのレビュー

チュートリアルの下のスクリプト例をレビューします。Analytics スクリプトは、各行が上から1行ずつ実行されます。このため、スクリプトを順番に下に実行し、各 `COMMENT` を読み取って、スクリプトロジックの実行内容を把握できます。

`COMMENT` 行はスクリプトロジックの一部ではなく、実行されません。

- 関数の処理を理解する

スクリプトに含まれる関数には特に注意します。関数は茶色でハイライトされます。各関数で実行される小さいタスクの詳細については、スクリプトの上のテーブルを参照してください。

前の関数チュートリアルを完了した場合は、スクリプトと実行するタスクのほとんどの関数を既に理解しています。

- スクリプトを実行する

スクリプトと含まれる関数を理解したら、スクリプトをコピーして、Analytics に貼り付け、実行してスクリプトがインタラクティブに動作する方法を確認します。

Analytics でサンプルスクリプトを実行する方法

1. 日付フィールドがある 1 つ以上のテーブルを含む Analytics プロジェクトを開きます。
Sample Project.ACL には日付フィールドがある複数のテーブルがあります。
2. 新しい空のスクリプトを作成します。
 - a. ナビゲーターでフォルダーまたは最上位のプロジェクト エントリをクリックし、**新規 > スクリプト**] を選択します。
 - b. 以下のスクリプト全体をコピーして、スクリプト エディターの新しいスクリプトに貼り付けます。
 - c. プロジェクトを保存します。
3. **実行**  をクリックして、スクリプトを実行します。
4. ダイアログボックス プロンプトに従い、テーブルと日付フィールドを選択し、開始日と終了日を指定して、数値小計フィールドを選択します。

スクリプト例：レコードのフィルタリングとグループ化

スクリプト例には 2 つの主要なことがあります。

- 指定した日付を使用して、テーブルのレコードを日付でフィルタリングする
- フィルターに含まれるレコードを月別にグループ化する

すべてのスクリプト構文を理解する必要はありません。要点は、スクリプトでさまざまな Analytics 関数が実行される様子を確認することです。

スクリプト構文の前の `COMMENT` は構文の処理を簡単に説明しています。

スクリプト例で使用される関数

スクリプト例で使用される各関数の目的は以下のとおりです。

スクリプトでは、関数は茶色でハイライトされます。

スクリプトの関数	目的
DATE()	<code>MIN1</code> および <code>MAX1</code> 変数を日付時刻から文字データ型に変換します。ダイアログボックスでテキスト文字列

スクリプトの関数	目的
	<p>の変数の内容を表示するには、文字データ型が必要です。</p> <p><code>MIN1</code> および <code>MAX1</code> は <code>STATISTICS</code> コマンドによって自動的に作成されたシステム変数です。選択した日付フィールドの最も古い日付と最も新しい日付が含まれます。</p>
<code>ALLTRIM()</code>	ダイアログボックスで表示される最も古い日付と最も新しい日付の余分なスペースを取り除きます。
<code>CTOD()</code>	<code>v_start_date</code> および <code>v_end_date</code> 変数を文字データ型から日付データ型に変換します。日付時刻データ型は、データの抽出または比較が必要です。
<code>CTOD()</code>	<code>v_start_date</code> および <code>v_end_date</code> 変数を文字データ型から日付データ型に変換し、 <code>v_date_field</code> 変数と一致するようにします。すべての <code>BETWEEN</code> 関数パラメーターは同じデータ型である必要があります。
<code>BETWEEN()</code>	指定した開始日と終了日に基づいて日付フィールドをフィルタリングします。
<code>MONTH()</code>	日付フィールドのすべての日付から数値として月を抽出します。
<code>CMOY()</code>	日付フィールドのすべての日付から文字値として月を抽出します。

スクリプト例：レコードを日付でフィルタリングし、フィルタリングされたレコードを月別にグループ化する

```

COMMENT
このスクリプトでは、日付フィルターを日付フィールドがあるアナリティクス テーブルに適用し、フィルターに
含まれるレコードを月別にグループ化します。
END

COMMENT Analytics プロジェクトでテーブルを選択する必要があります。
ACCEPT "日付フィールドがあるテーブルを選択してください:" FIELDS "xf" TO v_table_name

COMMENT 選択したテーブルを開きます。
OPEN %v_table_name%

COMMENT テーブルから日付フィールドを選択します。
ACCEPT "日付フィールドを選択してください:" FIELDS "D" TO v_date_field

COMMENT 選択した日付フィールドで最も古い日付と最も新しい日付を特定します。
STATISTICS ON %v_date_field%

COMMENT 最も古い日付と最も新しい日付を変数に割り当てます。変数は、日付フィルターの開始
日と終了日を指定するダイアログボックスで既存の日付範囲を表示するために使用されます。既存
の日付範囲がわかっている場合は、フィルター日付を指定する方が容易です。
ASSIGN v_min_date = ALLTRIM( DATE( MIN1, "YYYYMMDD" ) )
ASSIGN v_max_date = ALLTRIM( DATE( MAX1, "YYYYMMDD" ) )

```

```

COMMENT 日付フィルターの開始日と終了日を指定します。
DIALOG (DIALOG TITLE "User Dialog" WIDTH 484 HEIGHT 153 ) (BUTTONSET TITLE
"&OK;&Cancel" AT 370 12 DEFAULT 1 ) (TEXT TITLE "Specify a start date:" AT
12 16 ) (EDIT TO "v_start_date" AT 156 12 DEFAULT "YYYYMMDD" ) (TEXT TITLE
"Specify an end date:" AT 12 52 ) (EDIT TO "v_end_date" AT 156 48 DEFAULT
"YYYYMMDD" ) (TEXT TITLE "Date range in table:" AT 12 88 ) (TEXT TITLE "%v_
min_date%   to   %v_max_date%" AT 156 88 )

COMMENT ユーザー指定日付フィルターが 1 年を超えてまたがる場合に警告を表示します。
IF CTOD(v_end_date) - CTOD(v_start_date) > 365 OR CTOD(v_start_date) - CTOD
(v_end_date) > 365 DIALOG (DIALOG TITLE " Dialog" WIDTH 469 HEIGHT 100 )
(BUTTONSET TITLE "&OK;&Cancel" AT 348 8 DEFAULT 1 ) (TEXT TITLE "日付範囲が 1
年を超えています。月次グループに 1 年を超えるレコードが含まれている可能性があります。"AT
12 28 WIDTH 326 HEIGHT 33 ) (TEXT TITLE "Caution" AT 12 8 )

COMMENT ユーザーが指定した開始日が終了日の後の場合は警告を表示します。
IF CTOD(v_start_date) > CTOD(v_end_date) DIALOG (DIALOG TITLE "User Dialog"
WIDTH 469 HEIGHT 100 ) (BUTTONSET TITLE "&OK;&Cancel" AT 348 8 DEFAULT 1 )
(TEXT TITLE "開始日が終了日より後です2 つの日付間のレコードが含まれます。"AT 12 28
WIDTH 326 HEIGHT 33 ) (TEXT TITLE "Caution" AT 12 8 )

COMMENT 選択したテーブルと日付フィールドにフィルターを適用します。
SET FILTER TO BETWEEN(%v_date_field%, CTOD(%v_start_date%), CTOD(%v_end_
date%))

COMMENT 小計フィールドを選択します。
ACCEPT "各月で小計する数値フィールドを選択:" FIELDS "N" TO v_subtotal_field

COMMENT 月別にテーブルをグループ化し、結果を新しいテーブルに出力します。
SUMMARIZE ON MONTH(%v_date_field%) SUBTOTAL %v_subtotal_field% OTHER CMOY
(%v_date_field%, 9) TO "%v_table_name%_by_month.FIL" OPEN PRESORT

```

次の学習ステップ

"関数を使用する方法" ページ 1477と"高度な関数の使用" ページ 1499のすべてのチュートリアルを修了されたのなら、おめでとうございます!これで、あなたは Analytics における Analytics 関数の動作の基礎がみっちり理解できるようになりました。

関数の知識を高めるには、次のことをお勧めします。

- **探求を続ける**
 - 最も頻繁に使用される Analytics 関数と例の一覧については、"最もよく使用される 30 個の Analytics 関数" ページ 1517を確認してください。
 - "Analytics関数を使って検索とフィルタリングを行う" ページ 1235 Analytics 関数を使用して、テーブルのデータに対して、強力で効果的な検索とフィルタリングを実行する例を提供します。
 - Analytics "関数の概要" ページ 2139のセット全体を参照します。関数が実行できるあらゆる処理の概要を理解します。

- **関数を忘れない**

Analytics のデータ分析の課題に直面したときには、「1つの関数で解決できるか。複数の関数を組み合わせて解決できるか」を考えます。

Analytics コマンドを使用したデータ分析では、最も大きい課題は分析用のデータを準備することです。関数は単独でも組み合わせても、一般的に、準備において重要です。

最もよく使用される 30 個の Analytics 関数

ACLScript の上位 30 の関数は、さまざまなタスクで役に立ちます。これらの関数を定期的を使用し、スクリプトでデータを準備、解析、変換、および調整できます。

先頭と末尾のスペースを削除する

Analytics 関数 テーブルの文字フィールドは、一般的に、フィールド幅が固定長であるため、先頭または末尾にスペースがあります。文字フィールドからのデータを使用して演算を実行する必要があるときには、これらのスペースを削除し、文字列に実際のデータのみが含まれるようにします。

ALLTRIM()

入力文字列から先頭と末尾のスペースを除去した文字列を返します。

メモ

別の関数の入力として使用しているフィールドで `ALLTRIM()` を使用し、先頭または末尾のスペースが返される値に影響しないようにすることをお勧めします。

例

`Vendor_Number` フィールドには値 " 1254" があります。`Vendor_Number` からこの余分なスペースを削除し、フィールドを別のテーブルのデータと一致させる必要があります。

```
COMMENT "1254" を返します  
ALLTRIM(Vendor_Number)
```

英字の大文字と小文字を同期する

Analytics の文字列比較は大文字と小文字を区別するため、データを使用して比較、結合、または関係を実行する前に、フィールドのすべてのデータの大文字と小文字を同期することが有用です。

UPPER()

アルファベット文字を大文字に変換した文字列を返します。

例

Last_Name フィールドには値 "Smith" があります。この値を大文字にし、別のテーブルの大文字値と比較する必要があります。

```
COMMENT "SMITH" を返します  
UPPER>Last_Name)
```

LOWER()

アルファベット文字を小文字に変換した文字列を返します。

例

Last_Name フィールドには値 "Smith" があります。この値を小文字にし、別のテーブルの小文字値と比較する必要があります。

```
COMMENT "smith" を返します  
LOWER>Last_Name)
```

PROPER()

各単語の最初の文字を大文字に、残りの文字を小文字に設定した文字列を返します。

例

Last_Name フィールドには値 "smith" があります。出力に固有名詞として表示する必要があります。

```
COMMENT "Smith" を返します  
PROPER(Last_Name)
```

文字列の計算と分離

長い文字列からデータのセグメントを抽出するか、長さや内容などの文字列の情報をテストする必要があるときには、これらの関数を使用します。

SUBSTR()

文字列のうちの指定された部分文字列を返します。

例

`GL_Account_Code` フィールドには値 `"001-458-873-99"` があります。最初の3バイトまたは3文字を文字列から抽出する必要があります。

```
COMMENT "001" を返します  
ASSIGN v_start_pos = 1  
ASSIGN v_length = 3  
SUBSTR(GL_Account_Code, v_start_pos, v_length)
```

LAST()

文字列の末尾から指定された数の文字を返します。

例

`GL_Account_Code` フィールドには値 `"001-458-873-99"` があります。最初の2バイトまたは2文字を文字列から抽出する必要があります。

```
COMMENT "99" を返します  
ASSIGN v_num_chars = 2  
LAST(GL_Account_Code, v_num_chars)
```

SPLIT()

文字列のうちの指定された部分を返します。

例

GL_Account_Code フィールドには値 "001-458-873-99" があります。文字列からコードの2番目のセグメントを抽出する必要があります。

```
COMMENT "458" を返します
ASSIGN v_delimiter = "-"
ASSIGN v_segment_num = 2
SPLIT(GL_Account_Code, v_delimiter, v_segment_num)
```

AT()

文字値における部分文字列の特定の出現の開始位置を示す数値を返します。

例

GL_Account_Code フィールドには値 "001-458-873-99" があります。値 "458" の開始バイト位置を決定し、GLコードの2番目のセグメントが"458" (開始位置 "5") であるかどうかをテストする必要があります。

```
COMMENT "5" を返します
ASSIGN v_occurrence = 1
ASSIGN v_substring = "458"
AT(v_occurrence, v_substring, GL_Account_Code)
```

OCCURS()

部分文字列が指定された文字値内に現れる回数を数えて返します。

例

`GL_Account_Code` フィールドには値 `"001-458-873-99"` があります。データに3つのハイフン文字があることを確認し、GLコードが正しい形式であることを判定する必要があります。

```
COMMENT "3" を返します  
ASSIGN v_substring = "-"  
OCCURS(GL_Account_Code, v_substring)
```

LENGTH()

文字列に含まれている文字数を返します。

例

`GL_Account_Code` フィールドには値 `"001-458-873-99"` があります。データに14文字があることを確認し、GLコードが正しい形式であることを判定する必要があります。

```
COMMENT "14" を返します  
LENGTH(GL_Account_Code)
```

データ型の変換

Analytics テーブルを生成したデータソースとインポート文によっては、1つのデータ型から別のデータ型にフィールドの値を変換し、演算をできるようにする必要があります。たとえば、文字 (`"12345"`) としてインポートされたデータで演算を実行するには、数値に変換する必要があります。

STRING()

数値を文字列に変換します。

例

`Invoice_Amount` フィールドには値 `12345.67` があります。これを文字データに変換する必要があります。

```
COMMENT "12345.67"を返します  
ASSIGN v_str_length = 8  
STRING(Invoice_Amount, v_str_length)
```

VALUE()

文字列を数値に変換します。

ヒント

一般的に、VALUE() は ZONED() とともに使用され、先頭のゼロを追加します。

例

`Invoice_Amount` フィールドには値 `"12345.67"` があります。これを数値データに変換する必要があります。

```
COMMENT 12345.67を返します  
VALUE(Invoice_Amount, 2)
```

CTOD()

文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。

例

`Submission_Date` フィールドには値 `"April 25, 2016"` があります。これを日付時刻データに変換する必要があります。

```
COMMENT `20160425`を返します  
ASSIGN v_date_format = "mmm dd, yyyy"  
CTOD(Submission_Date, v_date_format)
```

DATE()

指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。

例

`Submission_Date` フィールドには値 `20160425` があります。これを文字データに変換する必要があります。

```
COMMENT "04/25/2016"を返します
ASSIGN v_date_format = "MM/DD/YYYY"
DATE(Submission_Date, v_date_format)
```

先頭のゼロの追加

先頭のゼロが必要なフィールドと一致させる必要があるときに、数値データを文字データに変換し、出力に先頭のゼロを追加します。

ZONED()

数値データを文字データに変換し、出力の先頭にゼロを追加します。

例

`Employee_Number` フィールドには値 `254879` があります。値を先頭ゼロ埋めの10桁の文字列に変換する必要があります。

ヒント

`VALUE()` 関数を使用して文字を数値データに変換してから、`ZONED()` への入力値として数値を使用します。

```
COMMENT "0000254879" を返します
ASSIGN v_str_length = 10
```

```
ASSIGN v_num_decimals = 0  
ZONED(VALUE(Employee_Number, v_num_decimals), v_str_length)
```

BINTOSTR()

ZONED または EBCDIC 文字データから変換された Unicode 文字データを返します。"Binary to String" の省略形です。

メモ

Unicode 版のみ。非 Unicode 版については、上記の `ZONED()` を参照してください。

例

`Employee_Number` フィールドには値 `"254879"` があります。値を先頭ゼロ埋めの 10 桁の文字列に変換する必要があります。

ヒント

`VALUE()` 関数を使用して文字を数値データに変換してから、`ZONED()` への入力値として数値を使用します。そこで `BINTOSTR()` を使用し、`ZONED()` から返された ASCII データを Unicode に変換します。

```
COMMENT "0000254879" を返します  
ASSIGN v_str_length = 10  
ASSIGN v_num_decimals = 0  
ASSIGN v_str_type = "A"  
BINTOSTR(ZONED(VALUE(Employee_Number, v_num_decimals), v_str_length),  
v_str_type)
```

日付時刻部分の抽出

これらの関数を使用して、日付時刻値の特定のコンポーネントを分離および抽出します。

MONTH()

指定された日付または日付時刻から月を抽出し、それを数値 (1 ~ 12) として返します。

例

`Transaction_Date` フィールドには値 ``20160815 100252`` があります。先頭にゼロがある文字データとして月を抽出する必要があります。

```
COMMENT "08" を返します
ASSIGN v_str_length = 8
ZONED(MONTH(Transaction_Date), v_str_length)
```

DAY()

指定された日付または日付時刻から日にちを抽出し、それを数値(1 ~ 31)として返します。

例

`Transaction_Date` フィールドには値 ``20160815 100252`` があります。文字データとして日を抽出する必要があります。

```
COMMENT "15" を返します
ASSIGN v_str_length = 8
STRING(DAY(Transaction_Date), v_str_length)
```

YEAR()

指定された日付または日付時刻から年を抽出し、それをYYYY書式の数値として返します。

例

`Transaction_Date` フィールドには値 ``20160815 100252`` があります。数値として年を抽出する必要があります。

```
COMMENT 2016を返します
YEAR(Transaction_Date)
```

HOUR()

指定された時刻または日付時刻から時間を抽出し、それを24時間制の数値として返します。

例

`Transaction_Date` フィールドには値 `20160815 100252` があります。数値として時間を抽出する必要があります。

```
COMMENT 10を返します  
HOUR(Transaction_Date)
```

MINUTE()

指定された時刻または日付時刻から分数を抽出し、それを数値として返します。

例

`Transaction_Date` フィールドには値 `20160815 100252` があります。数値として分を抽出する必要があります。

```
COMMENT 2を返します  
MINUTE(Transaction_Date)
```

SECOND()

指定された時刻または日付時刻から秒数を抽出し、それを数値として返します。

例

`Transaction_Date` フィールドには値 `20160815 100252` があります。数値として秒を抽出する必要があります。

```
COMMENT 52を返します  
SECOND(Transaction_Date)
```

CDOW()

指定された日付または日付時刻の曜日を返します。"Character Day of Week" の省略形です。

例

`Transaction_Date` フィールドには値 `20160815 100252` があります。文字データとして日の名前を抽出する必要があります。

```
COMMENT "Mon" を返します  
CDOW(Transaction_Date, 3)
```

CMOY()

指定された日付または日付時刻の月の名前を返します。"Character Month of Year" の省略形です。

例

`Transaction_Date` フィールドには値 `20160815 100252` があります。文字データとして月の名前を抽出する必要があります。

```
COMMENT "Aug" を返します  
CMOY(Transaction_Date, 3)
```

文字列の操作

これらの関数を使用して、文字フィールドのセグメントを削除または置換します。

INCLUDE()

指定した文字のみを含む文字列を返します。

例

`Address` フィールドには値 `"12345 ABC Corporation"` があります。住所番号を抽出し、会社の名前を除外する必要があります。

```
COMMENT "12345"を返します
ASSIGN v_chars_to_return = "0123456789"
INCLUDE(Address, v_chars_to_return)
```

EXCLUDE()

指定した文字を除外する文字列を返します。

例

`Address` フィールドには値 `"12345 ABC Corporation"` があります。会社名を抽出し、住所番号を除外する必要があります。

```
COMMENT "ABC Corporation" を返します
ASSIGN v_chars_to_exclude = "0123456789"
EXCLUDE(Address, v_chars_to_exclude)
```

REPLACE()

指定された文字列のすべてのインスタンスを新しい文字列で置き換えます。

例

`Address` フィールドには値 `"12345 Acme&Sons"` があります。"&"文字を単語 "and" で置換する必要があります。

```
COMMENT "12345 Acme and Sons"を返します  
ASSIGN v_target_char = "&"  
ASSIGN v_replacement_char = " and "  
REPLACE(Address, v_target_char, v_replacement_char)
```

OMIT()

指定した1つ以上の部分文字列が削除された文字列を返します。

例

`Address` フィールドには値 `"12345 Fake St"` があります。番地の接尾辞がない住所を抽出する必要があります。

```
COMMENT "12345 Fake"を返します  
ASSIGN v_chars_to_omit = "St"  
OMIT(Address, v_chars_to_omit)
```

REVERSE()

文字の順番を逆にした文字列を返します。

例

`Report_Line` フィールドには値 `"001 Correction 5874.39 CR "` があります。値を逆にし、先頭または末尾のスペースを省略する必要があります。

```
COMMENT "RC 93.4785 noitcerroC 100" を返します  
REVERSE(ALLTRIM(Report_Line))
```

BLANKS()

指定された数の空白スペースを含んでいる文字列を返します。

例

`region_code` フィールドの値に基づいて、領域名の演算フィールドを作成する必要があります。コマンドの最後に指定するデフォルト値が少なくとも最も長い入力値と同じ長さであることを確認する必要があります。

```
COMMENT BLANKS 8 " " 文字の文字列を返します
ASSIGN v_length = 8
DEFINE FIELD region COMPUTED

"Southern" IF region_code = 1
"Northern" IF region_code = 2
"Eastern" IF region_code = 3
"Western" IF region_code = 4
BLANKS(v_length)
```

このページは意図的に空白のままにしています

スクリプトの操作

Analytics スクリプトは、Analytics の一部であるスクリプト エディターで、プレーンテキストとして作成されます。すべてのプレーンテキストの内容のように、スクリプト エディターと他のプレーンテキスト ソースの間で自由にテキストをコピーして貼り付けることができます。

Analytics ユーザー インターフェイスのスクリプト

スクリプトは、ナビゲーターの **概要** タブで個別に視覚化されます。スクリプトは個別に視覚化されますが、Analytics プロジェクトのすべてのスクリプトは単一のプロジェクト ファイル (`*.acl`) 内に含まれます。

必要に応じて、スクリプトは、Analytics プロジェクトの外部に保存される個別の `.aclscript` ファイルとしてエクスポートすることができます。別ファイルとしてエクスポートされたスクリプトは、後で任意の Analytics プロジェクトにインポートできます。

以下のメイン Analytics ユーザー インターフェイスには、スクリプト エディターで最初のスクリプトが開いている状態で、ナビゲーターのさまざまなスクリプトが表示されます。

The screenshot displays the ACL DigiLink Travel Analytics application window. The title bar reads "ACL_DigiLink_Travel.ad - Analytics". The menu bar includes "ファイル(F)", "編集(E)", "インポート(I)", "データ(D)", "分析(A)", "機械学習(M)", "サンプリング(S)", and "アプリケーション(P)". The toolbar contains various icons for file operations and data analysis.

The left sidebar, titled "ナビゲーター", shows a tree view of the project structure:

- ACL_DigiLink_Travel.ad
 - ACL_DigiLink_Travel
 - Scripts
 - Delete_Temp_Tables** (selected)
 - Delete_Temp_Tables_A00
 - IMPORT_Travel_Data
 - PREP_Travel_Data
 - TNE01_Keyword_Search
 - TNE02_High_Risk_Exp_Category
 - TNE03_Outlier_Exp_Category

The main window, titled "Delete_Temp_Tables", shows a command line editor with the following script content:

```

1  COMMENT
2  *****
3  ScriptHub ID: Delete_Temp_Tables
4  Deletes all table layouts and linked FIL files with names
5  starting with T_.
6  *
7  LEGAL: These Scripts are provided "as is" and ACL does not
8  warrant that these
9  Scripts are free from errors. ACL does not provide Support
10 for Scripts, however,
11 assistance is provided through the ACL Support user forum.
12 By using these
13 Scripts you are agreeing to the ACL Script License
14 Agreement, the full document
15 can be found here: http://www.acl.com/legal
16 *****
17 END
18
19 COMMENT *** Set preferences to delete fil with table layout
20 SET DELETE_FILE ON
21 SET SAFETY OFF
22
23 COMMENT *** Obtain list of temporary files in the project
24 directory
25 CLOSE PRIMARY SECONDARY
26 DIRECTORY 'T_*.fil' TO File_List SUPPRESS
27
28 COMMENT *** Delete each table in the list
29 v_count_max = %WRITE1%
30 v_count = 1
31 OPEN File_List
32 DO Delete_Temp_Tables_A00 WHILE v_count <= v_count_max
33
34 COMMENT *** Clean up our own files
35 CLOSE PRIMARY SECONDARY
36 DELETE FORMAT File_List OK
37
38 COMMENT *** Return preferences to not delete data file with
39 table layout
40 SET DELETE_FILE OFF
41
42

```

At the bottom left of the editor, there are buttons for "総覧", "ログ", and "変数".

スクリプト記述ツール

スクリプトはさまざまな方法で作成、編集、またはデバッグすることができます。一部のツールでは、事前に構文を知らなくても、自動的にACLスクリプト構文を作成できます。

上記のツールを単独または組み合わせて使用すれば、新しいスクリプトを作成したり既存のスクリプトを編集したりすることができます。

ツール	説明
スクリプト エディター	ACLScript 構文を入力して、スクリプトを作成または編集します。
コマンド ログ	コマンド ログのエントリを選択して、自動的に ACL Script を作成します。プロジェクトで実行された全てのコマンドのレコードは保持されます。
スクリプト レコーダー	スクリプト記録が有効なときに、実行する ACLScript コマンドを記録して、ACLScript 構文を自動的に作成します。
構文キャプチャ	構文キャプチャが有効なときに、Analytics ユーザー インターフェイスでアクセスする ACLScript コマンドを記録して、ACLScript 構文を自動的に作成します。コマンドは実際に実行されません。
テーブルの履歴	ACL スクリプト コマンドまたは一連のコマンドの出力として作成された Analytics テーブルのテーブル履歴から自動的に ACLスクリプト 構文を作成します。
ScriptHub	Analytics スクリプト 項目の Web ライブラリである ScriptHub からスクリプトまたはスニペットをインポートします。
デバッグ機能	スクリプトをテストまたはデバッグするために、ブレークポイントを設定したり、スクリプトを一度に 1 行ずつステップ実行したりします。

スクリプトの作成と編集

スクリプトの作成にはいくつかのオプションがあります:

- **スクリプト エディタ**-スクリプト エディタ ウィンドウのスクリプト構文を入力する
- **コマンド ログ**-コマンド ログからスクリプト 構文をコピーして貼り付ける
- **スクリプト レコーダー**-実行時にコマンドの構文を記録する
- **構文キャプチャ**-コマンドを実行せずユーザー インターフェイスで選択する場合にコマンドの構文を挿入する
- **テーブル履歴**-出力テーブル履歴からスクリプトの構文をコピーする

これらのオプションは組み合わせで利用できます。たとえば、コマンド ログから構文をコピーして、**スクリプト エディター** ウィンドウに直接構文を追加してスクリプトを開始することができます。

Analytics 内の既存のスクリプトを編集するには、**スクリプト エディター**を使用する必要があります。

また、選択したテキスト エディターを使用して、Analytics スクリプトに構文をコピーして貼り付けることもできます。

構文の自動補完

スクリプト エディターで構文を入力すると、Analytics は、ACL スクリプト のコマンドとキーワードの自動補完、および関数パラメータの自動オンスクリーンヘルプを提供します。

オートコンプリートをオフにするには、**オプション** ダイアログ ボックスの **インターフェイス** タブ(**ツール > オプション**) で **スクリプトのオートコンプリートを無効にする**を選択します。関数パラメータのオンスクリーンヘルプを無効にすることはできません。

ScriptHub からスクリプト やスニペット をインポートする

スクリプトを作成するための別のオプションは、ScriptHubのコンテンツを利用することです。ScriptHub は、Diligent の社員と Diligent ユーザー コミュニティによって開発された Analytics スクリプト 項目の Web ライブラリです。サブスクリプションの一部として、ScriptHub のコンテンツをダウンロードおよび使用できます。

詳細については、"ScriptHub からのインポート" ページ 1606を参照してください。

スクリプト エディター でスクリプト を作成または編集する

必要な ACLScript 構文を直接**スクリプト エディター**に入力するか、あるいは既存のスクリプトに対してはその他の方法のいずれかを使用することで、スクリプトを作成または編集することができます。

メモ

スクリプトを作成または編集する場合、ACLスクリプトの各コマンドは必ず別々の行に入力してください。

方法の説明

スクリプトの作成または編集

1. **スクリプト エディター**でスクリプトを開くには、次の手順のいずれかを実行します。

- スクリプトを新規作成するには、**[ファイル > 新規作成 > スクリプト]**を選択します。
スクリプトは名前 **新しいスクリプト** で作成されます。名前を右クリックし、**名前の変更** をクリックすると、スクリプトの名前を変更します。

メモ

スクリプトの名前は、64文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- 既存のスクリプトを開くには、**ナビゲーターの [総覧]** タブでそのスクリプトをダブルクリックします。
2. **スクリプト エディター**で、ACLScript 構文を追加、変更、または削除します。

ヒント

一般的な操作には以下のショートカットキーを使用できます:

- Ctrl+Z** - キーを押すと、1段階以上の操作を元に戻します。
- Ctrl+Y** - キーを押すと、1段階以上の操作をやり直します。
- Ctrl+S** - キーを押すと、開いているスクリプトも含め、Analytics プロジェクトを保存します。

3. (オプション) スクリプト内の適切な場所にカーソルを置き、次のいずれかの手順を実行して、1つまたは複数の特定の項目を挿入します:

項目	手順
プロジェクト項目名 (テーブル、スクリプト、ビュー、ワークスペース、またはインデックス)	a. 挿入を右クリックして、 プロジェクト項目 をクリックします。 b. 項目のタイプ ドロップダウンリストから、項目の種類を選択します。 c. 1つ以上の項目名を選択し、 [OK] をクリックします。
フィールド名	a. スクリプト エディター のツールバーで、 [フィールドの挿入] をクリックします  。 b. 1つ以上のフィールド名を選択して [OK] をクリックします。
式	

項目	手順
	<p>a. スクリプト エディターのツールバーで、[式の挿入]をクリックします .</p> <p>b. 式を作成して OK]をクリックします。</p>
ダイアログボックス	<p>a. スクリプト エディターのツールバーで、[ダイアログの新規作成]をクリックします .</p> <p>b. カスタム ダイアログ ボックスを作成し、閉じる]、OK]の順にクリックします。 詳細については、"カスタム ダイアログ ボックスの作成" ページ 1588を参照してください。</p>
日付/時刻	<p>a. 挿入 > 日付時刻 を右クリックします。</p> <p>b. 日付、日付時刻、または時刻を入力または選択して OK]をクリックします。</p>
HighBondトークン	<p>a. 右クリックして 挿入 > HighBond トークン]を選択すると、HighBond アクセストークンがスクリプトに挿入されます。 API アクセストークンを管理します]ページがお使いのブラウザで開きます。まず Launchpad にサインインするように求められる場合があります。</p> <p>b. 次のいずれかを実行します。</p> <ul style="list-style-type: none"> • 既存のトークンを使用する - トークン]列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。 • 新しいトークンを作成する - トークンを作成 > HighBond API]をクリックし、お使いのHighBond アカウントのパスワードを入力します。 新しいHighBondトークンが作成されます。 <p>ヒント 新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。 既存のトークンを使用すると、管理する必要があるトークン数が減ります。</p> <p>c. コピー]をクリックしてトークンをコピーします。</p> <p>ヒント トークンが含まれているこのダイアログ ボックスは、そのトークンをスクリプトに貼り付けるまで閉じないでください。</p> <p>d. Analytics で、スクリプト内の適切なポイントに当該のトークンを貼り付けます。</p> <p>e. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。 新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。</p>

項目	手順
	<p>詳細については、アクセストークンの作成と管理を参照してください。</p> <p>注意 アカウント パスワードのようにアクセストークンを保護してください。 HighBond アカウント 固有の情報が含まれます。アクセストークンを共有しないでください。</p>

4. **[ファイル > プロジェクトを上書き保存]**の順にクリックします。
5. 確認ダイアログボックスで、**[はい]**をクリックします。

ダイアログボックスを使用してコマンド構文を編集する

スクリプト内のコマンドを手作業で編集するのではなく、関連するダイアログボックスを使用して編集することができます。

メモ

この方法は、ダイアログボックスを持つコマンドに対してのみ使用できます。

1. スクリプトで既存のACLスクリプトコマンドを選択します。
2. **スクリプト エディター**のツールバーで、**[コマンドの編集]**をクリックします 。
ダイアログボックスが表示されます。
3. ダイアログボックスでコマンド パラメータに必要な変更を行い、**[OK]**をクリックします。
スクリプト構文が更新されます。

スクリプトをテストする

スクリプトをテストするために実行またはステップ実行してみる場合は、**スクリプト エディター**ツールバーにある **実行**  または **ステップ**  をクリックします。

メモ

スクリプトを実行またはステップ実行している場合は、開いているすべてのスクリプトが自動的に保存されます。

コマンド ログからスクリプトを作成する

ナビゲーターの **ログ** タブにあるログ エントリを、スクリプトを新規作成するための出発点として、または既存のスクリプトに追加するためにコピーすることができます。Analytics で以前に実行された各コマンドの構文は個別のログ エントリに記録されます。

次のタイプのログ エントリを選択し、新しい、または既存のスクリプトにコピーします。

- 個別のエントリ
- テーブルに関連付けられた一連のエントリ
- セッションまたは期間に関連付けられたエントリのより大きいグループ

方法の説明

1. ナビゲーターで **ログ** タブをクリックしてログを表示します。
2. スクリプトに含めるログ エントリまたはエントリのグループを選択します。
ツリービュー内の上位レベルの項目を選択すると、サブエントリも自動的に選択されます。
3. 次のうちの1つを行います:

<p>新しいスクリプトを作成する</p>	<ol style="list-style-type: none"> a. ログ タブで右クリックして 選択項目の保存 > スクリプト]を選択します。 b. スクリプトを別名で保存]ダイアログ ボックスで、新しいスクリプトの名前を入力して OK]をクリックします。 <p>メモ スクリプトの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <p>新しいスクリプトがナビゲーターの 概要]タブに追加されます。スクリプトはアクティブな Analytics テーブルを含むフォルダー、またはテーブルが開いていない場合はルートプロジェクトに保存されます。</p>
<p>既存のスクリプトに構文をコピーする</p>	<ol style="list-style-type: none"> a. ログ]タブで右クリックして コピー]を選択します。 b. 既存のスクリプトがまだ開いていない場合は、そのスクリプトを開きます。 c. スクリプト エディター で、コピーした構文を挿入する位置にカーソルを置きます。 d. 貼り付け]を右クリックし、選択します。

スクリプト レコーダーを使用してスクリプトを作成する

Analytics **スクリプト レコーダー** では、Analytics ユーザー インターフェイスでテーブルに対して行った操作やコマンドを記録することによって、スクリプトを作成できます。**スクリプト レコーダー** を使用してスクリプトを作成する利点は、使用する ACLScript コマンドに対して構文を入力することや構文の知識が必要ないという点です。

スクリプト レコーダー が記録するのはコマンドだけです。原則として、コマンド ログに表示されるコマンドは **スクリプト レコーダー** に記録できます。

ヒント

スクリプト レコーダーは ACLScript を学ぶための有用なツールでもあります。スクリプト レコーダーを使用して、実行した一連の分析手順をスクリプトに記録しておけば、そのスクリプトを見ることで、実行した分析を再現するために必要な一連のコマンドおよび構文を確認することができます。

方法の説明

1. Analytics のメインメニューから、**ツール > スクリプト レコーダーをオンに設定**] を選択します。
スクリプト レコーダーアイコン  がオンであることを示すため、ステータスバーにはスクリプト レコーダーアイコンが表示され、メニュー項目の左側にはチェックボックスが表示されます。
2. レコードの分析や処理を実行します。
Analyticsは処理した各コマンドを新しいスクリプトに記録します。
3. データに対する解析または処理が終了したら、再度、**ツール > スクリプト レコーダーをオンに設定する**] を選択してスクリプト レコーダーをオフにします。
スクリプトを保存するよう指示されます。
4. スクリプトの内容が特定しやすい名前をテキストボックスに入力して、**OK**] をクリックします。

メモ

スクリプトの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

構文のキャプチャを使用してスクリプトを作成する

構文のキャプチャを使用すると、Analytics メニューとダイアログボックスを使用して、Analytics スクリプト 構文をスクリプトに自動的に挿入できます。

方法の説明

1. 作業したいスクリプトを開きます。
2. (オプション) 作業したいテーブルを開きます。

ヒント

テーブルを開く前に構文のキャプチャを開始すると、構文のキャプチャ中にコマンドが実際に実行されないため、**[表示]** タブではテーブルが物理的に開きません。ガイダンスのために開いたテーブル無しで後続のコマンドを視覚化するのは難しいかもしれませ

3. スクリプト エディターのツールバーで、**[構文のキャプチャを開始する]** をクリックします .

- レコードの分析や処理を実行します。
関連する構文が自動的にスクリプトに挿入されます。コマンド自体は実行されません。
- 構文のキャプチャを終了**]  をクリックして、スクリプトへのコマンド構文の挿入を停止します。

テーブル履歴からスクリプトを作成する

Analytics 出力または結果または結果テーブルに関連付けられた履歴に基づいてスクリプトを作成できません。

たとえば、Analytics プロジェクトに月次棚卸テーブルを追加し、関連する部署と項目を新しいテーブルに抽出する場合は、作成した最初の出カテーブルに関連付けられたテーブル履歴に基づいてこのプロセスをスクリプト化できます。

Analytics では、Analytics コマンドの実行結果の出力として作成されたテーブルごとに、そのテーブルの作成に使用したコマンドの記録を保持しています。これには、元の ACL テーブルと出カテーブル間に存在した中間テーブルを作成したコマンドも含まれます。後続の出カテーブルの作成を自動化するために使用できる新しいスクリプトにこのテーブル履歴をコピーできます。

方法の説明

- スクリプトで自動化するプロセスの結果である出カテーブルを開きます。
- ツール > テーブル履歴からのスクリプト作成**] を選択します。
デフォルト **_ビュー**] (Default View) がアクティブな場合、新しいスクリプトを実行するときに上書きされないようにするため、Analytics ではビューの名前を変更するよう促します。
- ビューの名前の変更を促された場合は、**名前の変更**] をクリックして新しい名前を入力し、**OK**] をクリックします。
- スクリプトを別名で保存**] ダイアログボックスで、新しいスクリプトの名前を入力して **OK**] をクリックします。

メモ

スクリプトの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。

- (オプション) スクリプトの動作を調整する場合は、新しいスクリプトを開いて編集します。
たとえば、元のテーブルを上書きする代わりに、別の名前のテーブルに結果を出力できます。

スクリプトのテストおよびデバッグ

Analytics スクリプト エディターには、Analytics スクリプトのテストやデバッグに役立つ機能がいくつか備わっています:

- スクリプト内のカーソル位置からスクリプトを実行する
- ブレークポイントを設定して、特定の行でスクリプトの実行を一時停止する
- スクリプトを一度に1行ずつステップ実行する
- スクリプト エラーを分離する
- 関連付けられた **変数** タブで、変数の作成や、変数への値の割り当てをリアルタイムで追跡する

これらの機能は、スクリプトがスクリプト エディターで開いていればいつでも利用できます。開いているスクリプトが1つ以上のサブスクリプトを呼び出す場合、それらのサブスクリプトは呼び出されたときに自動的に開かれます。

スクリプトをステップモードまたはブレークポイントモードで実行している間、スクリプトは読み取り専用となり、Analytics の他の機能はコマンドラインも含め、ほとんどが無効になります。スクリプト エラーが発生した場合は、エラーを修復できるように、スクリプトは編集可能となります。

スクリプトを実行すると、その実行方法に関係なく、開いているすべてのスクリプトが自動的に保存されます。

メモ

実行またはステップ実行が完了した時点でテーブルが開いたままである場合は、Analytics の表示領域はスクリプト エディターから、ビュー タブで開いているテーブルの表示へ自動的に切り替わります。スクリプトをテストまたはデバッグしている間は、そのままスクリプト エディターが表示され続けるようにしたい場合は、スクリプトの最後に一時的に CLOSE コマンドを含めることができます。

カーソルからスクリプトを実行する

スクリプトの先頭からスクリプトの実行またはステップ実行をしたくない場合は、スクリプトの実行を開始したい行にカーソルを位置付けてから、右クリックして **カーソルから実行**] または **カーソルからステップ**] を選択します。カーソル位置からスクリプトを実行またはステップ実行すると、スクリプトの特定の部分をテストでき、スクリプト全体を不必要に実行することで時間と労力が無駄に費やされることを避けられます。

スクリプトがいったん実行されたら、**カーソルから実行**] と **カーソルからステップ**] は使用できません。これらのオプションは、スクリプトの実行を開始する場合と、スクリプト エラーが発生した後またはエラーを修正した後で、スクリプトを再開する場合にのみ使用できます。

メモ

カーソルからのスクリプトの実行を使用してスクリプトの一部を回避する場合、そのセクションに後続のセクションに必要な前提条件となる操作が含まれている場合は、後続のセクションが正しく実行される見込みはありません。

ブレークポイントを設定する

Analytics スクリプト内に1つ以上のブレークポイントを設定して、特定の行でスクリプトの実行を一時停止させることができます。ブレークポイントによって、スクリプト全体を実行する必要なく、スクリプトの一部をテストできるようになります。また、スクリプト内の特定の時点における Analytics プロジェクトの状態を調べることが可能になります。スクリプトのより複雑、より重要な部分を開発およびテストしているとき、ブレークポイントは有用なツールとなるでしょう。

ブレークポイントからスクリプトを再起動する

ブレークポイントからスクリプトを再起動する場合、以下のオプションがあります:

- ブレークポイントからスクリプトをステップ実行する
- スクリプトを挿入した場合は次のブレークポイントまでスクリプトを実行する
- スクリプトを最後まで実行する
- スクリプトを終了する

空白行とコメント

ブレークポイントを空白行またはコメント行に設置した場合、スクリプトはその空白行またはコメント行の後にある最初のスクリプト行で一時停止します。

ブレークポイントの持続性

- ブレークポイントは、そのスクリプトを閉じても、スクリプト内に保持されます。
- **スクリプト エディター**内で右クリックして **すべてのブレークポイントをクリア**を選択すると、Analytics プロジェクト内のすべてのスクリプトからすべてのブレークポイントをいつでも削除することができます。
- すべてのブレークポイントは、Analytics を閉じるときに、プロジェクト内のすべてのスクリプトから自動的に削除されます。

手順

方法の説明

1つ以上のブレークポイントを設定する

1. ブレークポイントを設定したい Analytics スクリプトを開きます .
2. Analytics スクリプト内のターゲット行の左側に接しているブレークポイント列をクリックします。
ブレークポイント列は、行番号の列とスクリプトの左余白の間に配置されます。

ブレークポイントは、スクリプト内のターゲット行にカーソルを置いて **F9** キーを押すか、またはスクリプト エディター ツールバーにある **ブレークポイントの設定/解除**  をクリックしても設定できません。

3. ブレークポイントを削除するには、そのブレークポイントをクリックするか、ターゲット行にカーソルを置いてから **F9** キーを押す、または **ブレークポイントの設定/解除**  をクリックします。

ブレークポイントでスクリプトを実行する

1. スクリプトをブレークポイントまで実行するには、**実行**  をクリックするか **F5** キーを押します。
スクリプトが実行を開始し、ブレークポイントまで実行されます。スクリプトをブレークポイントモードで実行している間、スクリプトは読み取り専用となり、Analytics の他の機能はコマンドラインも含め、ほとんどが無効になります。
2. ブレークポイントを越えて移動するには、**実行**  をクリックするか **F5** キーを押します。
次のブレークポイントまでスクリプトを実行するか、あるいは他のブレークポイントがない場合は、スクリプトの実行を完了します。
3. ステップ矢印が赤色  に変わり行で停止している場合は、エラーを示しており、スクリプトが編集可能になります。エラーを修正し、その後次のいずれかを行うことができます。
 - エラーの発生場所からスクリプトの実行を続行するか、または適切な行にカーソルを置き、右クリックして **カーソルから実行**] を選択することによって、別の行から実行を続行する。
実行  をクリックするか **F5** キーを押して、スクリプトを先頭から
 - 再実行する。エラーが発生したときにテーブルが開いた状態である場合は、Analytics の表示領域はスクリプトエディターから、ビュータブで開いているテーブルの表示へ自動的に切り替わります。スクリプトエディターに切り替えて、エラーを修正します。
4. スクリプトが完了する前にスクリプトを終了したい場合は、**Esc** キーを押し、確認プロンプトで **[はい]** をクリックします。
Analytics を閉じることによって、スクリプトを終了することもできます。
5. ブレークポイントの後、またはエラーを修正した後、残りのスクリプトをステップ実行したい場合は、次のいずれかを行います。
 - ブレークポイントの後には、**ステップ**  をクリックするか、**F10** キーを押します。
 - エラーを修正した後は、適切な行にカーソルを置き、右クリックして **カーソルから実行**] を選択します。

スクリプトをステップ実行する

一度に1行ずつステップ実行することで、Analytics スクリプトを実行できます。スクリプトを実行すると、制御された方法で実行をテストし、あらゆるエラーをそれが発生する正確な行で見つけることができます。

ステップ矢印

緑色の矢印 - スクリプトをステップ実行しているとき、緑色のステップ矢印  が実行されようとしているスクリプト内の行を示します。矢印が行を越えて1ステップ進んだら、その行は実行済みです。

赤色の矢印 - 行に無効なコマンド構文や何かほかの種類誤りがある場合、スクリプトは停止し、ステップ矢印は赤色に変わって進まなくなり、 誤っている場所が強調表示されます。スクリプトは、ステップ実行している間は読み取り専用になりますが、エラーが発生した場合は、エラーを修復できるように編集可能となります。

手順

方法の説明

1. ステップ実行したい Analytics スクリプトを開きます。
2. 繰り返し、**[ステップ]**  をクリックするか F10 キーを押します。**[ステップ]** をクリックするか F10 キーを押すと、
スクリプトが開始します。さらに **[ステップ]** をクリックするか F10 キーを押すごとに、単一行が順番に実行されます。
スクリプトをステップモードで実行している間、スクリプトは読み取り専用となり、Analytics の他の機能はコマンドラインも含め、ほとんどが無効になります。
3. ステップ矢印が赤色  に変わった場合は、エラーを示しており、スクリプトが編集可能になります。エラーを修正し、その後次のいずれかを行うことができます。
 - エラーの発生場所からスクリプトのステップ実行を続行するか、または適切な行にカーソルを置き、右クリックして **[カーソルからステップ]** を選択することによって、別の行からステップ実行を続行する。
 - **[ステップ]**  をクリックするか F10 キーを押して、
 - スクリプトを再実行し、先頭からステップ実行を開始する。エラーが発生したときにテーブルが開いた状態である場合は、Analytics の表示領域はスクリプトエディターから、ビュータブで開いているテーブルの表示へ自動的に切り替わります。スクリプトエディターに切り替えて、エラーを修正します。
4. スクリプトが完了する前にスクリプトを終了したい場合は、**Esc** キーを押し、確認プロンプトで **[はい]** をクリックします。
Analytics を閉じることによって、スクリプトを終了することもできます。

5. どの時点でも、ステップ実行しないで残りのスクリプトを実行したい場合は、**実行**  をクリックするか F5 キーを押します。

スクリプト エラーを分離する

スクリプトが失敗する原因となるエラーが発生するスクリプトを Analytics で実行するたびに、エラーが発生する行はスクリプト エディター内で自動的に強調表示されます。スクリプト エディターが開いていない場合は、自動的に開きます。このスクリプト エラーの識別は、スクリプトをスクリプト エディターで直接実行したか、[ツール]メニューから、コマンドラインから、あるいはナビゲーター内でスクリプトを右クリックして実行したかに関係なく発生します。

この自動エラー識別は強力なトラブルシューティング機能で、特に、深く入れ子になっているサブスクリプトで発生するエラーにとって有用です。スクリプトを記述できる Analytics ユーザーであれば、エラーに遭遇したときに、そのエラーを直接修正することができます。スクリプト記述に慣れていないユーザーは、エラーが発生したスクリプトの名前と行番号を記録しておけば、スクリプトの問題について助けを求めることが容易になります。

変数]タブを使用する

ナビゲーターの **変数]タブ**では、変数の作成や、変数への値の割り当てをリアルタイムで追跡します。読み取り専用タブには、Analytics プロジェクトのすべての変数の名前、データ型、現在の値が表示されます。名前はアルファベット順に表示されます。

変数	データ型	値
v_tax_1	N	0.05
v_tax_2	N	0.06
v_tax_2_end	D	2021/12/31
v_tax_2_start	D	2021/07/01

ナビゲーター 

総覧 ログ **変数** 

スクリプトを1行ずつ進むと、スクリプトで定義された変数またはシステムで生成された変数が作成時に **変数]タブ**に表示されます。変数がすでに存在する場合は、スクリプト ロジックに基づいて値が動的に更新されます。(スクリプト エディターで、**[ステップ** ] オプションを使用してスクリプトを進めます。)

スクリプト変数で発生している正確な変更を発生時に監視できるのは、重要な診断ツールです。この機能では、スクリプト構文の調査だけでは検出が難しいスクリプト エラーをピンポイントで特定できます。

スクリプトを実行する場合、変数に関連するすべての変更は、ブレークポイントに達したとき、またはスクリプトが完了したときに **変数** タブに表示されます。

複数行のコマンド

GROUP、LOOP、または DEFINE FIELD ... COMPUTED などの複数行コマンドの内容をステップ実行することはできません。スクリプトをステップモードで実行していて複数行コマンドに遭遇した場合は、そのコマンドの内容全体が実行され、ステップ矢印は、複数行コマンドの直後にある行に位置付けられます。

ブレークポイントは、複数行コマンドの内部では認識されません。複数行コマンドの内部にブレークポイントを設定した場合、スクリプトは、複数行コマンドの直後にある行で一時停止されます。

ヒント

複数行コマンドの内容部分は、囲んでいるコマンド構文なしで内容を別のスクリプトへコピーすることによって、テストできるかもしれません。

PASSWORD アナリティクス タグを含むアナリティクスのテスト

スクリプトのアナリティクス ヘッダーに PASSWORD タグがある場合に、アナリティクス スクリプトを Analytics で実行して、テストすると、Analytics によって自動的に PASSWORD コマンドが生成されるため、ユーザーは該当するパスワードを入力するように求められます。この自動生成されたコマンドにより、テスト目的でアナリティクス スクリプトの スクリプト部で PASSWORD コマンドを挿入し、アナリティクス スクリプトをロボットにアップロードする前にそれを削除する必要がありません。自動生成された PASSWORD コマンドは、パスワード値なしでログに保存されます。

パスワード値は、Analytics でアナリティクス スクリプトを実行するときに保存されるため、カーソル位置からアナリティクスの実行またはステップなど、スクリプトを実行するたびにパスワードを指定する必要があります。

アナリティクス スクリプトはアナリティクス ヘッダーがある標準のスクリプトであり、HighBond プラットフォームのロボット アプリで実行できます。

スクリプトの実行

Analytics でスクリプトを実行する場合、スクリプト内の各コマンドは、そのスクリプトの最後まで順番どおりに処理されます。

スクリプトの実行中は、Analytics でほかの作業を行うことはできません。また、一度に実行できるスクリプトは1つのみです。ただし、DO SCRIPT コマンドを使用して、ほかのスクリプトを呼び出して実行するスクリプトを作成することはできます。

スクリプト状態

スクリプト実行中は、処理状態とスクリプトまたはサブスクリプトの名前がステータスバーに表示されます。スクリプトの実行が完了すると、ステータスバーにアイコンが表示され、実行が正常に完了した場合は 、失敗した場合は  が示されます。スクリプトが失敗した場合は、エラーの発生場所である行がスクリプトエディターで自動的に強調表示  されます。

必要であれば、Esc キーを押すか Analytics を閉じることによって、スクリプトの処理を停止することができます。

メインメニューからのスクリプトの実行

メインメニューからスクリプトを実行するには2つの方法があります。

- **アプリケーションメニュー** - カスタムメニューから特定のスクリプト名を選択します
詳細については、"Analytics のメインメニューへのカスタム項目の追加" ページ 155を参照してください。
- **ツールオプション** - プロジェクトのすべてのスクリプトのピックアップリストを表示します
 1. **[ツール > スクリプトの実行]** をクリックします。
 2. **[スクリプトの実行]** ダイアログボックスで表示される、現在のプロジェクト内で利用可能なスクリプトの一覧から、実行するスクリプトを選択します。
 3. スクリプトを実行するために真と評価される必要がある条件を指定する場合は、次のいずれかを実行します。
 - If テキストボックスに論理式を入力します。
 - **[If]** をクリックし、**式ビルダー**を使用して論理式を作成します。この論理式は、スクリプトを実行するかどうかを決定するために1回だけ評価されます。論理式が False と評価された場合、スクリプトは実行されません。
 4. **[OK]** をクリックします。

総覧]タブからスクリプトを実行する

ナビゲーターの **総覧]** タブで、ツリービューで実行対象のスクリプトを右クリックして **実行]** を選択します。

スクリプト エディターからスクリプトを実行する

スクリプト エディターからスクリプトを開き、スクリプト エディターのツールバーで **実行]**  をクリックします。スクリプト エディターでスクリプトに変更を加えた場合、その変更内容は、スクリプトを実行するときに自動的に保存されます。

Windows コマンドラインからのスクリプトの実行

Windows コマンドラインまたはバッチ ファイル (***.bat**) からスクリプトを実行できます。これにより、Windows タスク スケジューラーなどのユーティリティを使用してスクリプトをスケジュールし、無人で実行できます。

コマンドライン構文はこの基本フォームを使用します。

```
acl_executable_path acl_project_path </vVarName=value> /bScript_name </min>
```

例

以下のコマンドライン構文では、**Sample Project.ACL** が開き、スクリプト **Calculate_Median_Value** が実行されます。

```
"C:\Program Files (x86)\ACL Software\ACL for Windows 14\ACLWin.exe"  
"C:\Users\username\Documents\ACL Data\Sample Data Files\Sample  
Project.ACL" /vv_table_name="Ap_Trans" /vv_field_name="Invoice_Amount"  
/bCalculate_Median_Value
```

コマンドラインの構文

メモ

ファイル名およびファイル拡張子を含む Analytics 実行ファイルと、Analytics プロジェクトへの完全パスを指定します。パスにスペースが含まれている場合は、パスを引用符で囲んでください。

パラメーター	詳細と例
"ACL_exe_path_and_filename"	<p>Analytics 実行ファイル、および実行ファイル名 (ACLWin.exe) へのパスを指定します。</p> <pre>"C:\Program Files (x86)\ACL Software\ACL for Windows 14\ACLWin.exe"</pre>
"ACL_project_path_and_filename"	<p>Analytics プロジェクト ファイル、およびスクリプトを含むプロジェクトのファイル名 (*.acl) へのパスを指定します。</p> <pre>"C:\Users\username\Documents\ACL Data\Sample Data Files\Sample Project.ACL"</pre>
/v 省略可能	<p>変数名を指定し、値を割り当てます。Analytics プロジェクトが開くと、変数が自動的に初期化されます。このとき、/v スイッチと変数名の間にはスペースを入れません。たとえば、変数 v_table_name の場合は次のようになります。</p> <pre>/vv_table_name="Ap_Trans"</pre> <p>メモ</p> <p>割り当てられた値のデータ型は、スクリプトの変数のデータ型と一致する必要があります。データ型が一致しない場合、"式タイプ不一致" エラーが発生し、スクリプトが失敗します。引用符を使用して、文字列値を修飾し、バッククォートを使用して、日付時刻値を修飾します。</p> <p>文字変数</p> <pre>/vv_table_name="Ap_Trans" /vv_field_name="Invoice_Amount"</pre> <p>数値変数</p> <pre>/vv_materiality=10000</pre> <p>日付時刻変数</p> <pre>/vv_start_date=`20180101` /vv_end_dates=`20180331`</pre>

パラメーター	詳細と例
/b	<p>実行対象のスクリプトの名前を指定します。</p> <p>このとき、<code>/b</code> スイッチとスクリプト名の間にはスペースを入れません。</p> <pre style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;">/bCalculate_Median_Value</pre>
/min 省略可能	開くときに、Analytics が最小化するように指定します。

無人で実行されるスクリプトを作成するためのガイドライン

ユーザー操作の回避

次のユーザー操作コマンドを使用しないでください。

- `DIALOG`
- `ACCEPT`
- `PASSWORD`
- `PAUSE`

代わりに、上述のコマンドライン構文を使用して、必要な変数を指定し、値を割り当てます。

確認ダイアログボックスの抑制

スクリプトの始まりに `SET SAFETY OFF` コマンドを追加すると、確認ダイアログボックスを表示しないで、必要に応じてファイルを上書きできるようになります。デフォルトの動作に戻すには、スクリプトの終わりに `SET SAFETY ON` コマンドを追加します。

`DELETE` や `RENAME` などの通常は確認ダイアログボックスを表示するコマンドの後に、`OK` パラメーターを追加します。

Analytics を終了する

`QUIT` コマンドでスクリプトを終え、Analytics を終了します。

Windows ショートカットからスクリプトを実行する

Windows ショートカットからスクリプトを実行できます。

1. Analytics のショートカットを作成します。
2. ショートカットを右クリックし、`プロパティ` を選択します。

スクリプトの操作

3. **ターゲット** フィールドで、該当するコマンドライン構文(上記)を入力します。
4. **OK** をクリックします。
5. ショートカットをダブルクリックして、スクリプトを実行します。

ACLScript での変数の操作

Analytics でスクリプトを作成するときには、通常、変化する値を構成する必要があります。例：

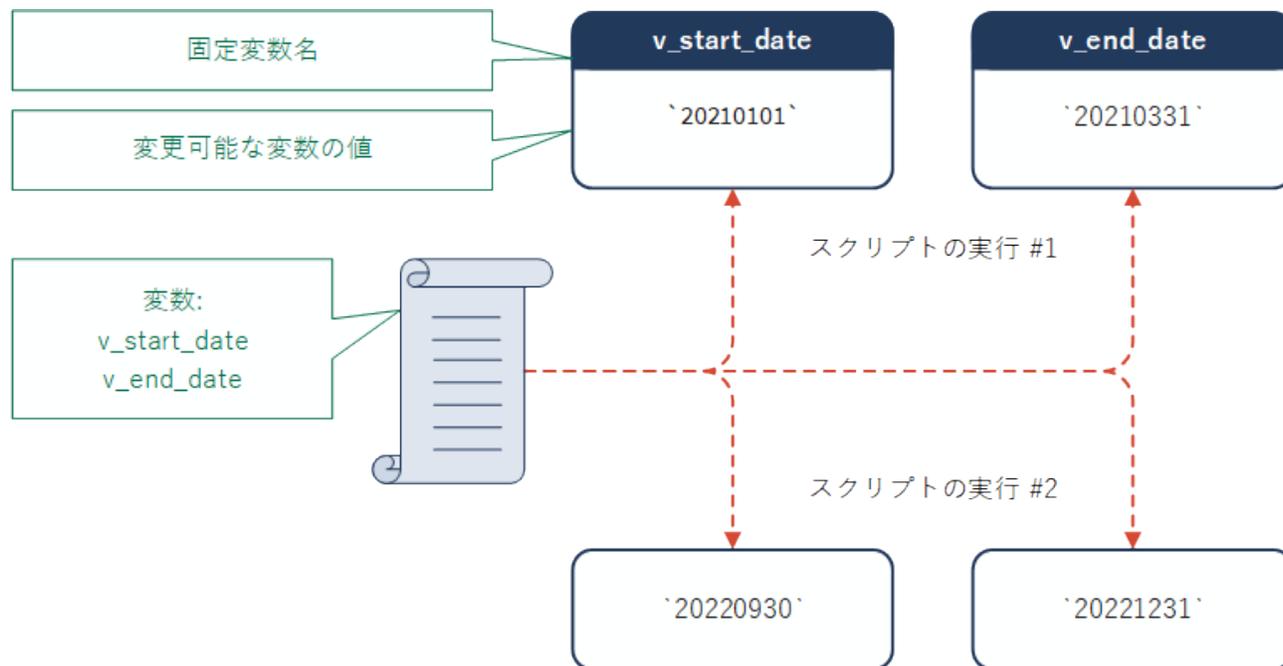
- **スクリプト構成** - すべての値の出現を更新せずにテーブルやフィールド名などのスクリプト ロジックの値をカスタマイズする簡単な方法が必要です。
- **ユーザー入力** - ユーザーがスクリプトを実行するときに、日付範囲やしきい値金額などの入力値をカスタマイズする必要があります。
- **一時保存** - スクリプトのロジックは、カウンターや合計などのスクリプトが進むに従って継続的に更新される値を一時的に保存する必要があります。

このような目的では、**変数**を使用します。

変数について

変数は、変化する値を格納または保存する名前付きコンテナであると考えられます。値は変化しますが、変数の名前は変わりません。スクリプト作成者は、スクリプト ロジック全体で固定の変数名を使用します。スクリプトが実行されるときには、変数名は処理されません。スクリプトは、現在変数に格納されている値にアクセスし、その値を処理します。

以下の例では、`v_start_date` および `v_end_date` は変数名で、日付は変数に格納されている値です。



変数の利点

スクリプト全体で、リテラル値ではなく、変数を使用できます。実際のテーブル名、フィールド名、数値、日付をそれらの項目を参照する変数で置換することで、スクリプトの柔軟性が大幅に向上します。1つの特定な状況でのみ動作するスクリプトを作成するのではなく、あらゆる状況で使用できるスクリプトを作成できます。

変数を使用する方法と変数を使用できる場所

変数が参照するリテラル項目のように、変数の作成、名前変更、削除、演算やさまざまな式での使用、エクスポート、抽出ができます。つまり、リテラル項目が有効であるすべての場所で、変数を使用できます。

変数を作成する方法

変数を作成するには、Analytics コマンドの ASSIGN コマンドまたは他のコマンドのいずれかを使用します。一般的には、コマンドはスクリプトに含めますが、Analytics コマンドラインで ASSIGN コマンドを入力すると、手動で変数を作成することもできます。

例

スクリプトにカスタマイズ可能な日付範囲を設定し、開始日と終了日の変数を作成する必要があります。

次の例では、現在、変数 `v_start_date` には日付値 ``20210101``、変数 `v_end_date` には日付値 ``20210331`` が格納されています。

これらの2つの変数を使用して、ユーザーが指定するすべての開始日と終了日を格納できます。日付が必要になる場合には、スクリプト ロジックで `v_start_date` および `v_end_date` を指定します。ただし、スクリプトが実行されるときには、ユーザーが指定した実際の日付が使用されます。

```
ASSIGN v_start_date = `20210101`  
ASSIGN v_end_date = `20210331`
```

ヒント

これらの変数のいずれかを作成するには、ASSIGN コマンドとパラメーターを Analytics コマンドラインにコピーし、Enter を押します。ナビゲーターで **変数** タブを開き、結果を表示します。

コマンドライン テキスト ボックスが表示されていない場合は、**ウィンドウ > コマンドライン** の順にクリックします。

Analytics プロジェクトでの変数の表示

変数を操作するときには、Analytics プロジェクトに存在する変数と現在の値を確認できると、非常に便利です。次の2つの方法を使用すると、変数と値を表示できます。

- 変数]タブ
- DISPLAY VARIABLES コマンド

変数]タブ

ナビゲーターの 変数]タブでは、変数の作成や、変数への値の割り当てをリアルタイムで追跡します。読み取り専用タブには、Analytics プロジェクトのすべての変数の名前、データ型、現在の値が表示されます。名前はアルファベット順に表示されます。

変数	データ型	値
v_tax_1	N	0.05
v_tax_2	N	0.06
v_tax_2_end	D	2021/12/31
v_tax_2_start	D	2021/07/01

スクリプトを1行ずつ進むと、スクリプトで定義された変数またはシステムで生成された変数が作成時に 変数]タブに表示されます。変数がすでに存在する場合は、スクリプト ロジックに基づいて値が動的に更新されます。(スクリプト エディターで、**ステップ** ) オプションを使用してスクリプトを進めます。)

スクリプト 変数で発生している正確な変更を発生時に監視できるのは、重要な診断ツールです。この機能では、スクリプト 構文の調査だけでは検出が難しいスクリプト エラーをピンポイントで特定できます。

スクリプトを実行する場合、変数に関連するすべての変更は、ブレークポイントに達したとき、またはスクリプトが完了したときに 変数]タブに表示されます。

DISPLAY VARIABLES コマンド

DISPLAY VARIABLES コマンドには、Analytics プロジェクトのすべての変数の名前、データ型、現在の値が表示されます。名前は、変数の作成日または更新日時に基づいて、日時の新しい順に表示さ

れます。

`DISPLAY 変数名` は、単一の変数に関連付けられた情報を返します。

`DISPLAY VARIABLES` コマンドは次の2つの方法で使用できます。

- **プロジェクト変数の現在の状態** - コマンドラインで `DISPLAY VARIABLES` と入力し、**Enter** キーを押します。
プロジェクト内のすべての変数とその現在値が、時系列を遡って画面に表示されます。
- **スクリプトの別の点にあるプロジェクト変数の状態** - プロジェクト変数の状態を取り込むスクリプトの時点で、`DISPLAY VARIABLES` をスクリプトに追加します。この目的でもブレークポイントを使用できます。ブレークポイントはスクリプトの実行を一時停止しますが、`DISPLAY VARIABLES` コマンドはスクリプトの実行を一時停止しないという点が違います。

スクリプトが完了した後、コマンドログで対応する `DISPLAY VARIABLES` エントリをダブルクリックします。指定したスクリプトの時点でのプロジェクト変数の状態が、日時の新しい順に画面に表示されます。

メモ

GROUP コマンド内では `DISPLAY VARIABLES` コマンドを使用できません。

ACLScript 変数に関して理解しておくべき 8 つの事項

他のプログラミング言語やスクリプト言語に詳しい場合は、すでに変数や変数のさまざまな使用方法についても熟知しています。ACLScript 変数の主な特長は次のとおりです。

スクリプトと変数を初めて使う場合は、変数の操作を始めると、次の情報を理解できるようになります。

- "1. 変数は、値を割り当てるときに自動的に作成される" 下
- "2. 変数を作成する別の方法が存在する" 見開きページ
- "3. 変数のデータ型は自動的に指定されます。" ページ 1559
- "4. 変数名は一部の制限がありますが柔軟です" ページ 1562
- "5. 変数は範囲でグローバルです" ページ 1563
- "6. 変数は Analytics プロジェクトが終了するまで永続します" ページ 1563
- "7. 変数は明示的に削除できます" ページ 1563
- "8. システム変数は一部の Analytics コマンドによって自動的に作成されます" ページ 1565

1. 変数は、値を割り当てるときに自動的に作成される

変数は、初期値を名前付き変数に割り当てるときに自動的に作成(宣言)されます。変数への初期値の割り当ては、変数の初期化とも呼ばれます。

次の例では、ASSIGN コマンドを使用して、値 "Ap_Trans" を変数に割り当てるときに、v_input_table 変数が自動的に作成されます。つまり、変数の作成と、変数への値の割り当て(宣言、初期化)は1つのステップで実行されます。

```
ASSIGN v_input_table = "Ap_Trans"
```

変数を作成するタイミングと場所

変数を最初に使用する前に変数が作成されるかぎり、変数はスクリプトのどの部分でも作成できます。ベストプラクティスとして、すべてのユーザー入力変数とスクリプト構成変数をスクリプトの先頭で作成することをお勧めします。このアプローチに従うと、ユーザーはスクリプトの開始時点ですべての必須入力値を入力し、スクリプトを実行させることができます。また、すべての必須のスクリプト構成変数がスクリプトの先頭の単一の整理された場所に配置されていることを保証できます。

ユーザー入力や詳細構成が必要ない変数は、スクリプトの変数が必要になる部分で作成できます。

必要に応じて、文字または日付時刻変数に空白の値を割り当て、変数だけ存在し、初期値が空になるようにできます。

```
ASSIGN v_input_table = ""
ASSIGN v_all_field_list = BLANKS(20000)
ASSIGN v_start_date = `19000101`
```

2. 変数を作成する別の方法が存在する

変数を作成する方法は、変数を使用する方法によって異なります。

使用方法	作成方法
スクリプト構成 一時ストレージ スクリプトの他の使用	<p>ASSIGN コマンド</p> <p>スクリプトの別の行で"ASSIGN コマンド" ページ 1660を使用し、変数を宣言して初期化します。割り当てられた値のデータ型は、変数のデータ型を示します。</p> <p>メモ</p> <p>ASSIGN キーワードを明示的に指定することをお勧めします。スクリプトが読みやすくわかりやすくなります。ただし、ASSIGN キーワードを省略し、次のように指定することもできます。</p> <pre>変数名 = 変数値</pre>
ユーザー入力	<p>ACCEPT コマンド</p> <p>"ACCEPT コマンド" ページ 1625を使用して、1つ以上のスクリプト入力値をインタラクティブにユーザーに確認する基本のダイアログボックスを作成します。各入力値は、それが表す内容に関係なく、名前付き文字変数に格納されます。</p>

使用方法	作成方法
	<p>"インタラクティブ スクリプトの作成" ページ 1586を参照してください。</p> <p>DIALOG コマンド</p> <p>"DIALOG コマンド" ページ 1736と関連付けられたダイアログビルダーを使用して、1つ以上のスクリプト入力値をインタラクティブにユーザーに確認する高度なダイアログボックスを作成します。各入力値は名前付き変数に格納されます。ほとんどの入力オプションは文字変数を使用します。</p> <p>ACCEPT コマンドで提供されるオプションのほかに、DIALOG コマンドではチェックボックス、ラジオボタン、ドロップダウンリストなどのその他のオプションが提供されます。</p> <p>"インタラクティブ スクリプトの作成" ページ 1586を参照してください。</p> <p>PARAM タグ、TABLE タグ、FIELD タグ</p> <p>ロボットで実行されるスクリプトのユーザー入力では、アナリティクス ヘッダーを作成し、"PARAM タグ" ページ 2616、"TABLE タグ" ページ 2629、または"FIELD タグ" ページ 2631を使用します。</p> <p>PARAM タグは1つの変数作成方法であり、関連付けられた変数のデータ型を明示的に指定する必要があります。</p> <p>TABLE タグとFIELD タグには、文字変数の入力値が格納されます。</p>
資格情報入力	<p>PASSWORD コマンド</p> <p>"PASSWORD コマンド" ページ 2015を使用して、パスワード、トークン、または他の機密情報をインタラクティブにユーザーに確認するダイアログボックスを作成します。PASSWORD コマンドは、保護されたメモリの場所に資格情報値の安全なデータ入力と暗号化された保存を提供します。</p> <p>資格情報の安全な処理の一部として、機密情報を含む変数は表示されず、直接管理できません。</p> <p>"インタラクティブ スクリプトの作成" ページ 1586を参照してください。</p> <p>SET PASSWORD コマンドを使用して、ユーザー入力せずに、直接パスワードを指定することもできます。ただし、パスワードはスクリプトにプレーンテキストとして表示されます。</p> <p>PASSWORD タグ</p> <p>ロボットで実行されるスクリプトの資格情報入力では、アナリティクス ヘッダーを作成し、"PASSWORD タグ" ページ 2626を使用します。</p>

3. 変数のデータ型は自動的に指定されます。

ASSIGN コマンドを使用するときには、変数に割り当てられた値のデータ型を示します。このプロセスは暗黙の型宣言と呼ばれます。ほとんどの場合、Analytics で変数のデータ型を明示的に指定することはありません。

ヒント

作成するスクリプトの変数のデータ型には十分に注意してください。スクリプト エラーの一般的な原因は、変数のデータ型と変数を使用している方法が一致していないことです。詳細については、"変数のデータ型の重要性" ページ 1565を参照してください。

変数に割り当てられる値の種類

リテラル値を変数に割り当てたり、フィールドを変数に割り当てたりすることができます。変数は、現在選択されているレコードのフィールド値で初期化されます。1つの変数の現在の値を別の変数に割り当てることもできます。

値のデータ型

Analytics では、すべてのデータ値が次のデータ型 (またはカテゴリ) のいずれかになっています。

- 文字
- 数値
- 日付時刻
- 論理

値のデータ型は次のいずれかの方法で定義されます。

- **リテラル値** - リテラル値のデータ型は、値を囲む修飾子の有無と、値を構成する英数字で定義されます。
- **フィールド** - フィールドの値のデータ型はフィールドのデータ型で定義されます。
- **変数** - "from" 変数とそこに含まれる値はすでに設定され、"to" 変数のデータ型になります。一般的に、作成した後は、変数のデータ型は変更しません (ただし、このような操作は禁止されていません)。

ASSIGN コマンドを使用して、別のデータ型の変数を作成する例

メモ

変数を作成する ACCEPT コマンド、または DIALOG コマンドのほとんどのオプションを使用するときには、含まれる値の型に関係なく、変数は文字データ型です。

変数と値のデータ型	例
<p>文字</p> <p>文字値には、英数字文字と特殊文字を含めることができます。</p> <p>文字リテラル値は、引用符または二重引用符で囲む必要があります。</p> <p>文字値に引用符が含まれる場合は、次のアプローチを使用してください。</p> <ul style="list-style-type: none"> ◦ 文字値に二重引用符がある場合は、値全体を一重引用符で囲んでください。 ◦ 文字値に一重引用符がある場合は、値全体を二重引用符で囲んでください。 	<p>文字の例</p> <p>文字値 "Vendor_number" を v_input_table 変数に割り当てます。</p> <pre>ASSIGN v_input_table = "Vendor_number"</pre> <p>現在選択されているレコードの文字 Vendor_number フィールドの値を v_identifier 変数に割り当てます。</p> <pre>ASSIGN v_identifier = Vendor_number</pre>

変数と値のデータ型	例																											
	<p>メモ</p> <p>修飾フィールド名を変数に割り当てること、非修飾フィールド名を変数に割り当てることの違いを理解することが重要です。</p> <p>"Vendor_number" を割り当てると、フィールド名を構成する文字の実際の文字列が割り当てられます。</p> <p>Vendor_number を割り当てると、Vendor_number フィールドに含まれる値のいずれかが割り当てられます。</p> <div data-bbox="792 541 1341 919" style="border: 1px solid #ccc; padding: 5px;"> <p>ナビゲーター + 閉</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">変数</th> <th style="text-align: left;">データ型</th> <th style="text-align: left;">値</th> </tr> </thead> <tbody> <tr> <td>v_identifier</td> <td>C</td> <td>11645</td> </tr> <tr> <td>v_input_table</td> <td>C</td> <td>Vendor_number</td> </tr> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table> <p style="text-align: right; margin-top: 5px;"> 総覧 ログ 変数 </p> </div>	変数	データ型	値	v_identifier	C	11645	v_input_table	C	Vendor_number																		
変数	データ型	値																										
v_identifier	C	11645																										
v_input_table	C	Vendor_number																										
<p>数値</p> <p>数値には、0 ~ 9 の数値、小数点、負号を含めることができます。</p> <p>リテラル数値は修飾子で囲まないでください。</p>	<p>数値の例</p> <p>数値 1000 を v_min_amount 変数に割り当てます。</p> <div data-bbox="753 1066 1344 1136" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>ASSIGN v_min_amount = 1000</pre> </div> <p>現在選択されているレコードの数値 Invoice_Amount フィールドの値を v_running_total 変数に割り当てます。</p> <div data-bbox="753 1262 1344 1331" style="border: 1px solid #ccc; padding: 5px;"> <pre>ASSIGN v_running_total = Invoice_Amount</pre> </div>																											
<p>日付時刻</p> <p>日付時刻値は日付、時刻、または日付時刻にすることはできます。値は Analytics でサポートされている形式を使用する必要があります。</p> <p>サポートされている日付時刻形式の詳細については、「日付時刻リテラルの書式」ページ 888 を参照してください。</p> <p>リテラル日付時刻値はバッククォートで囲む必要があります。</p>	<p>日付時刻の例</p> <p>日付値 `20210101` を v_start_date 変数に割り当てます。</p> <div data-bbox="753 1472 1344 1541" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>ASSIGN v_start_date = `20210101`</pre> </div> <p>現在選択されているレコードの日付時刻値 Invoice_Date フィールドの値を v_date 変数に割り当てます。</p> <div data-bbox="753 1667 1344 1736" style="border: 1px solid #ccc; padding: 5px;"> <pre>ASSIGN v_date = Invoice_Date</pre> </div>																											

変数と値のデータ型	例
<p>メモ</p> <p>Analytics は文字データ型を使用する日付時刻値もサポートします。詳細については、「日付時刻値について」ページ 1566を参照してください。</p>	
<p>論理</p> <p>論理値は <code>T</code> または <code>F</code> にすることができます。リテラル論理値は修飾子で囲まないとはいけません。</p>	<p>論理値の例</p> <p>論理値 <code>T</code> を <code>v_Q1</code> 変数を割り当てます。</p> <pre>ASSIGN v_Q1 = T</pre> <p>現在選択されているレコードの論理値 <code>Approved</code> フィールドの値を <code>v_approved</code> 変数に割り当てます。</p> <pre>ASSIGN v_approved = Approved</pre>

4. 変数名は一部の制限がありますが柔軟です

ほとんどの場合、目的に最適な方法で変数名を指定できます。

自発的な変換

これらの自発的な変換により、スクリプトが読みやすく、わかりやすくなります。

- スクリプトで変数が実行する役割に直接関連する、理解しやすくわかりやすい名前を変数に指定します。変数の目的がわかりにくい短く不可解な名前よりも、長めのわかりやすい名前の方が良いです。
- すべての変数名の先頭に `v_` を付けるなど、一般的な命名規則を採用します。

適用される制限事項

Analytics では、変数名について次の制限事項が適用されます。

- 最大長** - 31 文字
- 有効な文字** - 英数字とアンダースコア(`_`)。名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。

メモ

代入変数に使用される変数の名前に、`@`のような英語以外の文字は使用しないでください。変数名に英語以外の文字が含まれていると、変数の代入が失敗します。

このヘルプトピックの英語版以外のバージョン(フランス語版など)には、アクセント付き文字で変数名が表記されている場合があります。実際のスクリプトでは、変数代入で使用される変数名にアクセント付き文字を絶対に使用しないでください。

- 一意性 - 変数名は Analytics プロジェクトで一意的でなければなりません。変数はプロジェクト全体でグローバルに機能するためです。

変数名は大文字と小文字が区別されません。ACLScript では、`v_start_date` と `v_Start_Date` は同じ変数です。

5. 変数は範囲でグローバルです

Analytics スクリプトで作成された変数は、後続のすべての部分で使用できます。Analytics プロジェクトの他のスクリプトでも使用できます。親スクリプトで定義された子スクリプトで変数を使用する場合には、この動作が役立ちます。

ACLScript は変数のローカルスコーピングをサポートしません。

6. 変数は Analytics プロジェクトが終了するまで永続します

変数は、変数に含まれる Analytics プロジェクトが終了するか、変数を明示的に削除するまで、コンピューターのメモリに残ります。

プロジェクトを終了した後も変数を永続させ、プロジェクトをもう一度開くときに使用可能にする場合は、永続変数を作成できます。永続変数を作成するには、変数名の前にアンダースコアを付けます。例：`_v_start_date` 変数と変数に含まれる値の両方が永続します。永続変数を削除するには、明示的に削除するしかありません。

メモ

永続変数は、ロボットで実行されるスクリプトではサポートされません。

7. 変数は明示的に削除できます

Analytics プロジェクトで 1 つまたはすべての変数を明示的に削除するには、コマンドラインまたはスクリプトで DELETE コマンドを使用します。

特定の変数を明示的に削除する

```
DELETE 変数名 OK
```

すべての変数を明示的に削除する

```
DELETE ALL OK
```

削除される内容

変数を削除するときには、名前付きコンテナと底に含まれる値をコンピューターのメモリからのみ削除します。

変数を削除しても、スクリプトの変数に関連する項目には影響しません。メモリの変数を元に戻すには、コマンドラインのASSIGNコマンドを手動で実行し、変数割り当てを含むスクリプトを実行します。

変数とその値を削除しても、Analytics テーブルなどの変数外に存在する値によって参照される項目には影響しません。

例

次の2つの例は、変数と変数によって参照されている項目がどのように個別のエンティティになるのかを示しています。一方を削除しても他方には影響しません。

メモ

この例では、[変数代入](#)を使用して、変数名に変数の内容を代入します。

最初の例では、変数 `v_test_table` を削除すると、変数値 `"Running_totals"` が削除されます。実際の `Running_totals` テーブルは削除されません。

```
ASSIGN v_test_table = "Running_totals"

COMMENT 正常に実行されます
OPEN %v_test_table%

COMMENT 変数が削除されます
DELETE v_test_table OK

COMMENT 正常に実行されます
OPEN Running_totals
```

2番目の例では、`Running_totals` テーブルを削除しても `v_test_table` 変数の影響はありませんが、変数値は存在しないテーブルを参照しています。

```
ASSIGN v_test_table = "Running_totals"

COMMENT 正常に実行されます
```

```
OPEN %v_test_table%

COMMENT 変数によって参照されているテーブルが削除されます
DELETE FORMAT %v_test_table% OK

COMMENT "Table 'Running_totals' invalid" エラーで失敗します
OPEN %v_test_table%
```

8. システム変数は一部の Analytics コマンドによって自動的に作成されます

特定の Analytics コマンドを実行するときに、システム変数が自動的に作成されます。たとえば、COUNT コマンドは `COUNTn` システム変数を生成します。これには合計のレコード数が含まれます。スクリプトの後続の部分では、これらの変数を変数に含まれる値を使用できます。

たとえば、ユーザーがレコードのないテーブルを選択した場合に、このスクリプトは自動的に終了します。

```
ACCEPT "Select an input table:" FIELDS "xf" TO v_input_table
OPEN %v_input_table%
COUNT
PAUSE "選択したテーブルにはレコードがありません。スクリプトは終了します。" IF COUNT1 = 0
ESCAPE IF COUNT1 = 0
DO SCRIPT summarize_transactions
```

詳細については、「Analytics コマンドによって作成されたシステム変数」ページ 1582を参照してください。

変数のデータ型の重要性

Analytics コマンドまたは関数を変数を参照するときには、変数のデータ型がコマンドまたは関数で許可されたデータ型と一致する必要があります。データ型が一致しない場合、エラーが発生し、コマンドまたは関数の処理が失敗します。

このような状況でエラーを回避するために、Analytics 関数を使用して、変数の格納された値を使用可能なデータ型に変換できます。たとえば、次の式では、文字変数に格納された日付を日付時刻データ型に変換します。

```
CTOD(v_character_date)
```

次の式では、文字変数に格納された数値を数値データ型に変換します。

```
VALUE(v_character_number, 2)
```

詳細については、[変換関数](#)を参照してください。

メモ

日付および数値は、ユーザー入力が入力された場合は、文字変数に格納されます。詳細については、[変数を作成するためのさまざまな方法](#)を参照してください。

日付時刻値について

Analytics では、日付時刻データ型または文字データ型を使用する日付時刻値を処理できます。一般的に、日付時刻値に関連付けられたデータ型は、Analytics にインポートするときのデータの性質に基づいています。

日付、時刻、日付時刻値を含む変数の意味。簡潔に言うと、操作している基本データのデータ型を認識している必要があるということです。スクリプト構文は、使用中のデータ型に基づいて異なる必要があります。

例

次の2つの例では、ユーザーが入力した日付範囲に基づいて、日付フィールドをフィルタリングする構文を示します。

- 最初の例では、日付フィールドが日付時刻データ型を使用します。
- 2番目の例では、日付フィールドが文字データ型を使用します。
- いずれの例でも、`v_start_date` および `v_end_date` 変数は文字データ型を使用します。これらはACCEPT コマンドまたはDIALOG コマンドを使用して作成されたからです。

最初の例では、CTOD() 関数を使用して、日付範囲値を文字データ型から日付時刻データ型に変換する必要があります。(CTOD は「Character to Date」を表します。) 日付範囲値のデータ型が日付時刻日付フィールドのデータ型と一致するように、この変換を実行する必要があります。

```
SET FILTER TO BETWEEN(datetime_date_field, CTOD(v_start_date), CTOD(v_end_date))
```

2番目の例では、日付範囲値の変換を実行しません。文字日付フィールドのデータ型とすでに一致するからです。変換する場合は一致せず、スクリプト エラーが発生します。

```
SET FILTER TO BETWEEN(character_date_field, v_start_date, v_end_date)
```

代入変数

Analytics コマンドまたは関数の入力として文字変数を使用するときには、変数名に変数の値を代入する必要があります。この代入を実行しない場合、コマンドまたは関数は、変数に格納されている値ではなく実際の変数名で動作しようとし、エラーが発生します。

変数代入は、変数名の文字変数に格納されている値を代入する方法です。変数代入を使用するには、文字変数の名前をパーセント記号(%)で囲みます。たとえば、`%v_input_table%`は`v_input_table`に格納された実際のテーブル名を取得し、それを変数名に代入します。

ほとんどの場合、スクリプトのユーザー入力は文字変数に格納されるため、変数代入は、ユーザー入力をスクリプト ロジックを取り込むときに一般的に使用されます。

メモ

変数代入は、文字データを表す値を格納する文字変数での使用のみが想定されています。他のデータ型の変数、または数値、日付時刻、論理値を表す値を格納する文字変数で変数代入を使用すると、信頼できない結果またはエラーが生じる可能性があります。

例

スクリプトで、ユーザーに入力テーブルを選択できるようにします。ユーザーが選択したテーブルをスクリプトによって開きます。OPEN コマンドで変数代入を使用し、コマンドが正しいテーブル名に対して実行されるようにします。

`OPEN %v_input_table%`を指定すると、スクリプトによって次の内容が実際に表示されます。

```
OPEN Name_of_table_selected_by_user
```

```
ACCEPT "Select an input table:" FIELDS "xf" TO v_input_table
OPEN %v_input_table%
```

この例で変数代入を使用しない場合、OPEN コマンドがテーブル`v_input_table`を開こうとして、その名前のテーブルは存在しないため、通常はスクリプトが失敗します。

```
ACCEPT "Select an input table:" FIELDS "xf" TO v_input_table
OPEN v_input_table
```

変数代入の使用は注意が必要です

特定の状況では、変数代入が必要に思われる場合がありますが、実際には変数代入によりエラーが発生します。

前述の例の1つについて考えてみます。

```
SET FILTER TO BETWEEN(character_date_field, v_start_date, v_end_date)
```

次の例では、変数とフィールドの両方で文字データとして格納された日付値を比較しているため、変数で変数代入を使用すると動作するはずですが、

```
SET FILTER TO BETWEEN(character_date_field, %v_start_date%, %v_end_date%)
```

では、この状況で変数代入のエラーが発生する理由は何でしょうか。それは変数代入が絶対にリテラルだからです。エラーのトラブルシューティングを実行するには、変数に格納された正確な値を確認し、手動で式に代入して、結果の式を確認します。例：

```
SET FILTER TO BETWEEN(character_date_field, 20000101, 20001231)
```

この式は、文字データ型を使用する日付と、代入後に数値データ型を使用する数値を比較します。これはデータ型が一致していません。

BETWEEN() で文字フィールドを指定したため、この関数は `v_start_date` と `v_end_date` の値を自動的に文字値として解釈し、変数代入を使用する必要はありません。

詳細については、「トラブルシューティング」 ページ 1580を参照してください。

ルールの例外

場合によっては、数値変数で問題なく変数代入をできることがあります。この利点は、数値変数の値がログに取り込まれ、トラブルシューティングで役立つということです。

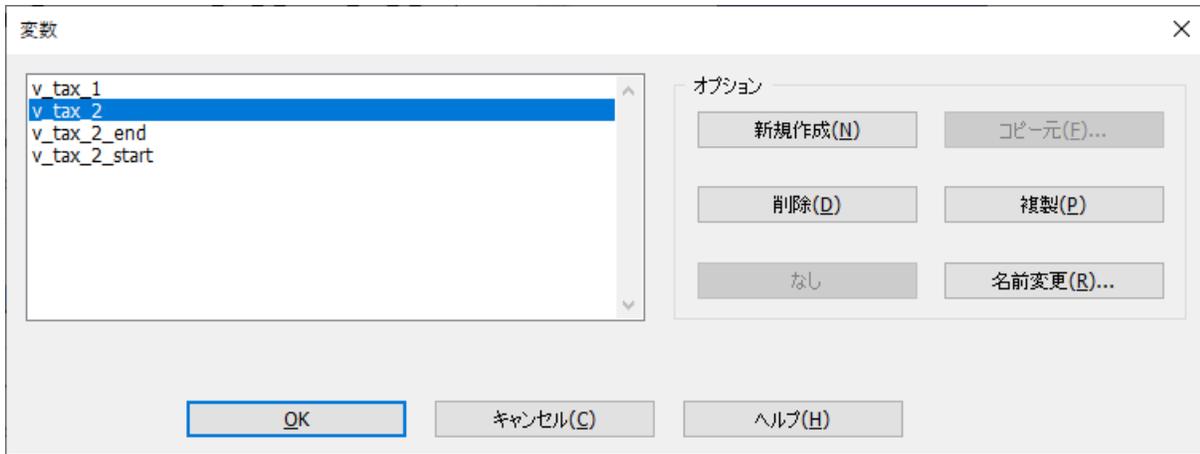
次の例はいずれも `COUNT1` 数値変数の値を `v_counter` 数値変数に割り当てます。ただし、最初の例では、実際の割り当てられた値はログに取り込まれません。2番目の例では、実際の値が取り込まれます。

```
COMMENT log entry: ASSIGN v_counter = COUNT1
ASSIGN v_counter = COUNT1
```

```
COMMENT log entry: ASSIGN v_counter = 500
ASSIGN v_counter = %COUNT1%
```

手動による変数の管理

Analytics メインメニューから **変数** ダイアログボックスを使用して手動で変数を管理できます。変数の作成、名前の変更、複製、または削除、あるいは変数値の更新ができます。



実行するメンテナンスは、メモリ内の変数の現在のインスタンスにのみ影響します。たとえば、変数の名前を変更する場合は、メモリの変数のインスタスの名前のみを変更しています。スクリプトの変数は変更していません。

メモ

ユーザーインターフェイスで手動で変数を管理する機能は、一部の状況で役立つ場合があります。手動の管理では、追加の機能がなく、コマンドを使用して変数を管理します。

ほとんどのスクリプト作成者はスクリプト エディターと、コマンドラインや **変数** タブなどの関連付けられた機能でのみ変数を操作します。

1. Analytics のメインメニューから、**編集 > 変数**] を選択します。

変数] ダイアログボックスが開き、プロジェクトに含まれるすべてのシステム変数とユーザー定義変数が表示されます。システム変数は、コマンドによって自動的に作成される変数です。

2. 次のいずれかを実行します。

新しい変数を作成する	<ol style="list-style-type: none"> a. 新規作成] をクリックして式ビルダーを開きます。 b. 変数に割り当てる式または値を 式] テキスト ボックスに入力します。 リテラル値は正しい形式と修飾子 (必要な場合) を使用する必要があります。詳細については、変数のデータ型は自動的に指定されますを参照してください。 c. 省略可能。 検証] をクリックし、値または式が有効であることを確認します。 d. 次の名前で作成] テキスト ボックスに変数の名前を入力します。 詳細については、次を参照してください。 <ul style="list-style-type: none"> • 変数名は一部の制限がありますが柔軟です • 変数は Analytics プロジェクトが終了するまで永続します e. OK] をクリックします。 変数が作成され、 変数] タブに表示されます。
------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

変数値を更新する	<p>a. リストの変数を選択し、OK]をクリックして、式ビルダーを開きます。</p> <p>b. 変数に割り当てられた式または値を 式]テキスト ボックスで更新します。 リテラル値は正しい形式と修飾子(必要な場合)を使用する必要があります。詳細については、変数のデータ型は自動的に指定されますを参照してください。</p> <p>c. 省略可能。 検証]をクリックし、値または式が有効であることを確認します。</p> <p>d. OK]をクリックします。 変数に割り当てられた値がコンピューターのメモリで更新されます。</p>
変数名を変更する	<p>a. リストの変数を選択し、名前の変更]をクリックします。</p> <p>b. 名前の変更]テキストボックスで、新しい名前を入力して、OK]をクリックします。 変数名がコンピューターのメモリで変更されます。</p> <p>c. キャンセル]をクリックして、変数]ダイアログ ボックスを閉じます。</p>
変数を複製する	<p>a. リストの変数を選択し、複製]をクリックします。 変数と値がコンピューターのメモリで複製されます。複製された変数の名前には、数値サフィックスが追加されます。</p> <p>b. 次のいずれかを実行します。</p> <ul style="list-style-type: none"> • 完了]をクリックして、変数]ダイアログ ボックスを閉じます。 • OK]をクリックして、変数に割り当てられた値または式を更新します。
変数の削除	<p>a. リストの変数を選択します。</p> <p>b. 省略可能。複数の変数を選択して削除するには、Shift キー + クリック、または Ctrl キー + クリックを使用します。</p> <p>c. 削除]をクリックします。</p> <p>d. もう一度 削除]をクリックします。 変数がコンピューターのメモリから削除されます。</p> <p>e. 完了]をクリックして、変数]ダイアログ ボックスを閉じます。</p>

変数の使用例

- "スクリプト構成の変数" 次のページ
- "ユーザー入力の変数" ページ 1572
- "ユーザー入力変数のデフォルト値" ページ 1575
- "一時保存の変数" ページ 1577

スクリプト構成の変数

売上税金額を計算し、定期的に変更される売上税率を処理するスクリプトを作成します。

必須の変数

異なる税率と日付範囲の構成を許可するには、"ASSIGN コマンド" ページ 1660を使用して、次の変数のセットを作成します。

変数名	変数の内容
v_tax_1	最初の税率
v_tax_2	2番目の税率
v_tax_2_start	2番目の税率が有効になる日付
v_tax_2_end	2番目の税率が終了する日付、または現在の日付(税率がまだ有効な場合)

スクリプト

```
COMMENT 売上税率の変数と日付範囲を作成して初期化します
COMMENT 必要に応じて、追加の税率と日付範囲変数を追加して、スクリプトを拡張できま
す
ASSIGN v_tax_1 = 0.05
ASSIGN v_tax_2 = 0.06
ASSIGN v_tax_2_start = `20210701`
ASSIGN v_tax_2_end = `20211231`

COMMENT 2つの異なる期間に対して別の売上税率を使用する条件付き演算フィールドを
作成します
DEFINE FIELD sales_tax COMPUTED

Invoice_Amount * v_tax_2 IF BETWEEN(Invoice_Date, v_tax_2_start, v_
tax_2_end)
Invoice_Amount * v_tax_1
```

変数]タブで作成および初期化された変数

スクリプトを実行した後は、作成および初期化された変数が **変数]タブ**に表示されます。

ASSIGN コマンドは変数を作成するため、割り当てられた値のデータ型は変数のデータ型を示します。この例では、2つの数値変数と2つの日付時刻変数があります。

変数	データ型	値
v_tax_1	N	0.05
v_tax_2	N	0.06
v_tax_2_end	D	2021/12/31
v_tax_2_start	D	2021/07/01

ユーザー入力の変数

ユーザーが Analytics プロジェクトで任意のテーブルを選択し、数値範囲と日付範囲でテーブルのレコードをフィルタリングして、フィルタリングされたレコードを新しいテーブルに出力するスクリプトを作成します。

必須の変数

スクリプトのユーザー入力を取り込むには、"ACCEPT コマンド" ページ 1625を使用して、次の変数のセットを作成します。

変数名	変数の内容
v_input_table	入力として使用する Analytics テーブルの名前
v_output_table	フィルタリングされたレコードの Analytics 出力テーブルの名前
v_numeric_field	数値範囲で使用する入力テーブルの数値フィールド
v_min_amount	数値範囲の下限
v_max_amount	数値範囲の上限

変数名	変数の内容
v_date_field	日付範囲で使用する入力テーブルの日付フィールド
v_start_date	日付範囲で最も早い日付
v_end_date	日付範囲で最も遅い日付

スクリプトの考慮事項

ACCEPT コマンドを使用して、対話型でユーザーにスクリプトの入力値を確認しているため、値が数値でも日付でも、各入力値は名前付きの文字変数に格納されます。この事実を考慮すると、スクリプトの特定の位置で Analytics 関数を使用して、変数値をスクリプト ロジックで必要なデータ型に変換する必要があります。

例：

- VALUE() 関数は、v_min_amount に格納された数値を文字データ型から数値データ型に変換します。

```
VALUE(v_min_amount, 2)
```

- CTOD() 関数は v_start_date に格納された日付を文字データ型から日付データ型に変換します。

```
CTOD(v_start_date)
```

これらの変数の値は、数値または日付時刻データ型を使用するフィールドの値と比較されているため、これらのデータ型変換を実行する必要があります。

変数代入を使用して、テーブルの実際の名前と変数の一部に格納されたフィールドにアクセスする必要もあります。

例：

```
OPEN %v_input_table%
```

スクリプト

```
COMMENT ダイアログボックスを作成し、入力テーブルと出力テーブルをユーザーに確認します
ACCEPT "入力テーブルを選択:" FIELDS "xf" TO v_input_table, "出力テーブル名
```

```
を指定 (スペースなし):" TO v_output_table

COMMENT ユーザーが選択した入カテーブルを開きます
OPEN %v_input_table%

COMMENT ダイアログボックスを作成し、数値フィールドと数値範囲をユーザーに確認します
ACCEPT "フィルタリングする数値フィールドを選択:" FIELDS "N" TO v_numeric_field,
"最小金額を指定:" TO v_min_amount, "最大金額を指定:" TO v_max_amount

COMMENT ダイアログボックスを作成し、日付フィールドと日付範囲をユーザーに確認します
ACCEPT "フィルタリングする日付フィールドを選択:" FIELDS "D" TO v_date_field, "
開始日を指定 (YYYYMMDD):" TO v_start_date, "終了日を指定 (YYYYMMDD):" TO
v_end_date

COMMENT ユーザーの数値および日付範囲に基づいて入カテーブルをフィルタリングします
SET FILTER TO BETWEEN(%v_numeric_field%, VALUE(v_min_amount, 2), VALUE
(v_max_amount, 2)) AND BETWEEN(%v_date_field%, CTOD(v_start_date),
CTOD(v_end_date))

COMMENT フィルタリングされたレコードのセットをユーザーが指定した出カテーブルに抽出しま
す
EXTRACT RECORD TO %v_output_table%

COMMENT フィルタリングされたレコードの出カテーブルを開きます
OPEN %v_output_table%
```

変数]タブで作成および初期化された変数

スクリプトを実行した後は、作成および初期化された変数が **変数]タブ**に表示されます。

ACCEPT コマンドは変数を作成するため、変数値が数値または日付を表す場合でも、変数は文字データ型を使用します。

WRITE1 はシステム変数です。この現在の状況では、値は出カテーブル(**v_output_table**)のレコード数を表します。詳細については、"Analytics コマンドによって作成されたシステム変数" ページ 1582を参照してください。

ナビゲーター		
変数	データ型	値
OUTPUTFOLDER	C	/Tables/Accounts_Payable
v_date_field	C	Invoice_Date
v_end_date	C	20000331
v_input_table	C	Ap_Trans
v_max_amount	C	5000
v_min_amount	C	1000
v_numeric_field	C	Invoice_Amount
v_output_table	C	Ap_Trans_filtered
v_start_date	C	20000101
WRITE1	N	6

総覧 ログ 変数

ユーザー入力変数のデフォルト値

ユーザーに空白フィールドを表示するのではなく、ユーザー入力フィールドにデフォルト値をあらかじめ入力します。デフォルト値は、入力値の正しい形式を表示することで、作業とエラーを減らすことができます。

デフォルト値を指定するには、最初に"ASSIGN コマンド" ページ 1660を使用して、デフォルト値で変数を初期化します。同じ変数名で"ACCEPT コマンド" ページ 1625を使用し、ユーザーが任意でデフォルト値を上書きできるようにします。

この例は上記の"ユーザー入力の変数" ページ 1572と同じです。ただし、数値および日付範囲のデフォルト値で変数を初期化するための予備セクションが追加されています。

デフォルト値で変数を初期化する

デフォルト値で変数を初期化するには、ASSIGN コマンドを使用して、次の変数を作成します。

変数名	変数の内容
v_min_amount	数値範囲の下限
v_max_amount	数値範囲の上限
v_number_of_months	日付範囲の長さ(月)
v_end_date	日付範囲で最も遅い日付(演算)

変数名	変数の内容
v_start_date	日付範囲で最も速い日付(演算)

スクリプトの考慮事項

ACCEPT コマンドによって作成されたデフォルト値をフィールドにあらかじめ入力しているため、これらの値は文字データ型を使用する必要があります別のデータ型の値を入力しようとすると、値が表示されません。

この事実を考慮して、Analytics 関数を使用し、デフォルト値を文字データ型に変換しなければならない場合があります。(スクリプトの後の部分では、他の関数を使用して、スクリプト ロジックで必要なデータ型に変数値を変換できます。)

例:

- STRING() 関数は 5000 という数値を文字列に変換します。この文字列は `v_min_amount` 変数に格納されます。

```
ASSIGN v_min_amount = STRING(5000, 4)
```

- DATE() 関数は計算された終了日を `v_end_date` 変数に格納された文字列に変換します。

```
ASSIGN v_end_date = ALLTRIM(DATE(EOMONTH(TODAY(), -1)))
```

日付範囲の計算

日付範囲の計算により、デフォルトの範囲が現在の日付のちょうど 12 か月前に設定されます。範囲の長さを変更するには、`v_number_of_months` 変数に割り当てられた値を更新します。日付範囲を計算するために使用される関数の詳細については、「EOMONTH() 関数」ページ 2257を参照してください。

スクリプト

```
COMMENT Analytics セッションの期間の日付形式を設定します
SET DATE "YYYYMMDD"

COMMENT 数値範囲の上限と下限のデフォルト値を指定します
ASSIGN v_min_amount = STRING(5000, 4)
ASSIGN v_max_amount = STRING(100000, 6)
```

```

COMMENT 計算された日付範囲の長さを指定します(月数)
ASSIGN v_number_of_months = 12

COMMENT 現在の日付と v_number_of_months で指定された長さに基づいて、デフォルト
の日付範囲の開始日と終了日を計算します。
ASSIGN v_end_date = ALLTRIM(DATE(EOMONTH(TODAY(),-1)))
ASSIGN v_start_date = ALLTRIM(DATE(EOMONTH(CTOD(v_end_date), -v_
number_of_months) + 1))

COMMENT ダイアログボックスを作成し、入力テーブルと出力テーブルをユーザーに確認します
ACCEPT "入力テーブルを選択:" FIELDS "xf" TO v_input_table, "出力テーブル名
を指定(スペースなし):" TO v_output_table

COMMENT ユーザーが選択した入力テーブルを開きます
OPEN %v_input_table%

COMMENT ダイアログボックスを作成し、数値フィールドと数値範囲をユーザーに確認します
ACCEPT "フィルタリングする数値フィールドを選択:" FIELDS "N" TO v_numeric_field,
"最小金額を指定:" TO v_min_amount, "最大金額を指定:" TO v_max_amount

COMMENT ダイアログボックスを作成し、日付フィールドと日付範囲をユーザーに確認します
ACCEPT "フィルタリングする日付フィールドを選択:" FIELDS "D" TO v_date_field, "
開始日を指定 (YYYYMMDD):" TO v_start_date, "終了日を指定 (YYYYMMDD):" TO
v_end_date

COMMENT ユーザーの数値および日付範囲に基づいて入力テーブルをフィルタリングします
SET FILTER TO BETWEEN(%v_numeric_field%, VALUE(v_min_amount, 2), VALUE
(v_max_amount, 2)) AND BETWEEN(%v_date_field%, CTOD(v_start_date),
CTOD(v_end_date))

COMMENT フィルタリングされたレコードのセットをユーザーが指定した出力テーブルに抽出しま
す
EXTRACT RECORD TO %v_output_table%

COMMENT フィルタリングされたレコードの出力テーブルを開きます
OPEN %v_output_table%

```

一時保存の変数

スクリプトを作成し、数値フィールドの値に基づいて合計を計算します。識別子フィールドの各一意の値の合計はゼロ(0.00)から開始します。

必須の変数

スクリプト入力は、次の3つの目的で変数を使用します。

- ユーザー入力
- 一時ストレージ
- スクリプト構成

ユーザー入力

スクリプトのユーザー入力を取り込むには、"ACCEPT コマンド" ページ 1625を使用して、次の変数のセットを作成します。

変数名	変数の内容
<code>v_input_table</code>	入力として使用する Analytics テーブルの名前
<code>v_output_table</code>	合計の Analytics 出力テーブルの名前
<code>v_id_field</code>	顧客 ID や業者 ID などの識別子値を格納する入力テーブルのフィールド。
<code>v_numeric_field</code>	合計を計算するために使用する入力テーブルの数値フィールド

値の一時保存、およびスクリプト構成

スクリプトがレコード単位で入力テーブルを処理するときに値の一時保存を許可するには、"ASSIGN コマンド" ページ 1660を使用して、次の2つの変数を作成および更新します。必要に応じて、ゼロ(0.00)以外の合計の開始点を構成できます。

変数名	変数の内容
<code>v_id_value</code>	スクリプトが現在処理しているレコードの識別子値
<code>v_running_total</code>	スクリプトが現在処理しているレコードの合計値

スクリプト

```
COMMENT
確認ダイアログボックスを表示せずにファイルの上書きを許可します
END
SET SAFETY OFF

COMMENT
```

```

ダイアログボックスを作成し、入力テーブルと出力テーブルをユーザーに確認します
END
ACCEPT "入力テーブルを選択:" FIELDS "xf" TO v_input_table, "出力テーブル名
を指定(スペースなし):" TO v_output_table
OPEN %v_input_table%

COMMENT
ダイアログボックスを作成し、文字識別子フィールドをユーザーに確認します
END
ACCEPT "文字識別子フィールドを選択:" FIELDS "C" TO v_id_field

COMMENT
ダイアログボックスを作成し、数値フィールドをユーザーに確認します
END
ACCEPT "数値フィールドを選択:" FIELDS "N" TO v_numeric_field

COMMENT
文字識別子フィールドでテーブルのインデックスを作成(並べ替え)します
END
INDEX ON %v_id_field% TO "ID_field_sorted"
SET INDEX TO "ID_field_sorted"

COMMENT
文字識別子フィールドの現在の値を v_id_value 変数に割り当てます。
初めて v_input_table が開かれるときの現在の値はフィールドの最初の値です。その後、
現在の値はスクリプトで現在処理されているレコードによって異なります。
END
ASSIGN v_id_value = %v_id_field%

COMMENT
合計値の開始値をゼロ(0.00)に設定します
END
ASSIGN v_running_total = 0.00

COMMENT
レコード単位でテーブルを処理します。一意の識別子ごとに、数値フィールドの合計を計算し
ます。
END
GROUP IF v_id_value = %v_id_field%
  ASSIGN v_running_total = v_running_total + %v_numeric_field%
  EXTRACT %v_id_field% %v_numeric_field% v_running_total AS "%v_
numeric_field% running total" TO %v_output_table%
ELSE
  ASSIGN v_id_value = %v_id_field%
  ASSIGN v_running_total = 0.00
  ASSIGN v_running_total = v_running_total + %v_numeric_field%

```

```

EXTRACT %v_id_field% %v_numeric_field% v_running_total AS "%v_
numeric_field% running total" TO %v_output_table%
END

COMMENT 計算された合計の出力テーブルを開きます
OPEN %v_output_table%

COMMENT ファイルを上書きする前に確認ダイアログボックスを表示するデフォルトの動作を復
元します
SET SAFETY ON

```

変数]タブで作成および初期化された変数

スクリプトを実行した後は、作成および初期化された変数が **変数]タブ**に表示されます。

ACCEPT コマンドで作成された変数は文字データ型を使用します。ASSIGN コマンドで作成された変数 (`v_id_value`、`v_running_total`) は割り当てられた値のデータ型を使用します。

`v_id_value` および `v_running_total` に割り当てられた値は、スクリプトの進行に従い、継続的に更新されます。**変数]タブ**には、スクリプトが完了する前の最終地が表示されます。

システム変数 (`WRITEn`) と番号体系については、「連続番号が付けられたシステム変数」ページ 1582を参照してください。

変数	データ型	値
OUTPUTFOLDER	C	/Tables
v_id_field	C	Batch
v_id_value	C	99999
v_input_table	C	Demo
v_numeric_field	C	Amount
v_output_table	C	Demo_running_total
v_running_total	N	6,569,918.43
WRITE1	N	500
WRITE3	N	477
WRITE8	N	23

総覧 ログ **変数**

トラブルシューティング

スクリプトで変数を操作するときには、次の2つの種類のエラーが多く発生します。

- **予期しない結果** - スクリプトは最後まで実行されて完了しましたが、変数を含む分析で想定された結果が得られません。
- **データ型の問題** - スクリプトは、「式の型が不一致」、「文字式が必要」などのデータ型に関連するエラーのため、失敗しました。

予期しない結果

初めて発生したエラーの場合には、次のトラブルシューティング手順を試してください。

1. ナビゲーターで **変数** タブを開きます。
2. Analytics のコマンドラインで **DELETE ALL OK** と入力し、既存の変数を削除します。
[コマンドライン] テキスト ボックスが表示されていない場合は、[ウィンドウ > コマンドライン] の順にクリックします。
3. [スクリプト エディター] で [ステップ ] オプションを使用し、スクリプトを 1 行ずつ進め、**変数** タブで各変数と割り当てられた値の作成状況を追跡します。

変数に割り当てられている値が想定している値とは異なり、スクリプトの調整が必要であることが判明する場合があります。詳細については、「スクリプトのテストおよびデバッグ」 ページ 1543を参照してください。

メモ

GROUP、LOOP、または DEFINE FIELD ... COMPUTED などの複数行コマンドの内容をステップ実行することはできません。この状況でスクリプト ロジックをテストするには、周辺のコマンド構文を除いたスクリプト ロジックを一時的に別のスクリプトにコピーします。

データ型の問題

変数を利用するスクリプトを開発するときには、多くの場合、データ型の不一致が原因のエラーが発生します。変換関数と変数代入を同じ式で組み合わせている場合は、特に、エラーが発生する確率が高くなります。

エラーの発生は普通のことであり、エラーを解消するには、別の方法をテストし、次の原則を覚えておく必要があります。

- **式のすべての要素**のデータ型に十分に注意する。式のすべてのデータ型が ACLScript で規定されている方法で一致しているか。
- 変数代入は**絶対にリテラル**である。変数代入でエラーが発生している疑いがある場合は、変数に格納された正確な値を確認し、手動でその値を式に代入します。シンプルなトラブルシューティング手順を実行するだけで、問題が明らかになる場合があります。
- スクリプトの別の場所に DISPLAY VARIABLES コマンドを挿入すると、プロジェクト変数のその時点の状態を取り込むことができます。
- スクリプトを 1 行ずつ進み、ブレークポイントを設定すると、トラブルシューティングが大幅に容易になります。詳細については、「スクリプトのテストおよびデバッグ」 ページ 1543を参照してください。

Analytics コマンドによって作成されたシステム変数

Analytics のダイアログ ボックスで情報を入力する、またはスクリプトを実行することで、特定のコマンドが実行されると、Analytics によってシステム変数が自動的に作成されます。これらの変数とそれに含まれる値は、後続の Analytics コマンドを処理する際に利用できます。

システム変数の値は、同じコマンドを再度実行すると、更新された値に置き換わります。

"Analytics システム変数" 次のページ Analytics によって作成されるシステム変数の一覧。

メモ

システム変数およびそれに含まれる値は、現在の Analytics セッションの間のみ有効です。

変数の現在の値の表示

次の方法のいずれでも、Analytics プロジェクト内のシステム変数を含むすべての変数の名前と現在の値を表示できます。

- ナビゲーターで **変数** タブを選択
- コマンド ラインで「`DISPLAY VARIABLES`」と入力

連続番号が付けられたシステム変数

"Analytics システム変数" 次のページのうち、 n が含まれているシステム変数名の場合、コマンドをコマンドのグループの外部で実行するときには、 n は常に 1 になります。たとえば、TOTAL1 のようになります。

複数のコマンドを実行するのにコマンドのグループを使用する場合、結果となるシステム変数には、その変数を生成したコマンドの行番号に基づいた番号が付けられます。グループ内の最初のコマンドの行番号は 2 となります。

例：

- TOTAL コマンドがグループ内の 3 番目のコマンドである場合、結果は変数 TOTAL4 に格納されます。
- TOTAL コマンドがグループ内で 5 番目のコマンドである場合、結果は変数 TOTAL6 に格納されます。

スクリプトで同じシステム変数の複数のバージョンが必要な場合は、グループを作成します。詳細については、"GROUP コマンド" ページ 1829を参照してください。

Analytics システム変数

次の表は、システム変数を自動的に作成する Analytics コマンドと、変数に割り当てられた値の一覧です。

コマンド	システム変数	割り当てられた値
テーブルまたはファイルを出力するコマンド	WRITE n	出力テーブルまたはファイル内のレコード数
Analytics テーブルを出力するコマンド	OUTPUTFOLDER	<p>出力テーブルを格納する、ナビゲーター内の Analytics プロジェクト フォルダーへのパス。</p> <p>これは DOS 形式のパスで、/フォルダー名/サブフォルダー名の形式を 사용합니다。最初のスラッシュ(/)は 総覧 タブのルート階層を表します。</p> <p>ヒント 別の出力フォルダーを指定する、または出力フォルダーを新規作成する場合は、SET FOLDER コマンドを使用します。</p>
COUNT	COUNT n	集計されたレコード数。
DISPLAY VERSION Analytics のバージョン番号はメジャー、マイナー、パッチという形式で表されます。	ACL_Ver_Major	現在実行している Analytics のメジャーバージョンです。
	ACL_Ver_Minor	現在実行している Analytics のマイナーバージョンです。
	ACL_Ver_Patch	現在実行している Analytics のパッチバージョンです。
	ACL_Ver_Type	現在実行している Analytics のエディションです。 <ul style="list-style-type: none"> ◦ 値が'0'の場合: 非 Unicode 版 ◦ 値が'1'の場合: Unicode 版
DUPLICATES	GAPDUP n	重複の合計数。
EVALUATE	MLE n	金額単位サンプリング 推定誤謬額(推定による虚偽表示) レコード サンプリング 最大誤謬率(計算される上限逸脱率)
	UEL n	金額単位サンプリング 最大誤謬額(虚偽表示上限)
EXECUTE	RETURN_CODE	<p>EXECUTE コマンドを使用した外部アプリケーションまたはプロセス実行によって返されるコード(リターンコード)。</p> <p>リターンコードは外部アプリケーションまたはプロセスによって生成される数値であり、Analytics に返されることで外部プロセスの結果を示します。Analytics はリターンコードを生成しません。</p>

コマンド	システム変数	割り当てられた値
		<p>通常、リターンコードは数値で、特定の通知やエラーメッセージが割り当てられています。たとえば、リターンコード "0" は "処理が正常終了した" ことを示します。リターンコード "2" は "指定されたファイルが見つからない" ことを示します。</p> <p>特定のリターンコードとその意味は外部アプリケーションまたはプロセスによって異なります。リターンコードは "エラーコード" や "終了コード" と呼ばれ、それらの意味は、関連する外部アプリケーションのドキュメントに記載されている可能性があります。リターンコードの一覧はインターネット上でも見ることができます。</p> <p>RETURN_CODE 変数は、EXECUTE コマンドが同期的に使用されているときは作成されますが、コマンドが非同期的に使用されているときは作成されません。</p>
FUZZYDUP	GAPDUP n	あいまい重複グループの合計数。
GAPS	GAPDUP n	ギャップの合計数。
SEQUENCE	WRITE n	順番検査エラーの数。
SIZE	SAMPINT n	必要なサンプリング間隔です。
	SAMPSIZE n	必要なサンプル数です。
STATISTICS	ABS n	指定した最初のフィールドの値の絶対値です。
	AVERAGE n	指定した最初のフィールドの値の平均値です。
	COUNT n	集計されたレコード数。
	HIGH n	<p>指定した最初のフィールドの値のうち、5 番目に高い値です。</p> <p>5 番目に高い値はデフォルト設定です。この設定を変更するには、統計 ダイアログボックスで 高値/低値の数 オプションを使用します。</p>
	LOW n	<p>指定した最初のフィールドの値のうち、5 番目に低い値です。</p> <p>5 番目に低い値はデフォルト設定です。この設定を変更するには、統計 ダイアログボックスで 高値/低値の数 オプションを使用します。</p>
	MAX n	指定した最初のフィールドの値の最大値です。

メモ

複数のフィールドに対して同時に STATISTICS コマンドを実行する場合、システム変数には、指定した最初のフィールドの値が含まれます。

メモ

Analytics で最も高い値が確認される際、重複値は除外されません。たとえば、降順の値リストが 100、100、99、98 の場合、3 番目に高い値は 98 でなく 99 になります。

メモ

Analytics で最も低い値が確認される際、重複値は除外されません。たとえば、昇順の値リストが 1、1、2、3 の場合、3 番目に低い値は 3 でなく 2 になります。

コマンド	システム変数	割り当てられた値
	MEDIAN n	指定した最初のフィールドの値の中央値です。
	MIN n	指定した最初のフィールドの値の最小値です。
	MODE n	指定した最初のフィールドの値のうち、最もよく使用している値です。
	Q25 n	指定した最初のフィールドの値のうち、最初の四分位数値(下位四分位数値)です。
	Q75 n	指定した最初のフィールドの値のうち、3番目の四分位数値(上位四分位数値)です。
	RANGE n	指定した最初のフィールドの値の最大値と最小値の差です。
	STDDEV n	指定した最初のフィールドの値の標準偏差です。
	TOTAL n	指定した最初のフィールドの値の総計です。
TOTAL	TOTAL n	指定した最初のフィールドの値の総計です。
VERIFY	WRITE n	データの妥当性エラーの数。

その他のシステム変数

以下の変数は、コマンドで生成されるのではなく、システムで生成されます。

- OUTPUTFOLDER - 現在の Analytics プロジェクト出力フォルダーです。

インタラクティブ スクリプト の作成

ユーザーに入力を確認するインタラクティブなスクリプトを作成できます。自動で実行される標準スクリプトとは異なり、インタラクティブなスクリプトは、必要な情報がユーザーによって入力されるまで実行を一時停止します。

インタラクティブであることの利点

インタラクティブな場合、適用性が広がり、柔軟なスクリプトを作成できます。あらかじめすべての入力情報を指定する必要はありません。このためには、テーブルやフィールド名などの情報を把握する必要があり、一般的には単一目的で絞り込まれたスクリプトになります。

スクリプト インタラクティブを使用すると、ユーザーがスクリプトを実行するときに、1つ以上のダイアログボックスを使用して、入力情報を収集できます。たとえば、インタラクティブを使用して次の入力のいずれかを収集できます。

- ユーザー名とパスワード
- テーブル名とフィールド名
- ファイル名
- 金額しきい値
- 日付範囲
- マーチャント コード、支店コード、業者および顧客 ID などの識別子
- コマンド パラメーター

インタラクティブの連続

可能であれば、スクリプトの先頭にすべてのインタラクティブ ダイアログ ボックスを置き、以降のスクリプトで中断することなくスクリプトを実行できるようにしてください。

インタラクティブなダイアログボックスがスクリプトの途中で発生した場合、ユーザーは入力が必要なところでスクリプトの実行に参加できないことがあります。入力があるまでスクリプトが停止します。

インタラクティブを作成するには3つの方法があります。

Analytics はスクリプトのインタラクティブを作成するための3つの方法を提供します。各方法は Analytics コマンドに関連付けられています。

ACCEPT および PASSWORD コマンドは ACL スクリプト構文を使用してのみ作成できます。DIALOG コマンドは ACL スクリプト構文を使用して作成できます。あるいは、視覚的ユーティリティのダイアログビルダーを使用して構文を自動生成できます。

コマンド	説明
<p>"ACCEPT コマンド" ページ 1625</p>	<p>ACCEPT コマンドはデフォルトのインタラクティブ ダイアログボックスを作成します。これはユーザー入力の2つの方法をサポートします。</p> <ul style="list-style-type: none"> ○ テキストボックス -は、日付、業者または顧客 ID など、ユーザーが入力する必要がある情報を収集します。 ○ プロジェクト項目一覧 -は、テーブル、フィールド、または変数などの Analytics プロジェクト項目のリストをユーザーに提示します <p>項目のリストはスクリプトが実行される Analytics プロジェクトの内容に基づいて動的に入力されます。</p> <p>項目ごとに個別のダイアログボックスを作成して、一度に1項目について入力を求めることもできますが、1つのダイアログボックスで複数項目について入力を求めることもできます。</p>
<p>"DIALOG コマンド" ページ 1736</p> <p>ダイアログビルダー</p>	<p>DIALOG コマンドはカスタム インタラクティブ ダイアログボックスを作成します。カスタム ダイアログボックスはより高度なレイアウト オプションと、ユーザー入力の5つの方法をサポートします。</p> <ul style="list-style-type: none"> ○ テキストボックス -は、日付、業者または顧客 ID など、ユーザーが入力する必要がある情報を収集します。 ○ チェックボックス -はユーザーに表示されるバイナリ選択肢です。つまり、関連付けられたオプションはオンまたはオフにできます。 ○ ラジオボタン -は相互に排他的なオプションをユーザーに表示します。表示されたオプションのいずれかのみを1度に選択できます。 ○ ドロップダウンリスト -は、選択可能なカスタム、テキストベース オプションの一覧をユーザーに提示します。 ○ プロジェクト項目一覧 -は、テーブル、フィールド、または変数などの Analytics プロジェクト項目のリストをユーザーに提示します <p>項目のリストはスクリプトが実行される Analytics プロジェクトの内容に基づいて動的に入力されます。</p> <p>項目ごとに個別のダイアログボックスを作成して、一度に1項目について入力を求めることもできますが、1つのダイアログボックスで複数項目について入力を求めることもできます。</p>
<p>"PASSWORD コマンド" ページ 2015</p>	<p>PASSWORD コマンドは、パスワードを入力するための単一フィールドがある簡単なダイアログボックスを作成します。</p> <p>パスワードを入力すると、ダイアログボックスには入力した文字がアスタリスク(*)で表示されます。パスワードは Analytics セッション中にメモリに保持されますが、スクリプトまたはログには表示されません。</p>

カスタム ダイアログ ボックスの作成

Analytics **ダイアログビルダー**では、1 つ以上のカスタム ダイアログボックスを作成し、スクリプトの実行中にユーザー入力を収集できます。

カスタム ダイアログボックスを使用して、次のようなさまざまな機能を実行します。

- ユーザーに対し、テーブル名、フィールド名、日付の範囲などを入力するように求める
- ユーザーが複数のオプションから選択できる
- 標準のメッセージ ボックスよりも多くの情報を表示する
- Analytics プロジェクト項目を動的に一覧表示する

メモ

カスタム ダイアログボックスを使用してパスワードを入力することは安全ではありません。代わりに "PASSWORD コマンド" ページ 2015 を使用してください。

ダイアログ ボックス コントロール

ダイアログビルダーでは、ユーザー入力コントロールをダイアログボックスに追加して、カスタム ダイアログボックスを設計および作成できます。

コントロールは小さいインタラクティブなソフトウェアコンポーネントであり、スクリプトによって要求されるユーザー入力を収集するためのさまざまな方法を提供します。1 つ以上のコントロールをダイアログボックスに追加し、ニーズに合わせて設定します。

次のオプションを使用できます。

- テキスト ラベル
- テキスト ボックス
- チェック ボックス
- ラジオ ボタン
- フォロップダウン リスト
- プロジェクト項目一覧

以下のサンプル カスタム ダイアログボックスは各タイプのコントロールの例を提供します。



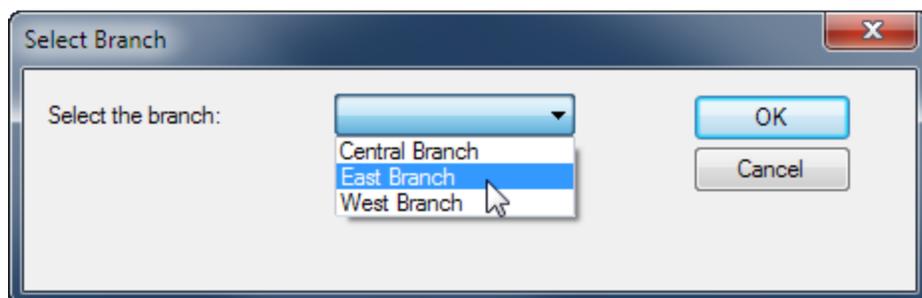
コマンドに自動的に変換されるダイアログボックス

新しく作成されたカスタム ダイアログボックスを保存するときには、Analytics は自動的に視覚的なダイアログボックスを一致する ACLスクリプト `DIALOG` コマンドに変換します。

コマンドは、カーソルが配置されるスクリプトの行に挿入されます。スクリプトを実行するときには、視覚的なダイアログボックスが表示されます。

以下の例は、1つのテキスト ラベルと1つドロップダウン リストの他に、対応する `DIALOG` コマンドがある簡単なダイアログボックスが表示されます。

カスタム ダイアログ ボックス



対応する DIALOG コマンド

```
DIALOG (DIALOG TITLE "Select Branch" WIDTH 464 HEIGHT 116 ) (BUTTONSET TITLE
"&OK;&Cancel" AT 336 12 DEFAULT 1 ) (TEXT TITLE "Select the branch:" AT 12
16 ) (DROPDOWN TITLE "Central Branch;East Branch;West Branch" TO "DROPDOWN1"
AT 156 12 )
```

カスタム ダイアログボックスの作成 - 一般的な手順

1. カスタム ダイアログ ボックスを追加する Analytics スクリプトを開きます。
2. スクリプト内で、`DIALOG` コマンドを挿入したい行にカーソルを配置します。

メモ

空白の行にカーソルを配置します。必要に応じて、空白のスペースを作成します。

3. **新しいダイアログの作成**  をクリックします。
ダイアログビルダーが "ユーザー ダイアログ" というデフォルトのタイトルで表示されます。
4. 次の手順を行って、カスタム ダイアログ ボックスを作成します。
 - **ダイアログビルダー**をダブルクリックして、ダイアログ ボックスのタイトルやサイズを変更します。
ダイアログボックスの幅と高さをピクセルで指定します。また、ダイアログボックスのサイズを変更するには、**ダイアログビルダー**の作業領域の右下端をドラッグします(**グリッドに合わせる**をオンにする必要があります)。
 - **グリッドに合わせる**  をクリックして、**ダイアログビルダー**のグリッドの表示をオンまたはオフにします。

グリッドを使用すると、レイアウト領域のコントロールの位置を調整できます。グリッドをオンにすると、各コントロールの左上隅を、最も近いグリッドポイントに合わせて配置します。

- **ダイアログビルダー**の左側で、コントロールアイコンをクリックしてからレイアウト領域内をクリックすると、そのコントロールがフォーム上に追加されます。

メモ

特定のコントロールを追加および設計するための手順が以下に表示されます。

5. 必要な数のコントロールを追加します。
6. 追加した後にコントロールを修正する必要がある場合は、コントロールをダブルクリックします。
7. **ダイアログビルダー**からコントロールを削除したい場合は、そのコントロールを選択して **削除** をクリックします。 

メモ

OK および **キャンセル** ボタンは削除できませんが、名前は変更できます(以下を参照)。

8. **閉じる** ボタンをクリックして**ダイアログビルダー**を終了します。
9. 変更を保存する場合は確認ダイアログボックスで **OK** をクリックして、終了します。

スクリプト エディターに対応する **DIALOG** のコマンドが表示されます。右にスクロールすると、カスタムダイアログボックスのすべてのコントロールに対応するスクリプト表現を見ることができます。

10. 省略可能。スクリプトの **DIALOG** コマンドで、OK またはキャンセルボタンのラベルテキストを編集します。

一般的に、OK およびキャンセルラベルは編集しないでください。ラベルを変更する場合は、必ず肯定的な値([はい] など) が否定的な値([いいえ] など) の前に来るようにしてください。

ラベルテキストのみを編集します。例: "&はい;&いいえ"

カスタム ダイアログ ボックスの修正

作成した後にカスタムダイアログボックスを修正する必要がある場合は、対応する **DIALOG** コマンドにカーソルを置き、**コマンドの編集**  をクリックします。

テキスト ラベルを追加する

テキスト コントロールを使用して、テキスト ラベルをカスタムダイアログボックスに追加します。

テキストラベルは次のいずれかの目的で使用できます。

- 別のコントロールを特定する
- 通知を提供する
- ユーザーに確認または指示する
- カスタムダイアログボックスで必要な他のテキスト情報を提供する

テキストラベルは表示専用であり、インタラクティブ機能に関連付けられていません。

手順

方法の説明

1. **ダイアログビルダー**  で、**テキスト]**をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。
テキスト]ダイアログボックスが開きます。
2. **ラベル]**フィールドに、カスタムダイアログボックスに表示するテキストを入力します。
 スペースを含め255文字以内で入力します。
3. 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定されたx(水平)およびy(垂直)値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー**をドラッグすることもできます。

4. 省略可能。コントロールのサイズを指定する場合は、**幅]**または**高さ]**フィールドの横にある**自動]**選択をオフにして、ピクセル単位で指定されている値を変更します。
 - **自動選択** -テキスト コントロールは、コントロールに含まれるテキストのサイズに合わせて自動的に調整されます
 - **自動選択** -テキスト コントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダーのサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

5. **左]**、**右]**または**中央]**の選択によって、コントロール内におけるテキストの配置を指定できます。
6. **OK]**をクリックして、コントロールを**ダイアログビルダー**に追加します。

テキストボックスの追加

編集ボックスコントロールを使用して、テキストボックスをカスタムダイアログボックスに追加します。

テキストボックスは、日付、業者または顧客IDなど、ユーザーが入力する必要がある情報を収集します。

編集ボックス変数

編集ボックスコントロールは、ユーザー入力を保存するための文字変数を作成します。

手順

方法の説明

1. **ダイアログビルダー**  で、 **編集ボックス]** をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。
編集ボックス] ダイアログボックスが開きます。
2. 省略可能。 **変数]** フィールドでは、カスタムダイアログボックスでユーザーが入力した値を格納する変数の名前を入力します。
EDITn デフォルト変数名を保持することを選択できます。
3. 省略可能。 **デフォルトテキスト]** フィールドで、テキストボックスのデフォルト入力値を指定します。ユーザーが入力値を指定しない場合は、デフォルト値が使用されます。
4. 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定されたx(水平)およびy(垂直)値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー** をドラッグすることもできます。

5. 省略可能。コントロールのサイズを指定する場合は、 **幅]** または **高さ]** フィールドの横にある **自動]** 選択をオフにして、ピクセル単位で指定されている値を変更します。
 - **自動選択** -編集ボックスコントロールはコントロールに含まれるテキストのサイズに合わせて自動的に調整されます
 - **自動選択** -編集ボックスコントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダー のサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

6. **OK]** をクリックして、コントロールを**ダイアログビルダー** に追加します。

チェックボックスを追加する

チェックボックスコントロールを使用して、チェックボックスをカスタムダイアログボックスに追加します。

チェックボックスはユーザーに表示されるバイナリ選択肢です。つまり、関連付けられたオプションはオンまたはオフにできます。たとえば、チェックボックスを使用すると、ユーザーが従業員テーブルから抽出されたデータに電子メールアドレスフィールドを含めるか除外できます。

オプションの組み合わせ

複数のチェックボックスを使用して、カスタムダイアログボックスのオプションの組み合わせを選択できます。相互に排他的なオプションの場合は、ラジオ ボタンを使用するようにしてください。

チェックボックス変数

チェックボックスコントロールは、ユーザー入力を保存するための論理変数を作成します。チェックボックスがオンの場合には True、チェックボックスがオフの場合には False の値が変数に格納されます。

手順

方法の説明

1. **ダイアログビルダー**  で、**チェックボックス** をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。
チェックボックス ダイアログボックスが開きます。
2. 省略可能。 **変数** フィールドでは、カスタムダイアログボックスでユーザーが入力した値を格納する変数の名前を入力します。
 のデフォルト変数名を保持することを選択できます。
3. **ラベル** フィールドに、チェックボックスのテキストを入力します。
 スペースを含め 255 文字以内で入力します。
4. 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定された x(水平) および y(垂直) 値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー** をドラッグすることもできます。

5. 省略可能。コントロールのサイズを指定する場合は、**幅** または **高さ** フィールドの横にある **自動** 選択をオフにして、ピクセル単位で指定されている値を変更します。
 - **自動選択** -チェックボックスコントロールは、コントロールに含まれるテキストのサイズに合わせて自動的に調整されます
 - **自動選択** -チェックボックスコントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダー のサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

6. **初期状態** の下で、カスタムダイアログボックスが最初に開いたときに、チェックボックスが**オフ**か**オン**かを指定します。

7. [OK]をクリックして、コントロールを**ダイアログビルダー**に追加します。

ラジオボタンの追加

ラジオボタンコントロールを使用して、2つ以上のラジオボタンをカスタムダイアログボックスに追加します。ラジオボタンは相互に排他的なオプションをユーザーに表示します。表示されたオプションのいずれかのみを1度を選択できます。たとえば、2つのラジオボタンを使用して、ユーザーが次のいずれかを選択できるようにします。

- \$5000未満の金額
- \$5000以上の金額

相互に排他的なオプション

複数のラジオボタンを使用すると、ユーザーがカスタムダイアログボックスでさまざまなオプションから1つのみを選択できます。相互に排他的なオプションでない場合は、チェックボックスを使用するようにしてください。

ラジオボタン変数

ラジオボタンコントロールは、ユーザー入力を保存するための数値変数を作成します。この変数の値は、最初のラジオボタンをクリックすると1、2番目のラジオボタンをクリックすると2という形になります。

手順

方法の説明

1. **ダイアログビルダー**  で、**[ラジオボタン]**をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。
[ラジオボタン]ダイアログボックスが開きます。
2. 省略可能。 **[変数]**フィールドでは、カスタムダイアログボックスでユーザーが入力した値を格納する変数の名前を入力します。
[RADIO_n]のデフォルト変数名を保持するを選択できます。
3. **[ラベル]**フィールドで、最初のラジオボタンのテキストを入力し、**[追加]**をクリックします。
スペースを含め255文字以内で入力します。
ラジオボタンは**ラベルリスト**に追加されます。
4. 任意の各追加のラジオボタンのラベルを追加します。
各追加のラジオボタンは**ラベルリスト**の最後に追加されます。

メモ

ラジオボタンコントロールは相互に排他的なオプションを作成するため、2つ以上のラジオボタンが必要です。

5. 省略可能。ラベルリストの最後にラジオボタンを追加する代わりに、次の他のオプションを使用できます。

オプション	説明
挿入	ラベルリストの任意の場所にラジオボタンを挿入できます。 挿入]をクリックする前に、新しいラジオボタンを挿入する場所の真下のリスト項目を選択します。
置換 (置換)	ラベルリストのラジオボタンを置換できます。置換は基本的にラジオボタン名を変更します。 置換]をクリックする前に、新しいラジオボタンで置換するリスト項目を選択します。
削除	ラベルリストからラジオボタンを削除できます。 削除したいリスト項目を選択し、削除]をクリックします。
デフォルトに設定	カスタムダイアログボックスが最初に開くときにデフォルトで選択されるラジオボタンを指定します。 デフォルトに設定するリスト項目を選択し、デフォルトに設定]をクリックします。

6. 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定されたx(水平)およびy(垂直)値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー**をドラッグすることもできます。

7. 省略可能。コントロールのサイズを指定する場合は、幅]または高さ]フィールドの横にある自動]選択をオフにして、ピクセル単位で指定されている値を変更します。

- **自動選択** -ラジオボタンコントロールは、コントロールに含まれるテキストのサイズに合わせて自動的に調整されます
- **自動選択** -ラジオボタンコントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダーのサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

8. **整列]**の下で、カスタムダイアログボックスでラジオボタンに**水平**または**垂直**整列があるかどうかを指定します。
9. **OK]**をクリックして、コントロールを**ダイアログビルダー**に追加します。

ドロップダウンリストを追加する

ドロップダウンリスト コントロールを使用して、ドロップダウンリストをカスタム ダイアログボックスに追加します。

ドロップダウンリストは、選択可能なカスタム、テキストベース オプションの一覧をユーザーに提示します。ユーザーは1度に1つのオプションのみを選択できます。たとえば、ドロップダウンリストを使用して、ユーザーが次の項目を選択できるようにします。

- 年の月
- カテゴリ
- 会社の部署または支店

相互に排他的なオプション

ドロップダウンリストのオプションは相互に排他的です。ラジオボタンを使用して、類似した結果を実現できますが、項目数が多いリストの場合、ドロップダウンの方がよりコンパクトになり、使用しやすくなります。

ユーザーが1度に複数のオプションを選択できるようにする場合は、代わりにチェックボックスを使用します。

ドロップダウンリスト 変数

ドロップダウンリスト コントロールは、ユーザー入力を保存するための文字変数を作成します。

手順

方法の説明

1. **ダイアログビルダー**  で、**[ドロップダウン リスト]**をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。
[ドロップダウン リスト]ダイアログボックスが開きます。
2. 省略可能。 **変数** フィールドでは、カスタム ダイアログボックスでユーザーが入力した値を格納する変数の名前を入力します。
DROPDOWNn のデフォルト 変数名を保持することを選択できます。
3. **ラベル** フィールドで、最初のドロップダウン リスト項目のテキストを入力し、 **[追加]** をクリックします。
スペースを含め255文字以内で入力します。
リスト項目はラベルリストに追加されます。
4. 任意の各追加のリスト項目のラベルを追加します。
各追加のリスト項目はラベルリストの最後に追加されます。

メモ

ドロップダウン リスト コントロールは相互に排他的なおプッシュのを作成するため、2 つ以上のリスト項目が必要です。

5. 省略可能。ラベルリストの最後にリスト項目を追加する代わりに、次の他のオプションを使用できます。

オプション	説明
挿入	ラベルリストの任意の場所にリスト項目を挿入できます。 [挿入]をクリックする前に、新しい項目を挿入する場所の真下でリスト項目を選択します。
置換 (名前の変更)	ラベルリストのリスト項目を置換できます。置換は基本的にリスト項目名を変更します。 [置換]をクリックする前に、新しい項目で置換するリスト項目を選択します。
削除	ラベルリストからリスト項目を削除できます。 削除したいリスト項目を選択し、[削除]をクリックします。
デフォルトに設定	カスタム ダイアログボックスが最初に開くときにデフォルトで選択されるリスト項目を指定します。 デフォルトに設定するリスト項目を選択し、[デフォルトに設定]をクリックします。

6. 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定されたx(水平)およびy(垂直)値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー**をドラッグすることもできます。

7. 省略可能。コントロールのサイズを指定する場合は、**幅**または**高さ**フィールドの横にある**自動**選択をオフにして、ピクセル単位で指定されている値を変更します。

- **自動選択** -ドロップダウン リスト コントロールは、コントロールに含まれるテキストのサイズに合わせて自動的に調整されます
- **自動選択** -ドロップダウン リスト コントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダーのサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

8. **OK**]をクリックして、コントロールを**ダイアログビルダー**に追加します。

プロジェクト項目一覧を追加する

プロジェクト項目一覧コントロールを使用して、プログラム項目一覧をカスタムダイアログボックスに追加します。

プロジェクト項目リストには、テーブルなどの Analytics プロジェクト項目が一覧表示されます。項目の一覧は、スクリプトが実行される Analytics プロジェクトの内容に基づいて動的に入力されます。

ユーザーは1度に1つのオプションのみを選択できます。たとえば、プロジェクト項目一覧を使用して、ユーザーが次の項目を選択できるようにします。

- 年のすべてのテーブルから、特定の月のテーブル
- テーブルのすべての数値フィールドから、特定の数値フィールド
- 複数の可能なスクリプトから、特定のスクリプト

プロジェクト項目のカテゴリ

プロジェクト項目一覧を作成するときには、実際のプロジェクト項目を指定しません。プロジェクト項目のカテゴリを指定します。たとえば、スクリプトを実行するときに開いているテーブルですべての文字フィールドのリストをユーザーに表示する場合は、文字フィールドカテゴリを指定します。

指定するカテゴリに基づき、Analytics は動的にカスタムダイアログボックスにプロジェクト項目一覧を入力します。

使用可能なカテゴリ

次のカテゴリを使用できます。

<ul style="list-style-type: none">◦ 文字フィールド◦ 数値フィールド◦ 日付時刻フィールド◦ 論理フィールド	<ul style="list-style-type: none">◦ 文字変数◦ 数値変数◦ 日付時刻変数◦ 論理変数	<ul style="list-style-type: none">◦ テーブル◦ ビュー◦ スクリプト◦ インデックス◦ ワークスペース
-------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------

プロジェクト項目一覧変数

プロジェクト項目一覧コントロールは、ユーザー入力を保存するための文字変数を作成します。

手順

方法の説明

1. ダイアログビルダー  で、[プロジェクト項目一覧]をクリックしてから配置したいレイアウトでクリックします。配置される位置はコントロールの左上隅が基準となっています。

[プロジェクト項目一覧]ダイアログボックスが開きます。

- 省略可能。**変数**フィールドでは、カスタムダイアログボックスでユーザーが入力した値を格納する変数の名前を入力します。

ITEMn デフォルト変数名を保持することを選択できます。

- カテゴリ**ドロップダウンリストで、プロジェクト項目一覧に入れるプロジェクト項目のカテゴリを選択し、**追加**をクリックします。

たとえば、**数値フィールド**を選択する場合は、スクリプトを実行するときに、プロジェクト項目一覧に開いているテーブルのすべての数値フィールドが含まれます。

カテゴリは**カテゴリリスト**に追加されます。

- 省略可能。任意のその他のカテゴリを追加します。
各追加のカテゴリは**カテゴリリスト**の最後に追加されます。

注意

テーブルとフィールド、またはスクリプトと変数などの異なるカテゴリを追加すると、ユーザーが混乱する可能性があります。ベストプラクティスは、文字フィールドと数値フィールドなどの類似したカテゴリのみを追加することです。

- 省略可能。**カテゴリリスト**の最後にカテゴリを追加する代わりに、次の他のオプションを使用できません。

オプション	説明
挿入	カテゴリリスト の任意の場所にカテゴリを挿入できます。 挿入 をクリックする前に、新しいカテゴリを挿入する場所の真下のカテゴリを選択します。
置換	カテゴリリスト でカテゴリを置換できます。 置換 をクリックする前に、新しいカテゴリで置換するカテゴリを選択します。
削除	カテゴリリスト からカテゴリを削除できます。 削除したいカテゴリを選択し、 削除 をクリックします。

- 省略可能。**デフォルト**フィールドで、カスタムダイアログボックスが最初に開くときにデフォルトで選択されるプロジェクト項目を指定します。

たとえば、特定のテーブル名またはフィールド名を指定できます。

メモ

カテゴリ名を指定しないでください。

アンダースコア(`_`)を含むプロジェクト項目のスペリングを正確に入力してください。

- 省略可能。コントロールの正確な位置を指定する場合は、ピクセルで指定されたx(水平)およびy(垂直)値を修正します。

ヒント

コントロールを配置するには、**ダイアログビルダー**をドラッグすることもできます。

8. 省略可能。コントロールのサイズを指定する場合は、**幅]**または **高さ]**フィールドの横にある **自動]**選択をオフにして、ピクセル単位で指定されている値を変更します。
 - **自動選択** -プロジェクト項目一覧コントロールは、コントロールに含まれるテキストのサイズに合わせて自動的に調整されます
 - **自動選択** -プロジェクト項目コントロールは、コントロールに含まれるテキストのサイズに関係なく、指定されたサイズのままです

ヒント

ダイアログビルダーのサイズ変更ハンドルを使用して、コントロールのサイズを変更することもできます。

9. **OK]**をクリックして、コントロールを**ダイアログビルダー**に追加します。

スクリプト エディターのカスタマイズ

スクリプト エディターは、右端で折り返すかどうかを切り替えたり、テキストの色やフォントを変更したり、編集領域の背景色を変更したりして、カスタマイズすることができます。キーワードのオートコンプリートを無効にすることもできます。関数パラメーターのオンスクリーンヘルプを無効にすることはできません。

以下のテキストタイプについてカスタマイズできます。

- デフォルト スタイル- コメントまたは ACLScript キーワード 以外のすべてのスクリプト テキスト
 - コメント スタイル- スクリプトのコメント
 - コマンド スタイル- ACLScript コマンドのキーワード
 - パラメーター スタイル- ACLScript パラメーターのキーワード
 - 関数スタイル- ACLScript 関数のキーワード
1. 右端での折り返しを有効または無効にするには、スクリプト エディターのツールバーにある **右端で折り返す**  をクリックします。
 2. スクリプト エディターの背景色やテキスト スタイルを変更するには、以下の手順を実行します。
 - a. [ツール > オプション] の順にクリックします。
 - b. [アプリケーション フォント] タブをクリックします。

メモ

[アプリケーション フォント] タブの固定幅フォントおよびプロポーショナルフォントの設定は、スクリプト エディター内のフォントだけでなく、アプリケーション全体のフォントに適用されます。

- c. [スクリプト エディターの設定] 領域で、テキスト スタイルまたは背景色を選択し、**色の変更**] をクリックします。
 - d. **色の設定**] ダイアログ ボックスで、**基本色**] 領域または色のパレットから希望の色を選択します。また、希望する色に対応する赤、緑、および青 (RGB) の値がわかっている場合は、それぞれのテキスト ボックスに値を入力します。
 - e. **OK**] をクリックします。
 - f. 選択したスタイルに太字属性を適用するには、**太字**] を選択します。
 - g. 選択したスタイルに斜体属性を適用するには、**斜体**] を選択します。
 - h. **OK**] をクリックします。
- Analytics が最初にインストールされたときに使用した背景色とテキスト スタイルの設定に戻りたい場合は、**オプション**] ダイアログ ボックスの下部にある **既定値に戻す**] ボタンをクリックします。**既定値に戻す**] をクリックすると、スクリプト エディター内の背景色やテキスト スタイルだけでなく、**オプション**] ダイアログ ボックスのすべてのタブにある全オプションをデフォルトの設定に戻します。
3. キーワードのオートコンプリートを無効にする場合は、次の手順を実行します。
 - a. [ツール > オプション] の順にクリックします。

スクリプトの操作

- b. [インターフェイス] タブをクリックします。
- c. [スクリプトのオートコンプリートを無効にする] を選択します。
- d. [OK] をクリックします。

スクリプトのコピ

ある Analytics プロジェクトから別のプロジェクトへスクリプトをコピーすることができます。同時に単一のスクリプト、または複数のスクリプトをコピーできます。

Analytics プロジェクトの外部に個別のファイルとして存在するスクリプトをインポートしたい場合は、「スクリプトのインポート」次のページを参照してください。

1. コピーされたスクリプトを格納するプロジェクトを開きます。
2. ナビゲーターの **総覧** タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダーを右クリックし、**別のプロジェクトからコピー > スクリプト** を選択します。

Analytics プロジェクトは、ツリービューにおける最上位のフォルダーです。

3. **プロジェクト ファイルの場所** ダイアログ ボックスで、スクリプトのコピー元となる Analytics プロジェクトを探して選択し、**開く** をクリックします。
4. **インポート** ダイアログ ボックスで、次のいずれかの作業を行って、1 つ以上のスクリプトを **インポート先 < プロジェクト名 >** リストに追加します。

- スクリプトをダブルクリックする。
複数のスクリプトを
- **Ctrl キー** を押しながらクリックし、右矢印 ボタンをクリックする。
- **すべて追加** をクリックして、すべてのスクリプトを追加する。

インポート先 プロジェクト名 リストからスクリプトを削除するには、個々のスクリプトをダブルクリックするか、複数のスクリプトを **Ctrl キー** を押しながらクリックした後に左矢印 ボタンをクリックするか、または **すべてクリア** をクリックします。

5. **OK** をクリックして、コピー先のプロジェクトにスクリプトをコピーします。

プロジェクト内に同じ名前のスクリプトが既に存在する場合は、コピーしたスクリプト名の末尾に増分数字が付けられます。

スクリプトのインポート

個別の `.aclscript` ファイルとして Analytics プロジェクトの外部に存在するスクリプトをインポートすることができます。一度にインポートできるスクリプトは1つだけです。

別の Analytics プロジェクトからスクリプトをインポートしたい場合は、「スクリプトのコピ」前のページを参照してください。

1. ナビゲーターの **総覧** タブで、Analytics プロジェクトのエントリまたはプロジェクト フォルダを右クリックし、**プロジェクト項目のインポート > スクリプト** を選択します。

Analytics プロジェクトは、ツリービューにおける最上位のフォルダです。

2. **プロジェクト** ダイアログボックスで、スクリプト ファイル (`.aclscript`) を探して選択し、**開く** をクリックします。
3. 確認ダイアログボックスで、**OK** をクリックします。

スクリプトがプロジェクトにインポートされます。プロジェクト内に同じ名前のスクリプトが既に存在する場合は、インポートしたスクリプト名の末尾に増分数字が付けられます。

ScriptHub からのインポート

ScriptHub は、Diligent の社員と Diligent ユーザーコミュニティによって開発された Analytics スクリプト項目の Web ライブラリです。サブスクリプションの一部として、ScriptHub のコンテンツをダウンロードおよび使用できます。

ScriptHub の内容：

- アナリティクス
- データをインポート、準備、または分析するスクリプト
- コード スニペット

ScriptHub には次の場所からアクセスできます。

- ACL for Windows
- Launchpad (www.highbond.com)
- ScriptHub のホームページ (scripts.highbond.com)

ScriptHub から Analytics にコンテンツをインポートする

以下の方法のいずれかを使用して、ScriptHub から Analytics にコンテンツをインポートできます。

Analytics から ScriptHub にアクセスする

メモ

Analytics バージョン 12(以上)が必要です。

1. Analytics プロジェクトを開きます。
2. ナビゲーターで最上位のプロジェクト項目をクリックし、**ScriptHub からインポート**]を選択します。
3. 必要に応じて、HighBond アカウントを使用して ScriptHub にサインインします。
4. ScriptHub で、インポートしたい項目を検索し、**詳細の表示**]をクリックします。

メモ

スクリプト詳細の内容を読んだことを確認します。これには、前提条件、データ要件、制限事項に関する重要な詳細が記載されています。

5. スクリプト ファイル パネルの上で、**すべてのスクリプト ファイルのダウンロード** ]をクリックします。ScriptHub コンテンツがダウンロードされ、ナビゲーターに表示されます。

ScriptHub に直接アクセスする

メモ

Analytics バージョン 11.4(以上)が必要です。

1. HighBond アカウントを使用して ScriptHub (scripts.highbond.com) にサインインします。
2. ScriptHub で、インポートしたい項目を検索し、**詳細の表示**]をクリックします。

メモ

スクリプト詳細の内容を読んだことを確認します。これには、前提条件、データ要件、制限事項に関する重要な詳細が記載されています。

3. **ScriptHub ID** テキスト ボックスに表示される ID をコピーします。
4. Analytics プロジェクトで、次のいずれかを実行します。
 - アナリティクスまたはスクリプトをインポートする場合は、新しいスクリプトを作成します。
 - コード スニペットをインポートする場合は、スニペットを貼り付ける既存のスクリプトを開きます。
5. **スクリプト エディター** ツールバーにある **ScriptHub アクセス**]  をクリックして、**ScriptHub コンテンツリンクの貼り付け**] ダイアログ ボックスを表示します。
6. **貼り付け**] をクリックして、ScriptHub ID をダイアログ ボックスに貼り付けます。
7. **完了**] をクリックします。

ScriptHub コンテンツがダウンロードされ、**ナビゲーター**に表示されます。

メモ

コード スニペット は開いているスクリプトに挿入されます。アナリティクスおよびスクリプトは、個別のスクリプトとして**ナビゲーター**に表示されます。この状態で、項目をインポートするために使用した空のスクリプトを削除できます。

スクリプトのエクスポート

スクリプトは、Analytics プロジェクトの外部に保存される個別の **.aclscript** ファイルとしてエクスポートすることができます。別ファイルとしてエクスポートされたスクリプトは、後で任意の Analytics プロジェクトにインポートできます。一度にエクスポートできるスクリプトは1つだけです。

1. ナビゲーターの **総覧** タブでスクリプトを右クリックし、**プロジェクト項目のエクスポート** を選択します。
2. **名前を付けて保存** ダイアログボックスで、スクリプトを保存する場所を選択し、必要であればスクリプトの名前を変更して、**保存** をクリックします。確認ダイアログボックスで **OK** をクリックします。

指定した場所にスクリプトがエクスポートされます。

メモ

スクリプト名を64文字の英数字(**.aclscript** 拡張子を含まない)に制限し、スクリプトが Analytics にインポートされるときに名前が切り捨てられないようにします。

名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。先頭に数字を使用することもできません。特殊文字、スペース、先頭の数字はすべて、スクリプトのインポート時にアンダースコア文字に置換されます。

テキストの検索と置換

Analytics スクリプト内で、文字列を検索し、さらにそれを置換することもできます。

1. **スクリプト エディター**内の、文字列の検索を開始する位置でクリックします。
スクリプト全体を検索する場合は、先頭行の先頭(最左端)でクリックします。
2. 右クリックして **データ検索**]を選択します。
3. **置換**]ダイアログボックスで、次の情報を入力します。
 - **検索対象**-検索する文字列または単語を指定します。
 - **置換後の文字列** -(任意) 検索した文字列を別の文字列に置き換える場合は、その置換後の文字列を指定します。
 - **単語単位で探す** - 検索する単語と厳密に一致する値のみを検索の対象とします。たとえば、"int" という文字列を検索する場合、デフォルトでは "integer" も一致する文字列となりますが、このオプションを適用すると、"int" のみが検索され、この文字列以外の長さの文字列は無視されます。
 - **大文字と小文字を区別する** - このオプションを選択すると、検索で大文字と小文字を区別します。たとえば、"Integer" という文字列を検索する場合、デフォルトでは "Integer" と "integer" が一致する文字列となりますが、このオプションを適用すると、"Integer" のみが検索されます。
4. 次のタスクのいずれかを完了します。
 - **次を検索**]をクリックすると、最初に一致した文字列の場所に移動します。もう一度クリックすると、次に一致した場所に移動します。このボタンを繰り返しクリックしていくと、一致する文字列の場所を順に移動することができます。
 - **置換して次に]**をクリックすると、現在強調表示されている一致文字列を **置換後の文字列]**の値に置き換えます。
 - **すべて置換]**をクリックすると、一致するすべての文字列を **置換後の文字列]**の値に置き換えます。完了すると、置換を行った数が表示されます。

コマンドの概要

ACLScript コマンドは、一般に範囲が広いデータに対して処理を実行します。

たとえば、SUMMARIZE コマンドはフィールドの同一の値に基づいてレコードをグループ化し、各グループの小計および統計値を計算します。

多くのコマンドは結果を新しい Analytics テーブルに出力します。他のコマンドはさまざまなアプリケーションタスクを実行します。

次のページに、Analytics で使用可能なコマンドを網羅したリストをカテゴリ別に列挙します。

- "インポートとエクスポート" ページ 1613
- "データのプロファイルと検証" ページ 1615
- "データの並べ替え" ページ 1616
- "データのグループ化" ページ 1616
- "データの結合" ページ 1617
- "サンプル データ" ページ 1618
- "機械学習" ページ 1619
- "フィールド、レコード、テーブル" ページ 1619
- "ユーザー操作と一般的なスクリプト" ページ 1621
- "HighBond API 要求" ページ 1622
- "レポート" ページ 1623
- "ファイルとシステム" ページ 1623

表記規則と使用方法

コマンド名の省略

注意

ACL は、スクリプトのコマンド名を省略せず、各名前前の完全なバージョンを使用することをお勧めします。

省略により、スクリプトが読み取りにくく、理解が困難になります。完全なコマンド名なしでは、オンラインヘルプでのコマンドの検索が難しくなります。

スクリプトが省略に精通していない人によって修正または継承される場合、特に省略が問題になります。

スクリプトのコマンドを指定するときには、名前を省略できます。コマンド名から十分な先頭文字を含め、すべての Analytics コマンドでコマンドを一意に特定する必要があります。

例：

- EXT は EXTRACT コマンドを一意に特定し、有効な略語にします。
- EX は EXTRACT コマンドを一意に特定しないため、エラーメッセージを生成します。



コマンドが一意に特定される場合は、必要に応じて略語を短くすることができます。

たとえば、次のすべての略語は OPEN コマンドで有効です。

- OPE
- OP
- O

メモ

略語が短くなると、他のユーザーが認識することが難しくなります。

ACL コマンドのパラメーターの順序

メモ

スクリプトの作成において、Analytics のスクリプト作成者は、Analytics のユーザーインターフェイスからコマンドを実行したときにコマンド ログに現れる順序と同じ順序でパラメーターを指定するようにしてください。

多くの Analytics コマンドは柔軟にパラメーターの順序を指定できます。たとえば、次に示す同じ CLASSIFY コマンドの 3 つのバリエーションはすべて同じ処理を実行し、すべて正しく実行されます。

```
CLASSIFY ON CUSTNO SUBTOTAL AMOUNT IF AMOUNT >= 100 TO "Classify_1.FIL" OPEN
APPEND KEY CODES STATISTICS
```

```
CLASSIFY ON CUSTNO SUBTOTAL AMOUNT KEY CODES IF AMOUNT >= 100 TO "Classify_
1.FIL" OPEN APPEND STATISTICS
```

```
CLASSIFY ON CUSTNO IF AMOUNT >= 100 SUBTOTAL AMOUNT STATISTICS KEY CODES TO
"Classify_1.FIL" APPEND OPEN
```

一部のコマンドでは、1つ以上のパラメーターが特定の順序で指定される必要があります。必要な順序はこれらのコマンドのトピックを参照してください。

メモ

コマンドのパラメーターの物理的な順序は、Analytics がパラメーターを処理する順序には影響しません。たとえば、範囲パラメーター(ALL、FIRST、NEXT、WHILE) は、パラメーターの相対的な位置に関係なく、IF パラメーターの前に適用されます。

コマンドドキュメントで使用される規則

表記	使用目的
大文字	<p>ACLScript のキーワード。</p> <p>一般的な構文セクションにおいて、山かっこ(<code>< ></code>) で囲まれていないキーワードは、必須の構文項目です。</p> <p>メモ Analytics のどのドキュメントでも、キーワードであるコマンドおよびパラメーターは大文字で記載されていますが、これはあくまでも書式設定の表記規則上、そう記載されているにすぎません。Analytics 自体では、キーワードを大文字で入力する必要はありません。</p>
斜体	ユーザーが指定するコマンド パラメーター。
 (縦棒)	山かっこまたは波かっこで囲まれた構文項目の選択肢を区切ります。項目のうち1つだけを使用できます。
<code>< ></code> (山かっこ)	オプションの構文項目。山かっこを入力しないでください。
<code>{ }</code> (波かっこ)	必須の構文項目。波かっこを入力しないでください。
<code><, ... n ></code>	前の項目が <i>n</i> 回繰り返されることを示します。出現はカンマによって区切られます。
<code><... n ></code>	前の項目が <i>n</i> 回繰り返されることを示します。出現は空白によって区切られます。
[ラベル] ::=	<p>構文のブロックの名前。</p> <p>この規則は、複数の場所で使用される可能性がある長い構文セクションや構文単位をグループ化してラベル(名前)を付ける際に使用します。構文のブロックが使用される場所は角かっこで囲んだラベルで示されます。例:[フィールド 構文]</p>

インポートとエクスポート

インポート コマンドでは、さまざまなデータソースからデータをインポートできます。

データソースによっては、インポートの一部として、ソースデータを定義します。データの定義は、フィールド名、フィールド長、フィールド データ型などの属性を指定することを意味します。

エクスポート コマンドでは、データをさまざまなファイル形式または HighBond のリザルトにエクスポートできます。

Analytics によるデータ アクセスは読み取り専用です

データソースに接続するとき、またはデータソースからインポートするときには、Analytics は厳密に読み取り専用です。Analytics はデータソースのデータを追加、更新、削除できません。また、いかなる方法でも、データソースを修正することはできません。この制限は、ファイルベースのデータソース、データベース、クラウド データ サービスといった Analytics がアクセスできるすべてのデータソースに適用されます。

インポートされたデータから作成される Analytics データ ファイル (.fil) も Analytics によって読み取り専用として処理されます。データソースからファイルを更新する以外に、Analytics は .fil ファイルを修正できません。

.fil ファイルは作成するために使用されるデータソースとは完全に別です。.fil ファイルを削除すると、データソースに影響しません。

コマンド説明

コマンド	説明
ACCESSDATA	さまざまな ODBC 準拠データソースからデータをインポートします。 64 ビットと 32 ビットのどちらの ODBC ドライバーを使用するかによって、このコマンドは ACCESSDATA64 または ACCESSDATA32 という形を取ります。
DEFINE TABLE DB	AX コネクタを使用してデータベーステーブルに接続し、Analytics サーバー テーブルを定義します。Microsoft SQL Server、Oracle、DB2 データベースに接続することができます。
EXPORT	指定されたファイル形式に、または HighBond 内のリザルト アプリまたはロボット アプリに、Analytics からデータをエクスポートします。
IMPORT ACCESS	Microsoft Access データベース ファイルを定義およびインポートして、Analytics テーブルを作成します。
IMPORT DELIMITED	区切り文字付きテキスト ファイルを定義およびインポートして、Analytics テーブルを作成します。
IMPORT EXCEL	Microsoft Excel ワークシートまたは名前付き範囲を定義およびインポートして、Analytics テーブルを作成します。
IMPORT GRCPROJECT	HighBond プロジェクト テーブルをインポートして、Analytics テーブルを作成します。
IMPORT GRCRESULTS	HighBond のリザルト テーブルまたは解釈をインポートして、Analytics テーブルを作成します。
IMPORT MULTIDELIMITED	複数の区切り文字付きファイルを定義およびインポートすることで、複数の Analytics テーブルを作成します。

コマンド	説明
IMPORT MULTIEXCEL	複数の Microsoft Excel ワークシートまたは名前付き範囲を定義およびインポートして、複数の Analytics テーブルを作成します。
IMPORT ODBC	ODBC データソースからデータを定義およびインポートし、Analytics テーブルを作成します。 ODBC は Open Database Connectivity であり、データベースにアクセスするための標準的な方法です。
IMPORT PDF	Adobe PDF ファイルを定義およびインポートして、Analytics テーブルを作成します。
IMPORT PRINT	印刷イメージ(レポート) ファイルを定義およびインポートして、Analytics テーブルを作成します。
IMPORT SAP	Direct Link を使用して、SAP システムからデータをインポートし、Analytics テーブルを作成します。
IMPORT XBRL	XBRL ファイルを定義およびインポートして、Analytics テーブルを作成します。
IMPORT XML	XML ファイルを定義およびインポートして、Analytics テーブルを作成します。
RETRIEVE	バックグラウンド処理に送信された Direct Link クエリの結果を取得します。

データのプロフィールと検証

プロフィールコマンドでは、レコードをカウントし、数値フィールドを合計し、データの統計プロフィールを作成できます。

検証コマンドは、データセットの整合性を検査するためのさまざまな方法を提供します。たとえば、データの有効性、データの順番、および重複を検査できます。

コマンド説明

コマンド	説明
BENFORD	フィールドに表示される最初の桁(1 ~ 9) や最初の桁の組み合わせの数をカウントし、実数と予測数を比較します。予測数はベンフォードの法則によって計算されます。
COUNT	現在のビュー内のレコードの合計数や、指定した条件を満たすレコード件数をカウントします。
DUPLICATES	重複値または重複レコードが Analytics テーブル内に存在するかどうかを検出します。
FUZZYDUP	文字フィールドでほぼ同一の値(あいまい重複)を検出します。
GAPS	Analytics テーブルの数値または日付時刻フィールドに1つ以上のギャップが連続して含まれるかどうかを検出します。
OUTLIERS	数値型フィールドにおける統計上の異常値を検出します。異常値の検出対象は、数値型フィールド

コマンド	説明
	全体であるか、あるいは1つまたは複数の文字型、数値型、または日付時刻型キーフィールドの値に基づく複数のグループです。
PROFILE	Analytics テーブルで1つまたは複数の数値フィールドまたは数値式について要約統計値を生成します。
SEQUENCE	Analytics テーブル内の1つ以上のフィールドが順番どおりに整列されているかどうかを確認したり、順序が正しくない項目を識別したりします。
STATISTICS	Analytics テーブルで、1つ以上の数値または日付時刻フィールドについて統計計算します。
TOTAL	Analytics テーブルの1つ以上のフィールドの合計値を計算します。
VERIFY	Analytics テーブル中のデータがテーブルレイアウトのフィールド定義と一致していることを確認することにより、1つ以上の Analytics テーブルのデータの妥当性エラーをチェックします。

データの並べ替え

並べ替えコマンドとしては、Analytics のレコードを異なる方法で並べ替えることができる、2つのコマンドがあります。[INDEX](#) コマンドは一時的に既存のテーブルを並べ替えます。[SORT](#) コマンドは、物理的に並べ替えられたレコードで新しいテーブルを生成します。

コマンド説明

コマンド	説明
INDEX	物理的ではなく連続的な順番でレコードへのアクセスを可能にする、Analytics テーブルのインデックスを作成します。
SORT	Analytics テーブルのレコードを、指定されたキーに基づいて昇順または降順に並べ替えます。結果は新しい物理的に並べ替えられた Analytics テーブルに出力されます。

データのグループ化

グループコマンドでは、同じ値または類似した値に基づいてレコードをグループ化できます。コマンドによって、テキスト値、数値、または日付、あるいはこれらの種類の組み合わせをグループ化できます。

コマンド説明

コマンド	説明
AGE	日付または日付時刻フィールドの値に基づいて、レコードを年齢調べ間隔でグループ化します。各期間のレコード数をカウントし、指定した数値フィールドの小計を期間ごとに求めます。
CLASSIFY	文字または数値フィールドの等しい値に基づいてレコードをグループ化する各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。
CLUSTER	1つ以上の数値フィールドの類似した値に基づいて、レコードをクラスターにグループ化します。クラスターは単次元または多次元です。
CROSSTAB	2つ以上の文字または数値フィールドの値の等しい組み合わせに基づいてレコードをグループ化し、結果のグループを行と列のグリッドに表示します。各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。
HISTOGRAM	文字フィールドまたは数値フィールドの値に基づいてレコードをグループ化し、各グループ内のレコード数をカウントして、グループとカウントを棒グラフで表します。
OUTLIERS	数値型フィールドにおける統計上の異常値を検出します。異常値の検出対象は、数値型フィールド全体であるか、あるいは1つまたは複数の文字型、数値型、または日付時刻型キーフィールドの値に基づく複数のグループです。
STRATIFY	数値フィールドの値に基づいて、レコードを数値間隔でグループ化します。各間隔のレコード数をカウントし、指定した数値フィールドの小計を間隔ごとに求めます。
SUMMARIZE	1つ以上の文字フィールド、数値フィールド、または日付時刻フィールドの同じ値に基づいて、レコードをグループ分けします。各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。

データの結合

データの結合コマンドは、Analytics 内でデータを結合するためのいくつかの方法を提供します。Analytics でのデータの結合の概要については、「データの結合」ページ 910を参照してください。

コマンド説明

コマンド	説明
APPEND	2つ以上の Analytics テーブルからのレコードを結合するには、それらのレコードを新しい Analytics テーブルに追加します。
DEFINE RELATION	2つの Analytics テーブル間の関連付けを定義します。

コマンド	説明
EXTRACT	Analytics テーブルからデータを抽出し、それを新しい Analytics テーブルに出力するか、または既存の Analytics テーブルへ追加します。レコード全体または選択したフィールドを抽出することができます。
FUZZYJOIN	曖昧一致を使用して、2 つの Analytics テーブルのフィールドを組み合わせることで 1 つの新しい Analytics テーブルにします。
JOIN	2 つの Analytics テーブルのフィールドを連結して 1 つの新しい Analytics テーブルにします。
MERGE	同一の構造を持つ 2 つの並べ替え済み Analytics テーブルのレコードを結合して、元のテーブルと同じ並べ替え順になっている新しい Analytics テーブルに出力します。

サンプル データ

Analytics には、次の 3 つのサンプリングの種類があります。

- レコード サンプリング(属性 サンプリング)
- 金額単位 サンプリング
- 従来の変数 サンプリング

選択するサンプリングの種類は、行う分析の性質とデータの性質によって決まります。

使用するサンプリングの種類に関する指針については、「データのサンプリング」ページ 1021 を参照してください。

サンプリング コマンド の 順序

サンプルコマンドは、特定の順番で使用されるように設計されています。

従来の変数 サンプリング の 順番

1. `CVSPREPARE` - 母集団を階層化し、各層の適切なサンプルサイズを計算します。
2. `CVSAMPLE` - レコードのサンプルを抽出します。
3. `CVEVALUATE` - レコードの母集団全体に対して、サンプルで見つかった誤謬を推定します。

金額単位 のサンプリングまたはレコードのサンプリングの順番

1. `SIZE` - 適切なサンプルサイズを計算します。
2. `SAMPLE` - レコードのサンプルを抽出します。
3. `EVALUATE` - レコードの母集団全体に対して、サンプルで見つかった誤謬を推定します。

コマンド説明

コマンド	説明
CVSPREPARE	母集団を階層化し、各層の統計的に有効なサンプルサイズを従来の変数サンプリングのために計算します。
CVSSAMPLE	従来の変数サンプリング方法を使用して、レコードのサンプルを抽出します。
CVSEVALUATE	従来の変数サンプリングのために、サンプル分析の結果を母集団全体に対して推定するための4つの方法があります。
SIZE	金額単位のサンプリングまたはレコードのサンプリングに対し、統計的に有効なサンプルサイズおよびサンプル間隔を計算します。
SAMPLE	レコード サンプリングまたは金額単位 サンプリング方法を使用して、レコードのサンプルを抽出します。
EVALUATE	レコード サンプリングまたは金額単位 サンプリングでは、サンプリングされたデータで決定された誤謬を基に母集団全体を予測し、逸脱率の上限または誤謬額を計算します。

機械学習

機械学習コマンドを使用すると、ラベルがないデータにおけるクラスまたは数値を予測したり、パターンを発見したりすることができます。

コマンド説明

コマンド	説明
CLUSTER	1つ以上の数値フィールドの類似した値に基づいて、レコードをクラスターにグループ化します。クラスターは単次元または多次元です。
TRAIN	自動的な機械学習を使用して、トレーニングデータセットに対する最適の予測モデルを作成します。
PREDICT	予測モデルをラベルがないデータセットに適用し、個別のレコードに関連付けられたクラスまたは数値を予測します。

フィールド、レコード、テーブル

このグループのコマンドは、Analytics でデータを整理および表示するために使用される中心的な項目であるフィールド、レコード、またはテーブルに対して、さまざまな処理を実行します。

コマンド説明

コマンド	説明
ACTIVATE	Analytics ワークスペースに保存されたフィールド定義を Analytics テーブルレイアウトのフィールド定義の既存のセットに追加します。
CREATE LAYOUT	特定のスクリプト作成状況で必要になる場合がある、空の Analytics テーブルレイアウトを作成します。
DEFINE COLUMN	1 つ以上の列を作成し、既存のビューに追加します。
DEFINE FIELD	Analytics テーブルレイアウトで物理データフィールドを定義します。
DEFINE FIELD...COMPUTED	Analytics テーブルレイアウトの演算フィールドを定義します。
DEFINE REPORT	新しいビューを作成するか、既存のビューを開きます。
DEFINE VIEW	新しいビューの定義または既存のビューを上書します。
EXTRACT	Analytics テーブルからデータを抽出し、それを新しい Analytics テーブルに出力するか、または既存の Analytics テーブルへ追加します。レコード全体または選択したフィールドを抽出することができます。
FIELDSHIFT	テーブルレイアウトでフィールド定義の開始位置を変更します。
FIND	インデックス付き文字フィールドで、指定した文字列と一致する最初の値を検索します。
IMPORT LAYOUT	外部のテーブルレイアウト ファイル(.layout) を Analytics プロジェクトにインポートします。
LIST	Analytics テーブルの 1 つ以上のフィールドのデータを、列で書式設定されたディスプレイに出力します。
LOCATE	指定された値または条件に一致する最初のレコードを検索するか、または指定されたレコード番号に移動します。
NOTES	Analytics テーブル内の個々のレコードに関連付けられている注釈(ノート)を作成、編集、または削除します。
OPEN	Analytics テーブルおよび関連するデータファイルを開きます。
REFRESH	関連するデータソースを基に、Analytics テーブル内のデータを更新します。
SAVE	Analytics テーブルのコピーを作成してそれを別の名前でも保存したり、Analytics プロジェクトを保存します。
SAVE LAYOUT	Analytics テーブルレイアウトを外部のテーブルレイアウト ファイル(.layout) に保存するため、またはテーブルレイアウトのメタデータを Analytics テーブルに保存します。

コマンド	説明
SAVE TABLELIST	Analytics プロジェクト のすべてのテーブルのリストを Analytics テーブルまたは CSV ファイルに保存します。
SAVE WORKSPACE	ワークスペースを作成して保存します。
SEEK	インデックス付き文字フィールドで、指定した文字式または文字列と一致する最初の値を検索します。
TOP	Analytics テーブルの最初のレコードに移動します。

ユーザー操作と一般的なスクリプト

ユーザー操作と一般的なスクリプト コマンドにより、Analytics スクリプト 作成者は、スクリプト の動作を構造化および制御するためのコマンドを使用できます。

コマンド説明

コマンド	説明
ACCEPT	1 つ以上のスクリプト 入力値をインタラクティブにユーザーに確認するダイアログ ボックスを作成します。各入力値は名前付き文字変数に格納されます。
ASSIGN	変数を作成し、変数に値を割り当てます。
CALCULATE	1 つ以上の式の値を計算します。
CLOSE	Analytics のテーブル、インデックス ファイル、またはログ ファイルを閉じ、 スクリプト 記録 セッションを終了します。
COMMENT	処理に影響を与えずに、スクリプト に注釈を追加します。
DELETE	Analytics プロジェクト 項目、テーブルレイアウト のフィールド、変数、1 つ以上のテーブル履歴エントリ、テーブル間の関係、または Windows フォルダーのファイルを削除します。ビューの列も削除します。
DIALOG	1 つ以上のスクリプト 入力値をインタラクティブにユーザーに確認するカスタム ダイアログ ボックスを作成します。各入力値は名前付き変数に格納されます。
DO SCRIPT	ACL スクリプト 内から第 2 のまたは外部スクリプト を実行します。
ESCAPE	Analytics を終了することなく、処理中のスクリプト、あるいはすべてのスクリプト を終了します。
EXECUTE	Analytics の外部にあるアプリケーションやプロセスを実行します。Windows の Run コマンドをエミュレートします。Windows のコマンド プロンプトを操作するために使用できます。
GROUP	テーブルの次のレコードに移動する前に、テーブルを 1 回通過するだけで、レコードに対して 1 つ以上

コマンド	説明
	の ACLScript コマンドを実行します。コマンド実行は条件によって制御できます。
IF	コマンドを実行するために true と評価する必要のある条件を指定します。
LOOP	指定された条件が真と評価されている間、一連の ACLScript コマンドをレコードで繰り返し実行します。
NOTIFY	受信者の中継する SMTP メールサーバーに電子メール通知メッセージを送信します。
PASSWORD	スクリプトの実行中にパスワードの入力をユーザーに求めるために、パスワード値を含まないパスワード定義を作成します。
PAUSE	スクリプトを一時停止し、ユーザーのダイアログボックスに情報を表示します。
RCOMMAND	Analytics テーブルを外部 R スクリプトにデータフレームとして渡し、外部 R スクリプトからの出力を使用して、Analytics プロジェクトで新しいテーブルを作成します。
RENAME	Analytics プロジェクト項目またはファイルの名前を変更します。
SET	構成可能な Analytics オプションを設定します。

HighBond API 要求

HighBond API 要求コマンドは HTTP 要求をアナリティクス スクリプトから HighBond API に送信します。回答は *.json ファイルで返されます。

スクリプト化された HighBond API 要求を使用して、ACL スクリプトと HighBond プラットフォームオブジェクトの間の自動連携を構築します。このスクリプトは Analytics でもロボットでも実行できます。

利用可能な HighBond API 要求およびエンドポイントの詳細については、[HighBond API リファレンス](#)を参照してください。

メモ

HighBond API 要求コマンドは HighBond API でのみ使用できます。その他の API では機能しません。

これらのコマンドは ACL スクリプトでのみサポートされています。Analytics のユーザーインターフェイスには同等のコマンドはありません。

コマンド説明

コマンド	説明
HB_API_DELETE	HighBond API に DELETE 要求を送信します。

コマンド

コマンド	説明
HB_API_GET	HighBond API に GET 要求を送信します。
HB_API_PATCH	HighBond API に PATCH 要求を送信します。
HB_API_POST	HighBond API に POST 要求を送信します。
HB_API_PUT	HighBond API に PUT 要求を送信します。

レポート

レポート コマンドでは、基本 Analytics レポートを書式設定、生成、および印刷できます。

コマンド説明

コマンド	説明
DO REPORT	指定された Analytics レポートを生成します。
PRINT	テキスト ファイル、Analytics ログ ファイル、その他外部ファイルとしてエクスポートされた Analytics プロジェクト項目 (スクリプト (.acscript)、テーブル (.layout)、ワークスペース (.wsp) など) を印刷します。コマンドで生成されたグラフも印刷できます。
REPORT	開いている Analytics テーブルに基づき、レポートを書式設定し、生成します。

ファイルとシステム

ファイルとシステム コマンドは、ファイル、プロジェクト、およびオペレーティングシステムレベルでさまざまな処理を実行します。

コマンド説明

コマンド	説明
DIRECTORY	指定されたディレクトリ内のファイルとフォルダーの一覧を生成します。
DISPLAY	指定された Analytics 項目タイプの情報を表示します。式の結果、または関数の出力も表示できません。
DUMP	ファイルや現在のレコードの内容を 16 進、ASCII、および EBCDIC 文字エンコードで表示します。

コマンド	説明
HELP	Analytics ヘルプ文書をブラウザで起動します。
QUIT	現在のセッションを終了し、Analyticsを終了します。
RANDOM	一連の乱数を生成します。
SAVE LOG	コマンド ログ全体、または現在の Analytics セッションのログ エントリを、外部ファイルに保存します。

ACCEPT コマンド

1つ以上のスクリプト入力値をインタラクティブにユーザーに確認するダイアログボックスを作成します。各入力値は名前付き文字変数に格納されます。

メモ

ACCEPT コマンドを使用してパスワードを入力することは安全ではありません。代わりに "PASSWORD コマンド" ページ 2015 を使用してください。

ACCEPT コマンドは、ロボットで実行されるスクリプトでサポートされません。

"DIALOG コマンド" ページ 1736では高度なインタラクティブダイアログボックスを作成できます。

構文

```
ACCEPT {メッセージテキスト <FIELDS プロジェクト項目カテゴリ> T0 変数名} <...n>
```

パラメーター

名前	説明
メッセージ テキスト	<p>入力するよう指示するために使用されるテキストとして、ダイアログボックスに表示されるラベルです。引用符で囲んだ文字列か、文字型変数を指定する必要があります。</p> <p>複数のプロンプトを入力する場合は、カンマで区切ることができます。カンマを使用すると、スクリプト判読性を向上させますが、必須ではありません。</p> <pre>ACCEPT "開始日を指定してください:" T0 v_start_date, "終了日を指定してください:" T0 v_end_date</pre>
FIELDS プロジェクト項目カテゴリ 省略可能	<p>テキストボックスではなく、ユーザー入力用のプロジェクト項目ドロップダウンリストを作成します。ユーザーはリストから、プロジェクト項目、フィールド、または変数の1つの値を選ぶことができます。</p> <p>プロジェクト項目カテゴリは、リストに表示する項目タイプを指定します。たとえば、xfと指定すると、リストにすべてのプロジェクトテーブルが表示されます。プロジェクト項目カテゴリは引用符で囲みます。</p> <pre>FIELDS "xf"</pre> <p>カテゴリを指定するために使用するコードについては、"プロジェクト項目カテゴリのコード" ページ 1628を参照してください。</p> <p>同じプロンプトで複数のコードを指定できますが、プロジェクト項目、フィールド、変数を混在させることはできません。</p>

名前	説明
TO 変数名	<p>ユーザー入力を格納するために使用する文字変数の名前。変数自体が存在していない場合は、変数が作成されます。</p> <p>この変数が既に存在する場合は、現在の値がデフォルト値としてダイアログボックスに表示されます。</p> <p>メモ</p> <p>代入変数に使用される変数の名前に、<code>@</code>のような英語以外の文字は使用しないでください。変数名に英語以外の文字が含まれていると、変数の代入が失敗します。</p> <p>ACCEPT コマンドは文字変数のみを作成します。別のデータ型の入力が必要な場合は、スクリプトの後の処理で、文字変数を必要な型に変換する必要があります。詳細については、「入力データ型」ページ 1629を参照してください。</p>

例

ユーザーに対し、開く Analytics テーブルを選択するよう求める

開くテーブルの名前を選択するようユーザーに求めるダイアログボックスが必要です。ユーザーが選択したテーブルをスクリプトによって開きます。

```
ACCEPT "開くテーブルの選択:" FIELDS "xf" TO v_table_name
OPEN %v_table_name%
```

パーセント記号を使用する理由は、開こうとするテーブルの名前が `v_table_name` 変数に格納されていることを示すためです。パーセント記号を省略した場合には、スクリプトは `"v_table_name"` というテーブルを開こうとします。

必要な入力を収集するために多数のダイアログボックスを使用する

スクリプト ユーザーが入力する必要がある各値のダイアログボックスを個別に作成したいとします。ACCEPT コマンドの各インスタンス内に1つのプロンプト文字列を指定します。スクリプトにより、以下の各項目を入力するダイアログボックスを別々に生成します。

- テーブル名
- サンプリングに使用するフィールド

- サンプル間隔
- ランダムな開始値

```
ACCEPT "分析するテーブル名を入力してください" TO v_table_name
OPEN %v_table_name%
ACCEPT "サンプルを抽出するフィールドを選択してください" FIELDS "N" to v_field_
to_sample
ACCEPT "サンプリング間隔を入力してください" TO v_sampling_interval
ACCEPT "ランダム開始値を入力してください" TO v_random_start_value
SAMPLE ON %v_field_to_sample% INTERVAL v_sampling_interval FIXED v_
random_start_value RECORD TO Sample_output OPEN
```

スクリプトを実行すると、次の動作が行われます。

1. 最初のダイアログボックスは、テーブルの名前を入力するように求めます。
2. 2番目のダイアログボックスでは、FIELDS "N" によって、数値フィールドのドロップダウンリストからのフィールドの選択を求めます。
3. 3番目のダイアログボックスでは、サンプリング間隔値の入力を求めます。
4. 4番目のダイアログボックスでは、ランダム開始値の入力を求めます。

必要な入力を収集するために多数のプロンプトを備えた単一のダイアログボックスを使用する

スクリプト ユーザーが入力する必要があるすべての値のダイアログボックスを作成したいとします。ユーザーに複数の入力値を要求する場合には、ACCEPT コマンド内に複数のプロンプトをカンマで区切って指定します。次の例では、日付範囲の開始日と終了日のプロンプトが同じダイアログボックスに表示されます。

```
ACCEPT "開始日を指定してください:" TO v_start_date, "終了日を指定してくさ
い:" TO v_end_date
```

備考

変数の使用の詳細については、「ACLScript での変数の操作」ページ 1554を参照してください。

インタラクティブ

インタラクティブ スクリプトを作成するには、ACCEPT コマンドを使用します。ACCEPT コマンドが処理される際にはスクリプトが中断し、Analytics が次の処理で使用する入力をユーザーに促すダイアログボックスが表示されます。

項目ごとに個別のダイアログボックスを作成して、一度に1項目について入力を求めることもできますが、1つのダイアログボックスで複数項目について入力を求めることもできます。

DIALOG と ACCEPT

DIALOG コマンドでは、次の1つ以上のタイプのコントロールを使用できるより高度なインタラクティブダイアログボックスを作成できます。

- テキストボックス
- チェックボックス
- ラジオボタン
- カスタマイズされた値のドロップダウンリスト
- プロジェクト項目一覧

ダイアログボックスのレイアウトは柔軟にカスタマイズできます。詳細については、"DIALOG コマンド" ページ 1736を参照してください。

プロジェクト項目カテゴリのコード

次のコードを使用して、ドロップダウンリストで表示するプロジェクト項目のカテゴリを指定します。

プロジェクト カテゴリ

コード	カテゴリ
xf	テーブル
xb	スクリプト
xi	インデックス
xr	ビューとレポート
xw	ワークスペース

フィールド カテゴリ

コード	カテゴリ
C	文字フィールド
N	数値フィールド
D	日付時刻フィールド
L	論理フィールド

変数カテゴリ

コード	カテゴリ
c	文字変数
n	数値変数
d	日付時刻変数
l	論理変数

入力データ型

ACCEPT はユーザー入力を 1 つ以上の文字変数として保存します。数値や日付時刻値として保存するには、VALUE() 関数や CTOD() 関数を使用します。文字変数の値がそれぞれ数値や日付時刻値に変換されます。

```
SET FILTER TO BETWEEN(%v_date_field%, CTOD(%v_start_date%), CTOD(%v_end_date%))
```

この例では、フィルターの開始日付および終了日付は文字値として保存されています。これらの日付は、日付時刻データ型を使用する日付フィールドで使用するには、日付型の値に変換する必要があります。

変数名をパーセント記号 (%) で囲むと、変数名の変数によって含まれる文字値を代入します。CTOD() 関数は文字値を日付値に変換します。

ACCEPT コマンドの位置

可能な限り、ACCEPT コマンドはすべてスクリプトの先頭部分に配置することをお勧めします。最初にすべての情報入力を要求し、必要な情報が入力されていれば、その後スクリプトを円滑に実行できます。

メモ

GROUP コマンド内では ACCEPT コマンドを使用できません。

ACCESSDATA コマンド

さまざまな ODBC 準拠データソースからデータをインポートします。

64ビットと32ビットのどちらのODBCドライバーを使用するかによって、このコマンドはACCESSDATA64またはACCESSDATA32という形を取ります。

構文

```
{ACCESSDATA64 | ACCESSDATA32} {CONNECTOR | ODBC {"Driver"|"Dsn"|"File"}}
NAME 値 <USER ユーザー ID> <PASSWORD num | PROMPT_PASSWORD> <PASSWORD 数値 AS
パスワード キー名 <...n>> TO テーブル名 CHARMAX 最大フィールド長 MEMOMAX 最大フィールド
長 <ALLCHARACTER> SOURCE (接続設定) <HASH(ソルト値, フィールド)>
SQL_QUERY
(SQL 構文)
END_QUERY
```

パラメーター

名前	説明
CONNECTOR ODBC {"Driver" "Dsn" "File"}	行いたい ODBC 接続の種類。 <ul style="list-style-type: none"> ○ CONNECTOR -- ネイティブの Analytics データ コネクタを使って接続する ○ ODBC "Driver" -- お使いのコンピューターにインストールされている Windows ODBC ドライバーを使って接続する ○ ODBC "Dsn" -- お使いのコンピューターに保存されている DSN(データ ソース名) を使って接続する ○ ODBC "File" -- DSN ファイル(保存されている .dsn ファイル) を使って接続する
NAME 値	Analytics データ コネクタ、ODBC ドライバー、または DSN の名前。 <p>例：</p> <ul style="list-style-type: none"> ○ NAME "Amazon Redshift" ○ NAME "Microsoft Access Driver (*.mdb, *.accdb)" ○ NAME "My Excel DSN" ○ NAME "excel.dsn"
USER ユーザー ID 省略可能	ユーザー ID が必要なデータソースのユーザー ID。
PASSWORD 数値	単一のパスワードが必要なデータソース：

名前	説明
PROMPT_PASSWORD 省略可能	<ul style="list-style-type: none"> ○ PASSWORD 数値 - 使用するパスワード定義 ○ PROMPT_PASSWORD - パスワード プロンプトを表示します。 詳細については、"ACCESSDATA でパスワード定義を使用する" ページ 1636を参照してください。 抑制されるパスワード値 Analytics のデータアクセス ウィンドウを使用して ACCESSDATA コマンドを実行し、パスワードを入力すると、パスワード値はログに記録されません。PROMPT_PASSWORD パラメーターが代用されます。 PASSWORD コマンドを PASSWORD 数値 とともに使用すると、PROMPT_PASSWORD の使用と同様の結果になります。両方のアプローチでユーザーにパスワードを要求します。PROMPT_PASSWORD にはユーザー ID を変更できるという利点があります。
PASSWORD 数値 AS パスワード キー名 <...n> 省略可能	複数のパスワードが必要なデータソースで使用するパスワード定義。 パスワード キー名は、SOURCE で指定された接続設定に表示されるように正確にパスワード キー名を複製する必要があります。 詳細については、"ACCESSDATA でパスワード定義を使用する" ページ 1636を参照してください。
TO テーブル名	コマンドの結果を送信する場所： <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例： <code>TO "Output.FIL"</code> デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <div style="border-left: 2px solid blue; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> </div>
CHARMAX 最大フィールド長	インポートしているソースの文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。 デフォルト値は 50 です。最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。 フィールドを切り詰める機能では、時折出現する長い値が原因で、インポート処理でサポートされるレコード全体の長さを超えることがなくなります。 <ul style="list-style-type: none"> ○ 32,767 文字 (非 Unicode Analytics) ○ 16,383 文字 (Unicode Analytics)
MEMOMAX 最大フィールド	インポートするテキスト、ノート、またはメモフィールドの文字の最大長。

名前	説明
長	デフォルト値は 100 です。最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。
ALLCHARACTER 省略可能	<p>インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。</p> <p>Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。</p> <p>ヒント</p> <p>ALLCHARACTER は、数値 ID 値を含むテーブルをインポートする際に有効な場合があります。ALLCHARACTER を使用すると、Analytics は自動的に数値データ型を、文字データ型を使用すべき値に割り当てることができます。</p>
SOURCE 接続設定	データソースへの接続に必要な接続設定 (接続文字列) 。
HASH(salt 値, フィールド) 省略可能	<p>暗号ハッシュ値として指定されたフィールドをインポートします。ハッシュ値は単方向の変換であり、フィールドのインポート後に復号化できません。</p> <ul style="list-style-type: none"> salt 値 - フィールドの値のハッシュを強化するためにソースデータ値と連結された英数字文字列。引用符で囲まれた文字列としてハッシュ値を入力します。 <p>ソルト値は 128 文字以下です。() " 文字は使用しないでください。</p> <ul style="list-style-type: none"> フィールド - ハッシュ化する 1 つ以上のフィールドのリスト。引用符で囲まれた文字列としてフィールドを入力し、各フィールドをカンマで区切ります。 <p>データソースの物理フィールド名ではなく、[データアクセス] ウィンドウプレビューとステージング領域に表示されるフィールド名を指定する必要があります。</p> <p>メモ</p> <p>[データアクセス] ウィンドウプレビューに表示されるフィールド名は、SQL クエリ ("フィールド名" AS "エイリアス") のフィールド エイリアス値です。エイリアス値を使用して、フィールドを参照する必要があります。</p> <pre>HASH("QZ3x7", "SSN_NO, CC_NO, Last_Name")</pre> <p>ACLScript でのハッシュ値のインポート中におけるハッシュ値の比較については、"ACCESSDATA でハッシュ化されたデータを ACLScript HASH() 関数でハッシュ化されたデータと比較する" ページ 1642を参照してください。</p>
SQL_QUERY (SQL構文) END_QUERY	<p>SQL のインポート文。</p> <p>丸括弧内はすべて SQL クエリの構成要素のため有効な SQL である必要があります。</p> <p>メモ</p> <p>SQL インポート文の本文では、ACLScript 構文 (コマンドまたは関数) を使用できません。有効な SQL 構文のみを使用する必要があります。</p>

例

ネイティブの Analytics データ コネクタを使用してデータをインポートする

Amazon Redshift クラウド データ サービスからのデータをインポートする必要があるとします。そうするには、Analytics Amazon Redshift データ コネクタを使用します。

```
ACCESSDATA64 CONNECTOR NAME "Amazon Redshift" USER "ACL_user" PROMPT_
PASSWORD TO "Entitlement_History.FIL" CHARMAX 50 MEMOMAX 100
SOURCE(
boolsaschar=0; cachesize=100; database=usage; declarefetchmode=0; maxbytea=
=255; maxlongvarchar=8190; maxvarchar=255; port=5439; servername=acl_
test.highbond.com; singlerowmode=1; sslmode=require; textaslongvarchar=0; -
usemultiplestatements=0; useunicode=1)
SQL_QUERY(
SELECT
"entitlement_history"."organization" AS "organization",
"entitlement_history"."user_email" AS "user_email",
"entitlement_history"."plan_id" AS "plan_id",
"entitlement_history"."date_from" AS "date_from",
"entitlement_history"."date_to" AS "date_to"
FROM
"prm"."entitlement_history" "entitlement_history"
) END_QUERY
```

Windows ODBC ドライバーを使用してデータをインポートする

Microsoft Access データベースからデータをインポートする必要があるとします。インポートを行うには、Windows ODBC ドライバーを使用して MS Access に接続します。

```
ACCESSDATA32 ODBC "Driver" NAME "Microsoft Access Driver (*.mdb)" TO
"Invoices.FIL" CHARMAX 50 MEMOMAX 100
SOURCE( dbq=C:\Users\lachlan_murray\Documents\ACL Data\Sample Data
Files\Sample.mdb; defaultdir=C:\Users\lachlan_murray\Documents\ACL
Data\Sample Data Files; driverid=281; fil=MS
```

```
Access;maxbuffersize=2048;maxscanrows=8;pagetimeout=5;safetransactions-
=0;threads=3;usercommitsync=Yes)
SQL_QUERY(
  SELECT
    `Customer`.`CustID` AS `CustID`,
    `Customer`.`Company` AS `Company`,
    `Customer`.`Address` AS `Address`,
    `Customer`.`City` AS `City`,
    `Customer`.`Region` AS `Region`,
    `Customer`.`PostalCode` AS `PostalCode`,
    `Customer`.`Country` AS `Country`,
    `Customer`.`Phone` AS `Phone`,
    `Orders`.`OrderID` AS `OrderID`,
    `Orders`.`CustID` AS `Orders_CustID`,
    `Orders`.`ProdID` AS `ProdID`,
    `Orders`.`OrderDate` AS `OrderDate`,
    `Orders`.`Quantity` AS `Quantity`,
    `Product`.`ProdID` AS `Product_ProdID`,
    `Product`.`ProdName` AS `ProdName`,
    `Product`.`UnitPrice` AS `UnitPrice`,
    `Product`.`Descript` AS `Descript`,
    `Product`.`ShipWt` AS `ShipWt`
  FROM
    (`Customer` `Customer`
  INNER JOIN
    `Orders` `Orders`
      ON `Customer`.`CustID` = `Orders`.`CustID`
    )
  INNER JOIN
    `Product` `Product`
      ON `Orders`.`ProdID` = `Product`.`ProdID`
  WHERE
    (
      `Customer`.`Region` = 'BC'
      OR `Customer`.`Region` = 'WA'
    )
) END_QUERY
```

Windows DSN(データソース名) を使用してデータをインポートする

Microsoft Excel ファイルからデータをインポートする必要があるとします。インポートを行うには、Windows DSN を使用して Excel に接続します。

```
ACCESSDATA32 ODBC "Dsn" NAME "Excel Files" TO "Trans_April_15_
cutoff.FIL" CHARMAX 50 MEMOMAX 100
SOURCE( dbq=C:\Users\lachlan_murray\Documents\ACL Data\Sample Data
Files\Trans_April.xls;defaultdir=C:\Users\lachlan_murray\Documents\ACL
Data\Sample Data Files;driverid=1046;maxbufferize=2048;pagetimeout=5)
SQL_QUERY(
SELECT
`Trans_Apr_`.`CARDNUM` AS `CARDNUM`,
`Trans_Apr_`.`AMOUNT` AS `AMOUNT`,
`Trans_Apr_`.`TRANS_DATE` AS `TRANS_DATE`,
`Trans_Apr_`.`CODES` AS `CODES`,
`Trans_Apr_`.`CUSTNO` AS `CUSTNO`,
`Trans_Apr_`.`DESCRIPTION` AS `DESCRIPTION`
FROM
`Trans_Apr$` `Trans_Apr_`
WHERE
(
`Trans_Apr_`.`TRANS_DATE` <= {ts '2003-04-15 00:00:00'}
)
) END_QUERY
```

備考

このコマンドの動作の詳細については、「データアクセス ウィンドウの作業」ページ 376を参照してください。

ACCESSDATA でパスワード定義を使用する

一般的に、ACCESSDATA コマンドで接続するデータソースでは、パスワード、トークン、または他のシークレット認証値を使用した認証が必要です。一部のデータソースでは、複数の認証値が必要です。

ACCESSDATA コマンドの一部として、複数のパスワード定義を指定してこの認証を提供できます。パスワード定義はパスワードや認証値自体ではありません。どちらかという、パスワード変数のようなものであり、以前に入力されたパスワードや認証値を安全に保存します。ACCESSDATA コマンドでパスワード定義を指定すると、SOURCE で指定された接続設定で、実際のパスワードをクリアテキストに表示しないようにすることができます。

パスワード定義の作成

インポート スクリプトでは、パスワード定義を作成した後に、ACCESSDATA コマンドで定義を使用できません。

Analytics で使用するパスワード定義の作成については、「PASSWORD コマンド」ページ 2015を参照してください。

ロボットで使用するパスワード定義の作成については、「PASSWORD タグ」 ページ 2626を参照してください。

パスワード定義を指定する2つのオプション

ACCESSDATA でパスワード定義を指定するときには2つのオプションがあります。

- **PASSWORD 数値** - 単一のパスワードが必要なデータソースに対して、単一のパスワード定義を指定します。
- **PASSWORD 数値 AS パスワード キー名** - 繰り返し使用して、複数の認証値が必要なデータソースで複数のパスワード定義を指定できます。

メモ

オプションは1つずつ使用することも、2つまとめて使用することもできます。

PASSWORD 数値の仕組み

データソースで単一のパスワードが必要な場合にのみ、PASSWORD 数値パラメーターを使用します。

以下の例:

1. `PASSWORD 1` コマンドはパスワードを入力するようにユーザーに指示し、入力されたパスワードを安全に保存するパスワード定義を作成します。
2. `ACCESSDATA` コマンドでは、`PASSWORD 1` パラメーターがパスワード定義を参照し、保存されたパスワード値を `SOURCE(auth_accesstoken=[$pwd])` で指定された接続設定に安全に渡します。

```
PASSWORD 1
ACCESSDATA64 CONNECTOR NAME "Concur" PASSWORD 1 TO "Concur_data_
import.FIL" CHARMAX 50 MEMOMAX 100
SOURCE( auth_accesstoken=[ $pwd ];auth_type=0Auth
2.0;enabledoublebuffer=1;host=www.concursolutions.com;useencryptedendp-
oints=1;userparam=all)
SQL_QUERY(
SELECT
  "List_Items"."Level_7_Code" AS "Level_7_Code",
  "List_Items"."Name" AS "Name",
  "List_Items"."Level_10_Code" AS "Level_10_Code",
  "List_Items"."Level_8_Code" AS "Level_8_Code",
  "List_Items"."URI" AS "URI",
  "List_Items"."Id" AS "Id",
  "List_Items"."Level_3_Code" AS "Level_3_Code",
  "List_Items"."List_Id" AS "List_Id",
  "List_Items"."Level_4_Code" AS "Level_4_Code",
  "List_Items"."Level_1_Code" AS "Level_1_Code",
```

```

"List_Items"."Parent_Id" AS "Parent_Id",
"List_Items"."Level_2_Code" AS "Level_2_Code",
"List_Items"."Level_5_Code" AS "Level_5_Code",
"List_Items"."Level_6_Code" AS "Level_6_Code",
"List_Items"."Level_9_Code" AS "Level_9_Code"
FROM
"Concur"."List_Items" "List_Items"
) END_QUERY

```

PASSWORD 数値 AS パスワード キー名の仕組み

データソースで複数のパスワードまたは認証値が必要な場合は、PASSWORD 数値 AS パスワード キー名パラメーターを使用します。

以下の例:

1. ロボットでは、アナリティクス ヘッダーの `//PASSWORD` タブは、ユーザーが認証値を入力する必要がある 4 つのパスワード パラメーターを作成します。4 つのパラメーターは、入力された値を安全に保存する 4 つのパスワード定義を作成します。
2. ACCESSDATA コマンドでは、4 つの `PASSWORD` パラメーターがパスワード定義を参照し、保存された認証値を `SOURCE` で指定された接続設定に安全に渡します。

- `oauthclientid=`
- `oauthclientsecret=`
- `oauthaccesstoken=`
- `OAuthAccessTokenSecret`

詳細については、「複数のパスワード定義で動作するように ACCESSDATA を構成する」次のページを参照してください。

```

COMMENT
//ANALYTIC TYPE IMPORT Twitter データをインポート
//PASSWORD 1 OAuth クライアント ID を入力:
//PASSWORD 2 OAuth クライアント シークレットを入力:
//PASSWORD 3 OAuth アクセス トークンを入力:
//PASSWORD 3 OAuth アクセス トークン シークレットを入力:
//RESULT TABLE Twitter_user_data
END

ACCESSDATA64 CONNECTOR NAME "Twitter" PASSWORD 1 AS oauthclientid
PASSWORD 2 AS oauthclientsecret PASSWORD 3 AS oauthaccesstoken
PASSWORD 4 AS oauthaccesstokensecret TO "Twitter_user_data.FIL"
CHARMAX 50 MEMOMAX 100
SOURCE(

```

```

oauthclientid=;oauthclientsecret=;oauthaccesstoken=;oauthaccesstokense-
cret=;readonly=true;drivertype=ACL Connector for
Twitter;connectonopen=true;convertdatetimetogmt=true;limitkeysize=255;-
maptolongvarchar=-
1;maptowvarchar=true;uppercaseidentifiers=false;supportenhancedsql=true-
e;proxyauthscheme=BASIC;proxyautodetect=true;_persist_=encrypted-dp
{AQA ... kX3E8yyh05HoG1rH4bm1lhWudUQ==}})
SQL_QUERY(
  SELECT
    "Users"."ID" AS "ID",
    "Users"."Name" AS "Name",
    "Users"."Screen_Name" AS "Screen_Name",
    "Users"."Location" AS "Location",
    "Users"."Profile_URL" AS "Profile_URL",
    "Users"."Lang" AS "Lang",
    "Users"."Created_At" AS "Created_At",
    "Users"."Friends_Count" AS "Friends_Count",
    "Users"."Followers_Count" AS "Followers_Count",
    "Users"."Favourites_Count" AS "Favourites_Count",
    "Users"."Statuses_Count" AS "Statuses_Count",
    "Users"."Time_Zone" AS "Time_Zone",
    "Users"."Following" AS "Following",
    "Users"."Contributors_Enabled" AS "Contributors_Enabled",
    "Users"."Follow_Request_Sent" AS "Follow_Request_Sent",
    "Users"."Listed_Count" AS "Listed_Count",
    "Users"."Description" AS "Description",
    "Users"."Default_Profile" AS "Default_Profile"
  FROM
    "Twitter"."Users" "Users"
) END_QUERY

```

複数のパスワード定義で動作するように ACCESSDATA を構成する

複数のパスワード定義で動作するように ACCESSDATA コマンドを構成するには、コマンドに PASSWORD パラメーターを挿入し、SOURCE パラメーターのパスワード キー名を PASSWORD パラメーターにコピーします。

1. Analytics でデータアクセス ウィンドウを使用して、複数の認証値が必要なデータソースからデータをインポートします。
2. ログの `ACCESSDATA` コマンドをスクリプト エディターで開いているスクリプトにコピーします。

一般的に、1つの認証値のみが `SOURCE` パラメーターでマスク (`[$pwd]`) されます。追加の値はクリアテキストで表示されます。例:

```
SOURCE( oauthclientid=cXQ ... dR4;oauthclientsecret=QUt ...
beo;oauthaccesstoken=913 ... cPn;oauthaccesstokensecret=[$pwd]; ... )
```

3. `SOURCE` パラメーターからクリアテキスト 認証値を削除し、パスワード キー名と等号のみを残します。

例：

```
SOURCE(
oauthclientid=;oauthclientsecret=;oauthaccesstoken=;oauthaccesstokensecret=[$pwd]; ... )
```

メモ

認証値は、PASSWORD コマンドまたは PASSWORD アナリティクス タグで作成されたパスワード定義を使用してユーザーが指定する必要があります。詳細については、「パスワード定義の作成」ページ 1636を参照してください。

4. 省略可能。マスクされた認証値の1つのパスワード キー名から `[$pwd]` を削除します。
このパスワード キー名では、2つの方法のいずれかを使用して、ACCESSDATA コマンドでパスワード定義を指定できます。詳細については、「パスワード定義を指定する2つのオプション」ページ 1637を参照してください。
5. `PROMPT_PASSWORD` パラメーターを `ACCESSDATA` コマンドから削除します。
6. `PROMPT_PASSWORD` を削除した場所に `PASSWORD` パラメーターを挿入し、`SOURCE` パラメーターのパスワード キー名をコピーして `PASSWORD` パラメーターに貼り付けます。

例：

```
ACCESSDATA64 CONNECTOR NAME "Twitter" PASSWORD 1 AS oauthclientid
PASSWORD 2 AS oauthclientsecret PASSWORD 3 AS oauthaccesstoken PASSWORD
4 AS oauthaccesstokensecret ...
SOURCE(
oauthclientid=;oauthclientsecret=;oauthaccesstoken=;oauthaccesstokensecret=; ... )
```

重要

パスワード キー名は、SOURCE パラメーターと PASSWORD パラメーターで同じでなければなりません。異なる場合、ACCESSDATA コマンドが失敗します。

7. マスクされた認証値のパスワード キー名から `[$pwd]` を削除しなかった場合、単一のパスワード定義を指定する方法を使用します。

例：

```
ACCESSDATA64 CONNECTOR NAME "Twitter" PASSWORD 1 AS oauthclientid
PASSWORD 2 AS oauthclientsecret PASSWORD 3 AS oauthaccesstoken PASSWORD
4 ...
SOURCE(
oauthclientid=;oauthclientsecret=;oauthaccesstoken=;oauthaccesstokensec-
ret=[$pwd]; ... )
```

結果 - ACCESSDATA コマンドと個別に作成されたパスワード定義をインポート スクリプトで使用できます。SOURCE 接続設定にはクリアテキストの認証値は表示されません。

多要素認証は適切ではありません

多要素認証 (MFA) が必要データソースにアクセスするには、ACCESSDATA コマンドを使用することはできません。スクリプトはこの方法で認証することができません。MFA で保護されたデータにアクセスする必要がある場合は、MFA が必要ではない汎用作業アカウントの使用が組織で許可されているかどうかを確認してください。

データ コネクターの更新

Analytics またはロボット エージェントをアップグレードする場合は、いずれかの Analytics データ コネクターを使ってデータをインポートするスクリプト (ACCESSDATA コマンド) をテストする必要があります。

データソースまたは ODBC ドライバーのサードパーティベンダーが加えた変更により、1 つまたは複数のデータ コネクターを更新しなければならない可能性があります。スクリプトに作り込んだデータ接続を引き続き正常に動作させるには、それを更新する必要があります。

- **インポートの再実行** - 接続機能を最も簡単に更新する方法は、更新されたバージョンの Analytics のデータアクセス ウィンドウを使って、インポートを手動で実行することです。ログに記録されている ACCESSDATA コマンドをコピーし、それを使ってスクリプトを更新します。

メモ

データソースに接続してインポートを再実行する前に、コネクターのキャッシュを消去することで、既存の一連のテーブル名を削除します。

データアクセス ウィンドウの **既存の接続** タブで、接続名の横にある  > **キャッシュの消去** を選択します。

- **フィールド指定の更新** - データソースまたは ODBC ドライバーのテーブルスキーマの変更と一致するように、スクリプト本体内のフィールド指定を更新しなければならない場合もあります。必要に応じてフィールド名、フィールドのデータ型、およびフィールド長とレコード長を変更します。
- **フィルター処理の結果の確認** - データインポートの一環として適用したフィルター処理の結果も確認する必要があります。インポートのフィルター処理によってレコードを正しく追加または除外できたかどうかを確認します。

ODBC 接続設定とSQL インポート文の作成

ODBC 接続設定とSQL インポート文は例に示すように通常、きわめて長く複雑です。

ACCESSDATA コマンドのこれらの部分を最も簡単に作成するには、まず Analytics の Data Access ウィンドウを使ってターゲットのデータソースに接続してからデータをインポートします。次に、ログから、接続設定やインポート文など ACCESSDATA コマンド全体をコピーし、必要に応じてこのコマンドをカスタマイズします。

ACCESSDATA ログファイル

2つのログファイルは ACCESSDATA コマンドに関連付けられたトランザクションを記録し、データ接続が失敗した場合にトラブルシューティングで使用できます。

- **ServerDataAccess.log** - はデータをインポートする前のすべての作業とエラーを記録します
場所: `C:\Users\<ユーザー アカウント>\AppData\Local\ACL\ACL for Windows\Data Access\ServerDataAccess.log`

メモ

`ServerDataAccess.log` の "Server" は Analytics がインストールされているコンピューターでローカルに実行されている Analytics のデータアクセスコンポーネントを参照します。

- **DataAccess.log** - には、インポート操作に関する情報と、
場所 `..\<Analytics プロジェクト フォルダー>\DataAccess.log`
にインポートしようとしているデータが含まれる Analytics プロジェクトに関する情報が記録されません。

ACCESSDATA でハッシュ化されたデータを ACLScript HASH() 関数でハッシュ化されたデータと比較する

ハッシュデータの未加工値を読み取れない場合でも、データの結合または分析時に役立ちます。

インポート中に ACCESSDATA でハッシュ化された値を、ACLScript の HASH() 関数を使用してハッシュ化された値と比較する場合は、Analytics の数値または日付時刻フィールドを文字値に変換し、データをハッシュ化する前に前後のスペースを切り取る必要があります。

文字に変換するときには、日付時刻フィールドは次の形式を使用する必要があります。

- 日付時刻 - "YYYY-MM-DD hh:mm:ss"
- 日付 - "YYYY-MM-DD"
- 時刻 - "hh:mm:ss"

次の例は、STRING() および ALLTRIM() 関数を使用して、ACLScript's HASH() 関数を使用して値をハッシュ化する前に、数値のクレジットカード番号フィールドを文字データに変換します。

コマンド

```
COMMENT ACL HASH 関数は、データのインポート後に使用されます  
HASH(ALLTRIM(String(CC_No, 16)), "QZ3x7")
```

Analytics 値をハッシュ化すると、ACCESSDATA コマンド インポートの一部としてハッシュ化された値を比較できます。

ACTIVATE コマンド

Analytics ワークスペースに保存されたフィールド定義を Analytics テーブルレイアウトのフィールド定義の既存のセットに追加します。

構文

```
ACTIVATE <WORKSPACE> ワークスペース名 <OK>
```

パラメーター

名前	説明
WORKSPACE ワークスペース名	アクティブにするワークスペースの名前。
OK 省略可能	アクションを確認せずに、項目を削除または上書きします。 アクティブ化されたワークスペースのフィールドと同じ名前のテーブルのフィールドがある場合は、確認なしで上書きされます。ただし、演算フィールドが参照しているフィールドを置き換えることはできません。

例

Analytics プロジェクトでワークスペースをアクティブにする

ComplexFormulas ワークスペースをアクティブにします。

```
ACTIVATE WORKSPACE ComplexFormulas OK
```

Analytics プロジェクトと同じフォルダー内のファイル (.wsp) として保存されたワークスペースをアクティブにする

.wsp ファイルとして保存された **ComplexFormulas** ワークスペースをアクティブにします。

```
ACTIVATE WORKSPACE ComplexFormulas.WSP OK
```

備考

機能の仕組み

ACTIVATE は、ワークスペースのフィールド定義をアクティブなテーブルで使用できるようにします。ワークスペースをアクティブにしたら、アクティブテーブルを閉じない限りは、テーブルでワークスペースのフィールドを使用できます。

テーブルレイアウトの編集

次の場合、ワークスペースのフィールドはテーブルレイアウトに永続的に追加されます。

- ワークスペースをアクティブにした後でテーブルレイアウトを編集した場合
- テーブルレイアウトが保存されるような変更を行った場合

ワークスペースのフィールドをテーブルレイアウトに保存すると、以下の操作を行えるようになります。

1. DEFINE COLUMN コマンドを使ってビューにそれらのフィールドを追加する
2. SAVE コマンドを使って変更を保存する

AGE コマンド

日付または日付時刻フィールドの値に基づいて、レコードを年齢調べ間隔でグループ化します。各期間のレコード数をカウントし、指定した数値フィールドの小計を期間ごとに求めます。

構文

```
AGE <ON> 日付フィールド <CUTOFF 締切日> <INTERVAL 日 <,...n>> <SUPPRESS>
<SUBTOTAL numeric_field <...n>|SUBTOTAL ALL <EXCLUDE 数値フィールド <...n>>>
<IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <TO
{SCREEN|filename|GRAPH|PRINT}> <KEY 内訳フィールド> <HEADER ヘッダーテキスト>
<FOOTER フッターテキスト> <APPEND> <STATISTICS>
```

パラメーター

名前	説明
ON 日付フィールド	年齢調べに用いる日付時刻フィールドまたは式の名前。 日付時刻フィールドを対象として年齢調べを行うことはできますが、日付時刻値の日付部分しか考慮されません。時刻部分は無視されます。時刻データだけで年齢調べを行うことはできません。
CUTOFF 締切日 省略可能	日付フィールドの値と比較される日付。 締切日は、日付フィールドの形式に関係なく、引用符で囲まれていないYYMMDDまたはYYYYMMDDの形式で指定する必要があります。例: <code>CUTOFF 20141231</code> 締切日を省略した場合は、締切日として、現在のシステム日付が使用されます。
INTERVAL 日数 <,...n> 省略可能	年齢調べ間隔の計算で使用する日付の間隔(つまり、日数)。 日数は、締切日から過去に遡る各年齢調べ間隔の始まりを表します。 <ul style="list-style-type: none"> 最初の日数値は、最初の年齢調べ間隔の始まりを決定します。 最初の日数値を'0'にした場合、最初の年齢調べ間隔は指定した締切日から始まります。 最後の日数値は、最終の年齢調べ間隔の終わりを決定します。 間隔は、引用符で囲まれていないカンマ区切りの文字列値として指定する必要があります。 <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">INTERVAL 0,90,180,270,365</div> デフォルトの年齢調べ間隔は"0"、"30"、"60"、"90"、"120"、"10,000"日です。10,000日の

名前	説明
	<p>期間は、無効と思われる日付が入っているレコードを分離するために使用されます。</p> <p>必要に応じて、ほかの内部年齢調べレポートを反映するように、日付の間隔をカスタマイズすることができます。</p>
<p>SUPPRESS</p> <p>省略可能</p>	<p>指定した年齢調べ間隔に該当しない日付をコマンドの出力から除外します。</p>
<p>SUBTOTAL 数値フィールド <...n> SUBTOTAL ALL</p> <p>省略可能</p>	<p>グループごとに小計を計算する1つ以上の数値フィールドまたは式。</p> <p>複数のフィールドはスペースで区切る必要があります。テーブル内のすべての数値フィールドについて小計を求める場合はALLを指定します。</p>
<p>EXCLUDE 数値フィールド</p> <p>省略可能</p>	<p>SUBTOTAL ALLを使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDEでは、指定したフィールドを除外して、SUBTOTAL ALLを微調整できます。</p> <p>EXCLUDEはSUBTOTAL ALLの直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>SUBTOTAL ALL EXCLUDE field_1 field_2</p> </div>
<p>IF テスト</p> <p>省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IFパラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト</p> <p>省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件がfalseと評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILEをFIRSTまたはNEXTとともに使用する場合は、1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲</p> <p>省略可能</p>	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>TO SCREEN ファイル名 GRAPH PRINT</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analyticsの表示領域に結果を表示します

名前	説明
	<p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code> GRAPH - は結果をグラフに表示し、それを Analytics の表示領域に表示します 印刷 - 通常使うプリンターに結果を送信します
KEY ブレークフィールド 省略可能	<p>小計計算をグループ化するフィールドまたは式。ブレークフィールドの値が変わるたびに、小計が計算されます。</p> <p>ブレークフィールドは、文字フィールドか式である必要があります。指定できるフィールドは1つだけですが、1つ以上のフィールドを含んでいる式を使用することができます。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> 同じフィールド 同じフィールド順序 一致するフィールドが同じ長さ 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
STATISTICS 省略可能	<p>メモ</p> <p>SUBTOTAL も指定されていない場合は使用できません。</p> <p>すべての SUBTOTAL フィールドの平均値、最小値、および最大値を計算します。</p>

例

小計された金額がある請求書の年齢調べ

Invoice_Date(請求日)フィールドに基づいてAr(売掛金)テーブルの年齢調べを行い、Invoice_Amount(請求金額)フィールドの小計を求めたいとします。

請求書は以下の30日の期間によってグループ化されます。

- 締切日からその29日前まで
- 締切日の30日前から59日前まで
- といった具合です。

次のコマンドにより、各期間の未払い請求金額の合計が出力されます。

```
OPEN Ar
AGE ON Invoice_Date CUTOFF 20141231 INTERVAL 0,30,60,90,120,10000
SUBTOTAL Invoice_Amount TO SCREEN
```

備考

このコマンドの動作の詳細については、「データの年齢調べ」ページ 1312を参照してください。

年齢調べ間隔

AGEコマンドは、日付または日付時刻フィールドに基づいて、レコードを年齢調べ間隔でグループ化します。出力結果には、期間ごとに1つのレコードが含まれ、各レコードには、ソーステーブル内でその期間に該当するレコードの数が含まれます。

間隔測定

年齢調べ間隔は、現在のシステム日付、あるいは会計年度末の日付など、指定した締切日から過去に遡る日数の間隔に基づきます。

将来の期間

日付の間隔に負の値を入力することによって、締切日より後の年齢調べ間隔を作成することができます。たとえば、次のコマンドでは、締切日から経過する年齢調べの間隔(マイナスの年齢調べ間隔)と、締切日から遡る年齢調べの間隔が作成されます。

```
INTERVAL -60, -30, 0, 30, 60, 90
```

この手法でさまざまな時点を使用すれば、テーブル内の全レコードの日付を含んだ日付プロファイルを作成することができます。

一般的な用途

年齢調べは、販売傾向の評価、取扱量の調査、未払い日数による請求書のグループ化などでよく使用されます。

Analytics は、指定された年齢調べ間隔に該当しない日付用として、1つまたは2つの追加年齢調べ間隔を自動的に作成します。ただし、これは SUPPRESS を使用していないことを前提とします。

APPEND コマンド

2つ以上の Analytics テーブルからのレコードを結合するには、それらのレコードを新しい Analytics テーブルに追加します。

構文

```
APPEND テーブル1, テーブル2, <...n> TO テーブル名 <OPEN> <ASCHAR> <ALLCHAR>
<SOURCETABLE>
```

パラメーター

名前	説明
テーブル1, テーブル2, <...n>	<p>追加するテーブル。</p> <p>各テーブル内のレコードが、テーブルを指定した順に追加されます。出力テーブルにはテーブル1のレコード、テーブル2のレコード、という順に格納されます。</p> <p>各ソーステーブルは、レコード構造が同じでも異なってもどちらでも構わず、また、並べ替えても並べ替えなくても構いません。</p>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> <code>TO "C:\Output.FIL"</code> <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
COMMONFIELDS	追加対象となるすべてのテーブルに共通のフィールドのみが、出力テーブルに追加されます。

名前	説明
省略可能	<p>COMMONFIELDS を省略した場合は、すべてのテーブルの全フィールドが出力テーブルに追加されます。ソース テーブルにフィールドがない場所は、出力テーブルでは空白値になります。</p> <p>ヒント これら 2 つのオプションの例を示した図と画面キャプチャについては、「テーブルの追加」 ページ 928を参照してください。</p> <p>メモ APPEND コマンドでは、演算フィールドの追加がサポートされていないためです。詳細については、「演算フィールドがサポートされていない」 ページ 1656を参照してください。</p> <p>フィールドが共通と見なされる条件とは</p> <p>フィールドが共通と見なされるには、フィールドが次のようである必要があります。</p> <ul style="list-style-type: none"> ◦ すべてのソース テーブルに出現している ◦ 同じ物理名を持つ ◦ 同じデータ型(下記)に属している: <ul style="list-style-type: none"> • 文字 • 数値 • 日付時刻 • 論理 <p>同じ名前、異なるデータ カテゴリ</p> <p>2 つのフィールドが同じ名前を持っているが、異なるデータ カテゴリに属している場合には、エラー メッセージが表示され、APPEND コマンドは実行されません。</p> <p>このエラー メッセージには、APPEND で指定された全テーブルにおけるデータ カテゴリの不一致のすべてが表示されます。このメッセージはコマンド ログに保存されます。</p> <p>メモ ASCHAR または ALLCHAR を使用してデータ カテゴリを一致させることで、このような状況を回避できます。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
ASCHAR 省略可能	<p>非文字フィールドを文字データ カテゴリに変換することで、同じ名前と異なるデータ カテゴリのフィールドを一致させます。</p> <p>たとえば、Employee_ID フィールドは 1 つのテーブルの文字データであり、もう 1 つの表の数値データである 2 つのテーブルを追加します。数値的 Employee_ID フィールドは文字データに変換され、2 つのフィールドはエラーなしで追加されます。Employee_ID フィールドは文字データに変換され、2 つのフィールドはエラーなしで最後に追加されます。</p> <p>ALLCHAR も指定されている場合、ASCHAR は無視されます。</p>
ALLCHAR 省略可能	<p>文字データ カテゴリに最後に追加されるすべてのテーブルのすべての非文字フィールドを変換します。</p>

名前	説明
	<p>この文字データへのグローバル変換により、すべて同じ名前のフィールドはエラーなしで最後に追加されます。</p> <p>メモ 最後に追加後は、フィールドに含まれているデータに応じて、最後に追加されたフィールド全体のデータカテゴリを変更することができます。</p>
<p>SOURCETABLE 省略可能</p>	<p>出力テーブルにソーステーブルフィールド(<code>Source_Table</code>)を含めます。</p> <p>出力テーブルの各レコードの [ソーステーブル] フィールドには、そのレコードのソーステーブルが表示されます。</p> <p>ヒント 追加するソーステーブルの名前を含めておくと、出力テーブルのデータを分析する際に役立つ可能性のある情報が得られます。</p>

例

3つの月次取引テーブルを最後に追加する

以下の例では、3つの月次取引テーブルが追加され、それらのテーブル内のすべてのフィールドを追加した四半期取引テーブルが出力されます。

```
APPEND Trans_Jan, Trans_Feb, Trans_Mar TO Trans_Q1
```

3つの従業員テーブルを最後に追加し、共通フィールドのみを含める

以下の例では、3つの課員テーブルが追加され、それらのテーブル内の共通フィールドのみを追加したマスター従業員テーブルが出力されます。

```
APPEND Employees_central, Employees_east, Employees_west TO Employees_master COMMONFIELDS
```

3つの従業員テーブルを最後に追加し、異なるデータカテゴリのフィールドを一致させる

以下の例では、3つの部門別従業員テーブルを最後に追加されており、ここでは、いくつかの同じ名前のフィールドは異なるデータカテゴリを使用しています。

最初の例では、一致のために必要な場合のみ、非文字フィールドを文字データカテゴリに変換しています:

```
APPEND Employees_central, Employees_east, Employees_west TO Employees_
master ASCHAR
```

2番目の例では、一致に必要なかどうか関わらず、すべての非文字フィールドを文字データカテゴリに変換しています:

```
APPEND Employees_central, Employees_east, Employees_west TO Employees_
master ALLCHAR
```

備考

このコマンドの動作の詳細については、「テーブルの追加」ページ 928を参照してください。

機能の仕組み

APPEND コマンドは、2つ以上のテーブル内のレコードを追加した新しいテーブルを作成することで、それらのテーブル内のレコードを結合します。ここで、追加とはレコードグループを別のレコードグループの末尾に付加することです。

同じ物理名と同じデータカテゴリを持つソーステーブルフィールドは、互いのフィールドに直接追加されません。

すべてのソーステーブルにおいて一意である物理名を持つフィールドは、出力テーブルには追加されますが、互いのフィールドには直接追加されません。

ヒント

名前が異なるフィールドを直接追加したい場合は、事前に各テーブルレイアウト内のフィールドの物理名を統一しておいてください(フィールドが同じデータカテゴリに属しているか、ASCHAR または ALLCHAR を使用してフィールドのデータカテゴリを一致させていると仮定します。)

APPEND の使用に適する場面

APPEND は、構造が同じまたは類似である複数のテーブル内のデータを結合したい場合に使用できます。たとえば、月次テーブルまたは四半期テーブルを年次テーブルに結合する場合に使用することをお勧めします。

ヒント

APPEND コマンドを 1 回実行しただけで、APPEND オプションを指定した EXTRACT コマンドの複数回の実行結果が置き換えられます。

JOIN や DEFINE RELATION の代替としては使用できない

APPEND は JOIN および DEFINE RELATION コマンドの代替としては使用できません。その理由は、共通キーフィールド内の一致する、または一致しない値に基づいてレコードを包含または除外することができないためです。APPEND では、すべてのソーステーブル内の全レコードが出力テーブルに追加されます。

類似性がまったくないテーブルの追加

類似性がまったくないテーブル、つまり共通のフィールドがない 2 つ以上のテーブルを追加することができます。類似性がないテーブルを追加することが分析目的に資する場合があります(これは APPEND コマンドの意図した主な用途ではありません)。

日付時刻フィールドの追加

2 つ以上の日付時刻フィールドを追加するには、次の条件を満たす必要があります。

- 物理名が同じ
- データカテゴリが同じ(日付時刻)
- データサブタイプが同じ(日付、日付時刻、時刻)
- タイムゾーンインジケータの使用有無の一致 - 追加対象となるすべてのフィールドで使用するか、使用しないかのいずれか

2 つの日付時刻フィールドが同じ名前を持っているが、その他の条件のいずれかを満たしていない場合には、エラーメッセージが表示され、APPEND コマンドは実行されません。

エラーメッセージには、APPEND で指定された全テーブルにおいて満たされなかったすべての条件が表示されます。このメッセージはコマンドログに保存されます。

メモ

異なる日付時刻フィールドを文字データカテゴリに変換してすることでこれらのフィールドを一致させ、それから最後に追加することができます。このアプローチでは、データを 1 つのテーブルに結合することができます。ただし、ソースデータの性質によっては、結合されたデータを日付時刻データに戻すことができない場合があります。

自動調整

次の場合、APPEND コマンドにより、フィールドを追加するために、フィールドが自動的に調整されます。

フィールドのデータカテゴリ	調整の実行
文字	<ul style="list-style-type: none"> 異なるフィールド長は調整されます。 フィールドを ASCII または UNICODE データ型に変換することで、カスタム、PCASCII、EBCDIC など様々な文字データ型が調整されます。
数値	<ul style="list-style-type: none"> 異なるフィールド長は調整されます。フィールドは ACL データ型に変換されます。 異なる小数点以下桁数が定義されている場合は、調整されます。小数点以下桁数は、必要に応じて数値の末尾にゼロが付加されて、最大の桁数に統一されます。フィールドは ACL データ型に変換されます。 フィールドを ACL データ型に変換することで、Print、Float、EBCDIC、Micro など複数のデータ型が調整されます。
日付時刻	<ul style="list-style-type: none"> フィールドを Analytics のデフォルト書式に変換することで、ソースデータ内にある複数の日付書式、日付時刻書式、時刻書式が調整されます。 <ul style="list-style-type: none"> YYYYMMDD YYYYMMDD hh:mm:ss hh:mm:ss

自動調整が実行されない場合とは

次の場合、フィールドは Analytics によって自動調整されません。エラーメッセージが表示され、追加操作は実行されません。

- 名前が同じ 2 つのフィールドが異なるデータカテゴリに属する
- 名前が同じ 2 つの日付時刻フィールドが異なる日付時刻サブタイプ(日付、日付時刻、時刻)に属する
- 名前が同じ 2 つの日付時刻フィールドにおいて、タイムゾーンインジケータの使用有無が一致していない

メモ

同じ名前で異なるデータカテゴリのフィールドのユーザー指定の一致については上記で説明されています。詳細については、"ASCHAR" ページ 1652と"ALLCHAR" ページ 1652を参照してください。

演算フィールドがサポートされていない

APPEND コマンドでは、演算フィールドの追加がサポートされていないためです。テーブルを追加する場合、ソーステーブル内のすべての演算フィールドは自動的に出力テーブルから除外されます。

あるソーステーブル内の演算フィールドが、別のソーステーブル内の物理フィールドと同じ名前を持っている場合には、エラーメッセージが表示され、APPEND コマンドは実行されません。

ヒント

演算フィールドを追加するには、まずこのようなフィールドを抽出して、物理フィールドに変換します(詳細については、"EXTRACT コマンド" ページ 1801を参照してください)。次に、演算フィールドの抽出先となったテーブルを追加操作で使用します。

別のアプローチとしては、追加された出力テーブルに演算フィールドを再作成します。

レコード ノート フィールドがサポートされていない

APPEND コマンドでは、レコード ノート フィールドの追加がサポートされていないためです。テーブルを追加する場合、ソーステーブル内のすべてのレコード ノート フィールドは自動的に出力テーブルから除外されます。

あるソーステーブル内のレコード ノート フィールドが、別のソーステーブル内の物理フィールドと同じ名前を持っている場合には、エラーメッセージが表示され、APPEND コマンドは実行されません。

レコード ノート フィールドは、レコードにノートを追加したときに Analytics によって自動的に生成されるものです。

レコード長

追加時にすべてのソーステーブル内の全フィールドを含めるには、出力テーブルのレコード長がソーステーブル内の最も長いレコードより長くなる可能性があります。

出力テーブルのレコード長が Analytics の上限である 32 KB を超えると、エラーメッセージが表示されません。

追加と小数点以下桁数

小数位がある数値フィールドの追加は、特定の動作によって制御されます。

小数位設定

APPEND コマンドでは、テーブルレイアウトのフィールド定義にある**小数位設定**に定義された小数点以下桁数が使用されます。

メモ

小数位設定は、ソースデータ内の実際の小数点以下桁数とは同じでない場合があります。**小数位設定**を超える小数点以下桁数は未定義のため、計算では丸められません。

統一されていない小数位設定

最後に追加された数値フィールドの**小数位設定**が統一されていない場合には、それらのフィールドは ACL データ型に変換され、最も長い**小数位設定**に自動的に調整されます。

最も長い小数位設定を超えるソース データ ファイル内の小数点以下桁数は、APPEND で生成される出力テーブルから除外されます。

統一されている小数位設定

追加された数値フィールドの小数位設定が統一されている場合には、データ型変換も調整も行われません。

小数位設定を超えるソース データ ファイル内の小数点以下桁数は、APPEND で生成される出力テーブルに追加されます。

並べ替え

ソーステーブルに存在するすべての並べ替え順は、出力テーブル内のそれぞれのレコード セットで別々に保持されます。

たとえ両方のテーブルのレコードが並べ替えられていても、結果として生じる結合されたテーブルは、並べ替えられていないテーブルと見なされます。これは、抽出されたレコードを出力先テーブルの末尾にグループとして追加するときに、出力先テーブルの既存の並べ替え順が何も考慮されないからです。

たとえば、月次または四半期テーブルを追加して年次テーブルを作成する場合は、月次または四半期データの内部並べ替えは保持されます。必要であれば、追加操作を実行した後で、出力テーブルを並べ替えることもできます。

フィールドの順序の使用方法

共通フィールド

ソーステーブルの共通フィールドは、同じ順序になっていなくても追加できます。

たとえば、以下のフィールドは順序が異なっていても正常に追加されます。

テーブル	フィールド
テーブル1	ラストネーム ファーストネーム ミドルネーム
テーブル2	ファーストネーム ミドルネーム ラストネーム

出力テーブル内でのフィールドの順序は、APPEND コマンドに指定した最初のテーブルによって決まります。したがって、上の例では、出力テーブルの順序は次のようになります。

- ラストネーム | ファーストネーム | ミドルネーム

共通でないフィールド

ソーステーブル内の共通でないフィールドは、選択したテーブルグループに出現している順に出力テーブルに出力されます。

たとえば、以下の2つのテーブルを追加する場合を考えます。

テーブル	フィールド
テーブル1	役職 ラストネーム ファーストネーム ミドルネーム
テーブル2	ファーストネーム ミドルネーム ラストネーム 生年月日

出力テーブルでの順序は次のようになります。

- 役職 | ラストネーム | ファーストネーム | ミドルネーム | 生年月日

代替列見出し

ソーステーブルの代替列見出しは、出力テーブルにも出力されます。複数のソーステーブルで同じフィールドに代替列見出しがある場合は、最初に選択したテーブルの代替列見出しが優先的に出力されま

す。

ASSIGN コマンド

変数を作成し、変数に値を割り当てます。

構文

```
ASSIGN 変数名 = 値 <IF テスト>
```

メモ

ASSIGN キーワードを明示的に指定することをお勧めします。スクリプトが読みやすくなりやすくなります。ただし、ASSIGN を省略し、次のように指定することもできます。

```
変数名 = 値
```

パラメーター

名前	説明
変数名	<p>値を割り当てる変数の名前。変数自体が存在していない場合は、変数が作成されます。変数が既に存在する場合は、新しい値で更新されます。</p> <p>代入変数に使用される変数の名前に、<code>@</code>のような英語以外の文字は使用しないでください。変数名に英語以外の文字が含まれていると、変数の代入が失敗します。</p> <p>メモ</p> <p>変数の名前は、31文字までの英数字に制限されます。名前にはアンダースコア文字(<code>_</code>)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
値	<p>変数に割り当てる値。新しい変数が作成される場合、変数の型は値のデータ型に基づいて決定されます。</p>
IF テスト 省略可能	<p>変数を作成したり変数に値を割り当てたりするには true となる必要のある条件式。</p>

例

変数に値を割り当てる

現在のレコード内の **Amount** フィールドの値を変数 `v_current_amount` に割り当てるとします。

```
ASSIGN v_current_amount = Amount
```

変数に条件付きで値を割り当てる

変数 `v_counter` が 10 未満の場合にのみ、変数 `v_quantity` の値を 1 に更新する必要があります。

`v_counter` が 10 以上の場合は割り当てが行われず、`v_quantity` の値は変わりません。

オプションの `ASSIGN` キーワードは省略されている点に注目してください。

```
v_quantity = 1 IF v_counter < 10
```

備考

変数の使用の詳細については、"ACLScript での変数の操作" ページ 1554 を参照してください。

変数の期間

変数は現在の ACL Analytics セッションの間のみ有効です。通常、Analytics プロジェクトを閉じると、プロジェクトのすべての変数が自動的に削除されます。

変数を Analytics プロジェクトで永続的に保存する場合は、変数名の先頭にアンダースコア(_) を付けてください。

```
ASSIGN 変数名 = 値
```

演算フィールドまたは GROUP で使用される変数の再割り当て

以下の場合に既存の変数に値を割り当てると、その新しい値は割り当てられるものの、前の値の長さおよび小数点以下桁数は保持されます。

- 変数が演算フィールド内で使用されている場合
- GROUP 内で変数に値が再度割り当てられる場合

必要に応じて、新しい値の長さを伸ばしたり切り詰めたり、小数点以下桁数が調整されます。

上記以外の場合に変数に値を再度割り当てると、前の値、その長さおよび小数点以下桁数は上書きされます。

Analytics コマンドによって作成される変数

Analytics のダイアログボックスで情報を入力する、またはスクリプトを実行することで、特定のコマンドが実行されると、Analytics によってシステム変数が自動的に作成されます。これらの変数とそれに含まれる値は、後続の Analytics コマンドを処理する際に利用できます。

システム変数の値は、同じコマンドを再度実行すると、更新された値に置き換わります。

詳細については、"Analytics コマンドによって作成されたシステム変数" ページ 1582を参照してください。

BENFORD コマンド

フィールドに表示される最初の桁(1 ~ 9) や最初の桁の組み合わせの数をカウントし、実数と予測数を比較します。予測数はベンフォードの法則によって計算されます。

構文

```
BENFORD <ON> 数値型フィールド <LEADING n> <IF テスト> <BOUNDS> <TO {SCREEN|テーブル名|GRAPH|PRINT}> <LOCAL> <HEADER ヘッダーテキスト> <FOOTER フッターテキスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <APPEND> <OPEN>
```

パラメーター

名前	説明
ON 数値フィールド	<p>分析する数値フィールド。</p> <p>メモ 取引金額などの「自然発生数」を含むフィールドを選択します。ベンフォード分析は、いかなる方法でも制約を受けた数値データには適していません。詳細については以下を参照 "ベンフォード分析を使用してテストできるデータ" ページ 1665</p>
LEADING <i>n</i> 省略可能	<p>分析する先頭桁数。<i>n</i>の値は1から6の範囲で指定します。</p> <p>LEADING を省略した場合は、デフォルト値の1が使用されます。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
BOUNDS 省略可能	<p>出力結果に計算された上限および下限値を含めます。</p> <p>出力結果の2つ以上のカウントが境界のいずれかを超える場合、データは操作された可能性があります、調査が必要になります。</p>
TO SCREEN テーブル名 GRAPH PRINT	<p>コマンドの結果を送信する場所:</p> <ul style="list-style-type: none"> SCREEN -- Analytics の表示領域に結果を表示します

名前	説明
省略可能	<p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例: <code>TO "Output.FIL"</code> デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <ul style="list-style-type: none"> ◦ GRAPH - は結果をグラフに表示し、それを Analytics の表示領域に表示します ◦ 印刷 - 通常使うプリンターに結果を送信します
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。 ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。 フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲	<p>処理するレコード数:</p>

名前	説明
省略可能	<ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>

例

結果をグラフに出力する

Amountフィールドに対してBENFORDコマンドを実行し、その結果をグラフに出力します。

```
BENFORD ON Amount LEADING 2 BOUNDS TO GRAPH
```

備考

このコマンドの動作の詳細については、「ベンフォード分析の実行」ページ 1373を参照してください。

ベンフォード分析を使用してテストできるデータ

ベンフォード分析は、会計金額、取引金額、費用、またはアドレス番号などの「自然発生数」から構成される数値データをテストする目的でのみ、使用してください。ベンフォード分析は、いかなる方法でも制

約を受けた数値データには適していません。

ベンフォード分析に適した数値データを特定するには、次のガイドラインに従ってください。

- **データセットのサイズ** - 有効は分布をサポートするには、データセットが十分に大きいサイズである必要があります。ベンフォード分析では、499レコード以下の場合には、信頼できる結果が得られない場合があります。
- **先頭の数の要件** -1 ~ 9 のすべての数値が、先頭の数として発生する可能性がなければなりません。
- **先頭の数のおよび分析対象の追加の数の要件** -0 ~ 9 のすべての数値が、先頭から2番目の数および分析対象の追加の数として発生する可能性がなければなりません。
- **制約されたデータ** - あらかじめ定義されたパターンに従って割り当てまたは生成された数値データは、ベンフォード分析に適していません。たとえば、次の分析では、ベンフォード分析を使用しないでください。
 - 連番の小切手または請求書番号
 - 特定のパターンにマッピングされる社会保障番号または電話番号
 - 特定の数字が出現しない範囲がある番号体系
- **乱数** -乱数生成器で生成された数値は、ベンフォード分析に適していません。

CALCULATE コマンド

1つ以上の式の値を計算します。

構文

```
CALCULATE {式 <AS 結果ラベル> <,...n>
```

パラメーター

名前	説明
式	<p>計算する式。</p> <p>式は次の4種類のいずれかにすることができます。</p> <ul style="list-style-type: none">○ 文字○ 数値○ 日付時刻○ 論理 <p>式が複数ある場合は、カンマで区切ります。</p> <pre>CALCULATE 4.7 * 18.5, 1 + 2, "a" + "b"</pre>
AS 結果ラベル 省略可能	<p>画面と Analytics コマンド ログに表示する結果の名前。</p> <p>結果ラベルパラメーターには、引用符で囲まれた文字列か有効な文字式を指定する必要があります。</p> <p>このパラメーターを指定しない場合は、計算を行う式が結果名として使用されます。</p>

例

単純な計算を行う

CALCULATE を使用して、4.70 を 18.50 で乗算し、結果の 86.95 を返します。

```
CALCULATE 4.70 * 18.50
```

計算結果に名前を付ける

定義済みの販売価格と単価のフィールドを使って、現在選択されているレコードの粗利益を導出するには、CALCULATE を使用します。

```
CALCULATE 販売価格 - 単価 AS "Margin"
```

結果は画面とログで、"Margin" と表示されます。

備考

機能の仕組み

CALCULATE には、Analytics の関数、変数、および現在選択されているレコード内データにアクセスしながら計算する機能があります。

コマンドの出力

CALCULATE をどこで実行するかによって、異なる場所に結果が出力されます。

- コマンドラインでコマンドを入力した場合：-結果は画面に表示されます。
- スクリプト内でコマンドを使用した場合：-結果はログに記録されます。

結果ラベルパラメーターの値は変数ではないのでスクリプト内で使用できません。これは単に画面またはログで計算を識別するために使用されます。

出力における小数点以下桁数

数値計算の結果の小数位は、式の中で小数位が最も多い式要素と同じになります。

1 が返されます。

```
CALCULATE 365/52/7
```

1.0027 が返されます。

```
CALCULATE 365.0000/52/7
```

テーブル入力の操作

式にフィールド値が含まれる場合は、そのフィールドが属するテーブルを開いておく必要があります。CALCULATE コマンドで分析するレコードに移動するためには、FIND、SEEK、または LOCATE コマンドを使用することができます。

CLASSIFY コマンド

文字または数値フィールドの等しい値に基づいてレコードをグループ化する各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。

構文

```
CLASSIFY <ON> キー フィールド <SUBTOTAL 数値フィールド <...n>|SUBTOTAL ALL <EXCLUDE
数値フィールド <...n>> <INTERVALS 数値> <SUPPRESS> <TO {SCREEN|テーブル名
|GRAPH|PRINT}>{15}<LOCAL> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
<HEADER ヘッダー テキスト> <FOOTER フッターテキスト> <KEY 内訳フィールド> <OPEN>
<APPEND> <STATISTICS>
```

パラメーター

名前	説明
ON キーフィールド	分類する文字または数値フィールド。 キーフィールドの最大長は2048文字です。 2048文字より長いキーフィールドを使用してテーブルを分類化したい場合は、SUMMARIZEコマンドを使用します。SUMMARIZEコマンドにはキーフィールドの長さ制限はありません。
SUBTOTAL 数値フィールド <...n> SUBTOTAL ALL 省略可能	グループごとに小計を計算する1つ以上の数値フィールドまたは式。 複数のフィールドはスペースで区切る必要があります。テーブル内のすべての数値フィールドについて小計を求める場合はALLを指定します。
EXCLUDE 数値フィールド 省略可能	SUBTOTAL ALLを使用するときのみ有効です。 コマンドから除外するフィールド。EXCLUDEでは、指定したフィールドを除外して、SUBTOTAL ALLを微調整できます。 EXCLUDEはSUBTOTAL ALLの直後に置く必要があります。例： <pre>SUBTOTAL ALL EXCLUDE field_1 field_2</pre>
INTERVALS 数 省略可能	出力結果のグループの最大数。 分類化されるフィールドにある同一値セットの数が指定された最大数を超える場合は、列の先頭から数えた同一値セット数のみの同一値セットが使用されます。 最大数を超える同一値セットは、"OTHER"(その他)というグループにまとめられます。

名前	説明
	<p>INTERVALS を省略した場合は、分類化されるフィールドの同一値セットごとに1つのグループが作成されます。</p> <p>メモ このパラメーターは Analytics のユーザー インターフェイスには実装されていません。ACLScript 構文の一部としてのみ使用することができます。</p>
<p>SUPPRESS 省略可能</p>	<p>メモ INTERVALS も指定されていない場合は使用できません。SUPPRESS は Analytics のユーザー インターフェイスには実装されていません。スクリプトまたはコマンドラインで、ACLScript 構文の一部としてのみ使用することができます。</p> <p>INTERVALS で指定された最大数を超える同一値セットは、コマンド出力から除外します。</p>
<p>TO SCREEN テーブル名 GRAPH PRINT</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <ul style="list-style-type: none"> ◦ GRAPH - は結果をグラフに表示し、それを Analytics の表示領域に表示します ◦ 印刷 - 通常使うプリンターに結果を送信します
<p>LOCAL 省略可能</p>	<p>Analytics プロジェクトと同じ場所に出力ファイルを保存します。</p>

名前	説明
	<p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
KEY ブレークフィールド 省略可能	<p>小計計算をグループ化するフィールドまたは式。ブレークフィールドの値が変わるたびに、小計が計算されます。</p> <p>ブレークフィールドは、文字フィールドか式である必要があります。指定できるフィールドは1つだけですが、1つ以上のフィールドを含んでいる式を使用することができます。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p>

名前	説明
	<p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
STATISTICS 省略可能	<p>メモ</p> <p>SUBTOTALも指定されていない場合は使用できません。</p> <p>すべてのSUBTOTALフィールドの平均値、最小値、および最大値を計算します。</p>

例

顧客ごとの総取引額

Customer_Number(顧客番号)フィールドに基づいてAr(売掛金)テーブルを分類化し、Trans_Amount(取引額)フィールドの小計を求めたいとします。出力結果は、顧客ごとにグループ化され、各顧客の総取引額が含まれます。

```
OPEN Ar
CLASSIFY ON Customer_Number SUBTOTAL Trans_Amount TO "Customer_
total.FIL"
```

顧客ごとの取引額の合計、平均、最小、および最大

前の例と同様に、Customer_Number(顧客番号)フィールドに基づいてAr(売掛金)テーブルを分類化し、Trans_Amount(取引額)フィールドの小計を求めます。

今回は、各顧客の取引額の平均値、最小値、および最大値を計算するためにSTATISTICSを追加します。

```
OPEN Ar
CLASSIFY ON Customer_Number SUBTOTAL Trans_Amount TO "Customer_
stats.FIL" STATISTICS
```

同じ請求金額

AP_Trans(買掛取引)テーブルに2回以上出現する請求金額を特定したいとします。

それには、テーブルを **Invoice_Amount**(請求金額) フィールドで分類化します。出力結果は請求金額によってグループ化され、2回以上出現する請求金額を特定するために使用できる関連カウントとともに表示されます。

```
OPEN AP_Trans
CLASSIFY ON Invoice_Amount TO "Grouped_invoice_amounts.FIL" OPEN
SET FILTER TO COUNT > 1
```

備考

このコマンドの動作の詳細については、「データの分類化」ページ 1319を参照してください。

機能の仕組み

CLASSIFY は、文字または数値フィールドに同じ値を持つレコードをグループ化します。

出力にはグループごとに、ソーステーブルのうち、そのグループに属するレコードの数が記録された特別なレコードが含まれます。

並べ替えとCLASSIFY

CLASSIFY では、データは並べ替えられていなくても処理できます。出力は、自動的に昇順に並べ替えられます。

自動生成された小計と統計フィールドの名前

STATISTICS を使用して、1つ以上のSUBTOTAL フィールドで統計演算を実行し、結果を Analytics テーブルに出力する場合は、パラメーターによって自動生成されたフィールドの名前は次のようになります。

コマンド

自動生成されたフィールドの説明	出力テーブルのフィールド名	出力テーブルの列見出し(表示名)
小計フィールド	集計対象となる、ソーステーブルのフィールド名	Total + 集計対象となる、ソーステーブルの代替列見出し
平均フィールド	a_ 集計対象となる、ソーステーブルのフィールド名	Average + 集計対象となる、ソーステーブルの代替列見出し
最小フィールド	m_ 集計対象となる、ソーステーブルのフィールド名	Minimum + 集計対象となる、ソーステーブルの代替列見出し
最大フィールド	x_ 集計対象となる、ソーステーブルのフィールド名	Maximum + 集計対象となる、ソーステーブルの代替列見出し

CLOSE コマンド

Analytics のテーブル、インデックス ファイル、またはログ ファイルを閉じ、**スクリプト記録セッション**を終了します。

構文

```
CLOSE <テーブル名|PRIMARY|SECONDARY|INDEX|LOG|LEARN>
```

パラメーター

名前	説明
テーブル名 PRIMARY SECONDARY INDEX LOG LEARN 省略可能	<p>閉じる対象となる項目は次のとおりです。</p> <ul style="list-style-type: none">○ テーブル名 - : 閉じる対象となる Analytics テーブルの名前○ PRIMARY - を指定すると、Analytics の主テーブルが閉じられます。 <p>CLOSE をパラメーターなしで使用すると主テーブルを閉じます。</p> <ul style="list-style-type: none">○ SECONDARY - では、Analytics の副テーブルが閉じられます。○ INDEX - では、Analytics テーブルに適用されている現在のインデックスが閉じられます。○ LOG - では、SET LOG コマンドを使用してデフォルト以外のログファイルが指定されている場合に、ログファイルをデフォルトのコマンド ログに再設定します。○ LEARN - では、アクティブな スクリプトレコーダー セッションを終了し、セッションが記録されたスクリプト ファイルを保存するかどうかを自身で確認するようにします。 <p>LEARN はスクリプト内で使用することができますが、その目的とする用途はコマンドラインにあります。スクリプトレコーダーは、Analytics のユーザー インターフェイスでダイアログボックスを使用して実行されるコマンドの ACLScript 構文を記録します。</p>

例

名前によってテーブルを閉じる

Inventory という名前のテーブルを閉じるには、次のようにします。

```
CLOSE Inventory
```

種類によってテーブルを閉じる

現在の副テーブルを閉じるには、次のようにします。

```
CLOSE SECONDARY
```

デフォルトの Analytics コマンド ログに戻す

スクリプトのデータ検証フェーズを別のログファイルに取り込んだ後で、ログファイルをデフォルトのコマンド ログに再設定するには、次のようにします。

```
SET LOG TO "DataVerificationPhase.log"  
COMMENT データ検証コマンドの実行  
CLOSE LOG
```

備考

CLOSE を使用しない場合とは

Analytics テーブルは通常、閉じる必要がありません。アクティブな Analytics テーブルは、ユーザーが別のテーブルを開けば自動的に閉じられるからです。主テーブルも、OPEN コマンドや QUIT コマンドを実行する前に自動的に閉じられます。

CLOSE で Analytics プロジェクトを閉じることはできません。代わりに QUIT を使用します。

関連するフィールドおよびテーブル

主テーブルや副テーブルを閉じると、関連するすべてのフィールド定義がメモリから解放されます。テーブルが閉じられる前に、テーブルレイアウトへの変更が保存されます。

テーブルの関連付けが定義された Analytics プロジェクトで CLOSE コマンドを実行すると、主テーブルと副テーブルがともに閉じられます。また、関連するテーブルも閉じられます。

CLUSTER コマンド

1 つ以上の数値フィールドの類似した値に基づいて、レコードをクラスターにグループ化します。クラスターは単次元または多次元です。

メモ

Analytics を 32 ビット コンピューターで実行する場合は、CLUSTER コマンドはサポートされません。このコマンドに必要な演算は、プロセッサの負荷が高いため、64 ビット コンピューターの方が適しています。

構文

```
CLUSTER ON キー フィールド <...n> KVALUE クラスター数 ITERATIONS 繰り返し数
INITIALIZATIONS 初期化数 <SEED シード値> <OTHER フィールド <...n>> TO テーブル名
<IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> OPEN {キーワードなし
|NOCENTER|NOSCALE}
```

パラメーター

名前	説明
ON キーフィールド<...n>	クラスター化する 1 つ以上の数値フィールド。複数のフィールドはスペースで区切る必要があります。
KVALUE クラスター数	出力結果で生成されるクラスター数。 詳細については、"クラスター数 (K 値) の選択" ページ 1368 を参照してください。
ITERATIONS 繰り返し数	クラスター計算が再実行される最大回数。
INITIALIZATIONS 初期化数	ランダム中心の初期セットを生成する回数。
SEED シード値 省略可能	Analytics の乱数ジェネレーターを初期化するために使用するシード値。 SEED を省略した場合は、シード値がランダムに選択されます。
OTHER フィールド<...n> OTHER ALL 省略可能	出力に含める 1 つ以上の追加フィールド。 <ul style="list-style-type: none"> OTHER フィールド <...n> 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。

名前	説明
	<ul style="list-style-type: none"> OTHER ALL - はテーブルのすべてのフィールドに含まれます フィールドは、テーブルレイアウトに表示される順序で含まれます。 <p>メモ キーフィールドは自動的に出力フィールドに含まれます。ただし、NOSCALEを指定しないと、値が調整されます。OTHERを使用すると、2番目の調整されていないキーフィールドを含めることができます。</p>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> テーブル名 - は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル (.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> <code>TO "C:\Output.FIL"</code> <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は 64 文字の英数字 (.FIL 拡張子を含まない) に制限されています。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター (WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>

名前	説明
OPEN 省略可能	コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。
キーワードなし NOCENTER NOSCALE	<p>クラスターを計算する前にキーフィールドの数値を前処理するメソッド。</p> <ul style="list-style-type: none"> ◦ キーワードなし - キーフィールド値の中央を平均ゼロ(0)にし、標準偏差で除算して調整します。これは、値をz得点相当(標準得点)に変換するプロセスです。 ◦ NOCENTER - 標準偏差で除算してキーフィールド値を調整しますが、中央を平均ゼロ(0)にしません。 ◦ NOSCALE - 元のキーフィールド値を使用します。中央合わせや調整は行いません。 <p>詳細については、"データ処理方法の指定" ページ 1370を参照してください。</p>

例

請求金額でのクラスター

Invoice_Amount フィールドで売掛金テーブルを階層化するほかに、同じフィールドでクラスター化することも決定します。

- 階層化により、\$1000 間隔などのあらかじめ定義された数値の境界を使用して、金額が階層にグループ化されます。
- クラスターにより、前もって数値の境界を決定せずに、データに存在する金額の有機グループを検出します。

```
OPEN Ar
CLUSTER ON Invoice_Amount KVALUE 8 ITERATIONS 30 INITIALIZATIONS 10
OTHER No Due Date Ref Type TO "Clustered_invoices" NOSCALE
```

各出力クラスターに含まれるレコード数を簡単に検出する方法として、Clustered_invoices 出力テーブルを **Cluster** フィールドで分類します。

```
OPEN Clustered_invoices
CLASSIFY ON Cluster TO SCREEN
```

備考

このコマンドの動作の詳細については、"データのクラスター" ページ 1364を参照してください。

COMMENT コマンド

処理に影響を与えずに、スクリプトに注釈を追加します。

構文

単一行コメント

```
COMMENT コメントテキスト
```

複数行コメント

```
COMMENT
  コメントテキスト
  <...n>
<END>
```

メモ

コメント テキストの行の前には、`^` キャラット文字を使用しないでください。キャラットは .acl プロジェクト ファイルにおいて特殊な使用方法があり、コメントの前にキャラットを置くと、個別の行のコメント テキストは保存されません。

パラメーター

名前	説明
コメントテキスト	追加するコメント。 <ul style="list-style-type: none"> 単一行のコメント -は、改行なしでコメント テキスト全体を入力します。 複数行のコメント -は、COMMENT コマンドの直後の行で始まるコメント テキストを任意の数だけ入力します 個別の行でEND キーワードを使用するか、改行を入れると、複数行コメントが終了します。
END 省略可能	複数コメント行を伴うCOMMENT コマンドの終了。 ENDを使用する場合は、これを最後のコメント行の直後の行に入力する必要があります。ENDを使用しない場合、最後のコメント行の直後には空白行を挿入する必要があります。

例

単一行コメント

単一行コメントは、スクリプトを将来保守するユーザーへの説明を追加するため、各コマンドの前に使用します。

```
COMMENT 標準偏差と平均を生成します。  
STATISTICS ON %v_amt% STD TO SCREEN NUMBER 5  
COMMENT 標準偏差と平均を格納するためにフィールドを作成します。  
DEFINE FIELD Standard_Dev COMPUTED STDDEV1  
DEFINE FIELD Average COMPUTED AVERAGE1
```

複数行コメント

作成する各スクリプトの先頭に、そのスクリプトの目的を説明する複数行コメントを記述することができます。

```
COMMENT  
この分析は取引日付値が等しいか、あるいは一日違いで  
共通の取引者 ID(ベンダー ID あるいは業者 ID など)を  
持つ複数のレコードを特定します。  
この分析は納品書の分割、注文書の分割、請求書の分割、  
そして企業カード取引の分割に使用できます。  
END
```

備考

COMMENT の用途

COMMENT の用途は、スクリプトの目的や使用するロジックに関する情報、またスクリプトで必要な入力、定義した各変数の目的などの情報を含めることです。

スクリプトが実行されるたびに、スクリプト中のコメントも Analytics コマンド ログに書き込まれます。

COUNT コマンド

現在のビュー内のレコードの合計数や、指定した条件を満たすレコード件数をカウントします。

構文

```
COUNT <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
```

パラメーター

名前	説明
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>

Analytics の出力変数

名前	含む
COUNT n	<p>コマンドによって計算されたレコード数。</p> <ul style="list-style-type: none"> 変数名が COUNT1 の場合、それは最近実行されたコマンドのレコード数を格納しています。 変数名 COUNTn の n が 1 より大きい場合、変数は GROUP コマンドの内で実行されたコマンドのレコード数を格納しています。 <p>n の値は GROUP 内のコマンドの行番号に基づいて割り当てられます。たとえば、コマンドが GROUP コマンドの 1 行下にある場合、値は COUNT2 が割り当てられます。コマンドが GROUP コマンドの 4 行下にある場合、値は COUNT5 が割り当てられます。</p>

例

COUNT1 を保管する

COUNT コマンドの結果は COUNT1 出力変数に保管されます。この変数の値を取得してユーザー定義変数に保管することができます。

COUNT コマンドが実行されるたびに、COUNT1 変数が上書きされます。したがって、この変数の値は、テーブルにフィルターを適用した後でこのコマンドが再度実行される前に、ユーザー定義変数に格納しておく必要があります。

```
OPEN CustomerAddress
COUNT
TotalRec = COUNT1
SET FILTER TO ModifiedDate > '20100101'
COUNT
TotalFilteredRec = COUNT1
```

備考

COUNT の用途

COUNT コマンドの用途は、Analytics テーブル内のレコード数をカウントしたり、特定のテスト条件を満たすレコードのみをカウントしたりすることです。テスト条件を指定しないと、Analytics テーブル内のレコードの合計が表示されます。

COUNT へのフィルターの影響

ビューにフィルターが適用されている場合は、フィルターリング条件の適用後、ビュー内に残っているレコード数がカウントされます。

CREATE LAYOUT コマンド

特定のスクリプト作成状況で必要になる場合がある、空の Analytics テーブルレイアウトを作成します。

構文

```
CREATE LAYOUT レイアウト名 WIDTH 文字 <RECORD 0|RECORD 1>
```

パラメーター

名前	説明
レイアウト名	レイアウトの名前。
WIDTH 文字数	レコード長の文字単位。
RECORD 0 RECORD 1 省略可能	<ul style="list-style-type: none">RECORD 0 を指定するか、またはこのパラメーターを省略した場合は、レコードもソースデータファイルもないテーブルレイアウトが作成されます。RECORD 1 を指定した場合は、1 件の空レコードとレイアウト名.fil というソースデータファイルを持つテーブルレイアウトが作成されます。

例

レコードのない空のテーブルレイアウトを作成する

レコード長が 100 文字である空のテーブルレイアウトを作成するには、次のようにします。

```
CREATE LAYOUT empty_table WIDTH 100
```

1 件のレコードを含む空のテーブルレイアウトを作成する

次を作成します。

- 1 件の空のレコードを含む空のテーブルレイアウト
- 50 文字のレコード長
- 対応する Analytics データ ファイルの名前は `empty_table.fil`

```
CREATE LAYOUT empty_table WIDTH 50 RECORD 1
```

備考

作成されるこの空のテーブルレイアウトには、**Field_1** という文字フィールドがあります。このフィールドの長さは、WIDTH で指定したレコード長と同じです。

CROSSTAB コマンド

2つ以上の文字または数値フィールドの値の等しい組み合わせに基づいてレコードをグループ化し、結果のグループを行と列のグリッドに表示します。各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。

構文

```
CROSSTAB {ON 行フィールド <...n>|ON ALL <EXCLUDE フィールド名 <...n>>} COLUMNS 列
フィールド <SUBTOTAL 数値フィールド <...n>|SUBTOTAL ALL <EXCLUDE 数値フィールド
<...n>>} TO {SCREEN|テーブル名|ファイル名|GRAPH|PRINT} <LOCAL> <IF test> <WHILE
テスト> <FIRST 範囲|NEXT 範囲> <APPEND> <COUNT> <OPEN> <HEADER ヘッダー テキスト>
<FOOTER フッター テキスト>
```

パラメーター

名前	説明
ON 行フィールド <...n> ON ALL	<p>結果として生じる行列グリッドの行に使用する、1つ以上の文字または数値フィールドまたは式。</p> <ul style="list-style-type: none"> ON 行フィールド <...n> - 指定したフィールドを使用します 複数のフィールドはスペースで区切る必要があります。また、異なるデータ型を指定できません。 1つ以上のフィールドを使用する場合、フィールドはリストの順序で含まれます。 ON ALL - テーブルのすべてのフィールドを使用します すべてのフィールドを使用する場合、フィールドはテーブルレイアウトに出現する順序で含まれます。
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p> <pre>ON ALL EXCLUDE field_1 field_2</pre>
COLUMNS 列フィールド	<p>結果として生じる行列グリッドの列に使用する、文字または数値フィールドまたは式。列に使用するフィールドや式は、1つしか指定できません。</p>

名前	説明
SUBTOTAL 数値フィールド <...n> SUBTOTAL ALL 省略可能	グループごとに小計を計算する1つ以上の数値フィールドまたは式。 複数のフィールドはスペースで区切る必要があります。テーブル内のすべての数値フィールドについて小計を求める場合はALLを指定します。
EXCLUDE 数値フィールド 省略可能	SUBTOTAL ALLを使用するときのみ有効です。 コマンドから除外するフィールド。EXCLUDEでは、指定したフィールドを除外して、SUBTOTAL ALLを微調整できます。 EXCLUDEはSUBTOTAL ALLの直後に置く必要があります。例： <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> SUBTOTAL ALL EXCLUDE field_1 field_2 </div>
TO SCREEN テーブル名 ファイル名 GRAPH PRINT	コマンドの結果を送信する場所： <ul style="list-style-type: none"> ○ SCREEN -- Analyticsの表示領域に結果を表示します <div style="margin-top: 10px;"> <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> </div> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となるAnalyticsテーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> デフォルトでは、テーブルデータファイル(.fil)は、Analyticsプロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <div style="margin-top: 10px;"> <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> </div> <ul style="list-style-type: none"> ○ ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。例：<code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analyticsプロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> ○ GRAPH - は結果をグラフに表示し、それをAnalyticsの表示領域に表示します

名前	説明
	<ul style="list-style-type: none"> 印刷 – 通常使うプリンターに結果を送信します
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> 同じフィールド 同じフィールド順序 一致するフィールドが同じ長さ 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
COUNT 省略可能	<p>レコード数が列に挿入されます。カウントは、SUBTOTAL を使用する場合に便利です。 小計フィールドが何も選択されていない場合は、自動的にカウントが含まれます。</p>
OPEN	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>

名前	説明
省略可能	
HEADER ヘッダーテキスト 省略可能	レポートの各ページの最上部に挿入されるテキスト。 ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。
FOOTER フッターテキスト 省略可能	レポートの各ページの最下部に挿入されるテキスト。 フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。

例

SUBTOTAL を使って売掛金 (Ar) テーブルをクロス集計する

顧客番号 (Customer_Number) フィールドと取引タイプ (Trans_Type) フィールドを基準にして売掛金 (Ar) テーブルをクロス集計したいとします。また、取引金額 (Trans_Amount) フィールドの値の小計も求めたいとします。

出力は顧客別にグループ化され、さらに各顧客内では取引タイプ別にグループ化されます。出力には、取引タイプごとに各顧客の総取引額が含まれます。

```
OPEN Ar
CROSSTAB ON Customer_Number COLUMNS Trans_Type SUBTOTAL Trans_Amount
COUNT TO SCREEN
```

売掛金 (Ar) テーブルをクロス集計することで重複取引を検出する

売掛金 (Ar) テーブルから重複取引の証拠を検出したいとします。

これを行うには、売掛金テーブルを取引金額 (Trans_Amount) フィールドと取引タイプ (Trans_Type) フィールドに基づいてクロス集計します。出力では、取引タイプごとに同一の取引額がグループ化され、カウントされます。

```
OPEN Ar  
CROSSTAB ON Trans_Amount COLUMNS Trans_Type TO SCREEN
```

備考

このコマンドの動作の詳細については、"データのクロス集計" ページ 1337を参照してください。

機能の仕組み

CROSSTAB は、2 つ以上の文字または数値フィールドの値で、同じ組み合わせを持つレコードをグループ化します。

出力には、ピボット テーブルと似た行と列で構成されるグリッドが含まれます。出力にはグループごとに1つの行と列の交差が含まれるほか、そのグループに属するソーステーブル内のレコードの数が含まれます。

並べ替えとCROSSTAB

CROSSTAB では、データは並べ替えられていなくても処理できます。出力の行フィールドと列フィールドは、どちらも自動的に昇順に並べ替えられます。

複数の行フィールドを指定する場合は、フィールドはネストされた並べ替えを使用します。ネストの順序は、指定するフィールド順序、または ON ALL を使用している場合はテーブルレイアウトのフィールド順序に従います。

CVSEVALUATE コマンド

従来の変数サンプリングのために、サンプル分析の結果を母集団全体に対して推定するための4つの方法があります。

構文

```
CVSEVALUATE BOOKED 簿価フィールド AUDITED 監査値フィールド ETYPE
{MPU|DIFFERENCE|RATIO SEPARATE|RATIO COMBINED|ALL} STRATA 境界値 <,...n>
POPULATION 層カウント,層簿価 <,...n> CONFIDENCE 信頼度 CUTOFF 金額,确实性層数,
确实性層簿価 ERRORLIMIT 数値 PLIMIT {BOTH|UPPER|LOWER} <BCUTOFF 値,确实性層
数,确实性層簿価> <TO {SCREEN|ファイル名}>
```

パラメーター

メモ

CVSPREPARE および CVSSAMPLE コマンドの出力結果を CVSEVALUATE コマンドの入力として使用している場合は、多数のパラメーター値が既に指定され、変数に格納されています。詳細については、"CVSPREPARE コマンド" ページ 1698と "CVSSAMPLE コマンド" ページ 1703を参照してください。

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
BOOKED 簿価フィールド	評価で使用する数値型の簿価フィールド
AUDITED 監査値フィールド	評価で使用する数値型の監査金額フィールド
ETYPE MPU DIFFERENCE RATIO SEPARATE RATIO COMBINED ALL	使用する推定タイプ: <ul style="list-style-type: none"> • MPU(平均) • 差異 • 比率分離 • 比率結合 • すべて <p>詳細については以下を参照 "どの推定タイプを使用すればよいですか?" ページ 1696</p>
STRATA 境界値 <,...n>	簿価フィールドを階層化するために使用される上限の境界値

名前	説明
POPULATION 層カウント, 層値 <,...n>	簿価フィールドの各層のレコード数と合計値
CONFIDENCE 信頼度	従来の変数サンプルの準備段階で使用される信頼度
CUTOFF 値, 確実性層数, 確実性層簿価	<ul style="list-style-type: none"> ○ 値: -従来の変数サンプルの準備およびサンプル段階で使用される上位確実性層カットオフ値 ○ 確実性層カウント: -確実性層のレコード数 ○ 確実性層簿価: -確実性層のレコードの合計簿価
ERRORLIMIT 数値	<p>サンプルで想定する最低誤謬数。</p> <p>メモ サンプルを分析したときに見つかった実際の誤謬数がERRORLIMIT数値より小さい場合は、使用可能な評価方法は平均推定のみです。</p>
PLIMIT BOTH UPPER LOWER	<p>使用する精度制限のタイプ:</p> <ul style="list-style-type: none"> ○ BOTH: 上限および下限 ○ UPPER: 上限 ○ LOWER: 下限 <p>詳細については、"CVSPREPARE コマンド" ページ 1698を参照してください。</p>
BCUTOFF 値, 確実性層数, 確実性層簿価 省略可能	<ul style="list-style-type: none"> ○ 値: -従来の変数サンプルの準備およびサンプル段階で使用される下位確実性層カットオフ値 ○ 確実性層カウント: -下位確実性層のレコード数 ○ 確実性層簿価: -下位確実性層のレコードの合計簿価
TO SCREEN ファイル名	<p>コマンドの結果を送信する場所:</p> <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ○ ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code>

例

レコードの母集団全体に対して、サンプルされたデータで見つかった誤謬を推定する

サンプルデータのテストを完了し、見つかった虚偽表示を記録しました。検出したすべての誤謬を母集団に対して投影できます。

以下の例は、サンプル分析の結果から母集団全体について推定するには、推定タイプ DIFFERENCE を使用します。

```
CVSEVALUATE BOOKED invoice_amount AUDITED AUDIT_VALUE ETYPE DIFFERENCE
STRATA 4376.88,9248.74,16904.52,23864.32 POPULATION
1279,3382131.93,898,5693215.11,763,9987014.57,627,12657163.59,479,1334-
6354.63 CONFIDENCE 95.00 CUTOFF 35000.00,36,1334318.88 ERRORLIMIT 6
PLIMIT BOTH TO SCREEN
```

備考

このコマンドの動作の詳細については、「従来の変数サンプリングの誤謬の評価」ページ 1139を参照してください。

どの推定タイプを使用すればよいですか？

使用しなければならない推定タイプは、データの性質、サンプル簿価、サンプル監査金額、およびそれらの関係によって決まります。

ガイドライン

推定タイプの選択に役立つガイドラインを次に示します。

ヒント

様々な推定タイプによって生成された結果同士を比較するには、すべての推定タイプを評価出力対象とするため、「ETYPE ALL」を指定します。

推定タイプ	虚偽表示の存在	虚偽表示のサイズ	簿価の符号	層間の比率の比較
平均推定	虚偽表示がないか、	該当なし	該当なし	該当なし

推定タイプ	虚偽表示の存在	虚偽表示のサイズ	簿価の符号	層間の比率の比較
	<p>非常に少ない虚偽表示しかない。</p> <p>監査したサンプル母集団に虚偽表示がないか、非常に少ない場合に、唯一有効な推定タイプ</p>			
差異	<p>虚偽表示が必要</p> <p>監査したサンプル母集団にはいくつかの虚偽表示が必要です。</p> <p>たとえば、サンプルの5%以上に虚偽表示が含まれているなどの場合です。</p>	<p>虚偽表示が比例しない場合</p> <p>虚偽表示が関連する簿価に比例しない場合、つまり虚偽表示のサイズが関連する簿価のサイズに比例しない場合に適しています。</p> <p>つまり、簿価の大小に関係なく、それに含まれる虚偽表示が大きかったり小さかったりする場合があります。</p>	該当なし	該当なし
比率分離		<p>虚偽表示が関連する簿価に比例する場合</p> <p>虚偽表示が関連する簿価に比例する場合、つまり虚偽表示のサイズが関連する簿価のサイズに比例する場合に適しています。</p> <p>つまり、簿価が小さければ虚偽表示が小さく、簿価が大きければ虚偽表示が大きくなる場合です。</p>	<p>すべての簿価が同じ符号を持つ</p> <p>サンプルの全簿価が同じ符号を持つ、つまりすべてプラスか、すべてマイナスかのどちらかである必要があります。</p>	<p>比率が異なる</p> <p>サンプルの監査金額の平均とサンプル簿価の平均との比率が層間で大きく異なる場合に適しています。</p>
比率結合				<p>比率間の差異があまりない</p> <p>サンプルの監査金額の平均とサンプル簿価の平均との比率が層間であまり差異がない場合に適しています。</p>

CVSPREPARE コマンド

母集団を階層化し、各層の統計的に有効なサンプルサイズを従来の変数サンプリングのために計算します。

構文

```
CVSPREPARE ON 簿価フィールド NUMSTRATA 数値 MINIMUM 層のサンプル サイズの最小値
PRECISION 値 CONFIDENCE 信頼度 <CUTOFF 値> <BCUTOFF 値> NCELLS 数値 PLIMIT
{BOTH|UPPER|LOWER} ERRORLIMIT 数値 <IF テスト> <MINSAMPSIZE 最低サンプル サイズ>
TO {SCREEN|ファイル名}
```

パラメーター

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
ON 簿価フィールド	従来の変数サンプルの準備の基準として使用する数値型の簿価フィールド。
NUMSTRATA 数値	簿価フィールドを数値的に階層化するために使用する層の数。 層の数は下限が1、上限が256です。 NUMSTRATA 1を指定しても CUTOFFを指定していない場合には、サンプルを抽出する前に母集団が階層化されなくなります。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> メモ 層の数は NCELLS に対して指定されたセルの数 55% を超過できません。 </div>
MINIMUM 層サンプルの最低数	各層からサンプリングする最小レコード数。 最低数を指定する理由が特でない限り、デフォルト値ゼロ(0)を使用してください。
PRECISION 値	許容虚偽表示および勘定で想定される虚偽表示の間の差異である金額。 <ul style="list-style-type: none"> ○ 許容虚偽表示 - サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。重大な虚偽表示額とまでは見なされません。 ○ 推定虚偽表示額 - サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。 精度は勘定が公正に表示されていることの許容度の範囲を決定します。

名前	説明
	<p>精度を下げると、サンプルサイズを大きくする必要がある許容度の範囲(誤謬のマージン)が小さくなります。</p>
<p>CONFIDENCE 信頼度</p>	<p>必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。</p> <p>たとえば、95を指定した場合は、サンプルが実際に95%の確率で母集団を代表しているとお客様が信頼したいということを意味します。信頼度は"サンプリングリスク"の補数です。信頼度が95%ということはサンプリングリスクが5%ということと同じです。</p> <ul style="list-style-type: none"> ○ <code>PLIMIT</code> に <code>BOTH</code> を指定した場合には、最低信頼度は10%、最大信頼度は99.5%になります。 ○ <code>PLIMIT</code> に <code>UPPER</code> または <code>LOWER</code> を指定した場合には、最低信頼度は55%、最大信頼度は99.5%になります。
<p>CUTOFF 値 省略可能</p>	<p>上位確実性層のカットオフ値。</p> <p>カットオフ値以上の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。CUTOFFを指定しない場合は、簿価フィールドの最大金額に等しいデフォルトのカットオフ値が使用されるので、上位確実性層にはレコードが追加されなくなります。</p>
<p>BCUTOFF 値 省略可能</p>	<p>下位確実性層のカットオフ値。</p> <p>カットオフ値以下の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。BCUTOFFを指定しない場合は、簿価フィールドの最大金額に等しいデフォルトのカットオフ値が使用されるので、上位確実性層にはレコードが追加されなくなります。</p>
<p>NCELLS 数値</p>	<p>簿価フィールドをあらかじめ階層化するために使用するセルの数。</p> <p>セルの数は、層の数より分割可能な数が少なくなります。階層化の前処理は、層の境界の位置を最適化する内部処理の一部です。最終的に階層化された出力には、セルは保持されません。</p> <p>セルの数は下限が2、上限が999です。</p> <p>メモ セル数は少なくとも層の <code>NUMSTRATA</code> 数の2倍である必要があります。</p>
<p>PLIMIT BOTH UPPER LOWER</p>	<p>使用する精度制限のタイプ。</p> <ul style="list-style-type: none"> ○ <code>BOTH</code> - 次の場合にこのオプションを指定します。 <ul style="list-style-type: none"> ● 勘定全体が過剰表示または過小表示である可能性がある ● 方向のいずれかの虚偽表示が指定された <code>PRECISION</code> を超過するかどうかを推定する ○ <code>UPPER</code> - 次の場合にこのオプションを指定します。 <ul style="list-style-type: none"> ● 全体としての勘定が過小評価されている可能性が高い ● 過小評価の合計金額が指定された <code>PRECISION</code> を超過するかどうかのみを推定する ○ <code>LOWER</code> - 次の場合にこのオプションを指定します。 <ul style="list-style-type: none"> ● 全体としての勘定が過大評価されている可能性が高い ● 過大評価の合計金額が指定された <code>PRECISION</code> を超過するかどうかのみを推定する <p>注意 指定するオプションがわからない場合は、<code>BOTH</code> を指定します。</p>

名前	説明
ERRORLIMIT 数値	<p>サンプルで想定する最低誤謬数。</p> <p>メモ サンプルを分析したときに見つかった実際の誤謬数がERRORLIMITに指定した数値より小さい場合は、使用可能な評価方法は平均推定のみです。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>注意 条件式を指定する場合、サンプルサイズの計算時とサンプルの抽出時とで、同じ条件式を使用する必要があります。 一方の段階で使用した条件を他方の段階で使用しない場合、つまり2つの条件が同一でない場合、サンプリング結果が通常、統計的に無効になります。</p>
MINSAMPLESIZE 最低サンプルサイズ 省略可能	<p>母集団全体からサンプリングする最低レコード数。 最低数を指定する理由が特にない限り、デフォルト値ゼロ(0)を使用してください。</p>
TO SCREEN ファイル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code>

Analytics の出力変数

名前	含む
CONFIDENCE	ユーザーが指定する信頼度。

名前	含む
ERRLIMIT	ユーザーが指定する最低誤謬数。
NSTRATA	ユーザーが指定する層の数。
PLIMIT	ユーザーが指定する精度限度のタイプ。
S_IF	ユーザーが指定する条件式
S_TOP	ユーザーが指定する上位確実性層カットオフ値。何も指定されていない場合は、コマンドで計算された上位層の上限境界。
SAMPLEFIELD	ユーザーが指定する簿価フィールド。
SBOTTOM	ユーザーが指定する下位確実性層カットオフ値。何も指定しない場合は、コマンドで計算された下位層の下限境界。
SBOUNDARY	コマンドで計算されたすべての層の上限境界。上位確実性層や下位確実性層は指定しません。
SPOPULATION	各層のレコード数のカウントと合計値。上位確実性層や下位確実性層は指定しません。
SSAMPLE	コマンドによって計算された、各層のサンプルサイズ。上位確実性層や下位確実性層は指定しません。

例

従来の変数サンプリングの準備

従来の変数サンプリングを使用して、請求書を含む勘定の金額虚偽表示の合計金額を推定することを決定しました。

サンプルを抽出する前に、母集団を階層化し、各層の統計的に有効なサンプルサイズを計算します。

Analytics によって抽出されるサンプルの 95% の時間が全体として母集団を表す信頼度が必要です。

指定された信頼度を使用して、以下の例は、**invoice_amount** フィールドを基準にしてテーブルを階層化し、上位確実性層と各層のサンプルサイズを計算します。

```
CVSPREPARE ON invoice_amount NUMSTRATA 5 MINIMUM 0 PRECISION 928003.97  
CONFIDENCE 95.00 CUTOFF 35000 NCELLS 50 PLIMIT BOTH ERRORLIMIT 6  
MINSAMPLESIZE 0 TO SCREEN
```

備考

このコマンドの動作の詳細については、「従来の変数サンプリングの準備」ページ 1121を参照してください。

数値の長さ制限

従来の変数サンプリングの準備段階では、いくつかの内部計算が行われます。これらの計算では、最大 17 桁の数値がサポートされます。計算結果が 17 桁を超える場合には、その計算結果が出力に含まれなくなるため、サンプリング処理を続行できなくなります。

注意: 17 桁未満のソースデータの数値から、17 桁を超える内部計算結果が生成される場合もあります。

CVSSAMPLE コマンド

従来の変数サンプリング方法を使用して、レコードのサンプルを抽出します。

構文

```
CVSSAMPLE ON 簿価フィールド NUMSTRATA 数値 <SEED シード値> CUTOFF 値 <BCUTOFF 値
> STRATA 境界値 <,...n> SAMPLESIZE 数値 <,...n> POPULATION 層カウント,層値
<,...n> <IF テスト> TO テーブル名
```

パラメーター

メモ

CVSPREPARE コマンドの出力結果を CVSSAMPLE コマンドの入力として使用している場合は、多数のパラメーター値が既に指定され、変数に格納されています。詳細については、「CVSPREPARE コマンド」ページ 1698を参照してください。

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
ON 簿価フィールド	サンプルの基準として使用する数値簿価フィールド。
NUMSTRATA 数値	簿価フィールドを階層化するために使用する層の数。
SEED シード値 省略可能	Analytics の乱数ジェネレーターを初期化するために使用するシード値。 SEED を省略した場合は、シード値がランダムに選択されます。
CUTOFF 値	上位確実性層のカットオフ値。 カットオフ値以上の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。
BCUTOFF 値 省略可能	下位確実性層のカットオフ値。 カットオフ値以上の簿価フィールドの金額が自動的に選択され、サンプルに取り込まれます。
STRATA 境界値 <,...n>	簿価フィールドを階層化するために使用される上限の境界値
SAMPLESIZE 数値 <,...n>	各層からサンプリングするレコード数。
POPULATION 層カウント,	各層のレコード数と各層の合計値。

名前	説明
層値 <...n>	
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>注意 条件式を指定する場合、サンプルサイズの計算時とサンプルの抽出時とで、同じ条件式を使用する必要があります。 一方の段階で使用した条件を他方の段階で使用しない場合、つまり2つの条件が同一でない場合、サンプリング結果が通常、統計的に無効になります。</p>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

Analytics の出力変数

名前	含む
S_TOPEV	<p>ユーザーが指定する上位確実性層カットオフ値。何も指定しない場合は、CVSPREPARE コマンドで以前に計算された上位層の上限境界。</p> <p>また、上位確実性層のレコード数とそれらのレコードの合計金額値も指定します。</p>
SBOTTOMEV	<p>ユーザーが指定する下位確実性層カットオフ値。何も指定しない場合は、CVSPREPARE コマンドで以前に計算された下位層の下限境界。</p> <p>また、下位確実性層と合計金額値のレコード数も指定します。</p>
SBOUNDARYEV	<p>コマンドによってあらかじめ入力、またはユーザーが指定したすべての層の上限の境界。上位確実性層や下位確実性層は指定しません。</p>

名前	含む
SPOPULATION	各層のレコード数のカウントと合計値。上位確実性層や下位確実性層は指定しません。

例

従来の変数サンプリングの抽出

従来の変数サンプリングを使用して、請求書を含む勘定の金額虚偽表示の合計金額を推定します。

母集団を階層化し、各層の統計的に有効なサンプルサイズを計算した後に、サンプルを抽出できます。

以下の例は、`invoice_amount` フィールドを基準にしてレコードの階層化サンプルを抽出し、その階層化されたレコードを `Invoices_sample` でテーブルに出力します。

```
CVSSAMPLE ON invoice_amount NUMSTRATA 5 SEED 12345 CUTOFF 35000.00
STRATA 4376.88,9248.74,16904.52,23864.32,35000.00 SAMPLESIZE
37,36,49,36,39 POPULATION
1279,3382131.93,898,5693215.11,763,9987014.57,627,12657163.59,479,1334-
6354.63 TO "Invoices_sample"
```

備考

このコマンドの動作の詳細については、「従来の変数サンプリングの実行」ページ 1131を参照してください。

システム生成フィールド

Analyticsにより、自動的に4つのフィールドが生成され、サンプル出力テーブルに追加されます。サンプルに追加される各レコードのフィールドには、以下の記述的情報が格納されます。

- **STRATUM** - レコードが割り当てられる層の数
- **ORIGIN_RECORD_NUMBER** - ソースデータテーブル内の元のレコード番号
- **SELECTION_ORDER** - 各層においてレコードがランダムに選択された順序
- **SAMPLE_RECORD_NUMBER** - サンプル出力テーブル内のレコード番号

DEFINE COLUMN コマンド

1つ以上の列を作成し、既存のビューに追加します。

構文

```
DEFINE COLUMN ビュー名 フィールド名 <AS 表示名> <POSITION n> <WIDTH 文字> <PIC 書式> <SORT|SORT D> <KEY> <PAGE> <NODUPS> <NOZEROS> <LINE n>
```

パラメーター

名前	説明
ビュー名	列を追加するビュー。
フィールド名	列を作成するフィールド。 関連テーブルのフィールドを使用する場合は、「 <code>テーブル名.フィールド名</code> 」の形式でフィールド名を指定します。
AS 表示名 省略可能	ビューにおけるフィールドの表示名(代替列見出し)。表示名をフィールド名と同じにしたい場合は、AS を使用しないでください。 表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。
POSITION <i>n</i> 省略可能	ビュー内の列の位置を、左から右へ数えた数字。 <ul style="list-style-type: none"> 省略する場合、列は追加した時点で一番右の列として配置されます。 指定した位置番号が見当たらない場合、列が連続して配置されるように列の位置を調整します。 位置番号が既に使用されている場合、新しい列は現在その位置番号を使用している列の左側に配置されます。
WIDTH 文字数 省略可能	フィールドの表示幅(文字数)。 指定した値によって、Analytics のビューおよびレポートにおけるフィールドの表示幅が決まります。表示幅はデータを変更するものではありませんが、表示幅がフィールドより短い場合にはデータが隠れる可能性があります。 WIDTH を省略した場合には、表示幅は、テーブルレイアウトのフィールドに指定された幅に設定されます。

名前	説明
	<p>メモ</p> <p>WIDTH で指定する文字数は、固定幅の文字数です。実際の文字の幅に関係なく、各文字には同じ空白量が割り当てられます。</p> <p>Analytics のビューでは、固定幅の文字間隔と対応しないプロポーショナルフォントがデフォルトで使用されます。</p> <p>WIDTH の値とビューの文字数との間に 1 対 1 の対応が必要な場合は、オプション] ダイアログボックスの [プロポーショナルフォント] 設定を Courier New などの固定幅フォントに変更することができます。</p>
PIC 書式 省略可能	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソース データの日付時刻値の物理形式 (日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソース データの物理形式と正確に一致する必要があります。たとえば、ソース データが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>
SORT SORT D 省略可能	<p>カラム(列) で並べ替えます。</p> <ul style="list-style-type: none"> 昇順 - SORT 降順 - SORT D
KEY 省略可能	<p>列はレポートのブレイクフィールドに指定されています。列の値が変更されると、レポートは小計され、分割されます。ブレイクフィールドに対し次のような制限があります。</p> <ul style="list-style-type: none"> 文字フィールドか式である必要があります。 フィールドにブレイク列を設定する場合、その列は一番左の列でなければなりません。 ビュー内の最後の列をブレイクフィールドにすることはできません。 複数の列をブレイクフィールドにする場合、追加するブレイクフィールドの左側にある列もすべてブレイクフィールドにしなければなりません。
PAGE 省略可能	<p>ブレイクフィールド値が変化するたびに改ページを挿入します。</p>
NODUPS 省略可能	<p>フィールドの繰り返し値に空白値を使用します。</p> <p>たとえば、請求書レコードごとに顧客名が記載されている場合、各顧客名の最初のインスタンスだけを記載すれば、レポートが読みやすくなるかもしれません。</p>
NOZEROS 省略可能	<p>フィールドのゼロ値に空白値を使用します。</p> <p>たとえば、レポートのフィールド大量のゼロ値がある場合、ゼロ以外の値のみを表示すると読みやすくなります。</p>
LINE n 省略可能	<p>列の行数。値が指定されない場合、列は単一行をデフォルトとします。n の値は 2 から 60 の間で指定する必要があります。</p>

例

6つの列を使ってビューを定義する

ARテーブルを開いた状態で、6つの列を使ってAR_Reportというビューを定義します。これらの列は、指定した順序で表示されます。

```
OPEN Ar
DEFINE VIEW AR_Report OK
DEFINE COLUMN AR_Report No AS "顧客番号" WIDTH 7 KEY
DEFINE COLUMN AR_Report Date AS "請求日" WIDTH 10
DEFINE COLUMN AR_Report Due AS "支払期日" WIDTH 10
DEFINE COLUMN AR_Report Reference AS "参照番号" WIDTH 6
DEFINE COLUMN AR_Report Type AS "取引タイプ" WIDTH 5
DEFINE COLUMN AR_Report Amount AS "取引金額" WIDTH 12 PIC "-9999999999.99"
```

DEFINE FIELD コマンド

Analytics テーブルレイアウトで物理データフィールドを定義します。

構文

```
DEFINE FIELD フィールド名 データ型 開始位置 長さ <小数点以下桁数|日付書式>
<NDATETIME> <PIC 書式> <AS 表示名> <WIDTH バイト数> <SUPPRESS> <フィールド メモ>
```

パラメーター

名前	説明						
フィールド名	<p>フィールドの名前。</p> <p>メモ フィールド名は256文字までの小文字の英字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。 Analyticsには多くの予約キーワードがあり、フィールド名にこのキーワードを使用することはできません。詳細については、「予約キーワード」ページ1421を参照してください。</p>						
データ型	<p>データを解釈するときに使用するデータ型。サポートされるデータ型の一覧については、「サポートされているデータ型」ページ1715を参照してください。</p> <p>たとえば、請求書番号がソース内に数値として格納されているとします。これらの値を数値ではなく、文字列として処理するには、文字データとしてフィールドを定義できます。</p>						
開始位置	<p>Analytics データファイル内のフィールドの開始バイトを指定します。</p> <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データでは、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						

名前	説明						
長さ	<p>フィールド長のバイト数。</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode データでは、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						
小数位 省略可能	数値フィールドの小数点以下の桁数						
日付書式 省略可能	<p>ソース日付フィールドでの日付書式。</p> <p>日付時刻フィールドまたは時刻フィールドについては、代わりに PIC 書式を使用します。PIC 書式は、日付フィールドにも使用できます。</p> <p>ソースデータにスラッシュなどの区切り文字が含まれている場合は、日付書式書式にその区切り文字を含める必要があります。たとえば、ソースデータが 12/31/2014 である場合は、書式を MM/DD/YYYY として入力します。日付書式を引用符で囲まないでください。</p>						
NDATETIME 省略可能	<p>数値フィールドに格納されている日付、日付時刻、または時刻の値を日付時刻データとして扱います。</p> <p>NDATETIME を使用するには、PIC 書式を使ってソースの日付時刻書式も指定されている必要があります。</p>						
PIC 書式 省略可能	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソースデータの日付時刻値の物理形式 (日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソースデータの物理形式と正確に一致する必要があります。たとえば、ソースデータが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>						
AS 表示名 省略可能	<p>ビューにおけるフィールドの表示名 (代替列見出し)。表示名をフィールド名と同じにしたい場合は、AS を使用しないでください。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン (;) を入れます。</p>						
WIDTH 文字数	フィールドの表示幅 (文字数)。						

名前	説明
省略可能	<p>指定した値によって、Analytics のビューおよびレポートにおけるフィールドの表示幅が決まります。表示幅はデータを変更するものではありませんが、表示幅がフィールド長より短い場合にはデータが隠れる可能性があります。</p> <p>このため、表示幅にはフィールド名や表示名の長さより短い値を指定しないでください。</p> <p>WIDTH を省略した場合には、表示幅はフィールド長の文字数に設定されます。</p> <p>メモ</p> <p>WIDTH で指定する文字数は、固定幅の文字数です。実際の文字の幅に関係なく、各文字には同じ空白量が割り当てられます。</p> <p>Analytics のビューでは、固定幅の文字間隔と対応しないプロポーショナルフォントがデフォルトで使用されます。</p> <p>WIDTH の値とビューの文字数との間に 1 対 1 の対応が必要な場合は、[オプション] ダイアログボックスの [プロポーショナルフォント] 設定を Courier New などの固定幅フォントに変更することができます。</p>
SUPPRESS 省略可能	<p>数値フィールドにのみ適用されます。</p> <p>Analytics レポートの数値フィールドの自動合計を抑制します。</p> <p>一部の数値フィールドの合計は適切ではありません。たとえば、単価フィールドまたは値引率フィールドです。</p>
フィールドノート 省略可能	<p>テーブルレイアウトでフィールド定義に追加するフィールドノートのテキスト。</p> <p>フィールドノートは、その他すべての必須およびオプションパラメーターよりも後の、最後に指定する必要があります。テキストは、複数行にすることはできません。引用符は必要ありません。</p>

例

文字フィールドの定義

文字フィールド **ProdDesc** を定義します。ビューでのこの列の見出しは、**製品説明** です。

非 Unicode 版 Analytics

- 開始: バイト 12 (文字位置 12)
- 長さ: 24 バイト (24 文字)

```
DEFINE FIELD ProdDesc ASCII 12 24 AS "製品説明"
```

Unicode 版 Analytics、拡張 ASCII (ANSI) データ

- 開始: バイト 12
- 長さ: 24 バイト (24 文字)

```
DEFINE FIELD ProdDesc ASCII 12 24 AS "製品説明"
```

Unicode 版 Analytics、Unicode データ

- 開始: バイト 13
- 長さ: 48 バイト (24 文字)

```
DEFINE FIELD ProdDesc UNICODE 13 48 AS "製品説明"
```

数値フィールドの定義

数値フィールド **QtyOH** を定義します。ビューでは、列は指定された表示書式を使用し、ビューの列見出しは **在庫数** です。

- 開始: バイト 61
- 長さ: 10 バイト
- 小数点以下桁数: なし

```
DEFINE FIELD QtyOH NUMERIC 61 10 0 PIC "(9,999,999)" AS "在庫数"
```

文字データからの日付時刻フィールドの定義

ソース文字データから、以下の最初の2つの例は、日付時刻フィールドの取引日を定義します。ソースデータでは、日付形式は DD/MM/YYYY です。列見出しは、指定されていないため、デフォルトでフィールド名になります。

- 開始: バイト 20
- 長さ: 10 バイト

ここでは、**日付形式** を使用して、日付形式が指定されます。

```
DEFINE FIELD Transaction_date DATETIME 20 10 DD/MM/YYYY
```

ここでは、PIC 形式を使用して、日付形式が指定されます。

```
DEFINE FIELD Transaction_date DATETIME 20 10 PIC "DD/MM/YYYY"
```

時刻データを含む日付時刻フィールドを定義するときには、PIC 形式を使用する必要があります。

以下の例は、日付時刻フィールドの**電子メールタイムスタンプ**を定義します。ソースデータでは、日付時刻形式は、YYYY/MM/DD hh:mm:ss-hh:mm です。

- 開始: バイト 1
- 長さ: 25 バイト

```
DEFINE FIELD email_timestamp DATETIME 1 25 PIC "YYYY/MM/DD hh:mm:ss-  
hh:mm"
```

数値データからの日付時刻フィールドの定義

次の例では、ソースの数値型データを使用して、そのソースデータに指定されている日付書式を持っていて、**Receipt_timestamp** という日付時刻型フィールドを定義しています。

- 開始: バイト 15
- 長さ: 15 バイト

```
DEFINE FIELD Receipt_timestamp DATETIME 15 15 PIC "YYYYMMDD.hhmmss"
```

"数値" 日付時刻フィールドの定義

次の例では、ソースの数値型データを使用して、そのソースデータに指定されている日付書式を持っていて、**Receipt_timestamp** という数値型フィールドを定義しています。

次の例では、NDATETIME パラメーターにより、Analytics が、数値フィールドに格納されている日付時刻値を日付時刻データとして扱うことができるようになっています。

- 開始: バイト 15
- 長さ: 15 バイト
- 小数点以下桁数: 6

```
DEFINE FIELD Receipt_timestamp PRINT 15 15 6 NDATETIME PIC  
"YYYYMMDD.hhmmss"
```

メインフレームのパックデータを読み取る物理データフィールドを定義する

NDATETIME オプションを使用して、パックされた数値フィールドから日付値を読み取る物理データフィールドを作成できます。

Analytics では、1 桁あたり 1 バイト未満に圧縮された、日付書式でない数値の日付は認識できません。したがって、NDATETIME で数値をアンパックしてすべての桁の数字を取得してから、PIC で日付書式を指定する必要があります。

それぞれの数字と年、月、日の対応を正確に示すには、パック型レコードレイアウトの日付書式と同じものを指定します。

```
DEFINE FIELD 日付フィールド名 NUMERIC 1 8 0 NDATETIME PIC "YYYYMMDD"
```

備考

このコマンドの動作の詳細については、「物理フィールドの定義」ページ 788を参照してください。

スクリプトを使ってフィールドを上書きする

既存のフィールドと同じ名前を使用するフィールドを定義することによって、テーブルレイアウト内のフィールドを上書きすることができます。SET SAFETY ON を実行した場合は、既存のフィールドを上書きする前に確認ダイアログボックスが表示されます。

スクリプトが中断されないようにするには、SET SAFETY OFF を実行します。これにより、追加の確認が行われることなく、既存のフィールドが上書きされます。

サポートされているデータ型

データ カテゴリ	データ型
文字	ASCII
	CUSTOM
	EBCDIC
	NOTE
	PCASCII
	UNICODE
数値	ACCPAC
	ACL
	BASIC
	BINARY
	FLOAT
	HALFBYTE
	IBMFLOAT
	MICRO
	NUMERIC
	PACKED
	PRINT
	UNISYS
	UNSIGNED
	VAXFLOAT
ZONED	
日付時刻	DATETIME
論理	LOGICAL

DEFINE FIELD ... COMPUTED コマンド

Analytics テーブルレイアウトの演算フィールドを定義します。

構文

演算フィールドを定義するには

```
DEFINE FIELD フィールド名 COMPUTED 式
```

オプション パラメーターを持つ演算フィールドを定義するには

```
DEFINE FIELD フィールド名 COMPUTED  
<IF テスト> <STATIC> <PIC 書式> <AS 表示名> <WIDTH 文字> <SUPPRESS> <フィールド  
メモ>  
式
```

条件付き演算フィールドを定義するには

```
DEFINE FIELD フィールド名 COMPUTED  
*** BLANK_LINE ***  
値 IF 条件  
<値 IF 条件>  
<...n>  
デフォルト値
```

オプション パラメーターを持つ条件付き演算フィールドを定義するには

```
DEFINE FIELD フィールド名 COMPUTED  
<IF テスト> <STATIC> <PIC 書式> <AS 表示名> <WIDTH 文字> <SUPPRESS> <フィールド  
メモ>  
値 IF 条件  
<値 IF 条件>  
<...n>  
デフォルト値
```

メモ

上記の一般的な構文は、以下の例に示すように、複数行の構文は正確に構造化される必要があります。

パラメーター

名前	説明
フィールド名	<p>演算フィールド名。</p> <p>メモ フィールド名は256文字までの小文字の英字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。 Analyticsには多くの予約キーワードがあり、フィールド名にこのキーワードを使用することはできません。詳細については、「予約キーワード」ページ1421を参照してください。</p>
式	<p>演算フィールドの値を定義する有効な Analytics 式。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
STATIC 省略可能	<p>異なる値を検出するまで、テーブル内の対象フィールドの各行にはすべて同じ値が表示されます。</p> <p>たとえば、次のソースデータに姓フィールドがある場合：</p> <ul style="list-style-type: none"> 最初のレコードが値 "Smith" を表示する 次の5レコードが空白の行を表示する 7番目のレコードが値 "Wong" を表示する <p>この場合、「Smith」は6つの連続する行に表示され、「Wong」は7番目の行に表示されず。</p>
PIC 書式 省略可能	<p>メモ 数値型のフィールドにのみ適用されます。</p> <p>Analytics のビューおよびレポートに含まれる数値の表示形式。 書式は引用符で囲む必要があります。</p>
AS 表示名 省略可能	<p>ビューにおけるフィールドの表示名(代替列見出し)。表示名をフィールド名と同じにしたい場合は、AS を使用しないでください。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p>

名前	説明
WIDTH 文字数 省略可能	<p>フィールドの表示幅(文字数)。</p> <p>指定した値によって、Analytics のビューおよびレポートにおけるフィールドの表示幅が決まります。表示幅はデータを変更するものではありませんが、表示幅がフィールド長より短い場合にはデータが隠れる可能性があります。</p> <p>このため、表示幅にはフィールド名や表示名の長さより短い値を指定しないでください。</p> <p>WIDTH を省略した場合には、表示幅はフィールド長の文字数に設定されます。</p> <p>メモ</p> <p>WIDTH で指定する文字数は、固定幅の文字数です。実際の文字の幅に関係なく、各文字には同じ空白量が割り当てられます。</p> <p>Analytics のビューでは、固定幅の文字間隔と対応しないプロポーショナルフォントがデフォルトで使用されます。</p> <p>WIDTH の値とビューの文字数との間に 1 対 1 の対応が必要な場合は、[オプション] ダイアログボックスの [プロポーショナルフォント] 設定を Courier New などの固定幅フォントに変更することができます。</p>
SUPPRESS 省略可能	<p>数値型のフィールドにのみ適用されます。</p> <p>Analytics レポートの数値演算フィールドの自動合計を抑制します。</p> <p>一部の数値フィールドの合計は適切ではありません。たとえば、単価フィールドまたは値引率フィールドです。</p>
フィールドノート 省略可能	<p>テーブルレイアウトでフィールド定義に追加するフィールドノートのテキスト。</p> <p>フィールドノートは、その他すべての必須およびオプションパラメーターよりも後の、最後に指定する必要があります。テキストは、複数行にすることはできません。引用符は必要ありません。</p>
値 IF 条件	<p>条件付き演算フィールドのみ。</p> <ul style="list-style-type: none"> ○ 値 -、条件が True と評価される場合に使用する演算フィールド値または式。 ○ 条件 -評価される論理テスト。
デフォルト値	<p>条件付き演算フィールドのみ。</p> <p>条件のどれも True と評価されない場合に演算フィールドで使用される値または式。</p> <p>メモ</p> <p>すべての数値演算値の小数精度は、デフォルト値の精度で統制されます。たとえば、既定値として 0.00 を指定すると、演算値はすべて小数点以下 2 桁まで計算され、必要に応じて四捨五入されます。精度を高めるには、デフォルト値の小数点以下桁数を増やします。</p>

例

演算フィールドを定義する

Cost フィールドと **Quantity** フィールドの積を計算した **Value** 演算フィールドを定義するには、次のように指定します。

```
DEFINE FIELD Value COMPUTED Cost * Quantity
```

オプションを使って演算フィールドを定義する

7つのオプションを定義することで、**Value_03** という演算フィールドを定義するとします。演算フィールドで処理されるレコードを制限する IF 条件を追加します。

```
DEFINE FIELD Value_03 COMPUTED
IF Product_Class = "03" PIC "($9,999,999.99)" AS "Value Prod Class 3"
値はコストを数量で乗算した値です
Cost * Quantity
```

条件付き演算フィールドを定義する

取引が発生した州に応じて異なる消費税を計算する条件演算フィールド **Sales_tax** を定義するとします。3つの州以外で発生した取引の消費税はデフォルトの \$0.00 です。

メモ

2番目の行は空白にする必要があります。省略可能なパラメーターはありません。

```
DEFINE FIELD Sales_tax COMPUTED

.0750 * Sale_amount IF State = "CA"
.0400 * Sale_amount IF State = "NY"
.0625 * Sale_amount IF State = "TX"
0.00
```

オプションを使って条件付き演算フィールドを定義する

取引が発生した州に応じて異なる消費税を計算する条件演算フィールド **Sales_tax_100** を定義します。このフィールドでは、\$100 以上の金額にかかる税のみを計算します。

3つの州以外で発生した取引の消費税はデフォルトの\$0.00です。

メモ

省略可能なパラメーターを指定するときには、改行を残さないでください。

```
DEFINE FIELD Sales_tax_100 COMPUTED
IF Sale_amount >= 100
.0750 * Sale_amount IF State = "CA"
.0400 * Sale_amount IF State = "NY"
.0625 * Sale_amount IF State = "TX"
0.00
```

備考

このコマンドの動作の詳細については、「演算フィールドの定義」ページ 795を参照してください。

2種類の演算フィールド

演算フィールドには2種類あります。

- 標準演算フィールド

標準演算フィールドはテーブルのすべてのレコードに対して同じ計算を実行します。

たとえば、Inventory テーブルでは、Cost フィールドの値を Quantity フィールドの値で乗算し、各レコードのコストで棚卸資産の価額を計算する演算フィールドを作成できます。

- 条件付き演算フィールド

条件付き演算フィールドは、指定する条件群に基づいて、テーブルのレコードに対して異なる計算を実行できます。レコードに対して実行された計算は、レコードが満たす条件によって異なります。

たとえば、Transactions テーブルで、取引が発生した州に基づいて調整されるレートを使用して、売上税を計算する条件付き演算フィールドを作成できます。IF State = "CA" や IF State = "NY" などの条件が各レコードをテストし、使用するレートを特定します。

条件付き演算フィールドの作成のためのガイドライン

メモ

条件付き演算フィールドを定義するときには、2行目にオプションのパラメーターをどれも指定しない場合は、2行目を空白のままにしておいてください。

条件付きの演算フィールドを作成するには、デフォルト値に加えて、少なくとも1つの条件値が必要です。条件付きの演算フィールドを定義するには、次のような複数行構文を使用する必要があります。

- オプションのパラメーターは2行目に配置します
- オプションのパラメーターがない場合、2行目が空白になります
- 最初の条件は3行目に配置します
- 各追加条件ステートメントは別々の行に配置します
- デフォルト値は最後の行に配置します

フィールド定義の上書き

既存のフィールドと同じ名前を使用するフィールドを定義することによって、テーブルレイアウト内のフィールド定義を上書きすることができます。

SET SAFETYをONにした場合は、既存のフィールドを上書きする前に確認ダイアログボックスが表示されます。スクリプトが中断されないようにするには、SET SAFETYをOFFに設定します。そのようにすると、既存のフィールドが、確認を求められることなく上書きされます。

DEFINE RELATION コマンド

2つのAnalyticsテーブル間の関連付けを定義します。

メモ

最大 18 の Analytics テーブルを関連付け、それらのテーブルのフィールドは単独のテーブルに存在しているかのように扱うことができ、任意に組み合わせてデータにアクセスし分析することが可能です。関連するテーブルの各ペアの個別の DEFINE RELATION コマンドを指定する必要があります。

構文

```
DEFINE RELATION キーフィールド名 WITH 関連テーブル名 INDEX インデックス名 <AS リレーション名>
```

パラメーター

名前	説明
キーフィールド名	<p>親テーブルのキーフィールド。 指定できるキーフィールドは、各リレーションにつき1つのみです。</p> <p>メモ 親テーブルと孫テーブルの関係を作成するときには、<code>テーブル名.フィールド名</code>の形式で完全修飾キーフィールド名を指定する必要があります。 "3つのテーブルを関連付ける" 次のページで、<code>Vouchers.created_by</code> を参照してください。</p>
WITH 関連テーブル名	関連テーブルの名前。
INDEX インデックス名	<p>関連テーブルのキーフィールドのインデックスの名前。 テーブルを関連付ける前に、キーフィールドを基に関連テーブルにインデックスを作成する必要があります。</p>
AS リレーション名 省略可能	<p>一意のリレーション名。 デフォルトでは、子テーブルの名前がリレーション名となります。同じ子テーブルに対し、さらに関連付けを追加して定義する場合は、一意な名前を指定する必要があります。</p>

例

2つのテーブルの関連付け

以下の例では、顧客番号フィールド (**CustNum**) をキーフィールドに指定して、現在開いているテーブルを **Customer** テーブルと関連付けます。

```
DEFINE RELATION CustNum WITH Customer INDEX Customer_on_CustNum
```

Customer_on_CustNum は、キーフィールドの子テーブルインデックスの名前です。子テーブルインデックスはテーブルを関連付けるときに必要です。

子テーブルインデックスが **DEFINE RELATION** コマンドを実行するときに存在していない場合は、エラーメッセージが表示され、関連付けが実行されません。

ヒント

Analytics のユーザー インターフェイスで関係を定義する場合、子テーブルインデックスが自動的に作成されます。

2つのテーブルを関連付ける前に子テーブルインデックスを作成できます。

必要に応じて、2つのテーブルを関連付ける直前に子テーブルインデックスを作成できます。次の例は、**Ar** テーブルを **Customer** テーブルに関連付ける前に、**Customer** 子テーブルでインデックスを作成する方法を示します。

```
OPEN Customer
INDEX ON CustNum TO Customer_on_CustNum
Open Ar
DEFINE RELATION CustNum WITH Customer INDEX Customer_on_CustNum
```

3つのテーブルを関連付ける

以下の例は、**ACL_Rockwood.ACL** サンプルプロジェクトで3つのテーブルを関連付けます。

- **Vouchers_items** - 親テーブル
- **Vouchers** - 子テーブル
- **Employees** - 孫テーブル

Vouchers テーブルを関係の中間テーブルとして使用すると、各伝票項目を、項目を処理した従業員に関連付けることができます。

```
OPEN Vouchers
INDEX ON voucher_number TO "Vouchers_on_voucher_number"
OPEN Vouchers_items
DEFINE RELATION voucher_number WITH Vouchers INDEX Vouchers_on_
voucher_number
OPEN Employees
INDEX ON employee_number TO "Employees_on_employee_number"
OPEN Vouchers_items
DEFINE RELATION Vouchers.created_by WITH Employees INDEX Employees_on_
employee_number
```

構文ロジックの説明

1. **Vouchers** テーブルを開き、**voucher_number** フィールドにインデックスを作成します。
2. **Vouchers_items** テーブルを開き、**voucher_number** をキーフィールドとして使用し、**Vouchers** テーブルに関連付けます。
3. **Employees** テーブルを開き、**employee_number** フィールドにインデックスを作成します。
4. **Vouchers_items** テーブルを開き、**Vouchers.created_by** をキーフィールドとして使用し、**Employees** テーブルに関連付けます。

メモ

Vouchers.created_by は2番目の関係でキーフィールドとして使用できません。最初の関係で **Vouchers_items** と **Vouchers** を既に関連付けているためです。

備考

このコマンドの動作の詳細については、"テーブルの関連付け" ページ 999を参照してください。

DEFINE REPORT コマンド

新しいビューを作成するか、既存のビューを開きます。

構文

```
DEFINE REPORT ビュー名
```

パラメーター

名前	説明
ビュー名	既存のビューまたは新しいビューの名前。 <ul style="list-style-type: none">新しいビュー-は、開いているテーブルで、指定された名前の空のビューを作成します。ビュー名のスペースは、アンダースコア文字に置換されます。既存のビュー-は、開いているテーブルで指定されたビューを開きます。

例

新しいビューを作成する

新しいビュー **Q4_AR_review** を使用します。

```
DEFINE REPORT Q4_AR_review
```

DEFINE TABLE DB コマンド

AX コネクタを使用してデータベース テーブルに接続し、Analytics サーバー テーブルを定義します。Microsoft SQL Server、Oracle、DB2 データベースに接続することができます。

構文

```
DEFINE TABLE DB {SOURCE データベース プロファイル <PASSWORD 数値> <PASSWORD 数値>
| SERVER サーバー プロファイル <PASSWORD 数値>} <FORMAT 書式名> SCHEMA スキーマ
<TITLED ACL テーブル名> <PRIMARY|SECONDARY> DBTABLE DB テーブル名 FIELDS {フィール
ド名|ALL} <...n> <WHERE 条件> <ORDER フィールド名>
```

パラメーター

<p>SOURCE データベース プロファイル</p>	<p>データベース エンジンにアクセスするために使用する Analytics データベース プロファイル。</p> <p>データベース プロファイルには、次のように、データベース エンジンに接続するために必要な情報があります。</p> <ul style="list-style-type: none"> ○ 関連付けられたサーバー プロファイルへの参照 ○ データベースのタイプ ○ データベースの名前 ○ ユーザー アカウント情報 <p>メモ</p> <p>DEFINE TABLE DB では、Microsoft SQL Server、Oracle、DB2 データベースにのみ接続できます。</p>
<p>PASSWOR- D 番号 省略可能</p>	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前にPASSWORD コマンドかSET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクス スクリプトをスケジュールしている場合、PASSWORD 2により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ○ "PASSWORD コマンド" ページ 2015 ○ SET PASSWORD コマンド ○ PASSWORD アナリティクス タグ <p>このパスワードは、データベース プロファイルにパスワードを指定して保存していない場合のみ指定する必要があります。SOURCE キーワードの後にPASSWORD を2回使用します。1番目のパスワードでサーバーにログオンし、2番目のパスワードでデータベースにログオンします。</p>

	<p>メモ</p> <p>Analytics は、サーバープロファイルのパスワード、データベースプロファイルのパスワード、またはメールサーバーのパスワードを指定または入力するときに最大 30 文字サポートしています。この長さを超えるパスワード値を指定すると、パスワードが切り詰められ、接続エラーになります。</p>
SERVER サーバープロ ファイル	<p>使用されなくなりました。</p> <p>バージョン 10.0 より前の Analytics では、ACL Server Edition for z/OS へ接続するときに使用されていました。バージョン 10.0 以降の Analytics には、ACL Server Edition for z/OS が同梱されなくなりました。</p>
FORMAT フォーマット名 省略可能	<p>使用したいテーブルレイアウトを持つ、Analytics テーブルまたはテーブルレイアウト ファイル(.layout) の名前。</p>
SCHEMA ス キーマ	<p>接続するスキーマ。スキーマ名は引用符で囲む必要があります。</p>
TITLED ACL テーブル名 省略可能	<p>作成する Analytics テーブルの名前。</p> <p>ACL テーブル名は引用符で囲んだ文字列として指定する必要があります。TITLED を省略すると、データベースのテーブル名が使用されます。一度に複数のテーブルにアクセスした場合は、最初のテーブル名が使用されます。</p>
PRIMARY SECONDAR Y 省略可能	<p>複数ファイルコマンド内でテーブルを主テーブルと副テーブルのどちらとして使用します。どちらのオプションも指定されない場合、デフォルト値の PRIMARY が使用されます。</p>
DBTABLE データベース テーブル名	<p>アクセスするデータベーステーブル。データベーステーブル名は引用符で囲んだ文字列として指定する必要があります。</p>
FIELDS フィールド名 ALL	<p>出力に含めるフィールド：</p> <ul style="list-style-type: none"> ○ FIELDS フィールド名 - 指定されたフィールド フィールド名は引用符で囲まれた文字列である必要があります。 ○ ALL - テーブルのすべてのフィールドを使用します。 <p>複数のテーブルからフィールドを使用するには：</p> <ol style="list-style-type: none"> a. 最初のテーブル名、そのテーブルのフィールドの順に入力します。 b. 次のテーブル名、そのテーブルのフィールドの順に入力します。 c. テーブルを追加するごとに手順 b を繰り返します。 <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <pre>DBTABLE "DSN1310" FIELDS "Field_A Field_B Field_C" DBTABLE "DSN2516" FIELDS "Field_L Field_M Field_N"</pre> </div> <p>メモ</p> <p>AX Connector を使用すると、無制限の関連したテーブルにアクセスすることができますが、5 つ以下のアクセスを推奨します。複数のテーブルにアクセスすると処理時間が増えます。</p>

WHERE 条件 省略可能	<p>対象となるデータを、指定した条件を満たすレコードに限定する SQL WHERE 節。 引用符で囲まれた文字列として入力された有効な SQL 構文を使用する必要があります。 複数のテーブルを結合しようとする、WHERE 句に結合条件が表示されます。</p> <pre>"Table_1.First_name = Table_2.First_name"</pre>
ORDER 複数のフィールド名 省略可能	<p>データベースエンジンがレコードの並べ替えに使用するフィールド。フィールド名は引用符で囲んだ文字列として指定する必要があります。 レコードを並べ替えるときにはコマンドの実行に時間がかかります。並べ替えが重要であるときにのみ ORDER を使用します。</p>

例

例

Microsoft SQL Server データベース内のデータに AX Connector 経由でアクセスしたいとします。それには、DEFINE TABLE DB コマンドを使用します。データベース プロファイルを使用して AX Connector に接続する SOURCE パラメーターも指定します。

```
DEFINE TABLE DB SOURCE "SQLServer_Audit" SCHEMA "HR" TITLED "Payroll"
DBTABLE "HR.Employee" FIELDS "EmployeeID" DBTABLE
"HR.EmployeePayHistory" FIELDS "Rate PayFrequency" WHERE
"HR.Employee.EmployeeID=HR.EmployeePayHistory.EmployeeID"
```

備考

機能の仕組み

Analytics サーバー テーブルは、データベース プロファイルを使用してデータベース テーブルに接続するためのクエリとして定義されます。

日付時刻値の時刻部分を非表示にする

日付時刻値の時刻部分を非表示にするには、DEFINE TABLE DB コマンドの前に SET SUPPRESSTIME コマンドを置きます。

コマンド

SET SUPPRESSTIME ON は、バージョン 10.0 より前の Analytics スクリプトで使用することを目的としています。それらのスクリプトでは、日付時刻値の時刻部分は切り捨てられることが前提となっています。それらのスクリプトは、SET SUPPRESSTIME ON を追加しない場合には、日付時刻型対応バージョンの Analytics では動作することができません。

詳細については、「SET コマンド」 ページ 2081の「SET SUPPRESSTIME」セクションを参照してください。

DEFINE VIEW コマンド

新しいビューの定義または既存のビューを上書します。

構文

```
DEFINE VIEW ビュー名 <RLINES n> <ALL> <SUPPRESS> <SUMMARIZED> <IF テスト>
<WHILE テスト> <HEADER ヘッダーテキスト> <FOOTER フッターテキスト> <TO レポートファイル名
<HTML> > <OK>
```

パラメーター

名前	説明
ビュー名	作成または上書きするビューの名前。 メモ ビューの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。
RLINES n 省略可能	ビューやレポートの詳細レコードの行間隔。デフォルトでは、詳細行の行間隔は 1 行です。
ALL 省略可能	アクティブな Analytics テーブルレイアウト 内のすべてのフィールドがビューに追加されます。
SUPPRESS 省略可能	ビューから生成されるレポートで、空白の詳細行を表示させません。レポートの生成時、空白の詳細行は出力から除外されます。このオプションは、複数行ビューを基にしたレポートに適用されます。
SUMMARIZED 省略可能	ビューから生成されるレポートに小計と合計は含めるが、詳細行を含めないことを指定します。 小計は、ビューで定義されているブレイクフィールドを基に生成されます。このオプションを指定しないと、レポートには、指定した各ブレイクフィールドの小計と併せて詳細行も含まれません。
IF テスト 省略可能	各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。

名前	説明
	<p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
TO レポートファイル名 HTML 省略可能	<p>このビューから作成されるレポートのファイル名と種類。</p> <p>HTML キーワードを使用して、このビューから生成されたレポートを HTML ファイル(.htm) として保存します。デフォルトでは、生成されるレポートは ASCII テキスト ファイルとして出力されます。</p>
OK 省略可能	<p>アクションを確認せずに、項目を削除または上書きします。</p>

例

ビューを作成する

次の例では、Ar テーブルを開き、そのテーブルレイアウトの全フィールドを含む AR_Report というビューを作成します。

```
OPEN Ar
DEFINE VIEW AR_Report HEADER "AR_Report" ALL OK
```

DELETE コマンド

Analytics プロジェクト 項目、テーブルレイアウトのフィールド、変数、1 つ以上のテーブル履歴エントリ、テーブル間の関係、または Windows フォルダーのファイルを削除します。ビューの列も削除します。

構文

目的	構文
Analytics プロジェクト 項目を削除するには	<code>DELETE 項目の種類 項目名 <OK></code>
テーブルレイアウトからフィールドを削除するには	<code>DELETE フィールド名 <OK></code>
ビューから列を削除するには	<code>DELETE COLUMN ビュー名 フィールド名 <ALL> <OK></code>
1 つまたはすべての変数を削除するには	<code>DELETE {変数名 ALL} <OK></code>
現在の Analytics テーブルの履歴を削除するには	<code>DELETE HISTORY <保持する履歴エントリ数> <OK></code>
2 つのテーブル間の関係を削除するには	<code>DELETE RELATION <子テーブル名 リレーション名> <OK></code>
ファイルを削除するには	<code>DELETE ファイル名 <OK></code>
開いているテーブルからすべてのレコード ノートと自動生成された RecordNote フィールドを削除するには	<code>DELETE NOTES <OK></code>

パラメーター

名前	説明
項目の種類 項目名	<p>削除対象となる項目の種類と名前。</p> <p>次の項目タイプのいずれかを指定します。</p> <ul style="list-style-type: none"> FOLDER: -項目名に指定するプロジェクト フォルダーとその内容

名前	説明
	<ul style="list-style-type: none"> ◦ FORMAT: -項目名に指定するテーブルレイアウトと、それに関連するビュー、インデックスおよび関連付け 関連するテーブルのその他のテーブルレイアウトは保持されます。 [オプション]ダイアログボックス([ツール > オプション]) の [テーブル] タブで、テーブルと一緒にデータファイルを削除する オプションがオンになっていなければ、テーブルレイアウトに関連付けられたデータファイル(.fil) は削除されません。 このオプションのオン/オフは、スクリプトまたはコマンドラインで <code>SET DELETE_FILE {ON OFF}</code> コマンドを使用して切り替えることもできます。詳細については、"SET コマンド" ページ 2081を参照してください。 <p>注意</p> <p>テーブルと一緒にデータファイルを削除する オプションをオンにする際には慎重に行ってください。テーブルレイアウトと共に元のデータファイルが削除される場合もあります。 データファイルは完全に削除されます。Windows のごみ箱には送られません。</p> <ul style="list-style-type: none"> ◦ REPORT -指定されたビュー ビューは、現在アクティブである場合には削除できません。 ◦ COLUMN - 項目名に指定する列 ◦ SCRIPT(または BATCH) - 項目名に指定するスクリプト ◦ WORKSPACE - 項目名に指定するワークスペース ◦ INDEX - 項目名に指定するインデックス ◦ NOTES - 開いているテーブル内のすべてのレコード ノートと、テーブルレイアウト内の RecordNote フィールド
フィールド名 ALL	<h3>フィールドの削除</h3> <p>現在の Analytics テーブルレイアウトから削除するフィールドの名前 フィールドは、現在のビューに含まれている場合でもテーブルレイアウトから削除できます。</p> <p>メモ</p> <p>演算フィールドから参照されているフィールドは、まずその演算フィールドを削除してからでないと削除できません。</p> <h3>列の削除</h3> <p>指定したビューから削除する列の名前</p> <p>メモ</p> <p>列表示名ではなく、物理フィールド名を使用します。</p> <ul style="list-style-type: none"> ◦ ALL included(すべて含める) - は、ビューにおける指定列のすべての出現箇所を削除します。 ◦ ALL omitted(すべて省略) - は、ビューにおける指定列の最初(一番左)の出現箇所を削除します。

名前	説明
ビュー名	列を削除するビューの名前。
変数名 ALL	<p>削除する変数の名前。すべての変数を削除する場合は、ALLを使用します。</p> <p>ALLを指定すると、プロジェクトから以下のタイプの変数の出現がすべて削除されます。</p> <ul style="list-style-type: none"> システム変数 一時的なユーザー定義変数 永続的なユーザー定義変数 <p>メモ 演算フィールドから参照されている変数は、まずその演算フィールドを削除してからでないと削除できません。</p>
HISTORY 保持する履歴エントリ数	<p>テーブルの全履歴エントリのうち、保持する履歴エントリ数で指定した数の直近のエントリより前の履歴エントリがすべて削除されます。</p> <p>すべてのエントリを削除する場合は、保持する履歴エントリ数を省略します。</p>
RELATION 子テーブル名 関連付け名	<p>依存関係がなく、アクティブなビューやアクティブな演算フィールドで参照される関連フィールドもない関係がすべて削除されます。</p> <p>削除する関連付けを指定するには、以下のオプションを使用します。</p> <ul style="list-style-type: none"> 子テーブル名 - 関連付けに固有の名前が付けられていない場合に(関連付けを作成したときのデフォルト名を)使用します。 関連付け名 - 関連付けが固有の名前を付けて作成されている場合に使用します。そうでない場合は、子テーブル名を使用します。 <p>どちらのオプションも使用しない場合は、最後に定義された関連付けが削除されます。</p>
ファイル名	<p>削除する物理ファイルの名前。</p> <p>削除するファイルへの絶対パスまたは相対パスを指定することができます。パスにスペースが含まれている場合は、パスを二重引用符で囲みます。</p>
OK 省略可能	確認ダイアログボックスを表示せずに項目を削除します。

例

日付フィールドの削除

次の例では、Ar テーブルに関連付けられているテーブルレイアウトから **Date** フィールドが削除されます。

```
OPEN Ar  
DELETE Date
```

ビューから複数の列を削除する

Ar テーブルに関連付けられている AR_Report ビューから2つの列を削除するとします。次の例では、スクリプト実行時に確認プロンプトが表示されないよう、2つのDELETEコマンドのどちらでもOKが指定されています。

```
OPEN Ar  
DELETE COLUMN AR_Report Date OK  
DELETE COLUMN AR_Report Invoice_Date OK
```

DIALOG コマンド

1つ以上のスクリプト入力値をインタラクティブにユーザーに確認するカスタム ダイアログ ボックスを作成します。各入力値は名前付き変数に格納されます。

メモ

DIALOG コマンドを使用してパスワードを入力することは安全ではありません。代わりに "PASSWORD コマンド" ページ 2015 を使用してください。

DIALOG コマンドは、ロボットで実行されるスクリプトでサポートされません。

"ACCEPT コマンド" ページ 1625では基本的なインタラクティブ ダイアログボックスを作成できます。

ヒント

カスタム ダイアログボックスを作成する最も簡単な方法は、**ダイアログビルダー**を使用することです。詳細については、"カスタム ダイアログ ボックスの作成" ページ 1588を参照してください。

構文

```
DIALOG (DIALOG TITLE タイトルテキスト WIDTH ピクセル数 HEIGHT ピクセル数) (BUTTONSET TITLE "&OK;&Cancel" AT x_pos y_pos <WIDTH ピクセル数> <HEIGHT ピクセル数> DEFAULT 項目番号 <HORZ>) <[ラベル構文]|[テキストボックス構文]|[チェックボックス構文]|[ラジオボタン構文]|[ドロップダウンリスト構文]| [[プロジェクト項目リスト構文]> <...n>
```

ラベル構文 ::=

```
(TEXT TITLE タイトルテキスト AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <CENTER|RIGHT>)
```

テキストボックス構文 ::=

```
(EDIT TO 変数名 AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <DEFAULT 文字列>)
```

チェックボックス構文 ::=

```
(CHECKBOX TITLE タイトルテキスト TO 変数名 AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <CHECKED>)
```

ラジオボタン構文 ::= (RADIOBUTTON TITLE 値リスト TO 変数名 AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <DEFAULT 項目番号> <HORZ>)

ドロップダウンリスト構文 ::= (DROPDOWN TITLE 値リスト TO 変数名 AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <DEFAULT 項目番号>)

プロジェクト項目一覧構文 ::= (ITEM TITLE プロジェクト項目の種類 TO 変数名 AT X座標 Y座標 <WIDTH ピクセル数> <HEIGHT ピクセル数> <DEFAULT 文字列>)

パラメーター

一般パラメーター

名前	説明
DIALOG TITLE タイトルテキスト	メインダイアログボックスとダイアログボックスタイトルを作成します。 タイトルテキストは引用符で囲んだ文字列として指定する必要があります。
BUTTONSET TITLE "&OK;&キャンセル"	ダイアログボックスの [OK] ボタンおよび [キャンセル] ボタンのラベル。 通常、ラベルのテキスト値は編集しないでください。変更する場合は、必ず肯定的な値が否定的な値の前に来るようにしてください。例: "&はい;&いいえ"
WIDTH ピクセル数	個々のコントロールの幅、またはダイアログボックスの幅 (DIALOG コントロールに対して指定された場合)。 値はピクセル数で指定します。値がコントロールに対して指定されない場合は、幅はコントロールに含まれる最も長い値に基づいて計算されます。
HEIGHT ピクセル数	個々のコントロールの高さ、または DIALOG コントロールに対して指定された場合はダイアログボックスの高さ。 値はピクセル数で指定します。
AT X座標 Y座標	カスタムダイアログボックス上のコントロールの左上隅の場所。 <ul style="list-style-type: none"> X座標はダイアログボックスの左端からの水平距離 (ピクセル) です。 Y座標はダイアログボックスの上端からの垂直距離 (ピクセル) です。
DEFAULT 項目番号	BUTTONSET 値で、デフォルトとして選択する項目に相当する数値。 たとえば、BUTTONSET 値が "&OK;&キャンセル" の場合、DEFAULT 1 を指定し、デフォルトで OK を選択します。

名前	説明
HORZ 省略可能	BUTTONSET コントロールで値が左右に並べて表示されます。値はデフォルトで上下に並べて表示されます。

メモ

ほとんどのコントロールタイプでは、DIALOG コマンドは、ユーザー入力を保存するための変数を作成します。代入変数に使用される変数の名前に、`@` のような英語以外の文字は使用しないでください。変数名に英語以外の文字が含まれていると、変数の代入が失敗します。

デフォルトでは、DIALOG 変数の一部は、文字変数として作成されます。文字変数を使用して、数値または日付時刻値を保存する場合は、スクリプトの後の処理で、変数を必要なデータ型に変換する必要があります。詳細については、「入力データ型」ページ 1743を参照してください。

ラベルパラメーター

名前	説明
TEXT	テキスト ラベルを作成して、特定、通知、または指示します。
TITLE タイトルテキスト	コントロールラベル。 タイトルテキストは引用符で囲んだ文字列として指定する必要があります。
CENTER RIGHT 省略可能	コントロール内のテキストの配置。 CENTER または RIGHT を省略する場合、左寄せがデフォルトで使用されます。

テキストボックスパラメーター

名前	説明
EDIT	ユーザー入力用のテキストボックスを作成します。
TO 変数名	ユーザーによって指定された入力を格納する文字変数の名前。 変数が既に存在する場合は、指定された値が割り当てられます。変数が存在しない場合、変数が作成され、指定された値が割り当てられます。
DEFAULT 文字列 省略可能	コントロールに表示するデフォルトのテキスト文字列。 文字列は引用符で囲んだ文字列として指定する必要があります。

チェックボックス パラメーター

名前	説明
CHECKBOX	チェックボックスを作成して、ユーザーへのオプションを表示します。
TITLE タイトルテキスト	コントロールラベル。 タイトルテキストは引用符で囲んだ文字列として指定する必要があります。
TO 変数名	ユーザーによって指定された True または False 値を格納する論理変数の名前。 変数が既に存在する場合は、指定された値が割り当てられます。変数が存在しない場合、変数が作成され、指定された値が割り当てられます。
CHECKED 省略可能	コントロールがデフォルトでオンにされるように設定します。

ラジオボタン パラメーター

名前	説明
RADIOBUTTON	ラジオボタンを作成して、相互に排他的なオプションをユーザーに表示します
TITLE 値リスト	コントロールに表示される値のリスト。 値は引用符で囲んだ文字列として指定する必要があります。各値はセミコロン(;)で区切ります。
TO 変数名	ユーザーによって選択されたラジオボタン値の数値位置を格納する数値変数の名前。 変数が既に存在する場合は、指定された値が割り当てられます。変数が存在しない場合、変数が作成され、指定された値が割り当てられます。
DEFAULT 項目番号 省略可能	項目リスト中で、デフォルトとして選択する項目に相当する数値。 たとえば、値リストは "赤;緑;青" であるとして、フォルトで "緑" が選択されるようにするには、DEFAULT 2 を指定します。
HORZ 省略可能	コントロールで値が左右に並べて表示されます。値はデフォルトで上下に並べて表示されません。

ドロップダウン リスト パラメーター

名前	説明
DROPDOWN	ドロップダウン リストを作成して、ユーザーへのオプションのリストを表示します

名前	説明
TITLE 値リスト	コントロールに表示される値のリスト。 値は引用符で囲んだ文字列として指定する必要があります。各値はセミicolon(;)で区切ります。
TO 変数名	ユーザーによって選択されたドロップダウンリスト値を格納する文字変数の名前。 変数が既に存在する場合は、指定された値が割り当てられます。変数が存在しない場合、変数が作成され、指定された値が割り当てられます。
DEFAULT 項目番号 省略可能	項目リスト中で、デフォルトとして選択する項目に相当する数値。 たとえば、値リストは"赤;緑;青"であるとして、ドロップダウンリストが表示されるときデフォルトで"緑"が選択されるようにするには、 <code>DEFAULT 2</code> を指定します。

プロジェクト項目一覧

名前	説明
ITEM	プロンプト項目リストを作成して、フィールドなどのAnalyticsプロジェクト項目のリストをユーザーに表示します。
TITLE プロジェクト項目カテゴリ	プロジェクト項目コントロールに含めるプロジェクト項目のカテゴリ。 1つ以上のカテゴリを指定できます。ユーザーはプロジェクト項目リストから1つの値を選ぶことができます。 <i>project_item_category</i> を引用符で囲みます。カテゴリ間にはスペースまたは句読点を入れません。 カテゴリを指定するために使用するコードについては、"プロジェクト項目カテゴリのコード"ページ 1742を参照してください。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>特に理由がない場合は、同じITEMコントロールに異なるカテゴリを混在させないでください。たとえば、テーブルとフィールドを混在させないでください。結果のプロジェクト項目一覧がユーザーにとって混乱するものになる可能性があります。</p> </div>
TO 変数名	ユーザーによって選択されたプロジェクト項目の名前を格納する文字変数の名前。 変数が既に存在する場合は、指定された値が割り当てられます。変数が存在しない場合、変数が作成され、指定された値が割り当てられます。
DEFAULT 文字列 省略可能	デフォルトとして選択するプロジェクト項目の正確な名前。 文字列は引用符で囲んだ文字列として指定する必要があります。

例

テーブルとスクリプトを選択するようユーザーに求める

スクリプトでは、解析の実行に使用する Analytics のテーブルおよびスクリプトを選択するようユーザーに求める必要があるとします。

ACL_Demo.ac1 プロジェクトの **Metaphor_Inventory_2012** テーブルを Analytics テーブルのデフォルトとして選択されるように指定しています(4行目)。ただし、ユーザーはプロジェクト内で任意のテーブルを選択することもできます。

実行するスクリプトも、Analytics プロジェクト内でのスクリプトのリストから選択する必要があります。

```
DIALOG ( DIALOG TITLE "在庫分析" WIDTH 500 HEIGHT 200 ) ( BUTTONSET  
TITLE "OK(&O);キャンセル(&C)" AT 370 12 DEFAULT 1 ) ( TEXT TITLE "分析する  
Analytics プロジェクトを選択してください。" AT 50 16 ) ( TEXT TITLE "テーブル:" AT  
50 50 ) ( ITEM TITLE "f" TO "v_table" AT 50 70 DEFAULT "Metaphor_  
Inventory_2012" ) ( TEXT TITLE "スクリプト:" AT 230 50 ) ( ITEM TITLE "b"  
TO "v_script" AT 230 70 )
```

その他の例

その他の DIALOG の例については、「スクリプト例:レコードを日付でフィルタリングし、フィルタリングされたレコードを月別にグループ化する」ページ 1514を参照してください。

備考

このコマンドの動作の詳細については、「カスタムダイアログボックスの作成」ページ 1588を参照してください。

変数の使用の詳細については、「ACLScript での変数の操作」ページ 1554を参照してください。

インタラクティブ

インタラクティブスクリプトを作成するには、DIALOG コマンドを使用します。DIALOG コマンドが処理されるときにはスクリプトが中断し、Analytics が次の処理で使用する入力をユーザーに促すダイアログボックスが表示されます。

項目ごとに個別のダイアログボックスを作成して、一度に1項目について入力を求めることもできますが、1つのダイアログボックスで複数項目について入力を求めることもできます。

ACCEPT と DIALOG

ACCEPT コマンドでは、次の1つ以上のタイプのコントロールを使用できる基本的なインタラクティブダイアログボックスを作成できます。

- テキストボックス
- プロジェクト項目一覧

基本的な対話性については、ACCEPT で十分であると考えられます。詳細については、"ACCEPT コマンド" ページ 1625を参照してください。

プロジェクト項目カテゴリのコード

次のコードを使用して、プロジェクト項目リストで表示するプロジェクト項目のカテゴリを指定します。

プロジェクト カテゴリ

コード	カテゴリ
f	テーブル
十億	スクリプト
i	インデックス
r	ビューとレポート
w	ワークスペース

フィールド カテゴリ

コード	カテゴリ
C	文字フィールド
N	数値フィールド
D	日付時刻フィールド
L	論理フィールド

変数カテゴリ

コード	カテゴリ
c	文字変数
n	数値変数
d	日付時刻変数
l	論理変数

入力データ型

DIALOG コマンドのコントロールの一部は、文字変数にユーザー入力を保存します。数値や日付時刻値として保存するには、VALUE() 関数や CTOD() 関数を使用します。文字変数の値がそれぞれ数値や日付時刻値に変換されます。

```
SET FILTER TO BETWEEN(%v_date_field%, CTOD(%v_start_date%), CTOD(%v_end_date%))
```

この例では、フィルターの開始日付および終了日付は文字値として保存されています。これらの日付は、日付時刻データ型を使用する日付フィールドで使用するには、日付型の値に変換する必要があります。

変数名をパーセント記号 (%) で囲むと、変数名の変数によって含まれる文字値を代入します。CTOD() 関数は文字値を日付値に変換します。

DIALOG コマンドの位置

可能な限り、DIALOG コマンドはすべてスクリプトの先頭部分に配置することをお勧めします。最初すべての情報入力を要求し、必要な情報が入力されていれば、その後スクリプトを円滑に実行できます。

メモ

GROUP コマンド内では DIALOG コマンド コマンドを使用できません。

DIRECTORY コマンド

指定されたディレクトリ内のファイルとフォルダーの一覧を生成します。

構文

```
DIRECTORY <ファイルスペック> <SUPPRESS> <SUBDIRECTORY> <APPEND> <TO テーブル名 | ファイル名>
```

パラメーター

名前	説明
ファイルスペック 省略可能	<p>含まれている情報を一覧表示および表示する Windows フォルダーまたはファイル。</p> <p>アスタリスクのワイルドカード(*)を使用して、特定の拡張子を持つすべてのファイルや、特定の文字列で始まるすべてのファイル、またはフォルダー内のすべてのファイルを選択できます。</p> <p>例：</p> <ul style="list-style-type: none"> *.fil - ファイル名の拡張子が .fil であるすべてのファイル(Analytics データ ファイル) が一覧表示されます。 Inv*. * - ファイル拡張子が何であれ、名前が "Inv" で始まるすべてのファイルが一覧表示されます。 Results* または Results*. * - Results フォルダー内のすべてのファイルが一覧表示されます。 <p>一覧表示されるファイルを特定のフォルダーに制限するには、Analytics プロジェクト フォルダーへの相対パスを指定するか、完全パスを指定できます。例：</p> <ul style="list-style-type: none"> Results*. * - Analytics プロジェクト フォルダーの Results サブフォルダーの内容が表示されます。 C:\ACL Data\Results*. * - 指定したフォルダーの内容が表示されます。 <p>メモ</p> <p>ワイルドカード文字は、指定されたファイルパスの中間レベルで使用できません。上記のように、パスの最終レベルでのみ使用できます。</p> <p>スペースを含んでいるパスまたはファイル名は、二重引用符で囲む必要があります。</p> <p>ファイルスペックを使用する場合は、ほかのパラメーターの前に配置する必要があります。ファイルスペックがほかの位置に表示されると、DIRECTORY コマンドが処理されず、エラーが発生します。</p> <p>ファイルスペックを省略した場合は、Analytics プロジェクトが入っているフォルダー内のすべてのファイルが一覧表示されます。ファイルスペックを省略したら、ほかのパラメーターはどれも使用できません。</p>

名前	説明
<p>SUPPRESS</p> <p>省略可能</p>	<p>出力にパス情報が表示されずに、ファイル名とプロパティのみが表示されます。</p>
<p>SUBDIRECTORY</p> <p>省略可能</p>	<p>サブフォルダーの内容が、ディレクトリの一覧に含まれます。</p> <p>たとえば、ファイルスペックが <code>Results*.fil</code> の場合は、Results フォルダーと、Results フォルダ一下のすべてのサブフォルダーで、<code>.fil</code> ファイルが検索されます。</p> <p>サブフォルダーで検索が行われるとき、一覧表示する必要のあるサブフォルダーとファイルの数によっては、SUBDIRECTORY を使用することで遅延が生じる可能性があります。Analytics により、コマンド実行の進捗を示すダイアログボックスが表示されます。</p>
<p>APPEND</p> <p>省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p> </div>
<p>TO テーブル名 ファイル名</p> <p>省略可能</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 <code>.FIL</code> を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル (<code>.fil</code>) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>テーブル名は 64 文字の英数字 (<code>.FIL</code> 拡張子を含まない) に制限されています。名前にはアンダースコア文字 (<code>_</code>) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> </div> <ul style="list-style-type: none"> ○ ファイル名 -は結果の保存先となるファイルです。 <p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。例：<code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定し</p>

名前	説明
	<p>ます。</p> <ul style="list-style-type: none"> TO "C:\Output.TXT" TO "Results\Output.TXT" <p>TO を省略した場合は、ディレクトリの一覧は Analytics の表示領域に表示されます。</p>

例

ファイルの様々な一覧表示方法

ファイルの一覧表示機能は、臨時の調査に有効であるほか、スクリプトへの組み込み対象として有効なものです。

DIRECTORY コマンドを使ってファイルを一覧表示する様々な方法を次に示します。

すべてのファイルを一覧表示

Analytics プロジェクトが入っているフォルダー内のすべてのファイルをリストアップする

```
DIRECTORY
```

特定のタイプのすべてのファイルを一覧表示

Analytics プロジェクトが入っているフォルダー内のすべての .fil ファイル(Analytics データファイル)をリストアップする

```
DIRECTORY *.fil
```

ワイルドカードを使用してファイルを一覧表示

Analytics プロジェクトが入っているフォルダー内の、"Inv" で始まるすべてのファイル名をリストアップする

```
DIRECTORY Inv*.*
```

Analytics プロジェクト フォルダーに相対的なサブフォルダーのすべてのファイルを一覧表示

Analytics プロジェクトが入っているフォルダー内の、**Results** サブフォルダーにあるすべてのファイルをリストアップする

```
DIRECTORY "Results\*"
```

指定したフォルダー内のすべてのファイルをリストアップする

Results サブフォルダーのすべてのファイルを一覧表示

```
DIRECTORY "C:\ACL Data\Results\*"
```

特定の場所の特定のタイプのすべてのファイルを一覧表示

指定したフォルダーとそのすべてのサブフォルダー内にある、すべての .fil ファイル(Analytics データファイル)を一覧表示する

```
DIRECTORY "C:\ACL Data\Results\*.fil" SUBDIRECTORY
```

指定されたフォルダーのすべてのファイルを一覧表示し、リストを Analytics テーブルに出力

Results フォルダー内のすべてのファイルを一覧表示し、そのリストを、Analytics プロジェクトが入っているフォルダー内の Analytics テーブルに出力する

```
DIRECTORY "C:\ACL Data\Results\*" TO Results_Folder_Contents.fil
```

新しいテーブル **Results_Folder_Contents** が開いているプロジェクトに追加されます。

1つのフォルダーのすべてのファイルを一覧表示し、リストを別のフォルダーの Analytics テーブルに出力

ACL Data\Results フォルダー内のすべてのファイルを一覧表示し、そのリストを、**GL Audit 2014\Results** フォルダーの Analytics テーブルに出力する

```
DIRECTORY "C:\ACL Data\Results\*" TO "C:\ACL Projects\GL Audit 2014\Results\Results_Folder_Contents.fil"
```

新しいテーブル **Results_Folder_Contents** が開いているプロジェクトに追加されます。関連するデータファイル (**Results_Folder_Contents.fil**) が指定した出力フォルダーに作成されます。このフォルダーは、Analytics プロジェクトが入っているフォルダーであってもなくてもかまいません。

備考

DIRECTORY によって表示されるプロパティ

DIRECTORY コマンドは、Windows の DIR コマンドに似ています。フォルダー内のファイルとサブフォルダーの一覧を表示することに加え、DIRECTORY コマンドでは、次のファイルプロパティおよびフォルダープロパティも表示します。

<ul style="list-style-type: none"> ファイルサイズ 属性 	<ul style="list-style-type: none"> 作成日 作成時刻 	<ul style="list-style-type: none"> アクセス日 アクセス時刻 	<ul style="list-style-type: none"> 更新日 更新時刻 指定された条件に一致するファイルおよびフォルダーの総数
-----------------------------------------------------------------------	---------------------------------------------------------------------	-------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------

スクリプト内での DIRECTORY の使用

スクリプト内で DIRECTORY コマンドを使用すると、ファイルシステムを調べることができます。たとえば、DIRECTORY をほかのコマンドと連携させて、ファイルの有無を検出したり、ファイルのサイズを調べたり、

コマンド

ほかのファイルプロパティに基づいて判断を下したりすることができます。

DIRECTORY の結果を出力する

コマンドラインからこのコマンドを実行する場合は、ディレクトリの一覧を画面に表示するか、一覧を Analytics テーブルまたは .txt ファイルに保存することができます。

DIRECTORY の結果を保存したテーブルを開く方法

DIRECTORY コマンドには OPEN パラメーターが含まれていません。スクリプトでコマンドを使用し、結果を Analytics テーブルに出力する場合、その結果テーブルを開きたいときは、DIRECTORY コマンドの後に OPEN コマンドを続けてください。例：

```
DIRECTORY "C:\ACL Data\Results\*" TO Results_Folder_Contents.fil  
OPEN Results_Folder_Contents
```

DISPLAY コマンド

指定された Analytics 項目タイプの情報を表示します。式の結果、または関数の出力も表示できます。

構文とパラメーター

構文	目的
DISPLAY	フィールド定義を表示し、現在アクティブな Analytics テーブルの関連付けられた子テーブルを表示する
DISPLAY OPEN	開いている Analytics のテーブルおよびプロジェクト ファイルの一覧を表示する <ul style="list-style-type: none"> Analytics のテーブル-は、テーブルレイアウトの名前ではなく、ソースデータファイルの名前として表示されます。 複数テーブルモード - PRIMARY として特定されたソースデータファイルが、現在アクティブなテーブルに関連付けられます。 関連テーブル- 親テーブルが開いている場合、[ビュー]タブで子テーブルが開いていなくても、親と子の両方のテーブルのソースデータファイルが表示されます。
DISPLAY {<PRIMARY> SECONDARY}	主テーブルまたは副テーブルの名前およびテーブルレイアウト情報を表示する <ul style="list-style-type: none"> PRIMARY(またはキーワードを指定しない場合) - 現在アクティブなテーブルの情報が表示されます。 SECONDARY - 副テーブルの情報が表示されます。 <p>複数テーブルモードでは、SECONDARY は現在アクティブなテーブルに関連付けられている副テーブルを参照します。</p> <p>以下の情報が表示されます。</p> <ul style="list-style-type: none"> テーブルレイアウト名 ソースデータファイル名 現在アクティブなテーブルとその他のテーブルとの間にあるあらゆる関連付け テーブルレイアウト内のフィールド定義情報
DISPLAY HISTORY	現在のアクティブな Analytics テーブルのテーブル履歴を表示する <p>メモ テーブルには、関連するテーブル履歴がある場合もない場合があります。</p>
DISPLAY RELATION	現在アクティブな Analytics テーブルの関連付け情報を表示する <ul style="list-style-type: none"> すべての子テーブルの名前 キーフィールド名 インデックス名

構文	目的
DISPLAY {変数名 VARIABLES}	<p>1つの変数あるいはすべての変数の値を表示する</p> <ul style="list-style-type: none"> ◦ 変数名 - 値の表示対象となる単独の変数の名前 ◦ VARIABLES - すべてのシステム変数およびユーザー定義変数の値が表示されます。また、変数を格納するために利用できる残りのメモリ量も表示されます。
DISPLAY VERSION	<p>インストールされた Analytics のバージョンに関する情報を次の書式で表示する</p> <ul style="list-style-type: none"> ◦ バージョン - メジャーバージョン番号.マイナーバージョン番号 ◦ パッチ - パッチ番号 ◦ 種類 - Analytics の非 Unicode 版 (000) または Unicode 版 (001) ◦ Build(ビルド) - ソフトウェアのビルド番号
DISPLAY {DATE TIME}	<p>オペレーティングシステムの現在の日付と時刻を表示する</p> <p>DATE TIME - いずれかのキーワードを指定します。どちらのキーワードを指定しても上記の機能が実行されます。</p>
DISPLAY {FREE SPACE}	<p>Analytics で使用できる物理メモリ(RAM)の量を表示する</p> <p>表示されるこの数値には、変数用に確保されたメモリは含まれません。デフォルトで、Analytics は変数を格納するために 60 KB の物理メモリを予約しますが、このサイズは必要に応じて自動的に増加します。</p> <p>FREE SPACE - いずれかのキーワードを指定します。どちらのキーワードを指定しても上記の機能が実行されます。</p>
DISPLAY 式	<p>式の結果を表示する</p> <p>式 - 結果の表示対象となる式</p>
DISPLAY 関数	<p>関数の出力を表示する</p> <p>関数 - 出力の表示対象となる関数</p>

例

Analytics テーブルのレイアウトを表示する

テーブルレイアウトを表示する機能は、いろいろな場合に使用できます。たとえば、2つ以上のテーブルを結合する場合や、フィールドの長さやデータ型を調べる場合に使用できます。

次の例では、Ap_Trans テーブルのレイアウトを表示しています。

```
OPEN AP_Trans
DISPLAY
```

DISPLAY コマンドにより行われる画面 への出力は下記のとおりです。

メモ

Analytics コマンド ラインに直接「DISPLAY」を入力する場合には、出力がすぐに表示されます。

これに対し、スクリプト内で DISPLAY を実行した場合には、コマンド ログ内の該当する DISPLAY エントリをダブルクリックすれば出力が表示されます。

画面 への出力

関連

インデックス 'Vendor_on_Vendor_No' を使用した 'Vendor' と 'Vendor_No' の関連付け

ファイル

PRIMARY ファイルは 'Ap_Trans.fil' (書式 'Ap_Trans') です。

レコード長は 59 です。

フィールド

名前	型	開始位置	長さ	小数点以下桁数	フィールドの説明
Vendor_No	ASCII	1	5		AS "業者;番号" WIDTH 7
Invoice_No	ASCII	6	15		AS "請求書;番号"
Invoice_Date	DATETIME	21	8		PICTURE "MM/DD/YY" AS "請求日" WIDTH 8
Invoice_Amount	NUMERIC	29	12	2	PICTURE "(9,999,999.99)" AS "請求;金額" WIDTH 12
Prodno	ASCII	41	9		AS "製品番号"
Quantity	MICRO	50	4	0	SET PICTURE "(9,999,999.99)"
Unit_Cost	NUMERIC	54	6	2	PICTURE "(9,999,999)" AS "単価" SUPPRESS

Analytics プロジェクト内の全変数の値を表示する

DISPLAY VARIABLES はナビゲーターの **変数** タブに表示されるのと同じ情報を出力します。DISPLAY VARIABLES を使用することの1つのメリットは、表示された情報をコピーして貼り付けることができることです。

次の例は、2つのユーザー定義変数と2つのシステム変数を作成して、それらすべての変数の値を表示しています。

```
ASSIGN v_table_name = "Ap_Trans"
ASSIGN v_field_name = "Invoice_Amount"
OPEN %v_table_name%
TOTAL FIELDS %v_field_name%
DISPLAY VARIABLES
```

DISPLAY コマンドにより行われる画面への出力は下記のとおりです。

メモ

Analytics コマンドラインに直接「**DISPLAY VARIABLES**」を入力する場合には、出力がすぐに表示されます。

これに対し、スクリプト内で **DISPLAY VARIABLES** を実行した場合には、コマンドログ内の該当する **DISPLAY VARIABLES** エントリをダブルクリックすれば出力が表示されます。

画面への出力

名前	型	値
TOTAL1	N	278,641.33
OUTPUTFOLDER	C	"/Tables/Accounts_Payable"
v_field_name	C	"Invoice_Amount"
v_table_name	C	"Ap_Trans"

式の結果を表示する

次の例により、選択したレコードにおける Sale_Price フィールドの値と Quantity_on_Hand フィールドの値を乗算した結果が表示されます。

```
DISPLAY Sale_Price * Quantity_on_Hand
```

関数の出力を表示する

次の例により、選択したレコードの Invoice_Date フィールドの日付以来経過した日数が表示されます。

```
DISPLAY AGE(Invoice_Date)
```

備考

コマンドの実行結果の場所

DISPLAY を Analytics のコマンドラインから実行した場合 - 結果は画面に表示されます。

DISPLAY をスクリプト内で実行した場合 - 結果は Analytics コマンド ログに書き込まれます。結果を画面に表示するには、コマンド ログの当該エントリをダブルクリックします。

DO REPORT コマンド

指定された Analytics レポートを生成します。

構文

```
DO REPORT レポート名
```

パラメーター

名前	説明
レポート名	生成し、レポートとして出力するビューの名前。

例

デフォルトのビューを印刷する

AP_Trans テーブルを開いてデフォルトのビューを印刷するとします。

```
OPEN AP_Trans  
DO REPORT デフォルト_ビュー
```

備考

コマンドラインでの DO REPORT の実行とスクリプト内での DO REPORT の実行との比較

レポートを印刷するために使用される設定は、コマンドを実行する場所によって異なります。

- コマンドラインから - [印刷] ダイアログボックスが開き、ページを選択して、レポートの他のオプションを印刷および構成できます。

- スクリプト内 -レポートのデフォルト設定を使用して直ちにレポートが印刷されます。

DO SCRIPT コマンド

ACL スクリプト内から第 2 のまたは外部スクリプトを実行します。

構文

```
DO <SCRIPT> スクリプト名 {<IF テスト>|<WHILE テスト>}
```

パラメーター

名前	説明
SCRIPT スクリプト名	<p>実行対象となるスクリプトの名前。Analytics プロジェクトで副スクリプトを実行するか、.aclscript、.txt、.bat などの拡張子があるテキスト ファイルに格納された外部スクリプトを実行できます。</p> <p>外部スクリプトへのファイルパスを指定できます。スペースがある場合は、パスを引用符で囲む必要があります。</p> <p>メモ 既に実行されているスクリプトを呼び出すことはできません。たとえば、ScriptA が ScriptB を呼び出す場合、ScriptB から ScriptA を呼び出すことはできません。ScriptA は、ScriptB が完了するのを待つ間も実行されていません。</p>
IF テスト 省略可能	<p>スクリプトを実行するかどうかを判断するために 1 回評価される条件式。条件が True と評価された場合はスクリプトを実行します。そうでない場合は実行しません。</p> <p>同じコマンド内で WHILE と一緒に使用することはできません。両方を使用する場合は、スクリプトが処理される際に WHILE が無視されます。コメントがログに入力されますが、スクリプトの実行は停止しません。</p>
WHILE テスト 省略可能	<p>スクリプトの実行後、再度スクリプトを実行するかどうかを判断するために評価される条件式。テストが True と評価された場合はスクリプトを再実行します。そうでない場合は実行しません。</p> <p>メモ WHILE を使用する場合は、テストが最終的に false として評価されるようにしてください。そうしないと、スクリプトが無限ループになります。無限ループになった場合にスクリプト処理を取り消すには、Esc キーを押します。</p> <p>同じコマンド内で IF と一緒に使用することはできません。両方を使用する場合は、スクリプトが処理される際に WHILE が無視されます。コメントがログに入力されますが、スクリプトの実行は停止しません。</p>

例

入力が有効とされるまでサブスクリプトを繰り返し実行する

ダイアログボックスを使ってユーザー入力を収集するサブスクリプトがあるとします。このサブスクリプトは以下を行います。

1. 必要な値の入力をユーザーに求める
2. そのユーザー入力を確認する
3. 入力値が有効とされたら、`v_validated` 変数を true に設定する

ユーザーが有効な入力を行ったことを確認するには、DO SCRIPT を使用し、入力が有効とされるまでスクリプトでこのコマンドが再実行されるように WHILE 条件を追加します。変数の値が有効になると、メインのスクリプトはその次のコマンドに進みます。

```
DO SCRIPT GetUserInput WHILE v_validated = F
```

共有ロケーションからのサブスクリプトの実行

共有ロケーションで複数のユーティリティサブスクリプトを管理することができます。分析時にサブスクリプトのいずれかが必要な場合は、共有ロケーションへの絶対パスを使ってサブスクリプトを参照します。

```
DO SCRIPT "C:\My utility scripts\GetUserInput.ac1script" WHILE v_validated = F
```

備考

関連コマンド

DO SCRIPT は、Analytics の旧リリースで作成されたスクリプト内で使用されている DO BATCH コマンドと同等です。

GROUP コマンド内で DO SCRIPT コマンドを使用することはできません。

外部スクリプトの利便性

スクリプトを外部に保存すれば、Analytics スクリプト内から呼び出せるので、同じサブスクリプトを別の Analytics スクリプトとプロジェクトで再利用することができます。

スクリプトの単一のコピーを1つの場所に保存し、複数の場所で管理するのではなく1つの場所で更新できます。

DUMP コマンド

ファイルや現在のレコードの内容を 16 進、ASCII、および EBCDIC 文字エンコードで表示します。

メモ

このコマンドは、コマンドラインでのみ入力できます。スクリプト内で使用することはできません。

構文

```
DUMP {RECORD|ファイル名} <SKIP バイト> <COLUMN バイト> <HORIZONTAL>
```

パラメーター

名前	説明
RECORD	選択したレコードの内容を表示する ファイル名に値を指定しない場合は必須です。
ファイル名	表示するファイルの名前。 RECORD を指定しない場合は必須です。 メモ Analytics テーブルの文字エンコードを表示するには、ソースデータファイルの名前とファイル拡張子を指定する必要があります。例: <code>Ap_Trans.fil</code>
SKIP バイト数 省略可能	ダンプを開始する前にバイパスするバイト数。デフォルトは 0 です。
COLUMN バイト 省略可能	出力における列の幅(バイト数)。 メモ バイトとして指定された数値は、Analytics のレコードまたはテーブルに含まれるバイト数を示します。 出力におけるエンコードされた文字数は、ビューにおける文字数とは 1 対 1 で対応しない場合もあります。たとえば、数字 1 の 16 進エンコードは <code>31</code> です。 縦方向に表示した場合の各列のデフォルト数は 16 バイトであるのに対し、横方向に表示した場合の 1 列のデフォルト数は 64 バイトです。指定できる最大バイト数は 255 です。

名前	説明
HORIZONTAL 省略可能	文字エンコードが、縦方向である隣接するブロック(デフォルト)でなく、横方向である行に表示されます。

例

Inventory テーブルの文字エンコードを表示する

次の例により、Inventory テーブル内のデータの 16 進、ASCII、EBCDIC 文字エンコードが表示されます。出力は水平行で配置されます(16 進数エンコーディングは 2 行使用します)。各行には、当該の Analytics テーブルからのデータが 97 バイト分、表現されています。

```
DUMP Inventory.fil COLUMN 97 HORIZONTAL
```

DUPLICATES コマンド

重複値または重複レコードが Analytics テーブル内に存在するかどうかを検出します。

構文

```
DUPLICATES {<ON> キー フィールド <D> <...n>|<ON> ALL <EXCLUDE フィールド名 <...n>>}
<OTHER フィールド <...n>|OTHER ALL <EXCLUDE フィールド名 <...n>>> <UNFORMATTED>
<ADDGROUP> <PRESORT> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <APPEND>
<OPEN> <TO {SCREEN|テーブル名|ファイル名|PRINT}> <LOCAL> <HEADER ヘッダー テキスト>
<FOOTER フッターテキスト> <ISOLocale ロケール コード>
```

パラメーター

名前	説明
ON キーフィールド D <...n> ON ALL	<p>重複がないかどうかをテストする 1 つまたは複数のキーフィールド、あるいは式。</p> <ul style="list-style-type: none"> ON - キーフィールドには、指定された 1 つまたは複数のフィールドが使用されます。複数のフィールドでテストする場合、レコードが重複として検出されるには、指定したすべてのフィールドの値が同一である必要があります。フィールドはリストの順序で出力結果に含まれます。キーフィールドを降順に並べ替える D を含めます。デフォルトのソート順は昇順です。 ON ALL - テーブル内のすべてのフィールドが使用されます。テーブル内のすべてのフィールドをテストする場合、レコードが重複として検出されるには、レコードが完全に同一である必要があります。フィールドは、テーブルレイアウトに現れる順序と同じ並びで出力結果に含まれます。ON ALL では昇順でしか並べ替えることができません。 <p>メモ レコードの未定義部分はテストされません。</p>
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用して重複をテストするときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p>

名前	説明
	<pre>ON ALL EXCLUDE field_1 field_2</pre>
OTHER フィールド <...n> OTHER ALL 省略可能	<p>出力に含める 1 つ以上の追加フィールド。</p> <ul style="list-style-type: none"> OTHER フィールド <...n> - 指定した 1 つまたは複数のフィールドが含まれます。 OTHER ALL - キーフィールドとして指定しなかった、テーブル内のすべてのフィールドが含まれます。
EXCLUDE フィールド名 省略可能	<p>OTHER ALL を使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、OTHER ALL を微調整できます。</p> <p>EXCLUDE は OTHER ALL の直後に置く必要があります。例：</p> <pre>OTHER ALL EXCLUDE field_1 field_2</pre>
UNFORMATTED 省略可能	<p>結果をファイルに出力する場合、ページ見出しや改ページは除去されます。</p>
ADDGROUP 省略可能	<p>出力テーブルにグループ番号フィールド (<code>GROUP_NUM</code>) を含めます。</p> <p>グループ番号フィールドは、一意の各重複値グループに対し、1 ずつ増える連番を割り当てたものです。</p> <p>ヒント 重複値グループを番号で参照するこの機能は、出力テーブル内のデータを分析する場合に使用できます。</p>
PRESORT 省略可能	<p>コマンドを実行する前にキーフィールドでテーブルを並べ替えます。</p> <p>メモ GROUP コマンドの内部では PRESORT を使用することができません。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター (WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲	<p>処理するレコード数：</p>

名前	説明
省略可能	<ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
TO SCREEN テーブル名 ファイル名 PRINT 省略可能	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analyticsの表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となるAnalyticsテーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analyticsプロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <ul style="list-style-type: none"> ◦ ファイル名 -は結果の保存先となるファイルです。

名前	説明
	<p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> <p>◦ 印刷 – 通常使うプリンターに結果を送信します</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ Analytics テーブルである出カファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
ISOLOCALE ロケールコード 省略可能	<p>メモ Analytics の Unicode 版にのみ適用されます。</p> <p>システム ロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「<code>fr-ca</code>」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

Analytics の出力変数

名前	含む
GAPDUP n	コマンドによって確認されたギャップ、重複、またはあいまい重複グループの合計数。

例

1つのフィールドの重複値のテスト

次の例：

- **Invoice_Number** フィールドの重複値のテスト
- 重複する請求書番号を含むレコードを新しい Analytics テーブルに出力する

```
DUPLICATES ON Invoice_Number OTHER Vendor_Number Invoice_Date Invoice_Amount PRESORT TO "Duplicate_Invoices.FIL"
```

2つ以上のフィールドの組み合わせの重複値のテスト

次の例：

- **Invoice_Number** および **Vendor_Number** フィールドの値の重複した組み合わせのテスト
- 同じ請求書番号と同じ業者番号を含むレコードを新しい Analytics テーブルに出力する

このテストと前のテストの間の違いは、2つの異なる業者番号からの同じ請求書番号が誤検出として報告されないことです。

```
DUPLICATES ON Invoice_Number Vendor_Number OTHER Invoice_Date Invoice_Amount PRESORT TO "Duplicate_Invoices.FIL"
```

重複するレコードを検索する

下記の例では以下が行われます。

- Inventory テーブルのすべてのフィールドに重複値がないかどうかをテストする
- 完全に同じレコードを新しい Analytics テーブルに出力する

```
DUPLICATES ON ProdNum ProdClass Location ProdDesc ProdStatus UnitCost CostDate SalePrice PriceDate PRESORT TO "Duplicate_Inventory_Items.FIL"
```

この構文は、`ALL` を使用すれば次のように簡素化できます。

```
DUPLICATES ON ALL PRESORT TO "Duplicate_Inventory_Items.FIL"
```

重複値の出カテーブルをグループ番号でフィルター処理する

複数のキーフィールドを組み合わせて使用することで、買掛金テーブルに重複レコードがないかどうかをテストすることができます。

- Vendor_No(ベンダー番号)
- Invoice_No(請求書番号)
- Invoice_Date(請求書番号)
- Invoice_Amount(請求金額)

結果の重複値出カテーブルをフィルター処理することで、いくつかの重複値グループのみを追加処理することができます。

キーフィールドの組み合わせを使ってフィルターを作成することは、次のように骨が折れます。例：

```
SET FILTER TO ((Vendor_No = "11475") AND (Invoice_No = "8752512") AND
(Invoice_Date = `20191021`) AND (Invoice_Amount = 7125.80)) OR
((Vendor_No = "12130") AND (Invoice_No = "589134") AND (Invoice_Date =
`20191117`) AND (Invoice_Amount = 10531.71)) OR ((Vendor_No = "13440")
AND (Invoice_No = "5518912") AND (Invoice_Date = `20191015`) AND
(Invoice_Amount = 11068.20))
```

このようにする代わりに、次のようにグループ番号を基準にしたフィルターを作成することでも、同じ結果が得られます。

```
SET FILTER TO MATCH(GROUP_NUM, 3 , 8, 11)
```

備考

このコマンドの動作の詳細については、「重複の検査」ページ 1265を参照してください。

並べ替えと重複

一般に、DUPLICATES コマンドは、並べ替えた1つまたは複数のキーフィールドに対してのみ実行する必要があります。キーフィールド内の重複値が、それら同士がすぐ隣に並んでいる場合にのみ検出できるためです。

キーフィールドを並べ替えずにDUPLICATES コマンドを実行しても、隣り合っていない重複値は重複値として報告されません。同じ重複値の塊が2つ以上ある場合でも、それらの塊は別々のグループの重複値として報告されます。

分析の目的によっては、キーフィールドを並べ替えずにDUPLICATES コマンドを実行することにも意味がある場合があります。たとえば、ソーステーブル内ですぐ隣に並んでいる重複値のみを検出して、隣り合っていない重複値は無視したい場合が、その例です。

ESCAPE コマンド

Analytics を終了することなく、処理中のスクリプト、あるいはすべてのスクリプトを終了します。

メモ

スクリプトを終了して、Analytics も終了するには、「QUIT コマンド」ページ 2027を参照してください。

構文

```
ESCAPE <ALL> <IF テスト>
```

パラメーター

名前	説明
ALL 省略可能	アクティブなすべてのスクリプトが強制終了されます。このパラメーターを省略した場合は、現在のスクリプトが強制終了されます。
IF テスト 省略可能	コマンドが実行されるのに True として評価される必要のあるテスト。テストが False と評価された場合、コマンドは実行されません。

例

スクリプトを条件付きで強制終了する

次の例では、テーブル内のレコード数をカウントし、カウントしたレコード数が 100 未満である場合は、ESCAPE コマンドを使ってスクリプトを強制終了します。

```
COUNT  
ESCAPE IF COUNT1 < 100
```

備考

ESCAPE を使用する場面

ESCAPE を使用する場面は、条件に基づいてスクリプトまたはサブスクリプトの実行を停止する場合や、実行中のすべてのスクリプトの実行を停止する場合です。

サブスクリプト内で ESCAPE を使用する

サブスクリプト内で ESCAPE を実行すると、そのサブスクリプトが実行を停止し、メインのスクリプトが、そのサブスクリプトを呼び出した DO SCRIPT コマンドから処理を再開します。

サブスクリプトの ESCAPE コマンドに ALL オプションを追加した場合は、サブスクリプトとメインのスクリプトがどちらも処理を停止します。

```
ESCAPE ALL
```

EVALUATE コマンド

レコード サンプルまたは金額単位 サンプルでは、サンプルされたデータで決定された誤謬を基に母集団全体を予測し、逸脱率の上限または誤謬額を計算します。

レコードのサンプル 金額単位のサンプル

構文

```
EVALUATE RECORD CONFIDENCE 信頼度 SIZE サンプルサイズ ERRORLIMIT 誤謬の数
<TO SCREEN|ファイル名>
```

パラメーター

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
RECORD	レコード サンプルで見つかった誤謬を評価する
CONFIDENCE 信頼度	サンプル サイズの計算時に指定したのと同じ信頼度
SIZE サンプル数	サンプルのレコード数。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>抽出した実際のサンプル サイズを指定します。これは、Analytics によって当初計算されたサンプル サイズと異なってもかまいません。</p> </div>
ERRORLIMIT 誤謬の数	サンプルで見つかった誤謬または逸脱の総数。
TO SCREEN ファイル名	コマンドの結果を送信する場所： <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> </div>

名前	説明
	<ul style="list-style-type: none"> ○ ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code>

Analytics の出力変数

名前	含む
MLEn	コマンドによって計算された最大誤謬率(計算された上限逸脱率)。

例

レコードの母集団全体に対して、サンプルされたデータで見つかった誤謬を推定する

サンプルデータのテストを完了し、見つかったコントロール逸脱を記録しました。検出したすべての誤謬を母集団に対して投影できます。

以下の例は、サンプリングされたデータで見つかった2つの誤謬を基に母集団全体を予測し、**最大誤謬率**(計算される上限逸脱率)を6.63%と算出しました。

```
EVALUATE RECORD CONFIDENCE 95 SIZE 297 ERRORLIMIT 7 TO SCREEN
```

Analytics で誤謬を評価する際に値を計算する方法の詳細については、「レコード サンプルの誤謬を評価する」ページ 1056を参照してください。

備考

このコマンドの動作の詳細については、"レコード サンプルの誤謬を評価する" ページ 1056を参照してください。

構文

```
EVALUATE MONETARY CONFIDENCE 信頼度 <ERRORLIMIT 簿価, 誤謬額 <,...n>>
INTERVAL 間隔値 <TO SCREEN|ファイル名>
```

パラメーター

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
MONETARY	金額単位サンプルで見つかった誤謬を評価する
CONFIDENCE 信頼度	サンプルサイズの計算時に指定したのと同じ信頼度
ERRORLIMIT 簿価,誤謬額	<p>サンプルで見つかった虚偽表示の全誤謬。</p> <p>虚偽表示額と金額の簿価をカンマで区切って指定します。たとえば、金額の簿価が\$1,000で、金額の監査額が\$930の場合は、<code>1000,70</code>と指定します。</p> <p>過剰計上は正の金額として、過少計上は負の金額として指定します。たとえば、金額の簿価が\$1,250で、金額の監査額が\$1,450の場合は、<code>1250,-200</code>と指定します。</p> <p>簿価と誤謬額のペアが複数組ある場合はカンマで区切ってください。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px; width: fit-content;"> <code>1000,70,1250,-200</code> </div>
INTERVAL 間隔値	<p>サンプルの抽出時に使用した間隔値。</p> <div style="margin-top: 10px;"> <p>メモ</p> <p>使用した間隔値は、Analyticsによって当初計算された間隔値と異なっていてもかまいません。</p> </div>
TO SCREEN ファイル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します

名前	説明
	<p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code>

Analytics の出力変数

名前	含む
MLEn	コマンドによって計算された推定誤謬額(推定による虚偽表示)。
UELn	コマンドによって計算された最大誤謬額(虚偽表示上限)。

例

レコードの母集団全体に対して、サンプルされたデータで見つかった誤謬を推定する

サンプルデータのテストを完了し、見つかった虚偽表示を記録しました。検出したすべての誤謬を母集団に対して投影できます。

以下の例は、サンプリングされたデータで見つかった誤謬を基に母集団全体を予測し、以下を含む7つの値を計算しています。

- 基準となる精度** - サンプリングリスクの基本的な許容誤謬(18,850.00)
- 推定誤謬** - 推定による母集団全体の虚偽表示額(1,201.69)
- 最大誤謬率** - 母集団の虚偽表示上限(22,624.32)

コマンド

```
EVALUATE MONETARY CONFIDENCE 95 ERRORLIMIT 1000,70,1250,-200,3200,900  
INTERVAL 6283.33 TO SCREEN
```

Analytics で誤謬を評価する際に値を計算する方法の詳細については、"金額単位サンプルの誤謬を評価する" ページ 1087を参照してください。

備考

このコマンドの動作の詳細については、"金額単位サンプルの誤謬を評価する" ページ 1087を参照してください。

EXECUTE コマンド

Analytics の外部にあるアプリケーションやプロセスを実行します。Windows の Run コマンドをエミュレートします。Windows のコマンド プロンプトを操作するために使用できます。

メモ

EXECUTE コマンドでは、Analytics 外部のオペレーティングシステムやアプリケーションを操作することができるため、Analytics のネイティブ機能の範囲を超えた技術的な問題が発生する可能性があります。

サポート チームでは、Analytics 内における EXECUTE コマンドの操作の手助けはできませんが、Analytics 外部のプロセスやアプリケーションで発生する問題についてはサポートの対象外です。

構文

```
EXECUTE Windows_Run_コマンド構文 <ASYN</pre>

```

パラメーター

名前	説明
Windows_Run_コマンド構文	<p>実行するアプリケーションの名前、開くフォルダーまたはファイル、あるいは実行するコマンドと、任意の必要な引数やコマンド スイッチ。</p> <p>引用符で囲まれた有効な Windows Run コマンドである必要があります。</p>
ASYN 省略可能	<p>このコマンドを非同期モードで実行します。</p> <p>非同期モードでは、Analytics スクリプトは、EXECUTE コマンドによって開始されたプロセスが完了するのを待たずに、実行を続行します。</p> <p>ASYN
C を省略した場合は、Analytics スクリプトが続行されるには、EXECUTE コマンドによって開始されたプロセスが完了する必要があります。外部プロセスが実行中の場合は、Analytics にはアクセスできません。</p> <p>メモ</p> <p>Analytics のコマンド ラインから EXECUTE を使用する場合は、ASYN
C を指定する必要があります。</p>

Analytics の出力変数

名前	含む
RETURN_CODE	<p>EXECUTE コマンドを使用した外部アプリケーションまたはプロセス実行によって返されるコード(リターンコード)。</p> <p>リターンコードの概要</p> <p>リターンコードは外部アプリケーションまたはプロセスによって生成される数値であり、Analytics に返されることで外部プロセスの結果を示します。Analytics はリターンコードを出力せず、入力として受け付けるだけです。</p> <p>一般的なリターンコード</p> <p>通常、リターンコードは数値で、特定の通知やエラーメッセージが割り当てられています。たとえば、リターンコード "0" は "処理が正常終了した" ことを示します。リターンコード "2" は "指定されたファイルが見つからない" ことを示します。</p> <p>特定のリターンコードの意味</p> <p>特定のリターンコードとその意味は外部アプリケーションまたはプロセスによって異なります。リターンコードは "エラーコード" や "終了コード" とも呼ばれ、それらの意味は、関連する外部アプリケーションのドキュメントに記載されている可能性があります。リターンコードの一覧はインターネット上でも見ることができます。</p> <p>デフォルト モードでのみ作成された変数</p> <p>RETURN_CODE 変数は、デフォルト モードでEXECUTE コマンドが実行されるときに、作成されます。変数は、コマンドが非同期モードで実行されるときには、作成されません。</p>

例

アプリケーションを開く

Microsoft Excel を開く:

```
EXECUTE "Excel"
```

Adobe Acrobat Reader を開く

```
EXECUTE "AcroRd32.exe"
```

アプリケーションを閉じる

Microsoft Excel を閉じる:

```
EXECUTE "TASKKILL /f /im Excel.exe"
```

メモ

/f スイッチは慎重に使用してください。これは、変更の保存を確認するダイアログボックスなど、一切のダイアログボックスを表示しないで、アプリケーションを強制的に閉じます。

ファイルを開く

Excel ワークブック **AP_Trans.xlsx** を開く

```
EXECUTE "'C:\ACL Projects\Source Data\AP_Trans.xlsx'"
```

新しいフォルダーを作成する

新しいフォルダーの**ソース データ**を作成する

```
EXECUTE 'cmd /c MD "C:\ACL Projects\Source Data"'
```

外部スクリプト、または Analytics の外部のバッチファイル(.bat)を実行する

スクリプト **My_Batch.bat** を実行する

```
EXECUTE '"C:\ACL Projects\Batch Files\My_Batch.bat"'
```

Analytics の外部のバッチファイルにパラメーターを渡す

2つのパラメーターを `My_Batch.bat` に渡します。パラメーターには、リテラルまたは Analytics 変数を指定できます。

```
EXECUTE '"C:\ACL Projects\Batch Files\My_Batch.bat" param1%v_param2%'
```

他の Analytics プロジェクトで Analytics スクリプトを実行する

`AP Trans Tests.ac1` で "AP_Trans_script" を実行する"

```
EXECUTE 'aclwin.exe "C:\ACL Projects\AP Trans Tests.ac1" /b AP_Trans_script'
```

メモ

別のプロジェクト内で Analytics スクリプトを実行すると、もう1つの Analytics インスタンスが起動されます。そのもう1つの Analytics インスタンスを閉じたら制御が1番目のインスタンスに戻るように、2番目のプロジェクト内のスクリプトの末尾に QUIT コマンドを記述しておく必要があります。

Analytics スクリプトに待ち時間を組み込む

どちらの例も、30秒の待ち時間を設けます。

```
EXECUTE "TIMEOUT /t 30"
```

```
EXECUTE "cmd /c PING -n 31 127.0.0.1 > nul"
```

備考

EXECUTE を使用して便利なタスクを実行する

EXECUTE コマンドは、Windows コマンドおよび DOS コマンドを Analytics のコマンド ラインから、または Analytics スクリプトから実行できるようにします。

この機能を使用すると、以下に挙げる多様で便利なタスクを実行することにより、Analytics スクリプトの自動化を向上させることができます。これらのタスクは、ACLScript 構文だけを使用しては不可能です。

EXECUTE 使用して開始できるタスクの例

他のプログラムやアプリケーションを開いて、Analytics スクリプトで必要なタスクを実行する	バッチ ファイルにパラメーターを渡す	ネットワークの場所からデータにアクセスする	Active Directory アカウント一覧を組み込む
任意のファイルとその既定のアプリケーションで開く	他の Analytics プロジェクトで Analytics スクリプトを実行する	FTP を使用して、リモートの場所からデータにアクセスする	VBScript と統合する
Analytics の外に存在するファイルやフォルダーのコピー、移動、作成、削除、または比較などのファイルおよびフォルダーの管理タスクを実行する	Analytics スクリプトに待ち時間を組み込む	データを圧縮または解凍する	SQL データベースと統合する
外部スクリプト、または Analytics の外部のバッチ ファイル(.bat) を実行する	Analytics スクリプトに Windows のタスクスケジュールを組み込む	データを暗号化または復号する	Web ページを開く

メモ

これらのタスクのいずれかを実行する方法の具体的な詳細は、Diligent ヘルプドキュメントで記載していません。サポートが必要な場合は、適切な Windows オペレーティングシステムのドキュメント、またはその他サードパーティのドキュメントを参照してください。

デフォルト モードと非同期モード

デフォルト モードまたは非同期モードで、EXECUTE コマンドを実行できます。

- デフォルト モード** - EXECUTE で開始するプロセスは、Analytics スクリプトを続行する前に、完了する必要があります。
 - 外部プロセスが実行中の場合は、Analytics にはアクセスできません。
- 非同期モード** - Analytics スクリプトは、EXECUTE コマンドによって開始されたプロセスが完了するのを待たずに、実行を続行します。

コマンド

外部プロセスが実行中の場合は、Analytics にはアクセスし続けることができます。
ASYNC 指定する場合、EXECUTE コマンドは非同期モードで実行されます。

使用するモード

EXECUTE コマンドを使用する Analytics スクリプトを作成する場合は、どちらのモードの操作が適切であるかを考慮する必要があります。

デフォルト モードを使用する	非同期モードを使用する / ASYNC
<ul style="list-style-type: none">ファイルおよびフォルダー管理タスク待機期間の指定後続のタスクが依存するタスクすぐ後のスクリプトの実行が RETURN_CODE 変数の結果に依存します	<ul style="list-style-type: none">外部タスクにより、アプリケーション インターフェイスまたはポップアップ ダイアログボックスが開きます

自動実行される Analytics スクリプト

EXECUTE コマンドを含む Analytics スクリプトを自動で実行する場合は、次のいずれかの方法を使用します。

- タスクで非同期モードを使用することにより、アプリケーション インターフェイスまたはポップアップ ダイアログボックスが開きます
- 自動スクリプトでインターフェイス要素を開かないようにします

メモ

インターフェイス要素が閉じるまで、実行中のプロセスを表します。これらのインターフェイス要素がデフォルト モードで EXECUTE とともに開く場合、Analytics スクリプトの後続の行は実行されず、スクリプトがハングします。

ロボットの EXECUTE コマンド

ロボットのアナリティクススクリプトで、EXECUTE コマンドを使用する場合は、特に実行するコマンドを構成する必要があります。詳細については、[ロボット エージェントの構成](#)を参照してください。

二重引用符

EXECUTE コマンドで使用する Windows の Run コマンドの構文は、一重引用符または二重引用符で囲む必要があります。

次の例は、Windows **MD** コマンドを使用して新しいフォルダーを作成しています。

```
EXECUTE 'cmd /c md C:\New_Data_Folder'
```

ネストされた二重引用符

Run コマンド構文内のパスにスペースが含まれている場合は、そのパスも引用符で囲む必要があります。引用符でパスを囲むときには、2つのオプションがあります。

- **単一引用符内の二重引用符** - 単一引用符を使用して、Run コマンド文字列全体を囲み、二重引用符を内部で使用し、パスを囲みます

```
EXECUTE 'cmd /c md "C:\New Data Folder"'
```

2番目の方法よりも、この方法の方が容易である場合があります。

メモ

入れ子の順序を逆にする、つまり、二重引用符で文字列全体を囲み、一重引用符でパスを囲むと、動作しません。

- **2つの二重引用符** - 二重引用符を使用して、Run コマンド文字列全体を囲み、2つの二重引用符を内部で使用してパスを囲みます。

```
EXECUTE "cmd /c md ""C:\New Data Folder"""
```

この2番目の方法を使用する場合、内部的に使用される2つの二重引用符は隣接する必要があり、スペースを含めることはできません。

内部コマンドおよび外部コマンド

Windows コマンドは、**内部**または**外部**のいずれかです。

- **内部コマンド** - は、コマンドプロンプトからのみ実行できます。これはつまり、コマンドを指定する前に、`cmd /c` または `cmd /k` を用いて、コマンドシェルを開いておく必要があるということです。
- **外部コマンド** - は、コマンドプロンプトからでも、直接 EXECUTE コマンドを用いても実行できます。これはつまり、コマンドシェルを開くことは任意であり、必須ではないということです。

次の例では、違いを説明するために、内部 Windows `DIR` コマンド(ディレクトリの内容を表示する)と外部 Windows `COMP` コマンド(2つのファイルを比較する)を使用しています。

```
EXECUTE 'cmd /k dir "C:\ACL DATA\Sample Data Files"'
```

```
EXECUTE 'comp C:\File_1.txt C:\File_2.txt'
```

Windows コマンドを含んでいる外部スクリプトまたはバッチファイルを作成し、バッチファイルを起動するためだけに EXECUTE コマンドを使用することによって、この複雑な事態を避けることができます。例：

```
EXECUTE 'C:\My_Batch.bat'
```

複数行のRun コマンド構文

EXECUTE コマンドは、複数行のRun コマンド構文をサポートしていません。Analytics スクリプトに複数行のRun コマンドを組み込むには、以下の方法のいずれかを使用します。

方法	例
Run コマンドごとに EXECUTE コマンドを繰り返します。	<pre>EXECUTE 'cmd /c md "C:\New Data Folder"' EXECUTE 'cmd /c copy C:\File_1.txt "C:\New Data Folder"'</pre>
'&'を使用して、複数のRun コマンドを結合します。	<pre>EXECUTE 'cmd /c md "C:\New Data Folder" & copy C:\File_1.txt "C:\New Data Folder"'</pre>
複数行のRun コマンドを含んでいる外部スクリプトまたはバッチファイルを作成し、バッチファイルを起動するためだけに EXECUTE コマンドを使用します。	<pre>EXECUTE 'C:\My_Batch.bat'</pre>

EXPORT コマンド

指定されたファイル形式に、または HighBond 内のリザルト アプリまたはロボット アプリに、Analytics からデータをエクスポートします。

メモ

EXPORT コマンドで古い Excel ファイル (*.xls) や Microsoft Access ファイル (*.mdb) を扱うには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります詳細については、"Microsoft Access データベースエンジンを除外するオプション" ページ 2664を参照してください。

構文

```
EXPORT { <FIELDS> フィールド名 <AS >エクスポート名 <...n> | <FIELDS> ALL <EXCLUDE >フィールド名 <...n> } <UNICODE> エクスポート形式 <SCHEMA> <PASSWORD >番号 TO { ファイル名 | aclgrc_id | highbond_api_url } <OVERWRITE> <IF >テスト <WHILE >テスト < { FIRST 範囲 | NEXT 範囲 } > <APPEND> <KEEPTITLE> <SEPARATOR >文字 <QUALIFIER >文字 <WORKSHEET >ワークシート名 <DISPLAYNAME >
```

パラメーター

名前	説明
FIELDS フィールド名 AS エクスポート名 <...n> FIELDS ALL	<p>エクスポートするフィールド。</p> <ul style="list-style-type: none"> FIELDS フィールド名 - 指定されたフィールドをエクスポートします フィールド名はスペースで区切ります。フィールドは一覧の順序でエクスポートされます。 任意で AS エクスポート名を使って、エクスポート ファイル内のフィールドとは異なる名前を追加することもできます。エクスポート名は引用符で囲みます。 HighBond リザルト (ACLGRC) へのエクスポートを行う場合は、AS と DISPLAYNAME パラメーターを組み合わせることもできます。詳細については、"HighBond のリザルトにエクスポートを行う際の DISPLAYNAME と AS の相互作用" ページ 1797を参照してください。 FIELDS ALL - はテーブルのすべてのフィールドをエクスポートします。 フィールドは、テーブルレイアウトに表示されるとおりの順でエクスポートされます。
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用してエクスポートするときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p>

名前	説明
	<p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p>FIELDS ALL EXCLUDE <i>field_1 field_2</i></p> </div>
<p>UNICODE 省略可能</p>	<p>Analytics の Unicode 版でのみ利用可能です。テキストファイル、区切り文字付きテキストファイル、XML ファイル、Windows クリップボード出力にのみ適用されます。(ASCII、DELIMITED、XML、CLIPBOARD)</p> <p>Unicode UTF-16 LE 文字エンコードが適用されている Analytics データをエクスポートします。</p> <ul style="list-style-type: none"> ○ UNICODE を指定 -エクスポートしているデータが拡張 ASCII(ANSI)によってサポートされていない場合 ○ エクスポートしているデータのすべての文字が拡張 ASCII(ANSI)によってサポートされている場合は、Unicode -を指定しないでください <p>エクスポートされたデータは拡張 ASCII(ANSI)としてエンコードされます。</p> <p>メモ サポートされていない文字はエクスポートされたファイルから省略されます。</p> <p>詳細については、"Diligent Unicode 製品" ページ 2672を参照してください。</p>
<p>エクスポート形式</p>	<p>出力ファイル形式または宛先。</p> <p>次のいずれかのオプションを指定します。</p> <ul style="list-style-type: none"> ○ ACCESS - Microsoft Access データベースファイル(*.mdb) デフォルトでは、データは Unicode としてエクスポートされます。 ○ ACLGRC - HighBond リザルト ○ ASCII - ASCII プレーンテキスト(*.txt) ○ CLIPBOARD - Windows クリップボード ○ DBASE - dBASE 互換ファイル(*.dbf) ○ DELIMITED - 区切り文字付きテキストファイル(.del)、またはカンマ区切り値ファイル(.csv) ○ EXCEL - Microsoft Excel 1997 ~ 2003 と互換性のある Excel ファイル(*.xls) ○ HBDATA - HighBond のロボット アプリの HighBond のロボットまたはワークフローロボットに対して圧縮されたカンマ区切り値ファイル(*.csv.gz) ○ JSON - JSON ファイル(*.json) ○ LOTUS - Lotus 123 ファイル ○ WDPF6 - Wordperfect 6 ファイル ○ WORD - MS Word ファイル(*.doc) ○ WP - Wordperfect ファイル ○ XLS21 - Microsoft Excel バージョン 2.1 のファイル(*.xls) ○ XLSX - Microsoft Excel ファイル(*.xlsx) <p>デフォルトでは、データは Unicode としてエクスポートされます。</p> <ul style="list-style-type: none"> ○ XML - XML ファイル(*.xml)
<p>SCHEMA 省略可能</p>	<p>XML ファイルへの出力のみに適用されます。(XML)</p> <p>エクスポートされた XML ファイルに XML スキーマを含めます。XML スキーマには、フィールドのデータ型など、XML ファイルの構造を記述するメタデータが含まれています。</p> <p>ファイルがエクスポートされたら、スキーマに対してファイルを検証することができます。</p>
<p>PASSWORD 番号</p>	<p>HighBond アプリにのみ適用されます。(ACLGRC、HBDATA)</p>

名前	説明					
省略可能	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前にPASSWORD コマンドかSET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジューリングしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ <p>PASSWORD 番号は、TO の直前またはコマンド構文の文字列の最後に配置する必要があります。</p> <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、「パスワード定義の作成とパスワード値の指定」ページ 1798を参照してください。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>PASSWORD 番号が必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="488 911 1247 1199" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td data-bbox="488 911 867 1052">Analytics (オンラインアクティベーション)</td> <td data-bbox="867 911 1247 1052">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="488 1052 867 1142">Analytics (オフラインアクティベーション)</td> <td data-bbox="867 1052 1247 1142" rowspan="2">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="488 1142 867 1199">ロボット</td> </tr> </tbody> </table> </div>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます					
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。					
ロボット						
TO ファイル名	<p>ファイルにエクスポートする場合にはファイルの名前。</p> <p>必要に応じて、絶対ファイルパスまたは相対ファイルパスのいずれかを含めることができますが、そのWindows フォルダは既に存在している必要があります。ファイル名の値は引用符で囲まれた文字列として指定する必要があります。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>カンマ区切り値ファイル(*.csv)にエクスポートするには、ファイル名の一部として.csv ファイル拡張子を指定する必要があります。例：<code>vendors.csv</code></p> </div>					
TO ACL GRC ID	<p>HighBond リザルト アプリにエクスポートするにはリザルトの宛先。(ACLGRC)</p> <p>ACL GRC ID 値には、リザルトの統制テストのID番号が含まれている必要があるほか、北米(US)以外のデータセンターへエクスポートする場合には、そのデータセンターのコードも含まれている必要があります。ACL GRC ID 値は引用符で囲む必要があります。</p> <p>統制テストID番号とデータセンターコードは、記号(@)で区切られる必要があります。例：<code>TO "99@eu"</code>。</p> <p>統制テストID番号がわからない場合は、Analyticsのユーザーインターフェイスを使用して、リザルトへのエクスポートを開始します。統制テストID番号を特定したら、エクスポートをキャンセルします。詳細については、「HighBondのリザルト アプリへの例外のエクスポート」ページ 216を参照してください。</p> <p>データセンターコードは、どの地域のHighBond サーバーにデータをエクスポートするのかを指定します。</p>					

名前	説明				
	<ul style="list-style-type: none"> af- アフリカ (南アフリカ) ap- アジア太平洋 (シンガポール) au- アジア太平洋 (オーストラリア) ca- 北米 (カナダ) eu- ヨーロッパ (ドイツ) sa- 南米 (ブラジル) us- 北米 (米国) <p>組織の HighBond インスタンスに対して承認されているデータセンターコードのみを使用できます。北米のデータセンターがデフォルトであるため、@us を指定するのはオプションです。</p>				
<p>TO HighBond API URL</p>	<p>HighBond ロボット アプリにエクスポートするにはロボットの宛先。(HBDATA)</p> <p>メモ ACLロボットにはエクスポートできません。HighBond ロボットまたはワークフローロボットにエクスポートする必要があります。</p> <p>HighBond API URL 値は、宛先ロボットの HighBond API 要求 URL です。以下のエクスポート例では、*.csv.gz ファイルが次の宛先にエクスポートされます。</p> <ul style="list-style-type: none"> ID 番号 52053 のロボット (ID 番号 1000236 の HighBond 組織、 HighBond 北米 (US) のデータセンター内) <pre>TO "https://apis-us.highbond.com/v1/orgs/1000236/robots/52053/working_files?env=development"</pre> <p>env クエリ文字列パラメーターを使用して、ファイルをロボットの開発モードにエクスポートするか本番モードにエクスポートするかを指定します。</p> <table border="1" data-bbox="399 1209 1414 1402"> <tr> <td data-bbox="399 1209 907 1346"> <p>開発モード (デフォルト)</p> </td> <td data-bbox="907 1209 1414 1346"> <ul style="list-style-type: none"> ?env=development または クエリ文字列を指定しない </td> </tr> <tr> <td data-bbox="399 1346 907 1402"> <p>本番モード</p> </td> <td data-bbox="907 1346 1414 1402"> <ul style="list-style-type: none"> ?env=production </td> </tr> </table>	<p>開発モード (デフォルト)</p>	<ul style="list-style-type: none"> ?env=development または クエリ文字列を指定しない 	<p>本番モード</p>	<ul style="list-style-type: none"> ?env=production
<p>開発モード (デフォルト)</p>	<ul style="list-style-type: none"> ?env=development または クエリ文字列を指定しない 				
<p>本番モード</p>	<ul style="list-style-type: none"> ?env=production 				
<p>OVERWRITE 省略可能</p>	<p>HighBond アプリにのみ適用されます。(ACLGRC、HBDATA)</p> <ul style="list-style-type: none"> OVERWRITE 指定 -対象統制テスト(テーブル)またはロボットのすべての既存のデータが、エクスポートされたデータによって上書きされます。データを上書きするには、ターゲット コレクションで、Professional 部門長ロールが必要です。 OVERWRITE 省略 -対象統制テスト(テーブル)のすべての既存のデータの最後に、エクスポートされたデータが追加されます。詳細については、"HighBond のリザルト へのエクスポート" ページ 1796を参照してください。 <p>ロボットの *.csv.gz ファイルでは、この追加動作はサポートされていません。</p> <p>対象とする統制テスト(テーブル)の解釈は、上書きか追加かには関係なく、インポートされたデータを反映するために動的に更新されます。</p>				
<p>IF テスト</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対して</p>				

名前	説明
省略可能	のみ実行されます。 メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。
WHILE テスト 省略可能	各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。 メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。
FIRST 範囲 NEXT 範囲 省略可能	処理するレコード数： <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します 範囲 は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。
APPEND 省略可能	テキストファイルと区切り文字付きテキストファイルのみに適用されます。(ASCII、 DELIMITED) コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。 メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。 <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。
KEEPTITLE 省略可能	テキストファイル、区切り文字付きテキストファイル、カンマ区切り値ファイルのみに適用されます(ASCII、 DELIMITED) Analytics フィールド名とエクスポートされたデータを含めます。省略された場合、フィールド名は出力ファイルに表示されません。
SEPARATOR 文字 省略可能	区切り文字付きテキストファイルとカンマ区切り値ファイルのみに適用されます。(DELIMITED) フィールド間の区切りとして使用する文字。文字は引用符で囲まれた文字列として指定する必要があります。 デフォルトでは、カンマが使用されます。カンマ区切り値ファイルにエクスポートしている場合は、カンマ以外の文字を指定しないでください。
QUALIFIER 文字 省略可能	区切り文字付きテキストファイルとカンマ区切り値ファイルのみに適用されます。(DELIMITED) フィールド値を折り返すためと識別するためにテキスト修飾子として使用する文字。文字は引用符で囲まれた文字列として指定する必要があります。

名前	説明
	デフォルトでは、二重引用符が使用されます。
WORKSHEET ワークシート名 省略可能	Microsoft Excel *.xlsx ファイルのみに適用されます。(XLSX) 新規または既存の Excel ファイルに作成される Excel ワークシートの名前。 デフォルトでは、ワークシート名としてエクスポートする Analytics テーブルの名前が使用されます。 ワークシート名には、英数文字とアンダースコア文字 (_) のみを含めることができます。名前に特殊文字や空白を使用したり、名前を数字で始めることはできません。値を引用符で囲むのは任意です。 エクスポートするときに Excel ワークブックとワークシートを上書きする詳細については、"WORKSHEET パラメーターと上書き" ページ 1795を参照してください。
DISPLAYNAME 省略可能	HighBond のリザルトに適用されます。(ACLGRC) フィールド名自体に影響を与えることなく表示名がリザルトの列見出しに表示されるように、フィールド名をフィールド名として、表示名を表示名としてエクスポートします。 DISPLAYNAME を AS と組み合わせて使用することもできます。詳細については、"HighBond のリザルトにエクスポートを行う際の DISPLAYNAME と AS の相互作用" ページ 1797を参照してください。

例

"Excel の例" 下

"区切り文字付きファイルの例" 見開きページ

"カンマ区切り値 (CSV) ファイルの例" ページ 1792

"リザルト アプリの例" ページ 1792

"ロボット アプリの例" ページ 1793

Excel の例

Excel .xlsx ファイルにデータをエクスポートします。

Vendor テーブル内の特定のフィールドを Excel ***.xlsx** ファイルにエクスポートするには、次のようにします。

```
OPEN Vendor
EXPORT FIELDS Vendor_No Vendor_Name Vendor_City XLSX TO "VendorExport"
```

ワークシート名を指定して、データを Excel .xlsx ファイルへエクスポートする

Vendor テーブル内の特定のフィールドを Excel *.xlsx ファイル内の Vendors_US というワークシートにエクスポートするには、次のようにします。

```
OPEN Vendor
EXPORT FIELDS Vendor_No Vendor_Name Vendor_City XLSX TO "VendorExport"
WORKSHEET Vendors_US
```

区切り文字付きファイルの例

すべてのフィールドを区切りファイルにエクスポートする

Vendor テーブル内のフィールドを以下の2つの区切り文字付きテキストファイルにエクスポートします。

```
OPEN Vendor
EXPORT FIELDS ALL DELIMITED TO "VendorExport"
```

フィールドのサブセットを区切りファイルにエクスポートする

テーブルのフィールドのサブセットを外部ファイルにエクスポートする際には、2つのオプションがあります。

- エクスポートする個別のフィールドを指定する
- FIELDS ALL を指定し、エクスポートから除外するフィールドを指定する

ヒント

最も作業が少ない方法を使用してください。

次の例は、8つのフィールドがある Vendor テーブルを参照します。

- Vendor_No(ベンダー番号)
- 業者名
- 業者番地
- 業者市区郡
- 業者都道府県
- 業者郵便番号
- 前回有効日
- レビュー日

エクスポートするフィールドを指定する

Vendor テーブル内のフィールドを以下のすべての区切り文字付きテキスト ファイルにエクスポートします。

```
OPEN Vendor
EXPORT FIELDS Vendor_No Vendor_Name DELIMITED TO "Vendors" KEEPTITLE
SEPARATOR "|" QUALIFIER ''
```

FIELDS ALL を指定し、除外するフィールドを指定する

Vendor テーブル内の前回有効日およびレビュー日フィールドを除くすべてのフィールドを、区切り文字付きファイルにエクスポートします。

```
OPEN Vendor
EXPORT FIELDS ALL EXCLUDE Vendor_Last_Active Vendor_Review_Date
DELIMITED TO "Vendor_addresses" KEEPTITLE SEPARATOR "|" QUALIFIER ''
```

GROUP を使用して、データを複数の区切り文字付きテキスト ファイルへエクスポートする

Vendor テーブル内の特定のフィールドを以下の2つの区切り文字付きテキスト ファイルにエクスポートします。

- "A" から "M" の業者名の1つのファイル
- "N" から "Z" の業者名の1つのファイル

GROUP コマンドとIF 条件を使って、各レコードの業者名をテストします。

```
GROUP
EXPORT FIELDS Vendor_No Vendor_Name DELIMITED TO "AtoM" IF BETWEEN
```

```
(UPPER(VENDOR_NAME), "A", "M")
  EXPORT FIELDS Vendor_No Vendor_Name DELIMITED TO "NtoZ" IF BETWEEN
(UPPER(VENDOR_NAME), "N", "Z")
END
```

カンマ区切り値 (CSV) ファイルの例

すべてのフィールドをカンマ区切り値ファイルにエクスポート

Vendor テーブル内のフィールドをカンマ区切り値ファイルにエクスポートします。

```
OPEN Vendor
EXPORT FIELDS ALL DELIMITED TO "VendorExport"
```

リザルト アプリの例

HighBond のリザルト アプリへのデータのエクスポート

エクスポートするフィールドを指定する

AR_Exceptions テーブル内の特定のフィールドを HighBond のリザルトにエクスポートするとします。対象統制テスト(テーブル)内の既存のデータを上書きします。

```
OPEN AR_Exceptions
EXPORT FIELDS No Due Date Ref Amount Type ACLGRC TO "10926@us"
OVERWRITE
```

ロボットでコマンドを実行するために必要なパスワード定義を作成します

EXPORT コマンドで PASSWORD Analytics タグとそれに対応する PASSWORD パラメーターの 2 つの部分からなるパスワード定義を作成します。

//PASSWORD 1 Analytics タグは、ロボット のタスクのパスワード入 力パラメーターを作成します。EXPORT コマンドの一致する PASSWORD 1 パラメーターは、タスクに保存され暗号化されたパスワード値を参照します。必須のパスワードは有効な HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" ページ 1798を参照してください。

```
COMMENT
//Analytics データのリザルト へのエクスポート
//PASSWORD 1 HighBond アクセストークン
//RESULT LOG
END

OPEN AR_Exceptions
EXPORT FIELDS No Due Date Ref Amount Type ACLGRC PASSWORD 1 TO
"10926@us" OVERWRITE
```

ロボット アプリの例

HighBond のロボット アプリへのデータのエクスポート

メモ

ACLロボットにはエクスポートできません。HighBond ロボット またはワークフローロボットにエクスポートする必要があります。

ロボットにエクスポートした後のデータへのアクセス方法については、[load_working_file\(\) method](#) を参照してください

エクスポートするフィールドを指定する

HighBond ロボットで **Trans_May** テーブルから特定のフィールドを、圧縮した CSV ファイル (*.csv.gz) にエクスポートします。開発モードで、ロボット 52053 に同じ名前のファイルが存在している場合、上書きされます。

```
OPEN Trans_May
EXPORT FIELDS CARDNUM AS 'CARDNUM' CODES AS 'CODES' DATE AS 'DATE'
CUSTNO AS 'CUSTNO' DESCRIPTION AS 'DESCRIPTION' AMOUNT AS 'AMOUNT'
HBDATA TO "https://apis-
us.highbond.com/v1/orgs/1000236/robots/52053/working_
files?env=development" OVERWRITE
```

ロボットでコマンドを実行するために必要なパスワード定義を作成します

EXPORT コマンドで PASSWORD Analytics タグとそれに対応する PASSWORD パラメーターの 2 つの部分からなるパスワード定義を作成します。

//PASSWORD 1 Analytics タグは、ロボット のタスクのパスワード入力パラメーターを作成します。EXPORT コマンドの一致する PASSWORD 1 パラメーターは、タスクに保存され暗号化されたパスワード値を参照します。必須のパスワードは有効な HighBond アクセストークンです。詳細については、「パスワード定義の作成とパスワード値の指定」 ページ 1798を参照してください。

```
COMMENT
//ロボットへの分析エクスポート Analytics データ
//PASSWORD 1 HighBond アクセストークン
//RESULT LOG
END

OPEN Trans_May
EXPORT FIELDS CARDNUM AS 'CARDNUM' CODES AS 'CODES' DATE AS 'DATE'
CUSTNO AS 'CUSTNO' DESCRIPTION AS 'DESCRIPTION' AMOUNT AS 'AMOUNT'
HBDATA PASSWORD 1 TO "https://apis-
us.highbond.com/v1/orgs/1000236/robots/52053/working_
files?env=development" OVERWRITE
```

すべてのフィールドのエクスポート

HighBond ロボットで Trans_May テーブルすべてのフィールドを圧縮した CSV ファイル (*.csv.gz) にエクスポートします。

```
OPEN Trans_May
EXPORT FIELDS ALL HBDATA PASSWORD 3 TO "https://apis-
us.highbond.com/v1/orgs/1000236/robots/52053/working_
files?env=development" OVERWRITE
```

FIELDS ALL を指定し、除外するフィールドを指定する

HighBond ロボットでベンダーテーブルから最終有効日およびレビュー日フィールドを除くすべてのフィールドを圧縮された CSV ファイル (*.csv.gz) にエクスポートします。

```
OPEN Vendor
EXPORT FIELDS ALL EXCLUDE Vendor_Last_Active Vendor_Review_Date HBDATA
PASSWORD 3 TO "https://apis-
```

```
us.highbond.com/v1/orgs/1000236/robots/52053/working_
files?env=development" OVERWRITE
```

備考

EXPORT と GROUP コマンドの使用

ほとんどのエクスポート形式は、GROUP コマンドを使ってデータを複数ファイルへ同時にエクスポートすることができます。

Microsoft Excel および Microsoft Access にデータをエクスポートする場合は、同時に1つのファイルのみ作成することができます。

Excel へのエクスポート

データを Excel ファイルにエクスポートするときには、以下の制限事項があります。

制限	詳細
レコード数	<ul style="list-style-type: none"> Excel 2007 以降 (*.xlsx) - 16,384 フィールドで最大 1,048,576 レコード (Excel がサポートする最大ワークシートサイズ) Excel 97 および 2003 - 最大レコード件数は 65,536 件 <p>この最大件数を超える Analytics テーブルでもエクスポートは行えますが、上限を超えた分のレコードは無視されエクスポートされません。</p>
フィールドの長さ	<ul style="list-style-type: none"> 特定のフィールド長上限なし フィールド長の合計が、レコード全体の長さ上限である 32 KB (非 Unicode 版 Analytics では 32,765 文字、Unicode 版 Analytics では 16,382 文字) を超えることはできません。 Excel 2.1 の場合は、最大 247 文字
フィールド名の長さ	<ul style="list-style-type: none"> 最大 64 文字 Excel 2.1 の場合は、最大 248 文字

WORKSHEET パラメーターと上書き

Analytics テーブルから Excel ファイルにエクスポートするときに WORKSHEET コマンドを使用する場合としない場合の結果は、以下で説明します。

一致	説明	使用される WORKSHEET パラメーター	使用されない WORKSHEET パラメーター
一致する Excel ファイル名がない	<ul style="list-style-type: none"> TO に指定されたファイル名の値が既存のどの Excel ファイルの名前とも一致しない 	指定された名前のワークシートで新しい Excel ファイルが作成される	エクスポートされた Analytics テーブルの名前を使用するワークシートで、新しい Excel ファイルが作成される
一致する Excel ファイル名がある 一致するワークシート名がある	<ul style="list-style-type: none"> TO に指定されたファイル名の値と既存の Excel ファイルの名前が一致する WORKSHEET に指定されたワークシート名が Excel ファイルのワークシート名と一致しない 	指定された名前のワークシートが既存の Excel ファイルに追加される	エクスポートされた Analytics テーブルの名前を使用するワークシートで、既存の Excel ファイルが新しい Excel ファイルによって上書きされる
Excel ファイル名とワークシート名が一致する	<ul style="list-style-type: none"> TO に指定されたファイル名の値と既存の Excel ファイルの名前が一致する WORKSHEET に指定されたワークシート名が Excel ファイルのワークシート名と一致する 	最初に Analytics から作成された場合は、指定された名前のワークシートが既存のワークシートを上書きします。 既存のワークシートが最初に直接 Excel で作成された場合は、エラーメッセージが表示され、エクスポート処理がキャンセルされます。	エクスポートされた Analytics テーブルの名前を使用するワークシートで、既存の Excel ファイルが新しい Excel ファイルによって上書きされる

HighBond のリザルト へのエクスポート

以下の表では、リザルトで統制テストにエクスポートする方法の詳細について説明しています。

項目	詳細
必要なアクセス許可	<p>リザルトの統制テストへ結果をエクスポートする機能は、特定の HighBond 役割の割り当て、または管理者特権を必要とします。</p> <ul style="list-style-type: none"> リザルト コレクションのプロフェッショナル ユーザーまたはプロフェッショナル部長の役割を担うユーザーは、そのコレクション内のあらゆる統制テストに結果をエクスポートすることができます。 <p>メモ 監査部門長の役割のユーザーのみが統制テストの既存のデータをエクスポートおよび上書きできます。</p> <ul style="list-style-type: none"> HighBond システム管理者およびリザルト管理者は、HighBond 組織、または管理する組織のすべてのコレクションで自動的にプロフェッショナル部門長の役割を取得します。 <p>詳細については、リザルト アプリアクセス許可を参照してください。</p>
エクスポートの制限事項	<p>統制テストにエクスポートする際、次の制限が適用されます。</p> <ul style="list-style-type: none"> エクスポートごとに最大 100,000 レコード

項目	詳細
	<ul style="list-style-type: none"> ○ 統制テストごとに最大 100,000レコード ○ レコードごとに最大 500 フィールド ○ フィールドごとに最大 256 文字 <p>同じ統制テストへは、複数回 ~ 全体の上 限回数までエクスポートを行うことができます。</p>
フィールドの追加 (OVERWRITE 指定なし)	<p>既存のフィールドと一致する物理フィールド名がある場合は、Analytics テーブル内の順序に関係なく、エクスポートされたフィールドは統制テストの既存のフィールドの最後に追加されます。</p> <p>Analytics では、物理フィールド名はテーブルレイアウトの名前です。既存のフィールド名に一致しないエクスポートされるフィールドは、リザルトのテーブルに追加列として追加されます。</p> <p>Analytics とリザルトのフィールドの表示名は、考慮されません。ただし、オプションの <code>AS エクスポート名</code> パラメーターを使用する場合に、<code>DISPLAYNAME</code> を使用しないときは、<code>エクスポート名</code> 値が物理フィールド名として使用されます。</p> <p>アンケート フィールドの末尾にデータを追加するときには、リザルトの列の表示名は、アンケート構成で指定された名前のままです。</p> <p>対象統制テストの主キーフィールドが指定されている場合、最後に追加は異なる方法で動作します。詳細については、"HighBond のリザルト アプリへの例外のエクスポート" ページ 216を参照してください。</p> <p>メモ</p> <p>リザルトと Analytics の間でデータを往復し、データがリザルトで不一致になる場合は、不一致のフィールド名がある可能性があります。</p> <p>詳細については、"リザルト データをインポート およびエクスポート するときのフィールド名の考慮事項" ページ 1906を参照してください。</p>
パスワード要件	"パスワード定義の作成とパスワード値の指定" 見開きページを参照してください。

HighBond のリザルトにエクスポートを行う際の DISPLAYNAME と AS の相互作用

下記のマトリクスは、Analytics からリザルトにフィールド名をエクスポート する際の DISPLAYNAME パラメーターと AS の相互作用を示しています。

	AS を指定しない場合	AS を指定した場合
DISPLAYNAME を指定しない場合	Analytics のフィールド名が、リザルトのフィールド名と表示名になります。	リザルトのフィールド名と表示名が、AS パラメーターの表示名になります。
DISPLAYNAME を指定した場合	Analytics のフィールド名が、リザルトのフィールド名になります。Analytics の表示名が、リザルトの表示名になります。	Analytics のフィールド名が、リザルトのフィールド名になります。リザルトの表示名が、AS パラメーターの表示名になります。

HighBond ロボット へのエクスポートに必要なアクセス許可

圧縮 CSV ファイル(*.csv.gz) をロボット アプリにアップロードする機能には、特定のロボット ロールの割り当て、または管理者権限が必要です。

- ロボットの編集者ロールまたは所有者ロールを持つユーザーは、圧縮された CSV ファイルをロボットにアップロードできます。レビュー担当者ロールを持つユーザーは、アップロードできません。
- ロボット管理者は自動的にすべてのロボットのコラボレーターになり、所有者のロールを持ちます。
- Professional サブスクリプションを持つ HighBond システム管理者は、自動的にロボット管理者になります。

詳細については、[ロボット アプリのアクセス許可](#) を参照してください。

パスワード定義の作成とパスワード値の指定

ロボットでリザルト アプリにデータをエクスポートするスクリプトを実行したり、ロボット アプリに圧縮された CSV ファイル(*.csv.gz) をアップロードする場合、EXPORT コマンドにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 次のページを参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクス タグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクス タグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクス タグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようとする、パスワードプロンプトが表示されます。 詳細については、"PASSWORD コマンド" ページ 2015 を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール>HighBond アクセストークン**]を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン**]を選択します。

API アクセストークンを管理します]ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理**]ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する - トークン**]列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する - トークンを作成 > Analytics**]をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー**]をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。

- **PASSWORDアナリティクスタグ-ACLロボット**の**タスクデザイナー**で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。

- **PASSWORD コマンド** -Analytics で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド** -Analytics で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。
- 新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
- 詳細については、[HighBond アクセストークンの作成と管理](#)を参照してください。

EXTRACT コマンド

Analytics テーブルからデータを抽出し、それを新しい Analytics テーブルに出力するか、または既存の Analytics テーブルへ追加します。レコード全体または選択したフィールドを抽出することができます。

構文

```
EXTRACT {RECORD|FIELDS フィールド名 <AS 表示名> <...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <EOF> <APPEND> <OPEN>
```

パラメーター

名前	説明
RECORD FIELDS フィールド名 FIELDS ALL	<p>出力に含めるフィールド:</p> <ul style="list-style-type: none"> ◦ RECORD - ソース データ ファイル内のレコード全体、つまり、テーブル内のすべてのフィールド、およびレコード内の未定義部分すべてが使用されます。 フィールドは、テーブルレイアウトに現れる順序と同じ並びで使用されます。 演算フィールドを保持します。 ◦ FIELDS フィールド名 - 指定されたフィールドを使用します フィールドは一覧の順序で使用されます。 演算フィールドを出力先テーブル内の適切なデータ型 (ASCII 型または Unicode 型 (Analytics のエディションによる)、ACL 型 (ネイティブの数値データ型)、日付時刻型、あるいは論理型) の物理フィールドに変換します。演算された実際の値を物理フィールドに設定します。 ◦ FIELDS ALL - テーブルのすべてのフィールドを使用します。 フィールドは、テーブルレイアウトに現れる順序と同じ並びで使用されます。 演算フィールドを出力先テーブル内の適切なデータ型 (ASCII 型または Unicode 型 (Analytics のエディションによる)、ACL 型 (ネイティブの数値データ型)、日付時刻型、あるいは論理型) の物理フィールドに変換します。演算された実際の値を物理フィールドに設定します。
AS 表示名 省略可能	<p>FIELDS フィールド名を使って抽出を行う場合にのみ有効です。</p> <p>新しい Analytics テーブルのビューにおけるフィールドの表示名 (代替列見出し)。表示名をフィールド名、またはソース テーブル内の既存の表示名と同じにしたい場合は、AS を使用しないでください。</p>

名前	説明
	<p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>メモ AS は新しいテーブルに抽出を行う場合にのみ使用します。既存のテーブルに追加している場合は、既存テーブルの代替列見出しが優先されます。</p>
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用して抽出するときのみ使用されます。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ Analytics テーブルである出カファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テ</p>

名前	説明
省略可能	<p>ブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
EOF 省略可能	<p>ファイルの終わりに達した後、コマンドをもう一度実行します。</p> <p>これにより、GROUP コマンド内でテーブルの最後のレコードが処理されることが保証されます。すべてのフィールドが以前のレコードを参照する演算フィールドである場合にのみ使用してください。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>

例

テーブル内のすべてのレコードを新しいテーブルへ抽出する

次の例では、既存のすべてのレコードを新しい Analytics テーブルへ抽出することにより、**AR_Customer** テーブルの正確な複製を作成します。すべての演算フィールドは演算フィールドとして保持されます。

```
OPEN AR_Customer  
EXTRACT RECORD TO "AR_Customer_2"
```

テーブル内のすべてのフィールドを新しいテーブルへ抽出する

AR_Customer テーブル内の定義済みのすべてのフィールドを新しい Analytics テーブルへ抽出します。すべての演算フィールドが、物理フィールドに変換され、実際に計算された値が格納されます。

```
OPEN AR_Customer  
EXTRACT FIELDS ALL TO "AR_Customer_2"
```

テーブルのすべてのレコードの抽出と、既存のテーブルへの追加

AR_Customer テーブルのすべてのレコードを抽出し、**AR_Customer_Master** テーブルの最後にグループとして追加します。

```
OPEN AR_Customer  
EXTRACT RECORD TO "AR_Customer_Master" APPEND
```

テーブルのすべてのレコードの抽出と、別のフォルダの既存のテーブルへの追加

AR_Customer テーブルのすべてのレコードを抽出し、**AR_Customer_Master** テーブルの最後にグループとして追加します。これは、Analytics プロジェクト フォルダ以外のフォルダです。

```
OPEN AR_Customer
EXTRACT RECORD TO "C:\Users\Customer Data\AR_Customer_Master" APPEND
```

テーブルから新しいテーブルへフィールドのサブセットを抽出する

テーブルのフィールドのサブセットを抽出する際には、2つのオプションがあります。

- 抽出する個別のフィールドを指定する
- FIELDS ALL を指定し、抽出から除外するフィールドを指定する

ヒント

最も作業が少ない方法を使用してください。

次の例は、7つのフィールドがある **AR_Customer** テーブルを参照します。

- 参照番号
- customer number
- customer name
- 取引形態
- Invoice_Date(請求書番号)
- 期日
- Invoice_Amount(請求金額)

抽出するフィールドを指定する

次の例では、**AR_Customer** テーブルから3つのフィールドを新しい Analytics テーブルへ抽出します。

```
OPEN AR_Customer
EXTRACT FIELDS Name Due Date TO "AR_Customer_Dates.fil"
```

FIELDS ALL を指定し、除外するフィールドを指定する

次の例では、**AR_Customer** テーブルから、Reference_num フィールドを除くすべてのフィールドを新しい Analytics テーブルへ抽出します。

```
OPEN AR_Customer
EXTRACT FIELDS ALL EXCLUDE Reference_num TO "AR_Customer_Dates.fil"
```

抽出されたフィールドの表示名を作成する

次の例では、**AR_Customer** テーブルから3つのフィールドを抽出し、新しいAnalytics テーブルにおけるそれらのフィールドの表示名を作成します。

```
OPEN AR_Customer
EXTRACT FIELDS Name AS "Customer;Name" Due AS "Due;Date" Date AS
"Invoice;Date" TO "AR_Customer_Dates.fil"
```

条件に基づいてフィールドを抽出する

次の例では、**Due** フィールドの日付が2014年7月1日より前の日付の場合に、**AR_Customer** テーブルから3つのフィールドを新しいAnalytics テーブルへ抽出します。

```
OPEN AR_Customer
EXTRACT FIELDS Name Due Date IF Due < `20140701` TO "Overdue.fil"
```

備考

このコマンドの動作の詳細については、「データの抽出」ページ 201または「データの抽出と追加」ページ 940を参照してください。

テーブルの抽出 (EXTRACT) とテーブルのコピーの比較

EXTRACT では、新しいテーブルレイアウトだけでなく、新しいソース データ ファイル (.fil) も作成されます。

これに対し、**ナビゲーター**(編集 > コピー) を使ってテーブルを作成すると、元のソース データ ファイルとも関連付けられた新しいテーブルレイアウトが作成されます。この場合、新しいデータ ファイルが作成されるわけではありません。

FIELDSHIFT コマンド

テーブルレイアウトでフィールド定義の開始位置を変更します。

構文

```
FIELDSHIFT START 開始位置 COLUMNS シフトバイト数 <FILTER データ フィルター名> <OK>
```

パラメーター

名前	説明						
START 開始位置	<p>シフトする最初のフィールド定義の開始バイト位置。 すべてのフィールド定義が、指定されたフィールド定義の右にシフトされます。 非開始バイト位置を指定する場合は、次の開始バイト位置が使用されます。</p> <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データでは、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						
COLUMNS シフトバイト数	<p>フィールド定義をシフトするバイト数。 フィールド定義を右にシフトする正の数値を入力します。フィールド定義を左にシフトする負の数値を入力します。</p> <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データでは、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						

名前	説明
FILTER データフィルター名 省略可能	特定のレコード定義に関連するフィールド定義を識別するためのフィルターの名前。
OK 省略可能	アクションを確認せずに、項目を削除または上書きします。

例

フィールド定義をシフトする

バイト 11 でフィールド定義と後続のフィールド定義 4 バイトを右にシフトします。

```
FIELDSHIFT START 11 COLUMNS 4
```

備考

このコマンドの動作の詳細については、"テーブルレイアウトにおけるフィールドのシフト" ページ 824を参照してください。

シフトされたフィールド定義は、レコード長内にある必要があります。

1つ以上のフィールド定義を左右にシフトするときには、フィールドがいずれの方向でもレコード長を超えることができません。

FIELDSHIFT は、指定されたフィールド定義と、任意のフィールド定義の両方を、指定された定義の右に移動します。定義のシフトされたブロックがいずれかの方向でレコード長を超える場合、エラーメッセージが表示され、コマンドは実行されません。

ヒント

レコードの最後を超過しているためにエラーメッセージが表示される場合は、最終フィールド定義を削除し、シフト対象のフィールド定義の場所を作ってください。

FIND コマンド

インデックス付き文字フィールドで、指定した文字列と一致する最初の値を検索します。

メモ

FIND コマンドとFIND() 関数は Analytics の 2 つの別個の機能であり、大きな違いがあります。関数の詳細については、"FIND() 関数" ページ 2267を参照してください。

構文

```
FIND 検索値
```

パラメーター

名前	説明
検索値	検索する文字列。 検索値は大文字と小文字を区別し、先頭のスペースを含めることができません。 引用符が検索されるデータの一部でない限り、値を引用符で囲まないでください。

例

特定の値の検索

文字型フィールド **Card_Number** の値のうち、"8590124" と完全に一致するか "8590124" で始まる最初の値を検索したいとします。

まず、**Card_Number** フィールドにインデックスを昇順に作成します。次に、FIND を実行します。これらを実行する例は次のようになります。

```
INDEX ON Card_Number TO "CardNum" OPEN
SET INDEX TO "CardNum"
FIND 8590124
```

備考

このコマンドの動作の詳細については、「一致する最初のレコードの選択」ページ 1229を参照してください。

FIND の用途

FIND コマンドを使用すると、テーブル内で、インデックス付きの文字フィールドに指定した検索値を含んでいる最初のレコードに直接移動できます。

INDEX を使用するための要件

このコマンドを使用するには、検索対象のテーブルに、昇順の文字フィールドに基づくインデックスが作成されている必要があります。

複数の昇順の文字フィールドを基にインデックスが作成されている場合は、そのインデックスに指定されている最初のフィールドだけが検索されます。このコマンドで、文字以外のインデックスフィールドや、降順でインデックス付けされた文字フィールドを検索することはできません。

部分一致

部分一致がサポートされます。インデックス付きフィールドに含まれる長い値の一部を検索値に指定できるのです。ただし、検索値は、一致を成すフィールドの先頭に現れる必要があります。

一致に応じた FIND の出力

FIND コマンドの結果は、検索値が見つかったかどうかに応じて、以下のいずれかになります。

- **検索値が見つかった場合** - テーブル内で最初の一致するレコードが選択されます。
- **検索値が見つからない場合** - テーブル内で、検索値よりも大きい値を持つ最初のレコードが、選択されます。

検索値よりも大きい値がインデックス付きフィールドに存在しない場合は、テーブルは最初のレコードに位置付けられます。どちらの場合も、「キーと一致するインデックスがありません」というメッセージが表示されます。

FIND コマンドは、**[正確な文字比較を行う] オプション**(SET EXACT ON/OFF) の影響を受けません。

FUZZYDUP コマンド

文字フィールドでほぼ同一の値(あいまい重複)を検出します。

メモ

曖昧一致を使用して、2つの Analytics テーブルのフィールドを組み合わせることで1つの新しい Analytics テーブルにする手順については、「FUZZYJOIN コマンド」ページ 1817を参照してください。

構文

```
FUZZYDUP ON キー フィールド <OTHER フィールド <...n>|OTHER ALL <EXCLUDE フィールド名 <...n>>> LEVDISTANCE 値 <DIFFPCT 割合> <RESULTSIZ 割合> <EXACT> <IF テスト>
TO テーブル名 <LOCAL> <OPEN>
```

パラメーター

名前	説明
ON キーフィールド	あいまい重複をテストする文字フィールドまたは式。
OTHER フィールド <...n> OTHER ALL 省略可能	出力に含める1つ以上の追加フィールド。 <ul style="list-style-type: none"> OTHER フィールド<...n>- 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。 OTHER ALL - キーフィールドとして指定しなかった、テーブル内のすべてのフィールドが含まれます。 フィールドは、テーブルレイアウトに現れる順序と同じ並びで含まれます。
EXCLUDE フィールド名 省略可能	OTHER ALL を使用するときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、OTHER ALL を微調整できます。 EXCLUDE は OTHER ALL の直後に置く必要があります。例： <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>OTHER ALL EXCLUDE field_1 field_2</pre> </div>
LEVDISTANCE 値	2つの文字列があいまい重複と認定されて結果に含まれるために、それらの文字列間で許容される最大のレーベンシュタイン距離。

名前	説明
	<p>LEVDISTANCE の値は 1 未満にすることや、10 を超えることはできません。LEVDISTANCE の値を大きくすると、あいまい度が高い値(相互の関連性が低い値)が含まれるため、結果の件数が多くなります。</p> <p>詳細については、「FUZZYDUP 動作」 ページ 1814を参照してください。</p>
<p>DIFFPCT 割合 省略可能</p>	<p>「相違のパーセント」、つまり、文字列のうち異なってもよい割合を制限するしきい値。</p> <p>あいまい重複の可能性のあるペアが結果に含まれるためには、そのペアに対して実行される Analytics 内部での計算によって出力されるパーセントは、DIFFPCT の値以下である必要があります。DIFFPCT の値は 1 未満にすることや、99 を超えることはできません。</p> <p>DIFFPCT が省略された場合、しきい値はオフになり、FUZZYDUP コマンドの処理時に相違のパーセントは考慮されません。</p> <p>詳細については、「FUZZYDUP 動作」 ページ 1814を参照してください。</p>
<p>RESULTSIZE 割合 省略可能</p>	<p>キーフィールドのレコード数の割合として、出力結果のセットの最大サイズ。</p> <p>たとえば、50,000 個の値を持つキーフィールドの場合、RESULTSIZE に 3 を設定すると、結果のあいまい重複数が 1500 を超えた場合 (50,000 x 0.03) に処理が終了します。処理が終了した場合、出力テーブルは作成されません。</p> <p>RESULTSIZE の値は 1 未満にすることや、1000 パーセントを超えることはできません。1000% の上限は多対多一致の本質に対応するためのものです。多対多一致では、元の検査データセットより大きい結果が生成されるかもしれません。</p> <p>RESULTSIZE が省略された場合、しきい値はオフになり、FUZZYDUP コマンドの処理時に結果のサイズは考慮されません。</p> <div style="border-left: 2px solid red; padding-left: 10px; margin-top: 10px;"> <p>注意</p> <p>RESULTSIZE を省略すると、過度に大きな結果のセットが生成され、処理時間が非常に長くなったり、使用可能なメモリの超過を引き起こして、処理が終了したりする可能性があります。結果が管理可能なサイズになると確信している場合にのみ、RESULTSIZE を省略してください。</p> </div>
<p>EXACT 省略可能</p>	<p>あいまい重複だけでなく完全な重複も結果に含まれます。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <div style="border-left: 2px solid blue; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p> </div>
<p>TO テーブル名</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを</p>

名前	説明
	<p>指定します。</p> <ul style="list-style-type: none"> TO "C:\Output.FIL" TO "Results\Output.FIL" <p>メモ テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ Analytics テーブルである出カファイルを含むサーバー テーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターはTO パラメーターの直後に置く必要があります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。</p>

Analytics の出力変数

名前	含む
GAPDUP n	コマンドによって確認されたギャップ、重複、またはあいまい重複グループの合計数。

例

あいまい重複の姓フィールドのテスト

姓フィールドであいまい重複を検査(**Last_Name** フィールドは、**ACL DATA\Sample Data Files\Metaphor_Employee_Data.ACL** の **Employee_List** テーブルにあります) します。結果は新しい Analytics テーブルに出カされます。

- 検査フィールドに加え、他のフィールドも結果に含まれます。
- 許容される最大のレーベンシュタイン距離は1です。
- 異なってもよい文字列の割合は50%に制限されています。
- 結果のサイズは、検査フィールドのサイズの20%に制限されています。
- あいまい重複に加え、完全な重複も含まれます。

```
OPEN Employee_List
FUZZYDUP ON Last_Name OTHER First_Name EmpNo LEVDISTANCE 1 DIFFPCT 50
RESULTSIZE 20 EXACT TO "Fuzzy_Last_Name" OPEN
```

備考

機能の仕組み

FUZZYDUP コマンドは、ほぼ同一の値(あいまい重複)の検出や、手作業で入力されたデータで一貫性のないつづりを見つけます。

ISFUZZYDUP() 関数は、単一の文字値に対するあいまい重複の完全な一覧を識別します。それとは異なり、FUZZYDUP コマンドでは、フィールド内のすべてのあいまい重複を識別し、それらを完全でないグループにまとめて、結果を出力します。

このコマンドの動作の詳細については、「あいまい重複分析」ページ 1276を参照してください。

「網羅的でない」とは

「完全でない」とは、結果のあいまい重複の個々のグループには、そのグループ所有者の指定された相違の度合いの範囲内にある、検査フィールドのすべてのあいまい重複が含まれない可能性があることを意味します。ただし、グループ所有者が検査フィールド内の別の値のあいまい重複である場合は、2つの値が結果のどこかのグループと一緒に現れます。このため、グループは完全ではない可能性があります、全体の結果は完全です。

分析において、検査フィールド内の特定の値に対するあいまい重複の、単一かつ完全な一覧を生成することが重要な場合は、この目的のためにISFUZZYDUP() 関数を使用することができます。

FUZZYDUP 動作

FUZZYDUP コマンドには、2つのパラメーターがあります。これらを指定して、あいまい重複間の相違の度合いや結果のサイズを制御することができます。

- LEVDISTANCE
- DIFFPCT

これらの2つのパラメーターの設定の組み合わせをいろいろ試してみて、特定のデータセットでどれが最も良い状態で機能するかを調べる必要があるかもしれません。

LEVDISTANCE (レーベンシュタイン距離)

データの処理中に、FUZZYDUP コマンドは、検査フィールドで評価される文字列のペアごとにレーベンシュタイン距離を計算し、相違のパーセントを求めます。レーベンシュタイン距離は、ある文字列を別の

文字列にするために必要な、1文字の編集の最小回数を示す値です。詳細については、"LEVDIST() 関数" ページ 2335を参照してください。

DIFFPCT (相違のパーセント)

相違のパーセントは、評価される2つの文字列の長さが異なるとき、それらのうち短い方に対する割合であり、2つの文字列間のレーベンシュタイン距離を使用する、次のような Analytics 内部での計算の結果です。

$\text{レーベンシュタイン距離} / \text{短い文字列内の文字数} \times 100 = \text{相違のパーセント}$

詳しい情報

あいまい重複のさまざまな設定、結果サイズの制御、およびあいまい重複のグループに関する詳細については、"あいまい重複分析" ページ 1276を参照してください。

大文字と小文字の区別

FUZZYDUPコマンドでは大文字と小文字が区別されないため、"SMITH" は "smith" と同じであると判断されます。

最後の空白は自動的に削除されます

FUZZYDUP コマンドは、キーフィールド内の末尾にあるスペースを自動的に除去するため、キーフィールドの単一フィールドを指定するときに TRIM() または ALLTRIM() 関数を使う必要はありません。

キーフィールドのフィールドを連結する場合は、以下のように、ALLTRIM() を使用してください。

FUZZYDUP の効果の改善

次の3つの手法はFUZZYDUP コマンドの効果を大幅に高めることができます。

- テストフィールド値の個別の要素を並べ替える
- テストフィールド値から汎用要素を削除する
- テストフィールドを連結する

これらの手法では、誤検出が少なく、真の検出が多い、より絞り込まれた結果セットを生成します。これらの手法は別々に使用するか組み合わせて使用できます。

テストフィールド値の個別の要素を並べ替える

SORTWORDS() 関数は、テストフィールド値の個別の要素を連続する順序に並べ替えることで、FUZZYDUP コマンドの効果を高めることができます。

アドレスの構成要素などの要素を並べ替えると、同じ情報を含む、形式が異なり、相互により密接に類似した2つの文字列を作成できます。より密接な類似性により、文字列の組み合わせが相互のあいまい重複として選択される確率が上がります。

詳細については、"SORTWORDS() 関数" ページ 2498を参照してください。

SORTWORDS() の概要を説明する動画については、「[Fuzzy Matching Using SORTWORDS\(\)](#)」(英語のみ)をご覧ください。

テスト フィールド 値 から汎用要素を削除する

OMIT() 関数は、テスト フィールド 値 から "Corporation" や "Inc."、カンマ、ピリオド、アンパサンド (&) 文字などの一般的要素を除去することによって、FUZZYDUP コマンドの効果を高めることができます。

一般的要素および句読点の除去により、FUZZYDUP の文字列比較は、意味のある違いが発生する可能性のある文字列の部分だけに集中されます。

詳細については、"OMIT() 関数" ページ 2390を参照してください。

テスト フィールド を連結する

2 つ以上の検査フィールドを連結すると、検査値の一意性の度合いが増すことによって、FUZZYDUP コマンドの有効性を高めることができます。

たとえば、住所フィールドと市区郡フィールドを連結することで、別の市の住所との曖昧一致を回避できます。

```
FUZZYDUP ON ALLTRIM(Address)+ALLTRIM(City) OTHER Address City Vendor_Name  
LEVDISTANCE 4 DIFFPCT 50 RESULTSIZE 20 EXACT TO "Vendor_Name_Fuzzy_Dupes"  
OPEN
```

その他の文字列比較方法

- **DICECOEFFICIENT() 関数** -は、文字または文字ブロックの相対位置を重視しない、または完全に無視して文字列を比較するための方法を提供します。
- **SOUNDSLIKE() 関数**および **SOUNDEX() 関数** -は、正字法の比較(綴り)ではなく、発音記号の比較(発音)に基づいて文字列を比較するための方法を提供します。

FUZZYJOIN コマンド

曖昧一致を使用して、2つの Analytics テーブルのフィールドを組み合わせて1つの新しい Analytics テーブルにします。

メモ

単一の文字フィールドでほぼ同一の値(あいまい重複)を検出するには、"FUZZYDUP コマンド" ページ 1811を参照してください。

完全に一致するキーフィールド値を使用してテーブルを結合するには、"JOIN コマンド" ページ 1971を参照してください。

構文

```
FUZZYJOIN {DICE PERCENT 割合 NGRAM n-gram 長 | LEVDISTANCE DISTANCE 値} PKEY
主キー フィールド SKEY 副キー フィールド {FIELDS 主キー フィールド | FIELDS ALL <EXCLUDE
主フィールド <...n>>} <WITH 副フィールド | WITH ALL <EXCLUDE 副フィールド <...n>>> <IF
テスト> <OPEN> TO テーブル名 <FIRSTMATCH> <WHILE テスト> <FIRST 範囲 | NEXT 範囲>
<APPEND>
```

メモ

サーバーのテーブルに対してローカルで FUZZYJOIN コマンドを実行することはできません。

FUZZYJOIN コマンドはその全体を指定する必要があります。簡略化することはできません。

パラメーター

名前	説明
DICE PERCENT 範囲 NGRAM <i>n-gram</i> 長 LEVDISTANCE DISTANCE 値	<p>使用するあいまい一致アルゴリズム</p> <p>DICE - ダイス係数アルゴリズムを使用します</p> <ul style="list-style-type: none"> PERCENT 割合 - あいまい一致と見なされるための、2つの文字列の許容可能な最低ダイス係数 <p>0.0000 ~ 1.0000 の小数を指定します(例:0.7500)。小数点4桁の最大値を使用します。</p> <p>値を小さくすると、あいまい度が高い値(相互の関連性が低い文字列)が含まれるため、一致の件数が多くなります。</p>

名前	説明
	<ul style="list-style-type: none"> ◦ NGRAM <i>n-gram</i> 長 - 使用する <i>n-gram</i> の長さ 1 以上の整数を指定します。 <i>n-gram</i> の長さを大きくすると、2 つ文字列の間の類似度の基準が厳しくなります。 <i>n-gram</i> は、ダイス係数計算の構成要素であり、比較対象となる 2 つの文字列にとって、構成要素であると同時に重なり合う、部分文字列 (文字ブロック) です。 <p>メモ DICE を指定するときには、FUZZYJOIN コマンドは、IF 文で DICECOEFFICIENT() 関数を使用し、キーフィールド値を条件付きで結合します。関数の詳細については、"DICECOEFFICIENT() 関数" ページ 2238 を参照してください。</p> <p>LEVDISTANCE - レーベンシュタイン距離アルゴリズムを使用します</p> <ul style="list-style-type: none"> ◦ DISTANCE 値 - あいまい一致であると見なされるための、2 つの文字列間の許容可能な最低レーベンシュタイン距離 1 以上の整数で指定します。 値を大きくすると、あいまい度が高い値 (相互の関連性が低い文字列) が含まれるため、一致の件数が多くなります。 <p>メモ LEVDISTANCE を指定するときには、FUZZYJOIN コマンドは、IF 文で LEVDIST() 関数を使用し、キーフィールド値を条件付きで結合します。関数の詳細については、"LEVDIST() 関数" ページ 2335 を参照してください。 関数とは異なり、FUZZYJOIN コマンドのレーベンシュタイン距離アルゴリズムは、自動的に先頭と末尾の空白を取り除きます。大文字と小文字は区別されません。</p>
PKEY 主キーフィールド	主テーブルの文字キーフィールドまたは式。 主キーフィールドは 1 つだけ指定できます。
SKEY 副キーフィールド	副テーブルの文字キーフィールドまたは式。 副キーフィールドは 1 つだけ指定できます。
FIELDS 主フィールド FIELDS ALL	結合先の出力テーブルに含める、主テーブル内のフィールドまたは式。 <ul style="list-style-type: none"> ◦ FIELDS 主フィールド - 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。 ◦ FIELDS ALL - はテーブルのすべてのフィールドに含まれます フィールドは、テーブルレイアウトに表示される順序で含まれます。 <p>メモ 結合テーブルに含める場合は、主キーフィールドを明示的に指定する必要があります。FIELDS ALL を指定しても含まれます。</p>
EXCLUDE 主フィールド	FIELDS ALL を使用して、曖昧結合を実行するときのみ有効です。

名前	説明
<p>省略可能</p>	<p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>FIELDS ALL EXCLUDE field_1 field_2</p> </div>
<p>WITH 副フィールド WITH ALL 省略可能</p>	<p>結合先の出カテーブルに含める、副テーブル内のフィールドまたは式。</p> <ul style="list-style-type: none"> ○ WITH 副フィールド - 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。 ○ WITH ALL - はテーブルのすべてのフィールドに含まれます フィールドは、テーブルレイアウトに表示される順序で含まれます。 <p>メモ 結合テーブルに含める場合は、副キーフィールドを明示的に指定する必要があります。WITH ALL を指定すると含まれます。</p>
<p>EXCLUDE 副フィールド 省略可能</p>	<p>WITH ALL を使用して、曖昧結合を実行するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、WITH ALL を微調整できます。</p> <p>EXCLUDE は WITH ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>WITH ALL EXCLUDE field_1 field_2</p> </div>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p> <p>メモ IF 条件は、主テーブル、副テーブル、または両方を参照できます。</p>
<p>OPEN 省略可能</p>	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。</p>
<p>TO テーブル名</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを</p>

名前	説明
	<p>指定します。</p> <ul style="list-style-type: none"> TO "C:\Output.FIL" TO "Results\Output.FIL" <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
<p>FIRSTMATCH</p> <p>省略可能</p>	<p>各主キー値が、最初に発生した副キー一致にのみ結合されることを指定します。</p> <p>最初の発生が完全一致であった場合は、後続の主キー値のあいまい一致は、結合された出力テーブルに含まれません。</p> <p>FIRSTMATCHを省略する場合、FUZZYJOINのデフォルト動作では、各主キー値をすべての副キー一致に結合します。</p> <p>FIRSTMATCHは、完全一致またはあいまい一致の一致が2つのテーブル間に存在するかどうかを確認し、すべての一致を特定するために必要な処理時間を減らしたい場合にのみ役立ちます。</p> <p>各主キー値に対する一致が副テーブルで多くても1つだけであることが確実な場合は、FIRSTMATCHを使用することもできます。</p>
<p>WHILE テスト</p> <p>省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件がfalseと評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILEをFIRSTまたはNEXTとともに使用する場合は、1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲</p> <p>省略可能</p>	<p>処理するレコード数：</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>APPEND</p> <p>省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> 同じフィールド 同じフィールド順序 一致するフィールドが同じ長さ 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>

名前	説明
ISOLOCALE ロケールコード 省略可能	<p>メモ Analytics の Unicode 版にのみ適用されます。</p> <p>システム ロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr_ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> 言語 - ISO 639 標準言語コード 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

例

あいまい一致を使用して、業者である可能性がある従業員を検索するための方法として、2つのテーブルを結合する

次の例は、共通キーフィールドとして住所 (Address および Vendor_Street フィールド) を使用し、Empmast および Vendor テーブルを結合します。

FUZZYJOIN コマンドは、完全一致またはあいまい一致の主および副レコードを含む新しいテーブルを作成します。結果は、同一の住所または類似した住所の従業員と業者のリストです。

FUZZYJOIN とダイス係数アルゴリズム

```

OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
FUZZYJOIN DICE PERCENT 0.8000 NGRAM 2 PKEY Address SKEY Vendor_Street
FIELDS Employee_Number First_Name Last_Name Address WITH Vendor_Number
Vendor_Name Vendor_Street OPEN TO "Employee_Vendor_Match"
    
```

FUZZYJOIN とレーベンシュタイン距離アルゴリズム

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
FUZZYJOIN LEVDISTANCE DISTANCE 5 PKEY Address SKEY Vendor_Street
FIELDS Employee_Number First_Name Last_Name Address WITH Vendor_Number
Vendor_Name Vendor_Street OPEN TO "Employee_Vendor_Match"
```

すべてのフィールドを含める

このバージョンの FUZZYJOIN コマンドは、主テーブルと副テーブルのすべてのフィールドを結合された出力テーブルに含めます。

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
FUZZYJOIN LEVDISTANCE DISTANCE 5 PKEY Address SKEY Vendor_Street
FIELDS ALL WITH ALL OPEN TO "Employee_Vendor_Match"
```

曖昧一致の効果の改善

次の例では、SORTWORDS() 関数を使用して、Address フィールドと Vendor_Street フィールド間の曖昧一致の効果を高めます。UPPER() 関数を使用すると、キーフィールド値の要素の並べ替えに大文字と小文字が影響しないことを保証します。

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
FUZZYJOIN LEVDISTANCE DISTANCE 5 PKEY SORTWORDS(UPPER(Address)) SKEY
SORTWORDS(UPPER(Vendor_Street)) FIELDS Employee_Number First_Name
Last_Name Address WITH Vendor_Number Vendor_Name Vendor_Street OPEN TO
"Employee_Vendor_Match"
```

備考

このコマンドの動作の詳細については、「曖昧結合」ページ 985を参照してください。

大文字と小文字の区別

FUZZYJOIN コマンドは、使用するあいまい一致アルゴリズムに関係なく、大文字と小文字を区別しません。このため、"SMITH" は "smith" と同じです。

先頭と末尾の空白

FUZZYJOIN コマンドは、使用するあいまい一致アルゴリズムに関係なく、フィールドの先頭と末尾の空白を自動的に取り除きます。主および副キーフィールドを指定するときに、TRIM() または ALLTRIM() 関数を使用する必要はありません。

FUZZYJOIN の効果の改善

次の3つの手法はFUZZYJOIN コマンドの効果を大幅に高めることができます。

- 主キーおよび副キーフィールド値の個別の要素を並べ替える
- 主キーおよび副キーフィールド値から汎用要素を削除する
- 主キーおよび副キーフィールド値を調整する

これらの手法では、より厳密な曖昧設定を使用して、同じ曖昧一致を取得しながら、誤検出一致の数を減らすことができます。これらの手法は別々に使用するか組み合わせて使用できます。

キーフィールド値の個別の要素を並べ替える

SORTWORDS() 関数では、主キーフィールド値と副キーフィールド値の個別の要素を連続する順序に並べ替えることで、FUZZYJOIN コマンドの効果を改善することができます。

アドレスの構成要素などの要素を並べ替えると、同じ情報を含む、形式が異なり、相互により密接に類似したキーフィールド値を作成できます。より密接な類似性により、キーフィールド値が相互の曖昧一致として選択される確率が上がります。

詳細については、「SORTWORDS() 関数」ページ 2498を参照してください。

SORTWORDS() の概要を説明する動画については、「[Fuzzy Matching Using SORTWORDS\(\)](#)」(英語のみ)をご覧ください。

メモ

キーフィールド値の要素の並べ替えは、レーベンシュタイン距離アルゴリズムを使用した曖昧一致に最適です。

ダイ係数アルゴリズムを使用しているときには、要素の並べ替えが有効な場合とそうでない場合があります。本番設定でダイ係数アルゴリズムと SORTWORDS() を使用するかどうかを決定する前に、サンプルデータのセットをテストしてください。

注意

FUZZYJOIN コマンドと SORTWORDS() を使用する場合は、SORTWORDS() を比較対象の両方の文字列または両方のフィールドに適用する必要があります。

キーフィールド値から汎用要素を削除する

OMIT() 関数は、主キーフィールド値および副キーフィールド値から "Corporation" や "Inc."、カンマ、ピリオド、アンパサンド(&) 文字などの一般的要素を除去することによって、FUZZYJOIN コマンドの効果を高めることができます。

一般的要素と句読点の除去により、曖昧一致は、意味のある違いが発生する可能性のあるキーフィールド値の部分だけに集中されます。

詳細については、"OMIT() 関数" ページ 2390を参照してください。

キーフィールド値を一致させる

REPLACE() または REGEXREPLACE() 関数は、主キーフィールド値と副キーフィールド値の同じ要素の変異形を一致させることで、FUZZYJOIN コマンドの効果を改善できます。たとえば、"Street"、"St."、"St" を一致させ、単一の値 "St" を使用できます。

要素を一致させると、同じ情報を含む、形式が異なり、相互により密接に類似したキーフィールド値を作成できます。より密接な類似性により、キーフィールド値が相互の曖昧一致として選択される確率が上がります。

詳細については、"REPLACE() 関数" ページ 2469(簡単な置換) および"REGEXREPLACE() 関数" ページ 2455(複雑な置換) を参照してください。

GAPS コマンド

Analytics テーブルの数値または日付時刻フィールドに1つ以上のギャップが連続して含まれるかどうかを検出します。

構文

```
GAPS <ON> キーフィールド <D> <UNFORMATTED> <PRESORT> <MISSING 制限> <HEADER ヘッダー テキスト> <FOOTER フッター テキスト> <IF テスト> <WHILE テスト> <FIRST範囲|NEXT 範囲> <TO {SCREEN|テーブル名|ファイル名|PRINT}> <LOCAL> <APPEND> <OPEN>
```

パラメーター

名前	説明
ON キーフィールドD	ギャップを検査するフィールドや式。 キーフィールドを降順に並べ替えるDを含めます。デフォルトのソート順は昇順です。
UNFORMATTED 省略可能	結果をファイルに出力する場合、ページ見出しや改ページは除去されます。
PRESORT 省略可能	コマンドを実行する前にキーフィールドでテーブルを並べ替えます。 メモ GROUP コマンドの内部ではPRESORTを使用することができません。
MISSING 制限 省略可能	ギャップの幅ではなく、欠落している個々の項目を出力結果に含めません。 制限値は、識別された各ギャップでレポートする欠落項目の最大数を指定します。デフォルト値は5です。制限を超えると、欠落項目はその特定のギャップの範囲としてレポートされます。 制限値は、レポートされる欠落項目の総数を制限するものではなく、特定のギャップの範囲内でレポートされる欠落項目の数を制限するだけです。
HEADER ヘッダーテキスト 省略可能	レポートの各ページの最上部に挿入されるテキスト。 ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数であるHEADER の値よりも優先されます。
FOOTER フッターテキスト	レポートの各ページの最下部に挿入されるテキスト。

名前	説明
省略可能	フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
TO SCREEN テーブル名 ファイル名 PRINT 省略可能	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ○ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は 64 文字の英数字(.FIL 拡張子を含まない) に制限されています。名前にはアンダースコア文字(_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

名前	説明
	<ul style="list-style-type: none"> ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code> 印刷 - 通常使うプリンターに結果を送信します
LOCAL 省略可能	Analytics プロジェクトと同じ場所に出カファイルを保存します。 メモ Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。
APPEND 省略可能	コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。 メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。 <ul style="list-style-type: none"> 同じフィールド 同じフィールド順序 一致するフィールドが同じ長さ 一致するフィールドが同じデータ型 出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。
OPEN 省略可能	コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。

Analytics の出力変数

名前	含む
GAPDUP n	コマンドによって確認されたギャップ、重複、またはあいまい重複グループの合計数。

例

欠落した請求書番号がないかどうかをテストする

次の例では、GAPSを使用して、**Invoices** テーブルから欠落している請求書番号がないことを確認しています。

```
OPEN Invoices
GAPS ON Inv_Num PRESORT TO "Invoices_Gaps.fil"
```

備考

このコマンドの動作の詳細については、「ギャップの検査」ページ 1256を参照してください。

文字フィールドに対して GAPS を使用する

数値または日付時刻フィールドをテストする他に、文字フィールドに存在する数値データに対してもギャップがないかどうかをテストすることができます。たとえば、通常、文字データとして書式設定される小切手番号などに対してテストを行うことができます。

文字フィールドに文字と数字が続けて現れる場合は、数字のみが検査され、文字は無視されます。

GROUP コマンド

テーブルの次のレコードに移動する前に、テーブルを 1 回通過するだけで、レコードに対して 1 つ以上の ACLScript コマンドを実行します。コマンド実行は条件によって制御できます。

構文

```
GROUP <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
  コマンド
  <...n>
<ELSE IF テスト>
  コマンド
  <...n>
<ELSE>
  コマンド
  <...n>
END
```

メモ

一部の Analytics コマンドは、GROUP コマンドとともに使用できません。詳細については、「GROUP コマンドの内部で使用することができるコマンド」ページ 1833を参照してください。

パラメーター

名前	説明
IF テスト 省略可能	各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。 メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。
WHILE テスト 省略可能	各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。 メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。
FIRST 範囲 NEXT 範囲	処理するレコード数:

名前	説明
省略可能	<ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
コマンド <...n>	<p>GROUP内で実行する1つまたは複数のACLScriptコマンド。GROUP内で使用できるコマンドの詳細な一覧については、"GROUPコマンドの内部で使用することができるコマンド" ページ 1833を参照してください。</p> <p>これらの各コマンドの前にIFまたはELSE IFがある場合には、テストの評価結果がtrueになる必要があります。</p> <p>これらの各コマンドがELSEより後に記述されている場合は、そのコマンドより前にあるどのコマンドでも処理されていないレコードが存在するときに、そのコマンドが実行されます。各コマンドは別々の行で開始して、複数のコマンドを含めることができます。</p>
ELSE IF テスト 省略可能	<p>GROUPコマンドのELSE IFブロックを開きます。この条件は、GROUPコマンド テスト、およびその条件より前にあるどのELSE IF テストにも適合しなかったレコードをテストするものです。</p> <p>複数のELSE IF テストを含めることができ、それらは上から下へ、レコードがTrueと評価されて、そのELSE IF ステートメントに続くコマンドが実行されるまで評価されます。</p>
ELSE 省略可能	<p>GROUPコマンドのELSEブロックを開きます。このブロックより前にあるすべてのテストがFalseと評価されたレコードに対し、このブロックに入力したコマンドが実行されます。</p>
END	GROUPコマンドの終わり。

例

単純な GROUP

単純なグループは、GROUPコマンドで始まり、その後の一連のコマンド群が続き、ENDコマンドで終わります。

```
GROUP
  COUNT
  HISTOGRAM ON Quantity MINIMUM 0 MAXIMUM 100 INTERVALS 10
  CLASSIFY ON Location SUBTOTAL Quantity
END
```

GROUP IF

条件付きグループでは、条件が True または False のどちらであるかに基づいてコマンドを実行します。たとえば、次の GROUP コマンドは、**Product_class** フィールドの値が 5 未満であるレコードに対してのみ実行されます。

```
GROUP IF Product_class < "05"  
  COUNT  
  HISTOGRAM ON Quantity MINIMUM 0 MAXIMUM 100 INTERVALS 10  
  CLASSIFY ON Location SUBTOTAL Quantity  
END
```

GROUP IF ...ELSE

ELSE ブロックを指定しない限り、条件を満たしていないレコードは無視されます。

ELSE ステートメントに続くコマンドの数に制限はありません。次の例では、条件を満たしていないレコードすべての **Quantity** フィールドを合計します。

```
GROUP IF Product_class < "05"  
  COUNT  
  HISTOGRAM ON Quantity MINIMUM 0 MAXIMUM 100 INTERVALS 10  
  CLASSIFY ON Location SUBTOTAL Quantity  
ELSE  
  TOTAL Quantity  
END
```

GROUP IF...ELSE IF...ELSE

ELSE IF ブロックは、各ブロックが別々の条件式であれば、グループ内に複数指定できます。次の例では、ELSE IF および ELSE ブロックは 4 つの合計を生成します。

```
GROUP IF Product_class < "05"  
  COUNT  
  HISTOGRAM ON Quantity MINIMUM 0 MAXIMUM 100 INTERVALS 10  
  CLASSIFY ON Location SUBTOTAL Quantity  
ELSE IF Product_class = "05"
```

```

TOTAL Quantity
ELSE IF Product_class = "06"
  TOTAL Quantity
ELSE IF Product_class = "07"
  TOTAL Quantity
ELSE
  TOTAL Quantity
END

```

入れ子の GROUP コマンド

入れ子のグループとは、別のグループの中に入っているグループのことをいいます。入れ子のグループを使用すると、レコードに対してどのコマンドを実行するかを強かに制御できます。ほとんどのアプリケーションではこうした高度な機能は必要ありませんが、必要に応じて使用することができます。

ほかの種類のグループと同様、入れ子のグループの終了には END コマンドを使用します。Analytics のデータ処理は、すべてのグループコマンドが終了してから開始されます。

```

GROUP IF Product_class < "05"
  COUNT
  STRATIFY ON Quantity SUBTOTAL Quantity MIN 0 MAX 100 INT 10
  GROUP IF Quantity > 0
    STATISTICS ON Quantity
    HISTOGRAM ON Quantity
  END
ELSE
  TOTAL Quantity
END

```

この例では、**Product_class** が 05 未満である場合にのみ、COUNT から次の GROUP までに含まれるコマンドすべてが実行されます。

STATISTICS コマンドと HISTOGRAM コマンドは、**Quantity** が 0 より大きい場合にのみ実行されます。ただし、2 番目の GROUP コマンドは入れ子になっているため、STATISTICS コマンドと HISTOGRAM コマンドは、**Product_class** < "05" の条件と **Quantity** > 0 の条件の両方を満たすレコードに対してのみ実行されます。

GROUP ブロック内でシステム変数を生成する

1つのGROUPコマンドで、複数のシステム変数を作成できます。

通常、TOTAL、COUNT、STATISTICSなどのコマンドを実行すると、ただ1つのシステム変数が生成されます。コマンドを実行するたびに、前回のコマンド実行によって生成されたシステム変数の値は上書きされます。GROUPブロック内で実行されるコマンドのインスタンスごとに、固有の変数が作成されます。

次の例では、TOTALコマンドにより、Metaphor_Trans_2002テーブル内の各製品クラスのAmountフィールドの合計が計算されます。下記のコードを実行すると、以下の変数が生成され、GROUPブロックの後続のコマンド内で使用できるようになります。

- TOTAL2 - 製品クラス 03 の Amount フィールドの合計
- TOTAL3 - 製品クラス 05 の Amount フィールドの合計
- TOTAL4 - 製品クラス 08 の Amount フィールドの合計
- TOTAL5 - 製品クラス 09 の Amount フィールドの合計

```
OPEN Metaphor_Trans_2002
GROUP
  TOTAL AMOUNT IF PRODCLS = "03"
  TOTAL AMOUNT IF PRODCLS = "05"
  TOTAL AMOUNT IF PRODCLS = "08"
  TOTAL AMOUNT IF PRODCLS = "09"
END
CLOSE Metaphor_Trans_2002
```

備考

ヒント

GROUP および LOOP コマンドの詳細なチュートリアルについては、「グループ化とループ処理」ページ 1470を参照してください。

GROUP コマンドの内部で使用することができるコマンド

以下の表は Analytics コマンドの内部で使用することができるコマンドを示しています。

コマンドが以下の一覧にない場合は、GROUP 内で使用できません。

AGE	ASSIGN	BENFORD
CLASSIFY	COMMENT	COUNT
CROSSTAB	DUPLICATES	EXPORT
EXTRACT	GAPS	GROUP

HISTOGRAM	JOIN	LIST
LOOP	MERGE	PROFILE
REPORT	SEQUENCE	STATISTICS
STRATIFY	SUMMARIZE	TOTAL
VERIFY		

グループ化とループ処理

GROUP コマンドを使用すると、テーブル内の次のレコードに移動する前にレコードにいくつかのコマンドを実行することができます。これにより、処理時間を大幅に減らすことができます。

レコードに対して一連のコマンドを複数回実行する必要がある場合には、GROUP コマンド内で LOOP コマンドを使用することができます。

GROUP での変数の使用

ユーザー定義変数

GROUP コマンド内で変数を使用するには、その GROUP ブロックを入力する前に、その変数を定義しておきます。

メモ

GROUP のブロック内で変数を定義して初期化することもできますが、お勧めしません。GROUP 内で初期化された変数を使用すると、予期しない結果が生成される場合があります。

グループで変数を評価するには、代入変数を使用します。変数の値は、GROUP 内に入る前の値から変わらなくなります。

GROUP 内で変数を定義した場合に、代入変数を使ってその変数を参照することはできません。

```
ASSIGN v_test = "hello"
GROUP
  ASSIGN v_test2 = "%v_test% world"
  COMMENT これは無効です: v_test3 = "%v_test2% again"
END
```

システム定義変数

TOTAL や STATISTICS など特定のコマンドでは、実行する計算に基づくシステム変数を生成します。これらのコマンドを実行するのに GROUP を使用すると、生成されるすべてのシステム変数には連続番号が付けられます。この連続番号は、GROUP ブロック内のコマンドの行番号で始まり、 n で終わります(空行は除く)。 n の値は GROUP ブロック内の行番号 1 つごとに 1 ずつ増加します。

メモ

GROUP ブロック内で作成されたシステム生成変数を使用するには、GROUP ブロックが完了するまで待つ必要があります。変数が使用可能になるには、コマンドがテーブル内のすべてのレコードに対して実行される必要があります。これらの変数は、GROUP ブロックを終了する END キーワードの後に使用してください。

次の例では、最初の TOTAL コマンドにより変数 TOTAL2 が生成され、2 番目の TOTAL コマンドにより変数 TOTAL4 が生成されます。これらの変数は両方とも、GROUP ブロックの完了後に後続のコマンドで使用できます。

```
GROUP
  TOTAL Discount IF Order_Priority = "Low"
  ASSIGN v_var = "test"
  TOTAL Discount IF Order_Priority = "High"
END
```

構文に関するメモ

- GROUP コマンドには複数行の構文の記述が必要なため、コマンドラインでこのコマンドを入力することはできません。
- 各 GROUP コマンドは END コマンドで終了する必要があります。
- スクリプト内で GROUP コマンドを使用するときは、グループ内に記載されたコマンドをインデントすることにより、コマンドブロックの読みやすさを向上させることができます。ただし、インデントは必須ではありません。

HB_API_DELETE コマンド

HighBond API に DELETE 要求を送信します。

構文

```
HB_API_DELETE HighBond API 要求 URL HEADERS ヘッダー情報 PASSWORD 番号 <TO 回答ファイル>
```

パラメーター

名前	説明
HighBond API 要求 URL	<p>HighBond リソースの要求詳細。</p> <p>HB_API_DELETE は要求詳細の標準部分を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、これらの要求要素を明示的に指定する必要はありません。</p> <ul style="list-style-type: none"> ホスト情報 HighBond のリージョン HighBond インスタンス ID(組織 ID) <p>特定の HighBond リソースの要求構文については、HighBond API リファレンスを参照してください。</p> <p>メモ ホスト情報を明示的に指定する場合、HighBond API との接続には HTTPS プロトコルを使用する必要があります。例：<code>https://apis-us.highbond.com</code></p>
HEADERS ヘッダー情報	<p>要求ヘッダー情報。</p> <p>HB_API_DELETE はデフォルトのヘッダー情報を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、ヘッダー情報を明示的に指定する必要はありません。</p> <p>デフォルトの HEADERS 値：</p> <pre>'{"content-type": "application/vnd.api+json"}'</pre>
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定し</p>

名前	説明						
	<p>たり入力したりしている場合、またはアナリティクス スクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2 番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" 見開きページを参照してください。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>PASSWORD 番号が必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">Analytics (オンライン アクティベーション)</td> <td style="padding: 5px;">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td style="padding: 5px;">Analytics (オフライン アクティベーション)</td> <td style="padding: 5px;">PASSWORD 番号は必要です。</td> </tr> <tr> <td style="padding: 5px;">ロボット</td> <td></td> </tr> </table> </div>	Analytics (オンライン アクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフライン アクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンライン アクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます						
Analytics (オフライン アクティベーション)	PASSWORD 番号は必要です。						
ロボット							
<p>TO 回答ファイル 省略可能</p>	<p>要求の回答を含むファイルの名前。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列として回答ファイルを指定します。 例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0; width: fit-content;"> <p>TO "response.json"</p> </div> <p>回答 JSON ファイルを保存する場所。</p> <p>回答 JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。</p> <p>Analytics でスクリプトを実行する</p> <p>デフォルトでは、回答 JSON ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>ファイルを別の既存のフォルダーに保存するには、ファイルパスにその名前を含めます。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0; width: fit-content;"> <p>TO "C:\HighBond API responses\response.json"</p> </div> <p>ロボットでスクリプトを実行する</p> <p>アナリティクス ヘッダーで //RESULTFILEタグを指定すると、回答 JSON ファイルは、ロボットでのそれぞれのタスク実行の出力として保存されます。</p> <p>ファイル名のみを指定します。ファイルパスを指定しないでください。</p>						

備考

パスワード定義の作成とパスワード値の指定

ロボットで HighBond API に要求を送信するスクリプトを実行する場合、要求を送信するコマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 下を参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクスタグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクスタグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクスタグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようすると、パスワードプロンプトが表示されます。 詳細については、"PASSWORD コマンド" ページ 2015を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。
 - Analytics メインメニューで **[ツール>HighBond アクセストークン]** を選択します。
 - **スクリプト エディター**内を右クリックし、 **挿入 > HighBond トークン]** を選択します。

[API アクセストークンを管理します] ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **[API トークンの管理]** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。
 - **既存のトークンを使用する - トークン]** 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する - トークンを作成 > Analytics]** をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **[コピー]** をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。
 - **PASSWORD アナリティクス タグ - ACL ロボット のタスクデザイナー**で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
 - **PASSWORD コマンド - Analytics** で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド - Analytics** で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。

5. Launchpad で、トークンが含まれているダイアログボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[HighBond アクセストークンの作成と管理](#)を参照してください。

HB_API_GET コマンド

HighBond API に GET 要求を送信します。

構文

```
HB_API_GET HighBond API 要求 URL HEADERS ヘッダー情報 PASSWORD 番号 <TO 回答ファイル>
```

パラメーター

名前	説明
HighBond API 要求 URL	<p>HighBond リソースの要求詳細。</p> <p><code>HB_API_GET</code> は要求詳細の標準部分を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、これらの要求要素を明示的に指定する必要はありません。</p> <ul style="list-style-type: none"> ホスト情報 HighBond のリージョン HighBond インスタンス ID(組織 ID) <p>特定の HighBond リソースの要求構文については、HighBond API リファレンスを参照してください。</p> <p>メモ ホスト情報を明示的に指定する場合、HighBond API との接続には HTTPS プロトコルを使用する必要があります。例：<code>https://apis-us.highbond.com</code></p>
HEADERS ヘッダー情報	<p>要求ヘッダー情報。</p> <p><code>HB_API_GET</code> はデフォルトのヘッダー情報を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、ヘッダー情報を明示的に指定する必要はありません。</p> <p>デフォルトの HEADERS 値：</p> <pre>'{"content-type": "application/vnd.api+json}"'</pre>
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定し</p>

名前	説明						
	<p>たり入力したりしている場合、またはアナリティクス スクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2 番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" ページ 1843を参照してください。</p> <div style="border-left: 2px solid blue; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>PASSWORD 番号は必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">Analytics (オンライン アクティベーション)</td> <td style="padding: 5px;">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td style="padding: 5px;">Analytics (オフライン アクティベーション)</td> <td style="padding: 5px;">PASSWORD 番号は必要です。</td> </tr> <tr> <td style="padding: 5px;">ロボット</td> <td></td> </tr> </table> </div>	Analytics (オンライン アクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフライン アクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンライン アクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます						
Analytics (オフライン アクティベーション)	PASSWORD 番号は必要です。						
ロボット							
<p>TO 回答ファイル</p>	<p>要求の回答を含むファイルの名前。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列として回答ファイルを指定します。 例：</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0; width: fit-content;"> <p>TO "response.json"</p> </div> <p>回答 JSON ファイルを保存する場所。</p> <p>回答 JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。</p> <p>Analytics でスクリプトを実行する</p> <p>デフォルトでは、回答 JSON ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>ファイルを別の既存のフォルダーに保存するには、ファイルパスにその名前を含めます。</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0; width: fit-content;"> <p>TO "C:\HighBond API responses\response.json"</p> </div> <p>ロボットでスクリプトを実行する</p> <p>アナリティクス ヘッダーで //RESULTFILEタグを指定すると、回答 JSON ファイルは、ロボットでのそれぞれのタスク実行の出力として保存されます。</p> <p>ファイル名のみを指定します。ファイルパスを指定しないでください。</p>						

例

組織のすべてのロボットのリストを返します

最初の例はコマンドの短縮名称を使用します。これは、HighBond API 要求からすべてのデフォルト値を省略します。

```
HB_API_GET "robots" TO all_robots.json
```

2 番目の例では、HighBond API 要求にすべてのデフォルト値が含まれていて、回答 JSON ファイルを Analytics プロジェクトフォルダー以外の場所に保存します。

```
HB_API_GET "https://apis-us.highbond.com/v1/orgs/11594/robots" HEADERS  
'{"content-type": "application/vnd.api+json"}' TO "C:\HighBond API  
responses\all_robots.json"
```

上記の両方の例で、**all_robots.json** 回答ファイルのロボットのリストは同じです。例：

```
{  
  "data": [  
    {  
      "id": "17504",  
      "type": "robots",  
      "attributes": {  
        "active_app_version": 4,  
        "app_versions_count": 5,  
        "name": "Concur T&E Data Integration",  
        "category": "acl",  
        "drive_system_user": "exYRZqYABvrjHCjV7E7j"  
      }  
    },  
    {  
      "id": "24202",  
      "type": "robots",  
      "attributes": {  
        "active_app_version": 2,  
        "app_versions_count": 2,  
        "name": "Test_Steele_Adverse_Media",  
        "category": "highbond",  
        "drive_system_user": "exYRZqYABvrjHCjV7E7j"  
      }  
    }  
  ]  
}
```

```

    }
  }
]
}

```

備考

パスワード定義の作成とパスワード値の指定

ロボットで HighBond API に要求を送信するスクリプトを実行する場合、要求を送信するコマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、「HighBond アクセストークンを取得」見開きページを参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクスタグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクスタグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクスタグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようとする、パスワードプロンプトが表示されます。 詳細については、「PASSWORD コマンド」ページ 2015を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール>HighBond アクセストークン**]を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン**]を選択します。

API アクセストークンを管理します]ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理**]ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する - トークン**]列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する - トークンを作成 > Analytics**]をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー**]をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。

- **PASSWORDアナリティクスタグ-ACLロボット**の**タスクデザイナー**で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。

コマンド

- **PASSWORD コマンド** -Analytics で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド** -Analytics で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。
- 新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
- 詳細については、[HighBond アクセストークンの作成と管理](#)を参照してください。

HB_API_PATCH コマンド

HighBond API に PATCH 要求を送信します。

構文

```
HB_API_PATCH HighBond API 要求 URL HEADERS ヘッダー情報 DATA ペイロードファイル
PASSWORD 番号 <TO 回答ファイル>
```

パラメーター

名前	説明
HighBond API 要求 URL	<p>HighBond リソースの要求詳細。</p> <p>HB_API_PATCH は要求詳細の標準部分を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、これらの要求要素を明示的に指定する必要はありません。</p> <ul style="list-style-type: none"> ◦ ホスト情報 ◦ HighBond のリージョン ◦ HighBond インスタンス ID(組織 ID) <p>特定の HighBond リソースの要求構文については、HighBond API リファレンスを参照してください。</p> <p>メモ ホスト情報を明示的に指定する場合、HighBond API との接続には HTTPS プロトコルを使用する必要があります。例：<code>https://apis-us.highbond.com</code></p>
HEADERS ヘッダー情報	<p>要求ヘッダー情報。</p> <p>HB_API_PATCH はデフォルトのヘッダー情報を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、ヘッダー情報を明示的に指定する必要はありません。</p> <p>デフォルトの HEADERS 値：</p> <pre>'{"content-type": "application/vnd.api+json"}'</pre>
DATA ペイロードファイル	<p>要求ペイロードを含むファイルの名前。</p> <p>要求ペイロードは HighBond に送信するデータです。JSONファイルにデータを格納し、DATA を使用して HighBond API 要求のファイルを参照します。特定の HighBond リソースのペイロードデータの構築については、HighBond API リファレンスを参照してください。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列としてペイロードファイルを指定しま</p>

名前	説明
	<p>す。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "payload.json"</pre> <p>メモ ロボットで実行するスクリプトの場合、アナリティクスヘッダーで、DATA ペイロードファイルパラメーターに対応する //FILEタグも指定する必要があります。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">COMMENT //ANALYTIC Test HB API コマンド //FILE payload.json END</pre> <p>ペイロード JSON ファイルを保存する場所 ペイロード JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。 Analytics でスクリプトを実行する Analytics プロジェクトを含むフォルダーまたは別のフォルダーにペイロード JSON ファイルを保存できます。 プロジェクトフォルダー以外のフォルダーにファイルを保存する場合、ペイロードファイルにファイルパスとファイル名を含める必要があります。</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "C:\HighBond API payloads\payload.json"</pre> <p>ロボットでスクリプトを実行する ペイロード JSON ファイルを、スクリプトを実行するロボットの [入力/出力] タブにアップロードします。スクリプトを実行する前に、ファイルがこのタブに存在する必要があります。 ペイロードファイルでファイル名のみを指定します。ファイルパスを指定しないでください。</p>
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ○ "PASSWORD コマンド" ページ 2015 ○ SET PASSWORD コマンド ○ PASSWORD アナリティクスタグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" 見開きページを参照してください。</p>

名前	説明						
	<p>メモ</p> <p>PASSWORD 番号が必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="607 375 1269 665"> <tr> <td data-bbox="607 375 937 514">Analytics (オンラインアクティベーション)</td> <td data-bbox="937 375 1269 514">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="607 514 937 606">Analytics (オフラインアクティベーション)</td> <td data-bbox="937 514 1269 606">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="607 606 937 665">ロボット</td> <td data-bbox="937 606 1269 665"></td> </tr> </table>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます						
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。						
ロボット							
<p>TO 回答ファイル 省略可能</p>	<p>要求の回答を含むファイルの名前。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列として回答ファイルを指定します。 例：</p> <div data-bbox="565 837 1344 905" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "response.json"</p> </div> <p>回答 JSON ファイルを保存する場所。</p> <p>回答 JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。</p> <p>Analytics でスクリプトを実行する</p> <p>デフォルトでは、回答 JSON ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>ファイルを別の既存のフォルダーに保存するには、ファイルパスにその名前を含めます。</p> <div data-bbox="565 1209 1344 1276" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "C:\HighBond API responses\response.json"</p> </div> <p>ロボットでスクリプトを実行する</p> <p>アナリティクス ヘッダーで //RESULTFILEタグ を指定すると、回答 JSON ファイルは、ロボットでのそれぞれのタスク実行の出力として保存されます。</p> <p>ファイル名のみを指定します。ファイルパスを指定しないでください。</p>						

備考

パスワード定義の作成とパスワード値の指定

ロボットで HighBond API に要求を送信するスクリプトを実行する場合、要求を送信するコマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセス トークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 下を参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクス タグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクス タグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクス タグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようすると、パスワードプロンプトが表示されます。 詳細については、"PASSWORD コマンド" ページ 2015を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール > HighBond アクセストークン** を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン** を選択します。

API アクセストークンを管理します ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。
 - **既存のトークンを使用する** - [トークン] 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - [トークンを作成 > Analytics] をクリックし、お使いの HighBond アカウントのパスワードを入力します。
新しい Analytics トークンが作成されます。
3. [コピー] をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログ ボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。
 - **PASSWORD アナリティクス タグ - ACL ロボットのタスクデザイナー** で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
 - **PASSWORD コマンド - Analytics** で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド - Analytics** で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[HighBond アクセストークンの作成と管理](#) を参照してください。

HB_API_POST コマンド

HighBond API に POST 要求を送信します。

構文

```
HB_API_POST HighBond API 要求 URL HEADERS ヘッダー情報 DATA ペイロードファイル
PASSWORD 番号 <TO 回答ファイル>
```

パラメーター

名前	説明
HighBond API 要求 URL	<p>HighBond リソースの要求詳細。</p> <p><code>HB_API_POST</code> は要求詳細の標準部分を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、これらの要求要素を明示的に指定する必要はありません。</p> <ul style="list-style-type: none"> ◦ ホスト情報 ◦ HighBond のリージョン ◦ HighBond インスタンス ID(組織 ID) <p>特定の HighBond リソースの要求構文については、HighBond API リファレンスを参照してください。</p> <p>メモ ホスト情報を明示的に指定する場合、HighBond API との接続には HTTPS プロトコルを使用する必要があります。例：<code>https://apis-us.highbond.com</code></p>
HEADERS ヘッダー情報	<p>要求ヘッダー情報。</p> <p><code>HB_API_POST</code> はデフォルトのヘッダー情報を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、ヘッダー情報を明示的に指定する必要はありません。</p> <p>デフォルトの <code>HEADERS</code> 値：</p> <pre>'{"content-type": "application/vnd.api+json"}'</pre>
DATA ペイロードファイル	<p>要求ペイロードを含むファイルの名前。</p> <p>要求ペイロードは HighBond に送信するデータです。JSONファイルにデータを格納し、<code>DATA</code>を使用して HighBond API 要求のファイルを参照します。特定の HighBond リソースのペイロードデータの構築については、HighBond API リファレンスを参照してください。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列としてペイロードファイルを指定しま</p>

名前	説明
	<p>す。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "payload.json"</pre> <p>メモ ロボットで実行するスクリプトの場合、アナリティクスヘッダーで、DATA ペイロードファイルパラメーターに対応する //FILEタグも指定する必要があります。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">COMMENT //ANALYTIC Test HB API コマンド //FILE payload.json END</pre> <p>ペイロード JSON ファイルを保存する場所 ペイロード JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。 Analytics でスクリプトを実行する Analytics プロジェクトを含むフォルダーまたは別のフォルダーにペイロード JSON ファイルを保存できます。 プロジェクトフォルダー以外のフォルダーにファイルを保存する場合、ペイロードファイルにファイルパスとファイル名を含める必要があります。</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "C:\HighBond API payloads\payload.json"</pre> <p>ロボットでスクリプトを実行する ペイロード JSON ファイルを、スクリプトを実行するロボットの [入力/出力] タブにアップロードします。スクリプトを実行する前に、ファイルがこのタブに存在する必要があります。 ペイロードファイルでファイル名のみを指定します。ファイルパスを指定しないでください。</p>
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ○ "PASSWORD コマンド" ページ 2015 ○ SET PASSWORD コマンド ○ PASSWORD アナリティクスタグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" ページ 1854を参照してください。</p>

名前	説明						
	<p>メモ</p> <p>PASSWORD 番号は必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="607 380 1271 667"> <tr> <td data-bbox="607 380 938 516">Analytics (オンラインアクティベーション)</td> <td data-bbox="938 380 1271 516">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="607 516 938 611">Analytics (オフラインアクティベーション)</td> <td data-bbox="938 516 1271 611">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="607 611 938 667">ロボット</td> <td data-bbox="938 611 1271 667"></td> </tr> </table>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます						
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。						
ロボット							
<p>TO 回答ファイル 省略可能</p>	<p>要求の回答を含むファイルの名前。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列として回答ファイルを指定します。 例：</p> <div data-bbox="565 840 1346 905" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "response.json"</p> </div> <p>回答 JSON ファイルを保存する場所。</p> <p>回答 JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。</p> <p>Analytics でスクリプトを実行する</p> <p>デフォルトでは、回答 JSON ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>ファイルを別の既存のフォルダーに保存するには、ファイルパスにその名前を含めます。</p> <div data-bbox="565 1209 1346 1274" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "C:\HighBond API responses\response.json"</p> </div> <p>ロボットでスクリプトを実行する</p> <p>アナリティクス ヘッダーで //RESULTFILEタグ を指定すると、回答 JSON ファイルは、ロボットでのそれぞれのタスク実行の出力として保存されます。</p> <p>ファイル名のみを指定します。ファイルパスを指定しないでください。</p>						

例

HighBond プロジェクトで問題を作成する

JSON 形式を使用する要求ペイロードを構築し、`create_issue.json` という名前のファイルにペイロードを保存します。`HB_API_POST` コマンドを使用して、コマンドでペイロードファイルを指定し、ID 19756 のプロジェクトに問題を作成します。

ヒント

ペイロード構文をすばやく構築するには、適切なペイロード構文ブロックを [HighBond API リファレンス](#) からコピーします。ペイロード ブロックをコピーした後、空のままにする予定のキーと値のペアを削除します。

```
HB_API_POST "projects/19756/issues" HEADERS '{"content-type":  
"application/vnd.api+json"}' DATA create_issue.json PASSWORD 1 TO hb_  
api_response.json
```

`create_issue.json` の内容:

```
{  
  "data": {  
    "type": "issues",  
    "attributes": {  
      "description": "Description of issue",  
      "owner": "Jane Sleaman",  
      "deficiency_type": "Deficiency",  
      "title": "Data retention and backup",  
      "severity": "High",  
      "published": true,  
      "identified_at": "2021-11-01T18:15:30Z"  
    }  
  }  
}
```

備考

パスワード定義の作成とパスワード値の指定

ロボットで HighBond API に要求を送信するスクリプトを実行する場合、要求を送信するコマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、["HighBond アクセストークンを取得"](#) 次のページを参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクスタグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクスタグを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクスタグ を参照してください。
PASSWORD コマンド (Analyticsで実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようとする、パスワードプロンプトが表示されます。 詳細については、「PASSWORD コマンド」ページ 2015を参照してください。
SET PASSWORD コマンド (Analyticsで実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール > HighBond アクセストークン** を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン** を選択します。

API アクセストークンを管理します ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロファイルからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。
 - **既存のトークンを使用する** - [トークン]列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - [トークンを作成 > Analytics] をクリックし、お使いの HighBond アカウントのパスワードを入力します。
新しい Analytics トークンが作成されます。
3. [コピー] をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。
 - **PASSWORDアナリティクスタグ** -ACLロボットの**タスクデザイナー**で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
 - **PASSWORD コマンド** -Analytics で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド** -Analytics で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
5. Launchpad で、トークンが含まれているダイアログボックスを閉じます。
新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
詳細については、[HighBond アクセストークンの作成と管理](#)を参照してください。

HB_API_PUT コマンド

HighBond API に PUT 要求を送信します。

構文

```
HB_API_PUT HighBond API 要求 URL HEADERS ヘッダー情報 DATA ペイロードファイル
PASSWORD 番号 <TO 回答ファイル>
```

パラメーター

名前	説明
HighBond API 要求 URL	<p>HighBond リソースの要求詳細。</p> <p><code>HB_API_PUT</code> は要求詳細の標準部分を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、これらの要求要素を明示的に指定する必要はありません。</p> <ul style="list-style-type: none"> ◦ ホスト情報 ◦ HighBond のリージョン ◦ HighBond インスタンス ID(組織 ID) <p>特定の HighBond リソースの要求構文については、HighBond API リファレンスを参照してください。</p> <p>メモ ホスト情報を明示的に指定する場合、HighBond API との接続には HTTPS プロトコルを使用する必要があります。例：<code>https://apis-us.highbond.com</code></p>
HEADERS ヘッダー情報	<p>要求ヘッダー情報。</p> <p><code>HB_API_PUT</code> はデフォルトのヘッダー情報を自動的にバックグラウンドで提供します。デフォルト値を上書きしたい場合を除き、ヘッダー情報を明示的に指定する必要はありません。</p> <p>デフォルトの <code>HEADERS</code> 値：</p> <pre>'{"content-type": "application/vnd.api+json"}'</pre>
DATA ペイロードファイル	<p>要求ペイロードを含むファイルの名前。</p> <p>要求ペイロードは HighBond に送信するデータです。JSONファイルにデータを格納し、<code>DATA</code>を使用して HighBond API 要求のファイルを参照します。特定の HighBond リソースのペイロードデータの構築については、HighBond API リファレンスを参照してください。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列としてペイロードファイルを指定しま</p>

名前	説明
	<p>す。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "payload.json"</pre> <p>メモ ロボットで実行するスクリプトの場合、アナリティクスヘッダーで、DATA ペイロードファイルパラメーターに対応する //FILEタグも指定する必要があります。例：</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">COMMENT //ANALYTIC Test HB API コマンド //FILE payload.json END</pre> <p>ペイロード JSON ファイルを保存する場所 ペイロード JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。 Analytics でスクリプトを実行する Analytics プロジェクトを含むフォルダーまたは別のフォルダーにペイロード JSON ファイルを保存できます。 プロジェクトフォルダー以外のフォルダーにファイルを保存する場合、ペイロードファイルにファイルパスとファイル名を含める必要があります。</p> <pre style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">DATA "C:\HighBond API payloads\payload.json"</pre> <p>ロボットでスクリプトを実行する ペイロード JSON ファイルを、スクリプトを実行するロボットの [入力/出力] タブにアップロードします。スクリプトを実行する前に、ファイルがこのタブに存在する必要があります。 ペイロードファイルでファイル名のみを指定します。ファイルパスを指定しないでください。</p>
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ○ "PASSWORD コマンド" ページ 2015 ○ SET PASSWORD コマンド ○ PASSWORD アナリティクスタグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、"パスワード定義の作成とパスワード値の指定" 次のページを参照してください。</p>

名前	説明						
	<p>メモ</p> <p>PASSWORD 番号は必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="607 378 1269 667"> <tr> <td data-bbox="607 378 937 516">Analytics (オンラインアクティベーション)</td> <td data-bbox="937 378 1269 516">PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="607 516 937 606">Analytics (オフラインアクティベーション)</td> <td data-bbox="937 516 1269 606">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="607 606 937 667">ロボット</td> <td data-bbox="937 606 1269 667"></td> </tr> </table>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windows レジストリに保管されている現在のユーザーの HighBond アクセストークンが自動的に使用されます						
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。						
ロボット							
<p>TO 回答ファイル 省略可能</p>	<p>要求の回答を含むファイルの名前。</p> <p>*.json ファイル拡張子を持つ引用符で囲まれた文字列として回答ファイルを指定します。 例：</p> <div data-bbox="565 840 1344 907" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "response.json"</p> </div> <p>回答 JSON ファイルを保存する場所。</p> <p>回答 JSON ファイルを保存する場所はスクリプトを実行する場所によって決まります。</p> <p>Analytics でスクリプトを実行する</p> <p>デフォルトでは、回答 JSON ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>ファイルを別の既存のフォルダーに保存するには、ファイルパスにその名前を含めます。</p> <div data-bbox="565 1209 1344 1276" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>TO "C:\HighBond API responses\response.json"</p> </div> <p>ロボットでスクリプトを実行する</p> <p>アナリティクス ヘッダーで //RESULTFILEタグ を指定すると、回答 JSON ファイルは、ロボットでのそれぞれのタスク実行の出力として保存されます。</p> <p>ファイル名のみを指定します。ファイルパスを指定しないでください。</p>						

備考

パスワード定義の作成とパスワード値の指定

ロボットで HighBond API に要求を送信するスクリプトを実行する場合、要求を送信するコマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 下を参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクスタグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクスタグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクスタグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようとする、パスワードプロンプトが表示されます。 詳細については、"PASSWORD コマンド" ページ 2015を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール > HighBond アクセストークン** を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン** を選択します。

API アクセストークンを管理します ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。

- **既存のトークンを使用する** - **トークン** 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - **トークンを作成 > Analytics** をクリックし、お使いの HighBond アカウントのパスワードを入力します。

新しい Analytics トークンが作成されます。

3. **コピー** をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログ ボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。

- **PASSWORD アナリティクス タグ - ACL ロボットのタスクデザイナー** で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
- **PASSWORD コマンド - Analytics** で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
- **SET PASSWORD コマンド - Analytics** で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。

5. Launchpad で、トークンが含まれているダイアログ ボックスを閉じます。

新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。

詳細については、[HighBond アクセストークンの作成と管理](#) を参照してください。

HELP コマンド

Analytics ヘルプ文書をブラウザで起動します。

構文

```
HELP
```

HISTOGRAM コマンド

文字フィールドまたは数値フィールドの値に基づいてレコードをグループ化し、各グループ内のレコード数をカウントして、グループとカウントを棒グラフで表します。

構文

```
HISTOGRAM {<ON> 文字フィールド|<ON> 数値フィールド MINIMUM 値 MAXIMUM 値
{<INTERVALS 数値>|FREE 間隔値 <...n> 最後の間隔}} <TO SCREEN|ファイル名
|GRAPH|PRINT}> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <HEADER ヘッダー
テキスト> <FOOTER フッター テキスト> <KEY 内訳フィールド> <SUPPRESS> <COLUMNS 数値>
<APPEND> <OPEN>
```

パラメーター

名前	説明
ON 文字フィールド	ヒストグラムに使用する文字フィールドまたは式。
ON 数値フィールド	ヒストグラムに使用する数値フィールドまたは式。
MINIMUM 値	数値型のフィールドにのみ適用されます。最初の数値間隔の最小値。 FREE を使用している場合、MINIMUM は省略可能です。それ以外の場合は必要になります。
MAXIMUM 値	数値型のフィールドにのみ適用されます。最後の数値間隔の最大値。 FREE を使用している場合、MAXIMUM は省略可能です。それ以外の場合は必要になります。
INTERVALS 数 省略可能	数値型のフィールドにのみ適用されます。 MINIMUM 値と MAXIMUM 値によって指定された範囲の間に Analytics が生成する、均等な間隔の数。間隔の数を指定しない場合は、デフォルトの数を使用されます。 デフォルトは、 [オプション]ダイアログボックスの [コマンド] タブの [間隔の数] によって決定されます。
FREE 間隔値 <...n> 最終 間隔 省略可能	数値型のフィールドにのみ適用されます。 各間隔の開始点と最後の間隔の終了点を指定することにより、カスタム サイズの間隔を作成することができます。

名前	説明
	<p>MINIMUM 値と MAXIMUM 値を指定した場合は、これらの値がそれぞれ最初の間隔の開始点と最後の間隔の終了点となり、各 間隔値 が範囲内に追加の間隔を生成します。指定する間隔値は、MINIMUM 値より大きく、かつ MAXIMUM 値以下である必要があります。</p> <p>間隔値は、数値順でなければならず、重複値を含めることはできません。次に例を示します。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、FREE -1000, 0, 1000, 2000, 3000 のように指定します。</p> </div> <p>FREE と INTERVALS の両方を指定する場合は、INTERVALS が無視されます。</p>
<p>TO SCREEN ファイル名 GRAPH PRINT</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ ファイル名 - は結果の保存先となるファイルです。 <p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> <ul style="list-style-type: none"> ◦ GRAPH - は結果をグラフに表示し、それを Analytics の表示領域に表示します ◦ 印刷 - 通常使うプリンターに結果を送信します <p>メモ</p> <p>ファイルに出力されたヒストグラムの結果は、棒グラフのテキスト表現として表示されます。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p>

名前	説明
	<p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
KEY ブレークフィールド 省略可能	<p>小計計算をグループ化するフィールドまたは式。ブレークフィールドの値が変わるたびに、小計が計算されます。</p> <p>ブレークフィールドは、文字フィールドか式である必要があります。指定できるフィールドは 1 つだけですが、1 つ以上のフィールドを含んでいる式を使用することができます。</p>
SUPPRESS 省略可能	<p>MAXIMUM 値より大きい値と MINIMUM 値より小さい値をコマンド出力から除外します。</p>
COLUMNS 数 省略可能	<p>ヒストグラムの結果をテキストファイルに出力する場合は、棒グラフをテキストで表示するときの x 軸の長さを指定します。</p> <p>数の値は、x 軸 (および y 軸のラベル) に使用する文字スペース (テキスト列) の数です。COLUMNS を省略した場合は、デフォルトの文字スペース数の 78 が使用されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>

例

時給の基本的なヒストグラム

次の例は、HISTOGRAM を使用して、時給 0 ~ 100ドルの間の賃金の分布を示すグラフを作成しています。

```
HISTOGRAM ON Rate MINIMUM 0 MAXIMUM 100 TO GRAPH
```

時給用に定義された間隔を使用したヒストグラム

直前の例に続けて HISTOGRAM を使用し、直前の例より意味のあるやり方でグラフの範囲を指定します。

賃金の大部分は時給 20 ~ 50ドルの範囲に含まれているため、グラフに含まれる間隔の数は以下のとおりです。

- 20 ~ 50 の範囲に3つ
- 0 ~ 20 に1つ
- 50 ~ 100 に1つ
- 100 より大きい範囲に1つ

```
HISTOGRAM ON Rate MINIMUM 0 MAXIMUM 100 FREE 20,30,40,50,100 TO GRAPH
```

備考

このコマンドの動作の詳細については、「ヒストグラムの作成」ページ 1343を参照してください。

最小値と最大値を設定する

HISTOGRAM コマンドを実行する前に、数値フィールドで STATISTICS コマンドまたは PROFILE コマンドを実行して、フィールド内の最小値と最大値を MINIMUM パラメーターと MAXIMUM パラメーターに自動的に設定することができます。

関連コマンド

文字フィールドを使用してヒストグラムを作成するのは、分類に似ています。数値フィールドを使用してヒストグラムを作成するのは、階層化に似ています。

Analytics 内の他のグループ化操作とは異なり、ヒストグラムは数値フィールドの小計をサポートしていません。

IF コマンド

コマンドを実行するために true と評価する必要がある条件を指定します。

構文

```
IF テスト コマンド
```

パラメーター

名前	説明
検査	コマンドを実行するために満たす必要がある条件。
コマンド	テストの評価結果が true の場合に実行する、任意の有効な ACLScript コマンド。

例

スクリプトを条件付きで実行する

`v_counter` 変数が 10 より大きい場合にのみ、テーブルに対して CLASSIFY を実行するには、次のように指定します。

```
IF v_counter > 10 CLASSIFY ON Location TO "Count_by_Location.fil" OPEN
```

ユーザーの決定に基づいてコマンドを実行する

テーブルを分類化するかどうかをスクリプトのユーザーに決定させたいとします。

スクリプトで、選択した場合には、CLASSIFY を実行できるというチェックボックスとともに、ダイアログボックスを含めます。チェックボックスには、論理変数 `v_classify_checkbox` に True または False 入力値が保存されます。

IF テストを使用して、`v_classify_checkbox` の値を決定します。値が True の場合は、CLASSIFY は次を実行します。

```
IF v_classify_checkbox=T CLASSIFY ON Location TO "Count_by_
Location.fil" OPEN
```

備考

ヒント

IF コマンドおよび IF パラメーターの詳細なチュートリアルについては、"スクリプトでの決定" ページ 1446 および "統制構造" ページ 1467 を参照してください。

IF コマンドと IF パラメーターの比較

IF コマンドのロジックは、ほとんどのコマンドでサポートされている IF パラメーターとは異なります。

- **IF コマンド**: -テスト式の値に基づいて、関連付けられたコマンドを実行するかどうかを判断します。
- **IF パラメーター**: -Analytics テーブル内の各レコードに対し、テスト式の値に基づいて、コマンドを実行するかどうかを判断します。

スクリプト内で判断を行う

スクリプトでは、一連の IF コマンド条件を入力して、その結果を基に異なるコマンドを実行できます。さらに処理を行うかどうかを判断するために、IF コマンドを使用して変数の値をテストすることができます。

IMPORT ACCESS コマンド

Microsoft Access データベース ファイルを定義およびインポートして、Analytics テーブルを作成します。

メモ

IMPORT ACCESS コマンドを使用するには、32 ビットの Microsoft Access データベース エンジンがインストールされている必要があります。詳細については、"Microsoft Access データベースエンジンを除外するオプション" ページ 2664を参照してください。

構文

```
IMPORT ACCESS TO テーブル <PASSWORD >数値> インポート ファイル名 FROM ソース ファイル
名 TABLE 入力テーブル名 CHARMAX 最大フィールド長 MEMOMAX 最大フィールド長
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できませんが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
PASSWORD 番号 省略可能	<p>パスワード保護された Access ファイルでのみ使用します。</p> <p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に 2 つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクス スクリプトをスケジュールしている場合、<code>PASSWORD 2</code> により、2 番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> "PASSWORD コマンド" ページ 2015 SET PASSWORD コマンド PASSWORD アナリティクス タグ

名前	説明
インポートファイル名	<p>作成する Analytics データファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL"</p> <p>デフォルトでは、データファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ◦ "C:\data\Invoices.FIL" ◦ "data\Invoices.FIL"
FROM ソースファイル名	<p>ソース データファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> ◦ "C:\data\ソース ファイル名" ◦ "data\ソース ファイル名"
TABLE 入力テーブル名	<p>インポートする Microsoft Access データベースファイルのテーブルの名前。</p>
CHARMAX 最大フィールド長	<p>インポートしているソースの文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>1 ~ 255 文字を指定できます。</p> <p>フィールドを切り詰める機能では、時折出現する長い値が原因で、インポート処理でサポートされるレコード全体の長さを超えることがなくなります。</p> <ul style="list-style-type: none"> ◦ 32,767 文字 (非 Unicode Analytics) ◦ 16,383 文字 (Unicode Analytics)
MEMOMAX 最大フィールド長	<p>インポートするテキスト、ノート、またはメモフィールドの文字の最大長。</p> <p>1 ~ 32767 文字 (非 Unicode 版の Analytics)、または 16383 文字 (Unicode 版の Analytics) を指定できます。</p>

例

このコマンドの動作の詳細については、"Microsoft Access データベースファイルのインポート" ページ 263 を参照してください。

テーブルへのインポート

Microsoft Access ファイル `Acceptable_Codes.mdb` があります。[Acceptable_Codes] テーブルを Analytics からインポートする必要があります。このためには、次のコマンドを使用して、`acc_codes` テーブルを Analytics で作成します。

インポートされた文字またはメモフィールドの長さが、フィールドの最も長い値の長さまたは指定された文字の最大数の短い方に設定されます。

```
SET ECHO NONE
SET PASSWORD 1 TO "qr347wx"
SET ECHO ON
IMPORT ACCESS TO acc_codes PASSWORD 1 "C:\ACL DATA\Sample Data
Files\acc_codes.fil" FROM "Acceptable_Codes.mdb" TABLE "[Acceptable_
Codes]" CHARMAX 60 MEMOMAX 70
```

IMPORT DELIMITED コマンド

区切り文字付きテキスト ファイルを定義およびインポートして、Analytics テーブルを作成します。

構文

```
IMPORT DELIMITED TO テーブル インポート ファイル名 FROM ソース ファイル名 <SERVER プロ  
ファイル名> ソースの文字エンコード SEPARATOR {文字|TAB|SPACE} QUALIFIER {文字|NONE}  
<CONSECUTIVE> STARTLINE 行番号 <KEEPTITLE> <CRCLEAR> <LFCLEAR> <REPLACENULL>  
<ALLCHAR> {ALLFIELDS | [フィールド構文] <...n> <IGNOREフィールド番号> <...n>}
```

```
フィールド構文 ::=  
FIELD 名前 型 AT 開始位置 DEC 値 WID バイト PIC 書式 AS 表示名
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
インポートファイル名	<p>作成する Analytics データ ファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL"</p> <p>デフォルトでは、データ ファイル(.FIL) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータ ファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> "C:\data\Invoices.FIL" "data\Invoices.FIL"
FROM ソースファイル名	<p>ソース データ ファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データ ファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p>

名前	説明															
	<ul style="list-style-type: none"> "C:\data\ソース ファイル名" "data\ソース ファイル名" 															
SERVER プロファイル名 省略可能	インポートしたいデータが置かれている AX Server のサーバー プロファイル名。															
ソースの文字エンコード	<p>ソース データの文字セット および文字エンコード。</p> <p>お使いの Analytics エディションとソース データのエンコードに応じて、次のうち該当するコードを指定してください。</p> <table border="1"> <thead> <tr> <th>コード</th> <th>Analytics のエディション</th> <th>ソース データのエンコード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>非 Unicode 版</td> <td>すべてのデータ</td> </tr> <tr> <td>0</td> <td>Unicode 版</td> <td>ASCII データ</td> </tr> <tr> <td>2</td> <td>Unicode 版</td> <td>Unicode データ、UTF-16 LE エンコード</td> </tr> <tr> <td>3 数値コード*</td> <td>Unicode 版</td> <td>Unicode データ、UTF-16 LE エンコード ソース データのエンコードに適合する数値コードを決定するには、データ定義ウィザードを使用してインポートを実行し、[エンコードされたテキスト] オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、③、1つのスペース、数値コードを順に指定します。</td> </tr> </tbody> </table>	コード	Analytics のエディション	ソース データのエンコード	0	非 Unicode 版	すべてのデータ	0	Unicode 版	ASCII データ	2	Unicode 版	Unicode データ、UTF-16 LE エンコード	3 数値コード*	Unicode 版	Unicode データ、UTF-16 LE エンコード ソース データのエンコードに適合する数値コードを決定するには、 データ定義ウィザード を使用してインポートを実行し、 [エンコードされたテキスト] オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、③、1つのスペース、数値コードを順に指定します。
コード	Analytics のエディション	ソース データのエンコード														
0	非 Unicode 版	すべてのデータ														
0	Unicode 版	ASCII データ														
2	Unicode 版	Unicode データ、UTF-16 LE エンコード														
3 数値コード*	Unicode 版	Unicode データ、UTF-16 LE エンコード ソース データのエンコードに適合する数値コードを決定するには、 データ定義ウィザード を使用してインポートを実行し、 [エンコードされたテキスト] オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、③、1つのスペース、数値コードを順に指定します。														
SEPARATOR 文字 TAB SPACE	<p>ソース データのフィールド間で使用される区切り文字。文字は引用符で囲まれた文字列として指定する必要があります。</p> <p>タブまたは空白を区切り文字として指定するには、それらを二重引用符で囲んで入力するか、またはそれらのキーワードを使用します。</p> <ul style="list-style-type: none"> SEPARATOR " " または SEPARATOR SPACE SEPARATOR " " または SEPARATOR SPACE 															
QUALIFIER 文字 NONE	<p>ソース データ内のフィールド値を折り返すためと識別するために使用するテキスト修飾子文字。文字は引用符で囲まれた文字列として指定する必要があります。</p> <p>文字をテキスト修飾子として指定するには、二重引用符を一重引用符で囲む必要があります。QUALIFIER '''</p> <p>テキスト修飾子がないことを指定するには、次のいずれかの方法を使用します。</p> <ul style="list-style-type: none"> QUALIFIER "" QUALIFIER NONE 															
CONSECUTIVE 省略可能	連続したテキスト修飾子を単一の修飾子として扱います。															
STARTLINE 行番号	<p>ファイルの読み取りを開始する行番号。</p> <p>たとえば、ファイルの先頭 3 行に不要なヘッダー情報が含まれている場合に、4 行目からデータの読み取りを開始するには、「STARTLINE 4」を指定します。</p>															

名前	説明
KEEPTITLE 省略可能	<ul style="list-style-type: none"> ○ KEEPTITLE と ALLFIELDS を組み合わせて使用する - STARTLINE に指定した行番号がデータでなくフィールド名として扱われます。 KEEPTITLE を指定しない場合は、汎用フィールド名が使用され、STARTLINE に指定した行番号はデータとして扱われます。 ○ KEEPTITLE と各 FIELD 構文を組み合わせて使用する - STARTLINE に指定した行番号がインポートされなくなります。"FIELD 名前" の "名前" にはフィールド名を指定します。 KEEPTITLE を指定しない場合には、STARTLINE に指定した行番号はデータとして扱われます。"FIELD 名前" の "名前" にはフィールド名を指定します。
CRCLEAR 省略可能	<p>テキスト修飾子間で発生するすべての CR 文字 (キャリッジリターン) をスペース文字で置換します。CRCLEAR を使用するには、QUALIFIER と文字値を指定する必要があります。</p> <p>CRCLEAR と LFCLEAR の両方を使用する場合は、CRCLEAR を最初に記述する必要があります。</p>
LFCLEAR 省略可能	<p>テキスト修飾子間で発生するすべての LF 文字 (ラインフィード) をスペース文字で置換します。LFCLEAR を使用するには、QUALIFIER と文字値を指定する必要があります。</p> <p>CRCLEAR と LFCLEAR の両方を使用する場合は、CRCLEAR を最初に記述する必要があります。</p>
REPLACENULL 省略可能	<p>スペースで区切られたファイルで発生するすべての NUL 文字を置換します。置き換えられたすべての NULL 文字の数がログに記録されます。</p>
ALLCHAR 省略可能	<p>インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。</p> <p>ヒント インポートされたすべてのフィールドに文字のデータ型を割り当てると、区切り文字付きテキスト ファイルのインポート処理が容易になります。 Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。 Analytics によって識別子のフィールドに数値のデータ型が自動的に割り当てられたテーブルをインポートする際、実際には文字のデータ型を使用する必要がある場合には、ALLCHAR を使用することができます。</p>
ALLFIELDS	<p>ソース データ ファイルの全フィールドがインポートされます。</p> <p>ALLFIELDS を使用する際に Analytics によってデータ型が割り当てられる方法の詳細については、"備考" ページ 1879を参照してください。</p> <p>メモ ALLFIELDS を指定する場合、個別の FIELD 構文や IGNORE 指定しないでください。</p>
FIELD 名前 型	<p>インポートするソース データ ファイル内の個別フィールドの名前およびデータ型。フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。</p> <p>型については、"フィールド データ型の識別子" ページ 1879を参照してください。</p> <p>メモ ALLCHAR を指定した場合には、型は無視されます。</p>

名前	説明				
AT 開始位置	<p>Analytics データファイル内のフィールドの開始バイトを指定します。</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
DEC 値	<p>数値フィールドの小数点以下の桁数</p> <p>メモ</p> <p>DEC を指定した場合には、PIC は無視されます。</p>				
WID バイト	<p>Analytics テーブルレイアウトにおけるフィールドの長さ(バイト数)</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
PIC 書式	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソースデータの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソースデータの物理形式と正確に一致する必要があります。たとえば、ソースデータが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p> <p>メモ</p> <p>ALLCHAR を指定した場合には、PIC は無視されます。</p>				
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にス</p>				

名前	説明
	ペースがないことを確認してください。
IGNORE フィールド番号 <...n> 省略可能	<p>テーブルレイアウトからフィールドを除外します。</p> <p>フィールド番号は、ソースデータファイルにおける除外対象フィールドの位置を指定します。たとえば、IGNORE 5 は、ソースデータファイルの5番目のフィールドを Analytics テーブルレイアウトから除外します。</p> <p>メモ</p> <p>このフィールドのデータは、インポートは行われますが、定義されないため、新しい Analytics テーブルには表示されません。必要に応じ、データを後で定義して、テーブルに追加することができます。</p> <p>特定のフィールドをインポート対象から完全に除外するには、個別に各フィールドを指定する際にそのフィールドを指定しないでください。</p>

例

すべてのフィールドをインポートする

カンマ区切りファイルから **Employees** という名前の Analytics テーブルにすべてのフィールドをインポートするとします。このファイルでは、テキスト修飾子として二重引用符が使用されています。データ型は、「備考」ページ 1879に示した規則セットに基づいて自動的に割り当てられます。

```
IMPORT DELIMITED TO Employees "Employees.fil" FROM "Employees.csv" 0
SEPARATOR "," QUALIFIER '"' CONSECUTIVE STARTLINE 1 KEPTITLE
ALLFIELDS
```

すべてのフィールドをインポートし、文字のデータ型を自動的に割り当てる

カンマ区切りファイルから **Employees** という名前の Analytics テーブルにすべてのフィールドをインポートするとします。このファイルでは、テキスト修飾子として二重引用符が使用されています。インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。

```
IMPORT DELIMITED TO Employees "Employees.fil" FROM "Employees.csv" 0
SEPARATOR "," QUALIFIER '"' CONSECUTIVE STARTLINE 1 KEPTITLE
ALLFIELDS
```

指定したフィールドをインポートし、文字のデータ型を自動的に割り当てる

タブ区切りファイルから **Employees** という名前の Analytics テーブルに指定のフィールドをインポートするとします。このファイルでは、テキスト修飾子として二重引用符が使用されています。インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。

```
IMPORT DELIMITED TO Employees "Employees.fil" FROM "Employees.csv" 0
SEPARATOR TAB QUALIFIER '"' CONSECUTIVE STARTLINE 1 KEPTITLE ALLCHAR
FIELD "First_Name" C AT 1 DEC 0 WID 25 PIC "" AS "First Name" FIELD
>Last_Name" C AT 26 DEC 0 WID 25 PIC "" AS "Last Name" FIELD "CardNum"
C AT 51 DEC 0 WID 16 PIC "" AS "Card Num" FIELD "EmpNo" C AT 67 DEC 0
WID 6 PIC "" AS "Emp Num" FIELD "HireDate" C AT 73 DEC 0 WID 10 PIC ""
AS "Hire Date" FIELD "Salary" C AT 83 DEC 0 WID 5 PIC "" AS "" FIELD
"Bonus_2016" C AT 88 DEC 0 WID 10 PIC "" AS "Bonus 2016"
```

指定したフィールドをインポートし、フィールドごとにデータ型を割り当てる

セミコロン区切りファイルから **Employees** という名前の Analytics テーブルに指定のフィールドをインポートするとします。このファイルではテキスト修飾子は使用されていません。インポートするフィールドごとにデータ型を指定します。

```
IMPORT DELIMITED TO Employees "Employees.fil" FROM "Employees.csv" 0
SEPARATOR TAB QUALIFIER '"' CONSECUTIVE STARTLINE 1 KEPTITLE ALLCHAR
FIELD "First_Name" C AT 1 DEC 0 WID 25 PIC "" AS "First Name" FIELD
>Last_Name" C AT 26 DEC 0 WID 25 PIC "" AS "Last Name" FIELD "CardNum"
C AT 51 DEC 0 WID 16 PIC "" AS "Card Num" FIELD "EmpNo" C AT 67 DEC 0
WID 6 PIC "" AS "Emp Num" FIELD "HireDate" C AT 73 DEC 0 WID 10 PIC ""
AS "Hire Date" FIELD "Salary" C AT 83 DEC 0 WID 5 PIC "" AS "" FIELD
"Bonus_2016" C AT 88 DEC 0 WID 10 PIC "" AS "Bonus 2016"
```

備考

このコマンドの動作の詳細については、"区切り文字付きテキスト ファイルのインポート" ページ 265を参照してください。

ALLFIELDS を使用する際に Analytics によってデータ型が割り当てられる仕組み

個別にフィールドを定義する代わりに ALLFIELDS パラメーターを使用すると、区切り文字付きファイルの先頭にあるレコードのサブセットが Analytics によって調べられ、次に示す規則セットに基づいてフィールドにデータ型が割り当てられます。

Analytics にインポートされたデータのフィールドには、必要に応じて数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。

区切り文字付きファイル内のフィールド値の説明	例	割り当てられるデータ型
テキスト修飾子で囲んだ値	"ABC 社のサプライヤー" "6,990.75"	文字
フィールド内の任意の位置の値には、数値の区切り文字として使用されるカンマとピリオド、および負号 (-) を除く、数字以外の文字を 1 つ使用できます。	\$995 (995)	文字
値には数値、数値の区切り文字、および負号 (-) のみを使用できます。	6,990.75 -6,990.75 995	数値
フィールドには 1 つまたは複数の空白値が出現します。		文字
区切り文字や英字の月名を含む複数の日付時刻値	2016/12/31 2016 年 12 月 31 日	文字
数字のみから成る日付時刻値	20161231	数値

フィールド データ型の識別子

以下の表は、FIELD でデータ型を指定するときに使用する必要がある文字の一覧を示します。各文字は Analytics データ型の 1 つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、"C": FIELD "Last_Name" C と指定します。

詳細については、"Analytics のデータ型" ページ 811を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C"(CHARACTER 型)が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS

コマンド

文字	Analytics データ型
Z	ZONED

IMPORT EXCEL コマンド

Microsoft Excel ワークシートまたは名前付き範囲を定義およびインポートして、Analytics テーブルを作成します。

メモ

IMPORT EXCEL コマンドで古い Excel ファイル (*.xls) を扱うには、32 ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、「Microsoft Access データベースエンジンを除外するオプション」ページ 2664 を参照してください。

構文

```
IMPORT EXCEL TO テーブル インポートファイル名 FROM ソースファイル名 TABLE 入力ワークシートまたは名前付き範囲 <KEEPTITLE> <STARTLINE >行番号> <ALLCHAR>
{ALLFIELDS|CHARMAX 最大フィールド長|[フィールド構文] <...n> <IGNORE >フィールド番号>
<...n>} <OPEN>
```

```
フィールド構文 ::=
FIELD インポート名 型 {PIC 書式|WID 文字数 DEC 値} AS 表示名
```

メモ

IMPORT EXCEL パラメーターは、上記と完全に同じ順序で、以下の表にあるように、指定する必要があります。

ワークブックの保護ビューが有効な場合、analytics は Excel ワークブックからインポートできません。まず、ワークブックで編集を有効にし、ワークブックを保存して閉じてから、インポートを実行する必要があります。

パラメーター

名前	説明
TO テーブル	データをインポートする Analytics テーブルの名前。

名前	説明
	<p>メモ</p> <p>テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
<p>インポートファイル名</p>	<p>作成する Analytics データファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL"</p> <p>デフォルトでは、データファイル (.FIL) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ○ "C:\data\Invoices.FIL" ○ "data\Invoices.FIL"
<p>FROM ソースファイル名</p>	<p>ソース データファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データ ファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> ○ "C:\data\ソース ファイル名" ○ "data\ソース ファイル名"
<p>TABLE ワークシートまたは名前付き範囲</p>	<p>Excel のソースデータファイルからインポートするワークシートまたは名前付き範囲。</p> <p>要件:</p> <ul style="list-style-type: none"> ○ ワークシート名の末尾には、"\$" 記号を追加する 例: TABLE "Corp_Credit_Cards\$" ○ 名前付き範囲は、Excel にあるとおりに指定する 例: TABLE "Employees_Sales" ○ ワークシートまたは名前付き範囲は、引用符で囲んだ文字列として指定する
<p>KEEPTITLE 省略可能</p>	<ul style="list-style-type: none"> ○ KEEPTITLE と ALLFIELDS または CHARMAX を組み合わせて使用する - STARTLINE に指定した行番号がデータでなくフィールド名として扱われます。 KEEPTITLE を指定しない場合は、汎用フィールド名が使用され、STARTLINE に指定した行番号はデータとして扱われます。 ○ KEEPTITLE と各 FIELD 構文を組み合わせて使用する - STARTLINE に指定した行番号がインポートされなくなります。"FIELD 名前" の "名前" にはフィールド名を指定します。 KEEPTITLE を指定しない場合には、STARTLINE に指定した行番号はデータとして扱われます。"FIELD 名前" の "名前" にはフィールド名を指定します。
<p>STARTLINE 行番号 省略可能</p>	<p>ワークシートの読み取りを開始する行番号。</p> <p>たとえば、ワークシートの先頭 3 行に不要なヘッダー情報が含まれている場合に、4 行目からデータの読み取りを開始するには、「STARTLINE 4」を指定します。</p> <p>STARTLINE を指定しない場合には、開始行はワークシートの 1 行目となります。</p>

名前	説明
	<p>メモ</p> <p>名前付き範囲の開始行は、STARTLINE の設定値に関係なく、常に名前付き範囲の 1 行目になります。</p>
ALLCHAR 省略可能	<p>インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。</p> <p>ヒント</p> <p>インポートされたすべてのフィールドに文字データ型を割り当てると、Excel ファイルのインポート処理が容易になります。</p> <p>Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。</p> <p>Analytics によって識別子のフィールドに数値のデータ型が自動的に割り当てられたテーブルをインポートする際、実際には文字のデータ型を使用する必要がある場合には、ALLCHAR を使用することができます。</p>
ALLFIELDS	<p>ソース データ ファイルの全フィールドがインポートされます。</p> <p>メモ</p> <p>ALLFIELDS を指定する場合、個別の FIELD 構文、CHARMAX、IGNORE 指定しないでください。</p>
CHARMAX 最大フィールド長	<p>Excel のソースデータファイル内の文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>Excel ファイル内の最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。</p> <p>Excel ファイルの全フィールドが、データ型に関係なく、インポートされます。</p> <p>メモ</p> <p>CHARMAX を指定する場合には、個別の FIELD 構文、ALLFIELDS、IGNORE はいっさい指定しないでください。</p>
FIELD インポート名 型	<p>インポートするソース データ ファイル内の個別フィールドの名前およびデータ型。</p> <p>インポート名が Analytics テーブルのフィールド名になります。インポート名は、ソース データ ファイル内のフィールド名と同じでも同じでなくてもどちらでも構いません。</p> <p>ヒント</p> <p>インポート名とは異なる表示名を指定するには、追加で AS を使用します。</p> <p>型が Analytics テーブルのフィールドのデータ型になります。型は、ソース データ ファイル内のフィールド データ型と同じでも同じでなくてもどちらでも構いません。型の詳細については、"フィールド データ型の識別子" ページ 1889を参照してください。</p> <p>メモ</p> <p>ALLCHAR を指定した場合には、型は無視されます。</p> <p>個別の FIELD 構文を指定する場合には、ALLFIELDS や CHARMAX は指定しないでください。</p>

名前	説明
	<p>フィールドの除外</p> <p>フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。また、除外するフィールドに対しては、IGNOREを指定することも必要です。</p>
PIC 書式	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソースデータの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソースデータの物理形式と正確に一致する必要があります。たとえば、ソースデータが12/31/2014である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>
WID 文字	Analytics テーブルレイアウトにおけるフィールドの長さ(文字)。
DEC 値	数値フィールドの小数点以下の桁数
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にスペースがないことを確認してください。</p>
IGNORE フィールド番号 <...n> 省略可能	<p>テーブルレイアウトからフィールドを除外します。</p> <p>フィールド番号は、ソースデータファイルにおける除外対象フィールドの位置を指定します。たとえば、IGNORE 5 は、ソースデータファイルの5番目のフィールドを Analytics テーブルレイアウトから除外します。</p> <p>メモ</p> <p>フィールド番号が除外対象フィールドの位置を正確に指すように注意してください。追加対象フィールド(フィールド定義)を表しているか、存在しないフィールド位置を表しているフィールド番号を指定すると、インポートが正しく行われなくなります。</p> <p>FIELD パラメーター数と IGNORE パラメーター数の合計が、ソースデータテーブル内の総フィールド数と一致する必要があります。これらの数が一致しないと、インポートが正しく行われません。</p> <p>ALLFIELDS または CHARMAX を指定する場合は、IGNORE を指定しないでください。</p>
OPEN	コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。

名前	説明
省略可能	

例

指定されたフィールドをインポートする

Credit_Cards という名前の新しい Analytics テーブルを定義するインポートを行うとします。このインポートでは、Excel データの 1 行目をフィールド名として使用します。

この Analytics テーブルにはソース テーブルの 12 個のフィールドのうち、3 つのフィールドを定義、追加し、9 個のフィールドは除外します。

```
IMPORT EXCEL TO Credit_Cards "Credit Cards.fil" FROM "Credit_Cards_
Metaphor.xls" TABLE "Corp_Credit_Cards$" KEPTITLE FIELD "CARDNUM" C
WID 16 AS "Card Number" FIELD "EXPDT" D WID 10 PIC "YYYY-MM-DD" AS
"Expiry Date" FIELD "PASTDUEAMT" N WID 6 DEC 2 AS "Past Due" IGNORE 2
IGNORE 3 IGNORE 5 IGNORE 6 IGNORE 7 IGNORE 9 IGNORE 10 IGNORE 11
IGNORE 12
```

すべてのフィールドをインポートする

May_Transactions という名前の新しい Analytics テーブルを定義するインポートを行うとします。このインポートでは、Excel データの 1 行目をフィールド名として使用します。

Analytics テーブルにはソース テーブルのすべてのフィールドを含め、デフォルトのフィールド定義を使用します。

最も長い値に設定されたフィールド長

最初の例では、ソース データ ファイルの文字データから生成されるフィールドの長さは、そのフィールド内で最長の値の長さに設定されます。

```
IMPORT EXCEL TO May_Transactions "May_Transactions.fil" FROM "Trans_
May.xls" TABLE "Trans1_May$" KEPTITLE ALLFIELDS
```

制約されたフィールド長

2 番目の例では、ソースデータファイルの文字データから生成されるフィールドの長さは、そのフィールド内で最長の値の長さの 50 文字の CHARMAX 値のうち、短い方に設定されます。

```
IMPORT EXCEL TO May_Transactions "May_Transactions.fil" FROM "Trans_
May.xls" TABLE "Trans1_May$" KEPTITLE CHARMAX 50
```

すべてのフィールドを文字データ型としてインポートする

May_Transactions という名前の新しい Analytics テーブルを定義するインポートを行うとします。数値や日付を含むすべてのフィールドが文字データ型としてインポートされます。

```
IMPORT EXCEL TO May_Transactions "May_Transactions.fil" FROM "Trans_
May.xls" TABLE "Trans1_May$" KEPTITLE ALLCHAR ALLFIELDS
```

すべてのフィールドを文字データとしてインポートし、ヘッダー情報をスキップする

Past_Due_Report という名前の新しい Analytics テーブルを定義するインポートを行うとします。Excel ファイルのうち、レポート ヘッダー情報が含まれる最初の 2 行スキップし、フィールド名が含まれる 3 行目からファイルの読み取りを開始します。数値や日付を含むすべてのフィールドが文字データ型としてインポートされます。

```
IMPORT EXCEL TO Past_Due_Report "Past_Due_Report.fil" FROM "Past_Due_
Report.xlsx" TABLE "Sheet1$" KEPTITLE STARTLINE 3 ALLCHAR ALLFIELDS
```

備考

このコマンドの動作の詳細については、「Microsoft Excel データのインポート」ページ 250を参照してください。

個別にフィールドを定義するか、デフォルト定義を使用してすべてのフィールドをインポートします。

Analytics テーブルに Excel ファイルをインポートするとき、FIELD パラメーターを使用して各フィールドを個別に定義するか、あるいは ALLFIELDS パラメーターまたは CHARMAX パラメーターを使用して、デフォルトの Analytics フィールド定義を基にすべてのフィールドをインポートすることができます。

パラメーターの組み合わせを変えると、異なる結果が生成されます。以下の表は様々な組み合わせと結果をまとめたものです。

メモ

"定義する"とは、フィールド名、データ型、長さ、日付時刻書式などの項目を手動で指定することを意味しています。

目的	使用すべきパラメーター	使用してはいけないパラメーター
<ul style="list-style-type: none"> すべてのフィールドをデフォルトの定義を使って自動的にインポートする フィールドを Analytics にインポートした後に必要に応じてフィールドを定義する 	ALLFIELDS	CHARMAX、FIELD
<ul style="list-style-type: none"> すべてのフィールドをデフォルトの定義を使って自動的にインポートする フィールドを Analytics にインポートした後に必要に応じてフィールドを定義する 長い文字フィールドを切り捨てる 	CHARMAX	ALLFIELDS、FIELD
<ul style="list-style-type: none"> フィールドをインポート前に定義する 	FIELD	ALLFIELDS、CHARMAX
<ul style="list-style-type: none"> フィールドをインポート前に定義する 一部のフィルターをインポート対象から除外する 	FIELD IGNORE	ALLFIELDS、CHARMAX
<ul style="list-style-type: none"> フィールドをインポート前に部分的に定義する すべてのフィールドを自動的に文字データとしてインポートする 	ALLCHAR FIELD	ALLFIELDS、CHARMAX
<ul style="list-style-type: none"> ワークシートの冒頭にあるヘッダー情報や空白行を除外する 	STARTLINE	
<ul style="list-style-type: none"> ワークシートの先頭行をフィールド名として使用する 	KEEPTITLE	
<ul style="list-style-type: none"> STARTLINE に指定したワークシート行をフィールド名として使用する 	KEEPTITLE STARTLINE	

ALLFIELDS または CHARMAX を使用する際に Analytics によってデータ型が割り当てられる仕組み

個別にフィールドを定義する代わりに ALLFIELDS または CHARMAX パラメーターを使用すると、Excel ファイルの先頭にあるレコードのサブセットが Analytics によって調べられ、一連の内部規則に基づいてフィールドにデータ型が割り当てられます。

Analytics にインポートされたデータのフィールドには、必要に応じて数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。

データ インポートの最大サイズ

ファイル形式 .xlsx または .xlsm

.xlsx または .xlsm ファイルからインポートできる Excel の列の最大数、およびフィールド内の最大文字数は特定の数に制限されません。

これらの Excel ファイルの種類からのインポートは、Analytics データファイル(.fil) における 32 KB のレコード長の制限によって制御されます。ソース Excel ファイル内のいずれかのレコードによって 32 KB より長い Analytics レコードが作成される場合、インポートは失敗します。

ファイル形式 .xls

.xls(Excel 97 - 2003) ファイルのインポートは違う種類の処理を用いており、次を最大数としています。

- 255 列
- フィールドにつき 255 文字
- レコードにつき 32 KB
- 65,000 行

フィールド データ型の識別子

以下の表は、FIELD でデータ型を指定するときを使用する必要がある文字の一覧を示します。各文字は Analytics データ型の 1 つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、"C": FIELD "Last_Name" C と指定します。

詳細については、「Analytics のデータ型」ページ 811を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C"(CHARACTER 型) が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS

コマンド

文字	Analytics データ型
Z	ZONED

IMPORT GRCPROJECT コマンド

HighBond プロジェクト テーブルをインポートして、Analytics テーブルを作成します。

構文

```
IMPORT GRCPROJECT TO テーブル インポートファイル名 PASSWORD番号 FROM 組織 ID タイプ
ID <FIELD 名前 AS 表示名 <...n>> <CHARMAX 最大フィールド長>
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字()を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
インポートファイル名	<p>作成する Analytics データ ファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL".</p> <p>デフォルトでは、データファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> "C:\data\Invoices.FIL" "data\Invoices.FIL"
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、PASSWORD 2により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p>

名前	説明						
	<ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、「パスワード定義の作成とパスワード値の指定」ページ 1896を参照してください。</p> <p>メモ</p> <p>PASSWORD 番号は必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="605 575 1271 863"> <tr> <td data-bbox="605 575 938 716">Analytics (オンラインアクティベーション)</td> <td data-bbox="938 575 1271 716">PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="605 716 938 804">Analytics (オフラインアクティベーション)</td> <td data-bbox="938 716 1271 804">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="605 804 938 863">ロボット</td> <td data-bbox="938 804 1271 863"></td> </tr> </table>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます						
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。						
ロボット							
FROM 組織IDタイプID	<p>組織とインポート中のデータを定義する情報のタイプ。</p> <ul style="list-style-type: none"> ◦ 組織ID - データをインポートしているHighBond組織 ◦ タイプID - インポートする情報のタイプ。 <p>組織ID値とタイプID値はスペースを入れずにスラッシュで区切る必要があります。</p> <pre>FROM "125@-eu/audits"</pre> <p>文字列全体を引用符で囲む必要があります。</p> <p>組織ID</p> <p>組織IDには、組織のID番号が含まれている必要があるほか、北米(米国)以外のデータセンターへインポートする場合には、そのデータセンターのコードも含まれている必要があります。組織ID番号とデータセンターのコードは、アットマーク(@)とハイフン(-)で区切る必要があります。</p> <pre>FROM "125@-eu"</pre> <p>メモ</p> <p>北米(US)データセンターコードを指定する場合は、ハイフンを含まない少し異なる形式が使用されます。</p> <pre>FROM "125@us"</pre> <p>データセンターコードは、どの地域のHighBondサーバーからデータをインポートするかを指定します。</p> <ul style="list-style-type: none"> ◦ af - アフリカ(南アフリカ) ◦ ap - アジア太平洋(シンガポール) ◦ au - アジア太平洋(オーストラリア) ◦ ca - 北米(カナダ) ◦ eu - ヨーロッパ(ドイツ) ◦ sa - 南米(ブラジル) ◦ us - 北米(米国) 						

名前	説明
	<p>組織の HighBond インスタンスに対して承認されているデータセンターコードのみを使用できます。北米のデータセンターがデフォルトであるため、@us を指定するのはオプションです。</p> <p>組織 ID 番号がわからない場合は、Analytics のユーザー インターフェイスを使用して、プロジェクトからテーブルをインポートします。組織 ID 番号はログのコマンドに含まれます。詳細については、"HighBond プロジェクト データのインポート" ページ 759を参照してください。</p> <h3>タイプ ID</h3> <p>タイプIDはインポートする情報のタイプを指定します。プロジェクトの情報は一連の関連するテーブルに含まれます。</p> <p>タイプIDでは、次のリストの値のいずれかを使用します。必要に応じて、表示される値を正確に入力します(アンダースコアを含む)。</p> <ul style="list-style-type: none"> ○ 監査 - プロジェクト ○ 統制テスト計画 - 統制テスト計画 ○ 統制テスト - 統制テスト ○ 統制 - 統制 ○ - Actions - アクション ○ 調査結果 - 問題 ○ 軽減 - リスクコントロールの関連付け ○ 説明文 - 説明文 ○ 目標 - 目標 ○ リスク - リスク ○ ウォークスルー - ウォークスルー <h3>ヒント</h3> <p>プロジェクトのテーブルが関連付けられる方法および Analytics にインポートした後、テーブルを結合するために使用できるキーフィールドについては、"HighBond プロジェクト データのインポート" ページ 759を参照してください。</p>
<p>FIELD 名前 AS 表示名 <...n> 省略可能</p>	<p>インポートするソースデータの個別のフィールド。名前を指定します。</p> <p>FIELD を省略すると、すべてのフィールドがインポートされます。</p> <ul style="list-style-type: none"> ○ 名前は、プロジェクトのテーブルの物理フィールド名と正確に一致する必要があります。大文字小文字も区別されます。 ○ 表示名は、新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)です。各 FIELD 名の表示名を指定する必要があります。表示名の値は引用符で囲まれた文字列。 <p>列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>Analytics の他のいくつかの IMPORT コマンドとは異なり、FIELD 名を表示名として使用する方法として、空の表示名を指定することはできません。</p> <h3>ヒント</h3> <p>物理フィールド名を取得するには、Analytics のユーザー インターフェイスを使用して、プロジェクトから該当するテーブルをインポートします。物理フィールド名はログのコマンドに含まれます。</p> <p>後続のインポートをスクリプト化できます。</p>
<p>CHARMAX 最大フィールド長 省略可能</p>	<p>プロジェクトテーブル内の文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>プロジェクトテーブル内の最大フィールド長を超えるデータは、Analytics にインポートすると切り</p>

名前	説明
	<p>捨てられます。</p> <p>フィールドを切り詰める機能では、時折出現する長い値が原因で、インポート処理でサポートされるレコード全体の長さを超えることがなくなります。</p> <ul style="list-style-type: none"> 32,767 文字 (非 Unicode Analytics) 16,383 文字 (Unicode Analytics)

例

指定したフィールドをプロジェクト テーブルからインポートする

組織 286(米国地域)に属するすべてのアクティブなプロジェクトに関するプロジェクト テーブル内の指定したフィールドを、Analytics テーブル **All_Projects** にインポート します。

```
IMPORT GRCPROJECT TO All_Projects "C:\HighBond Projects Data\All_
Projects.fil" FROM "286@us/audits" FIELD "id" AS "Id" FIELD
"description" AS "Description" FIELD "name" AS "Name" FIELD "start_
date" AS "Start date" FIELD "status" AS "Status" FIELD "created_at" AS
"Created at"
```

プロジェクト テーブルからすべてのフィールドをインポートする

組織 286(米国地域)に属するすべてのアクティブなプロジェクトに関するプロジェクト テーブル内のすべてのフィールドを、Analytics テーブル **All_Projects** にインポート します。接続を認証するため、番号付けされたパスワード定義を追加します。以上を行うコマンドは次のようになります。

```
IMPORT GRCPROJECT TO All_Projects "C:\HighBond Projects Data\All_
Projects.fil" PASSWORD 1 FROM "286@us/audits"
```

プロジェクトテーブルからインポートするときのフィールド長の制限

プロジェクト テーブル内のすべてのフィールドを、Analytics テーブル **All_Projects** にインポートするとします。CHARMAX パラメーターを使用してフィールド長を制限します。

```
IMPORT GRCPROJECT TO All_Projects "C:\HighBond Projects Data\All_
Projects.fil" PASSWORD1 FROM "286@us/audits" CHARMAX 200
```

結果として得られた Analytics テーブルでは、プロジェクト内の文字データから発生するすべてのフィールドは、指定した 200 文字長に制限されます。制限を超えるフィールド値は 200 文字に切り詰められます。

すべてのフィールドを問題テーブルからインポートする

この例では、組織 11594(欧州地域)に属するすべてのアクティブなプロジェクトに関する問題テーブルのすべてのフィールドを、Analytics テーブル **All_Issues** にインポートします

```
IMPORT GRCPROJECT TO All_Issues "C:\HighBond Projects Data\All_
Issues.fil" FROM "11594@-eu/findings"
```

備考

このコマンドの動作の詳細については、"HighBond プロジェクト データのインポート" ページ 759を参照してください。

パスワード定義の作成とパスワード値の指定

ロボットで HighBond プロジェクトテーブルをインポートするスクリプトを実行する場合、IMPORT GRCPROJECT コマンドを使用してパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件が Analytics で実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセストークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 次のページを参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクスタグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクスタグを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクスタグ を参照してください。
PASSWORD コマンド (Analyticsで実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようとする、パスワードプロンプトが表示されます。 詳細については、「PASSWORD コマンド」ページ 2015を参照してください。
SET PASSWORD コマンド (Analyticsで実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBondに接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール > HighBond アクセストークン** を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン** を選択します。

API アクセストークンを管理します ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロファイルからこのページにアクセスすることもできます。

2. 次のいずれかを実行します。
 - **既存のトークンを使用する** - [トークン] 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- **新しいトークンを作成する** - [トークンを作成 > Analytics] をクリックし、お使いの HighBond アカウントのパスワードを入力します。
新しい Analytics トークンが作成されます。
3. [コピー] をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

4. 使用しているパスワード定義方法に応じて次のいずれかを実行します。
 - **PASSWORD アナリティクス タグ** - ACL ロボットの **タスクデザイナー** で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
 - **PASSWORD コマンド** - Analytics で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - **SET PASSWORD コマンド** - Analytics で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
5. Launchpad で、トークンが含まれているダイアログボックスを閉じます。
新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
詳細については、[HighBond アクセストークンの作成と管理](#) を参照してください。

インポート デバッグ機能

HighBond からのインポートには、簡易デバッグ機能があります。

インポートされたデータは、対象 Analytics プロジェクトを含むフォルダーの JSON 中間ファイルに一時的に格納されます。Analytics プロジェクトを含むフォルダーでは、データが Analytics にインポートされた後に削除する代わりに、JSON ファイルを保持するテキストファイルを作成できます。

- **JSON ファイルが存在する** - HighBond からのインポートが失敗し、JSON ファイルがコンピューターにある場合は、問題が HighBond 側ではなく、Analytics 側にあることがわかっています。
- **JSON ファイルが存在しない** - HighBond からのインポートが失敗し、JSON ファイルがコンピューターにない場合は、問題が HighBond 側にあることがわかっています。

この情報はトラブルシューティングで役立ちます。

JSON 中間ファイルの保持を設定する

ターゲット Analytics プロジェクトを含むフォルダーで、`_grc_import_debug.txt` という名前の空のテキストファイルを作成します。

HighBond のリザルトまたはプロジェクトからインポートするときには、JSON 中間ファイルは `results.json` という名前で保持されます。ファイルは後続の各 HighBond からのインポートするたびに上書きされます。

IMPORT GRCRESULTS コマンド

HighBond のリザルト テーブルまたは解釈をインポートして、Analytics テーブルを作成します。

構文

```
IMPORT GRCRESULTS TO テーブル インポートファイル名 PASSWORD番号 FROM リザルトリソースパス <FIELD 名前 AS 表示名 <...n>> <CHARMAX 最大フィールド長>
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
インポートファイル名	<p>作成する Analytics データファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL".</p> <p>デフォルトでは、データファイル (.FIL) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> "C:\data\Invoices.FIL" "data\Invoices.FIL"
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に 2 つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、PASSWORD 2 により、2 番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p>

名前	説明						
	<ul style="list-style-type: none"> ○ "PASSWORD コマンド" ページ 2015 ○ SET PASSWORD コマンド ○ PASSWORD アナリティクス タグ <p>必須のパスワード値は、HighBond アクセストークンです。詳細については、「パスワード定義の作成とパスワード値の指定」ページ 1906を参照してください。</p> <p>メモ</p> <p>PASSWORD 番号は必要な場合と不要な場合があります。スクリプトを実行する環境によって異なります。</p> <table border="1" data-bbox="605 573 1271 863"> <tr> <td data-bbox="605 573 938 716">Analytics (オンラインアクティベーション)</td> <td data-bbox="938 573 1271 716">PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます</td> </tr> <tr> <td data-bbox="605 716 938 804">Analytics (オフラインアクティベーション)</td> <td data-bbox="938 716 1271 804">PASSWORD 番号は必要です。</td> </tr> <tr> <td data-bbox="605 804 938 863">ロボット</td> <td data-bbox="938 804 1271 863"></td> </tr> </table>	Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます	Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。	ロボット	
Analytics (オンラインアクティベーション)	PASSWORD 番号は不要です。 Windowsレジストリに保管されている現在のユーザーのHighBondアクセストークンが自動的に使用されます						
Analytics (オフラインアクティベーション)	PASSWORD 番号は必要です。						
ロボット							
FROM リザルトリソースパス	<p>インポートしているデータへのリザルトパス。</p> <p>パスの形式は、インポートしているデータによって異なります。パスの形式の詳細については、「リザルトパス」ページ 1904を参照してください。</p> <p>メモ</p> <p>リザルトパスの形式はAPIによって提供され、変更されることがあります。パスの正確な現在の構文を取得する最も簡単で信頼できる方法は、対象データの手動インポートを実行し、コマンドログからパスをコピーすることです。</p>						
FIELD 名前 AS 表示名 <...n> 省略可能	<p>インポートするソースデータの個別のフィールド。名前を指定します。</p> <p>FIELD を省略すると、すべてのフィールドがインポートされます。</p> <p>名前</p> <p>名前は、リザルトのテーブルの物理フィールド名と正確に一致する必要があります。大文字小文字も区別されます。物理フィールド名を表示するには、次のいずれかを実行します。</p> <ul style="list-style-type: none"> ○ リザルトで、テーブルビューの列見出しをクリックします。[フィールド名]の後に物理フィールド名が表示されます。 ○ Analytics では、リザルトのテーブルをインポートするときに、フィールドを選択できるダイアログボックスの表示名の後にかっこ付きで物理フィールド名が表示されます。 <p>メモ</p> <p>リザルトの物理フィールド名は、テーブルビューの列見出しに使用される表示名ではありません。</p> <p>"リザルトデータをインポートおよびエクスポートするときのフィールド名の考慮事項" ページ 1906を参照してください。</p> <p>表示名</p>						

名前	説明
	<p>表示名は、新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)です。各 FIELD 名の表示名を指定する必要があります。表示名の値は引用符で囲まれた文字列。</p> <p>列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>Analytics の他のいくつかの IMPORT コマンドとは異なり、FIELD 名を表示名として使用する方法として、空の表示名を指定することはできません。</p>
CHARMAX 最大フィールド長 省略可能	<p>リザルトテーブルまたは解釈内の文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>リザルトテーブルまたは解釈内の最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。</p> <p>フィールドを切り詰める機能では、時折出現する長い値が原因で、インポート処理でサポートされるレコード全体の長さを超えることがなくなります。</p> <ul style="list-style-type: none"> 32,767 文字(非 Unicode Analytics) 16,383 文字(Unicode Analytics)

例

リザルトのテーブル内の指定したフィールドをインポートする

リザルトのテーブルから Analytics テーブル **T and E exceptions** に指定されたフィールドをインポートしています。

```
IMPORT GRCRESULTS TO T_and_E_exceptions "C:\Secondary Analysis\T_and_E_exceptions.fil" PASSWORD 1 FROM "results/api/orgs/11594/control_tests/185699/exceptions" FIELD "metadata.status" AS "Status" FIELD "EmpNo" AS "Employee Number" FIELD "DATE" AS "Date" FIELD "CARDNUM" AS "Card Number" FIELD "CODES" AS "MC Codes" FIELD "AMOUNT" AS "Amount" FIELD "DESCRIPTION" AS "Description"
```

リザルトのテーブル内のすべてのフィールドをインポートする

次の例では、リザルトのテーブルから Analytics テーブル **T and E exceptions** にすべてのフィールドをインポートしています。

```
IMPORT GRCRESULTS TO T_and_E_exceptions "C:\Secondary Analysis\T_and_E_exceptions.fil" PASSWORD 1 FROM "results/api/orgs/11594/control_tests/185699/exceptions"
```

リザルトのテーブルからインポートするときのフィールド長の制限

次の例では、リザルトのテーブルから Analytics テーブル **TandE exceptions** にすべてのフィールドをインポートしています。`CHARMAX` パラメーターを使用してフィールド長を制限します。

```
IMPORT GRCRESULTS TO T_and_E_exceptions "C:\Secondary Analysis\T_and_E_exceptions.fil" PASSWORD1 FROM "results/api/orgs/11594/control_tests/185699/exceptions" CHARMAX 200
```

結果として得られた Analytics テーブルでは、リザルト内の文字データから発生するすべてのフィールドは、指定した 200 文字長に制限されます。制限を超えるフィールド値は 200 文字に切り詰められます。

リザルトの解釈内のデータをインポートする

次の例では、リザルトの解釈を Analytics テーブル **T and E exceptions filtered** にインポートしています。

```
IMPORT GRCRESULTS TO T_and_E_exceptions_filtered "C:\Secondary Analysis\T_and_E_exceptions_filtered.fil" FROM "results/api/orgs/11594/control_tests/185699/interpretations/22699/exceptions"
```

備考

このコマンドの動作の詳細については、「HighBond リザルト データのインポート」ページ 764を参照してください。

並べ替え順やフィルターを保持する

リザルトからデータをインポートする場合、並べ替えやフィルターなどのデータカスタマイズは、データのインポート方法に応じて、インポート後の Analytics テーブルでは保持されるか、破棄されます。

- **テーブルのインポート** - データカスタマイズが破棄されます。省略するように選択したフィールドを除き、テーブルのすべてのデータがインポートされます。
- **解釈のインポート** - データカスタマイズが保持されます。

リザルト パス

メモ

リザルト パスの形式は API によって提供され、変更されることがあります。パスの正確な現在の構文を取得する最も簡単で信頼できる方法は、対象データの手動インポートを実行し、コマンド ログからパスをコピーすることです。

FROM パラメーターのリザルト パスは次の一般的な形式を使用します。

```
FROM "results <-地域コード>/api/orgs/<組織 ID>/control_tests/<統制テスト ID>/exceptions"
```

例：

```
FROM "results/api/orgs/11594/control_tests/4356/exceptions"
```

組織 ID は Launchpad にログインするときにブラウザーのアドレスバーに表示されます。統制テスト ID および解釈 ID は、リザルトでテーブルを表示するときにアドレスバーに表示されます。

以下の表は、リザルト パスのバリエーションを示します。

インポートするには	次の形式のリザルト パスを使用します。
統制テスト (テーブル) のデータ	<pre>FROM "results/api/orgs/11594/control_tests/4356/exceptions"</pre>
統制テスト (テーブル) の監査証跡	<pre>FROM "results/api/orgs/11594/control_tests/4356/audit_trail"</pre>
統制テスト (テーブル) のコメント	<pre>FROM "results/api/orgs/11594/control_tests/4356/comments"</pre>

<p>インポートするには</p>	<p>次の形式のリザルトパスを使用します。</p>
<p>解釈</p>	<pre>FROM "results/api/orgs/11594/control_tests/4356/interpretations/1192/exceptions"</pre>
<p>デフォルト地域 (us) 以外の HighBond 地域のデータ</p>	<ul style="list-style-type: none"> ○ アフリカ(南アフリカ) <pre>FROM "results-af/api/orgs/11594/control_tests/4356/exceptions"</pre> ○ アジア太平洋(シンガポール) <pre>FROM "results-ap/api/orgs/11594/control_tests/4356/exceptions"</pre> ○ アジア太平洋(オーストラリア) <pre>FROM "results-au/api/orgs/11594/control_tests/4356/exceptions"</pre> ○ 北米(カナダ) <pre>FROM "results-ca/api/orgs/11594/control_tests/4356/exceptions"</pre> ○ 欧州(ドイツ) <pre>FROM "results-eu/api/orgs/11594/control_tests/4356/exceptions"</pre> ○ 南米(ブラジル) <pre>FROM "results-sa/api/orgs/11594/control_tests/4356/exceptions"</pre>

システムで生成された情報列

リザルトからデータをインポートするときには、次で示すシステムで生成された情報列を1つ以上インポートすることができます。

システムで生成された列は次のいずれかです。

- リザルトのテーブルの一部であり、個々のレコードに関連する処理情報を含んでいます。
- 追加情報 - コレクション名、テーブル名、またはレコード ID 番号

以下に示すように、システムで生成された列のフィールド名を正確に指定する必要があります。Analytics のユーザーインターフェイスを使用してリザルトからインポートするときには、デフォルトの表示名が適用されます。インポート処理のスクリプトを作成する場合は、表示名を自由に変更できます。

フィールド名	デフォルトの表示名
metadata.priority	優先度
metadata.status	進捗状況
metadata.publish_date	公開済み
metadata.publisher	公開者名
metadata.assignee	割り当てられたユーザー
metadata.group	グループ
metadata.updated_at	更新
metadata.closed_at	クローズ済み
extras.collection	コレクション
extras.results_table	結果テーブル
extras.record_id	レコード ID

リザルト データをインポートおよびエクスポートするときのフィールド名の考慮事項

リザルトとAnalytics間でデータを往復する場合は、リザルト テーブルのすべてのフィールド名がより厳しいAnalyticsフィールド名要件を満たすことを確認する必要があります。そうでない場合、Analyticsとリザルトデータが一致しないおそれがあります。

たとえば、リザルトフィールド名の特殊文字は、Analyticsにインポートされるときに自動的にアンダースコアに変換されます。これは、フィールド名がリザルトの元の名前と一致しないことを意味します。後からAnalyticsデータをリザルトの元のテーブルにエクスポートする場合は、フィールドが正しく一致しません。

往復するデータでこの問題を回避するには、CSVまたはExcelファイルからリザルトにデータをアップロードする前に、以下のAnalyticsフィールド名要件を満たしていることを確認します。

- 特殊文字またはスペースがない
- 数字で始まらない
- 英数文字またはアンダースコア文字(_)のみを含む

パスワード定義の作成とパスワード値の指定

ロボットでHighBondリザルトテーブルまたは解釈をインポートするスクリプトを実行する場合、IMPORT GRCRESULTS コマンドとともにパスワード定義を含める必要があります。オフラインアクティベーションを使用する場合、同じ要件がAnalyticsで実行されるスクリプトに適用されます。

どの方法を用いてパスワード定義を作成したかにかかわらず、パスワードには値として HighBond アクセス トークンを指定する必要があります。このトークンは Launchpad で生成できます。詳細については、"HighBond アクセストークンを取得" 下を参照してください。

パスワード定義方法

方法	説明
PASSWORD アナリティクス タグ (ロボットで実行するスクリプト用)	PASSWORD アナリティクス タグを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。ロボットでスクリプトを実行するタスクを作成すると、 タスクデザイナー の入力フィールドで実際のパスワードを指定できます。 詳細については、 PASSWORD アナリティクス タグ を参照してください。
PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されません。スクリプトで接続しようすると、パスワードプロンプトが表示されます。 詳細については、"PASSWORD コマンド" ページ 2015を参照してください。
SET PASSWORD コマンド (Analytics で実行するスクリプトでは、オフラインアクティベーション)	SET PASSWORD コマンドを使用して、HighBond に接続するための番号付けされたパスワード定義を作成した場合、スクリプトでパスワードの値が指定されるため、パスワードプロンプトは表示されません。この方法は自動実行用に作成されたスクリプトに適していますが、スクリプトに実際のパスワードがクリアテキストで表示されるため、この状況には適していません。 詳細については、 SET PASSWORD コマンド を参照してください。

HighBond アクセストークンを取得

注意

生成されるアクセストークンは HighBond にサインインするために使用されるアカウントと一致します。他のユーザーが使用するスクリプトを作成する場合は、そのスクリプト内に自身のアクセストークンを指定することは不適切です。

アクセストークンはアカウントパスワードと同じように保護してください。

1. 次のいずれかを実行します。

- Analytics メインメニューで **ツール > HighBond アクセストークン** を選択します。
- **スクリプト エディター**内を右クリックし、**挿入 > HighBond トークン** を選択します。

API アクセストークンを管理します ページがお使いのブラウザで開きます。最初に HighBond にサインインすることを求められる場合があります。

ヒント

Analytics からの **API トークンの管理** ページへのアクセスが便利です。HighBond にサインインし、Analytics を使用せずにユーザープロフィールからこのページにアクセスすることもできます。

- 次のいずれかを実行します。
 - 既存のトークンを使用する** - [トークン] 列で、使用する一部がマスクされたトークンをクリックし、HighBond アカウントのパスワードを入力します。マスクされていないトークンが表示されます。

ヒント

新しいトークンを作成する理由がない場合は、既存のトークンを使用してください。既存のトークンが機能しない場合は、新しいトークンを作成してください。

既存のトークンを使用すると、管理する必要があるトークン数が減ります。

- 新しいトークンを作成する** - [トークンを作成 > Analytics] をクリックし、お使いの HighBond アカウントのパスワードを入力します。
新しい Analytics トークンが作成されます。
- [コピー] をクリックしてトークンをコピーします。

ヒント

トークンが含まれているこのダイアログボックスは、そのトークンを貼り付けるまで閉じないでください。

- 使用しているパスワード定義方法に応じて次のいずれかを実行します。
 - PASSWORD アナリティクス タグ - ACL ロボットのタスクデザイナー** で、コピーしたトークンをパスワードパラメーターフィールドに貼り付けます。
 - PASSWORD コマンド - Analytics** で、コピーしたトークンを、スクリプト実行中に表示されるパスワードの確認に貼り付けます。
 - SET PASSWORD コマンド - Analytics** で、スクリプトの SET PASSWORD コマンド構文内の適切な位置にコピーしたトークンを貼り付けます。
- Launchpad で、トークンが含まれているダイアログボックスを閉じます。
新しいトークンを作成した場合は、部分的にマスクされたバージョンのトークンが、トークンのリストに追加されます。
詳細については、[HighBond アクセストークンの作成と管理](#) を参照してください。

インポート デバッグ機能

HighBond からのインポートには、簡易 デバッグ機能があります。

インポートされたデータは、対象 Analytics プロジェクトを含むフォルダーの JSON 中間ファイルに一時的に格納されます。Analytics プロジェクトを含むフォルダーでは、データが Analytics にインポートされた後に削除する代わりに、JSON ファイルを保持するテキスト ファイルを作成できます。

- **JSON ファイルが存在する** -HighBond からのインポートが失敗し、JSON ファイルがコンピューターにある場合は、問題が HighBond 側ではなく、Analytics 側にあることがわかっています。
- **JSON ファイルが存在しない** - HighBond からのインポートが失敗し、JSON ファイルがコンピューターにない場合は、問題が HighBond 側にあることがわかっています。

この情報はトラブルシューティングで役立ちます。

JSON 中間ファイルの保持を設定する

ターゲット Analytics プロジェクトを含むフォルダーで、`_grc_import_debug.txt` という名前の空のテキストファイルを作成します。

HighBond のリザルトまたはプロジェクトからインポートするときには、JSON 中間ファイルは `results.json` という名前で保持されます。ファイルは後続の各 HighBond からのインポートするたびに上書きされます。

大きいテーブルのインポート

単一の IMPORT GRCRESULTS コマンドを使用して、多数のフィールドがあるテーブルは正常にインポートされない場合があります。リザルト外で多数のフィールドを含む単一のテーブルを操作する必要がある場合は、次のいずれかの方法を使用します。

- **テーブルの分割** -は、2 つ以上の IMPORT GRCRESULTS コマンドを使用して、フィールドのサブセットをインポートしてから、JOIN コマンドを使用して、Analytics の結果テーブルを結合します。
- **テーブルをファイルにエクスポート** -は、CSV 形式にエクスポートを使用してから、IMPORT DELIMITED コマンドを使用して、Analytics に結果ファイルをインポートします。

IMPORT LAYOUT コマンド

外部のテーブルレイアウト ファイル(.layout) を Analytics プロジェクトにインポートします。

メモ

バージョン 11 より前の Analytics では、外部テーブルレイアウト ファイルのファイル拡張子には .fmt を使用していました。拡張子を手動で指定すれば、拡張子 .fmt のテーブルレイアウト ファイルを今後もインポートすることができます。

構文

```
IMPORT LAYOUT 外部レイアウトファイル TO テーブルレイアウト名
```

パラメーター

名前	説明
外部レイアウトファイル	<p>外部テーブルレイアウト ファイルの名前。ファイル名 やパスにスペースが含まれている場合は、<code>"Ap Trans.layout"</code> のように引用符で囲む必要があります。</p> <p>ファイル拡張子 .layout はデフォルトで使用されるので、指定する必要はありません。 .fmt など、別のファイル拡張子を必要に応じて使用することもできます。</p> <p>レイアウト ファイルが Analytics プロジェクトと同じフォルダーにない場合は、ファイルの場所を指定するために絶対パス(例: <code>"C:\Saved layouts\Ap_Trans.layout"</code>) または相対パス(例: <code>"Saved layouts\Ap_Trans.layout"</code>) を使用する必要があります。</p>
TO テーブルレイアウト名	<p>インポートされたテーブルレイアウトの、Analytics プロジェクトにおける名前。たとえば、<code>"Ap Trans May"</code> と指定します。テーブルレイアウト名にスペースが含まれる場合は、引用符で囲まれた文字列として指定する必要があります。外部レイアウトファイルの名前とは異なるテーブルレイアウト名を指定することができます。</p>

例

```
外部テーブルレイアウト ファイルのインポート
```

次の例では、Analytics プロジェクトに **Ap_Trans.layout** という名前の外部テーブルレイアウト ファイルをインポートし、**Ap_Trans_May** という新しいテーブルレイアウトを作成しています。

```
IMPORT LAYOUT "C:\Saved layouts\Ap_Trans.layout" TO "Ap_Trans_May"
```

備考

IMPORT LAYOUT の使用に適する場面

外部テーブルレイアウト ファイルをインポートし、データファイルに関連付けると、新しいテーブルレイアウトをゼロから作成する手間を省くことができます。

- インポートしたテーブルレイアウトに特定の Analytics データファイル (.fil) との関連が指定されており、プロジェクトを含んでいるフォルダーに同じ名前のデータファイルが存在する場合には、フォルダー内のそのデータファイルとインポートしたテーブルレイアウトは自動的に関連付けられます。
- プロジェクト フォルダーに同じ名前のデータファイルが存在しない場合は、インポートしたテーブルレイアウトを新しいデータソースにリンクする必要があります。

テーブルレイアウトとソース データ ファイルが対応している必要性

インポートするテーブルレイアウトと、それに関連付けるデータファイルは、対応している必要があります。つまり、データファイル内のデータの構造は、テーブルレイアウトのメタデータによって規定されるフィールド定義に対応している必要があります。

データ構造は、データファイルに含まれるデータ要素(フィールド)、フィールド数やフィールドの並び順、各フィールドのデータ型や長さを示します。テーブルレイアウトとデータファイルが一致していない場合は、雑然としたデータになったり、データが欠落する結果となります。

IMPORT MULTIDELIMITED コマンド

複数の区切り文字付きファイルを定義およびインポートすることで、複数の Analytics テーブルを作成します。

構文

```
IMPORT MULTIDELIMITED <TO インポート フォルダー> FROM {ソース ファイル名|ソース フォル
ダー} ソースの文字エンコード SEPARATOR {文字|TAB|SPACE} QUALIFIER {文字|NONE}
<CONSECUTIVE> STARTLINE 行番号 <KEEPTITLE> <CRCLEAR> <LFCLEAR> <REPLACENULL>
<ALLCHAR>
```

メモ

IMPORT MULTIDELIMITED パラメーターを上記と完全に同じ順序で、以下のテーブルで指定する必要があります。

複数の区切り文字付きファイルを完全な形でインポートするには、それらすべてのファイルのインポート前の構造が同一である必要があります。

詳細については、「同一のファイル構造が必要」ページ 1918を参照してください。

パラメーター

名前	説明
TO インポート フォルダー 省略可能	<p>データをインポートするフォルダー。</p> <p>フォルダーを指定するには、絶対ファイルパス、または Analytics プロジェクトを含むフォルダーに相対的なファイルパスを使用します。インポート フォルダーは引用符で囲んで指定します。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p style="text-align: center; margin: 0;">例</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0; text-align: center;">TO "C:\Point of sale audit\Data\Transaction working data"</div> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0; text-align: center;">TO "Data\Transaction working data"</div> </div>

名前	説明
	<p><code>T0</code>を省略すると、データは Analytics プロジェクトを含むフォルダーにインポートされます。</p>
<p>FROM ソース ファイル名 ソース フォルダー</p>	<p>ソース データ ファイル、またはソース データファイルを含むフォルダーの名前。 ソース ファイル名 またはソース フォルダは引用符で囲んで指定します。 本コマンドでは、以下の4種類の区切り文字付きファイルがサポートされています。</p> <ul style="list-style-type: none"> ○ *.csv ○ *.dat ○ *.del ○ *.txt <p>メモ</p> <p>区切り文字付きファイルの64文字を超える名前は、Analytics にインポートする際に切り捨てられます。Analytics のテーブル名は、最長で64文字の英数字(拡張子.FILを含まない)に制限されます。</p> <p>また、区切り文字付きファイル名に含まれる以下の文字は、Analytics テーブル名ではアンダースコア文字(_)に置き換わります。</p> <ul style="list-style-type: none"> • アンダースコア文字以外の特殊文字 • スペース • 先頭の数字 <p>ルートの Analytics プロジェクト フォルダのソース データ ファイル</p> <p>複数のファイルを指定するには、ファイル名の中の一意の文字の代わりに、ワイルドカード文字(*)を使用します。ワイルドカード文字は、任意の文字、数字、または特殊文字のゼロ(0)回以上の出現を表します。</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center;">例</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0; text-align: center;"> <pre>FROM "Transactions_FY*.csv"</pre> </div> <p>次のファイルが選択されます。</p> <p style="color: red;">Transactions_FY18.csv</p> <p style="color: red;">Transactions_FY17.csv</p> </div> <p>ワイルドカードは、ファイル名およびファイル拡張子における複数の位置にも使用できます。</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center;">例</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0; text-align: center;"> <pre>FROM "Transactions_FY*.*"</pre> </div> </div>

名前	説明
	<div data-bbox="410 270 1414 428" style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p>次のファイルが選択されます。</p> <p>Transactions_FY18.txt</p> <p>Transactions_FY17.csv</p> </div> <p>ルートの Analytics プロジェクト フォルダー以外の場所にあるソース データ ファイル</p> <p>ソース データ ファイルが Analytics プロジェクトと同じディレクトリに位置していない場合、ファイルの位置を指定するために、絶対ファイルパス、またはプロジェクトを含むフォルダからの相対ファイルパスを使用する必要があります。</p> <div data-bbox="410 621 1414 1005" style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <h3 style="text-align: center;">例</h3> <div data-bbox="501 779 1304 873" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>FROM "C:\Point of sale audit\Data\Transaction master files\Transactions_FY*.csv"</pre> </div> <div data-bbox="501 919 1304 982" style="border: 1px solid #ccc; padding: 5px;"> <pre>FROM "Data\Transaction master files\Transactions_FY*.csv"</pre> </div> </div> <p>ソース データ ファイルを含むフォルダー</p> <p>ソース データ ファイル名を指定する代わりに、そのファイルを含むフォルダーの名前だけを指定することもできます。フォルダーに含まれるファイルのうち、サポートされている区切り文字付きファイル(*.csv、*.dat、*.del、*.txt) がすべてインポートされます。</p> <p>ソース データ フォルダーを指定するには、絶対ファイルパス、または Analytics プロジェクトを含むフォルダーに相対的なファイルパスを使用します。</p> <div data-bbox="410 1283 1414 1640" style="border: 1px solid #ccc; padding: 10px;"> <h3 style="text-align: center;">例</h3> <div data-bbox="501 1444 1304 1507" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>FROM "C:\Point of sale audit\Data\Transaction master files"</pre> </div> <div data-bbox="501 1554 1304 1617" style="border: 1px solid #ccc; padding: 5px;"> <pre>FROM "Data\Transaction master files"</pre> </div> </div>
<p>ソースの文字エンコード</p>	<p>ソース データの文字セットおよび文字エンコード。</p> <p>お使いの Analytics エディションとソース データのエンコードに応じて、次のうち該当するコードを指定してください。</p>

名前	説明															
	<table border="1" data-bbox="407 270 1409 764"> <thead> <tr> <th data-bbox="407 270 609 354">コード</th> <th data-bbox="609 270 808 354">Analytics のエディション</th> <th data-bbox="808 270 1409 354">ソースデータのエンコード</th> </tr> </thead> <tbody> <tr> <td data-bbox="407 354 609 417">0</td> <td data-bbox="609 354 808 417">非 Unicode 版</td> <td data-bbox="808 354 1409 417">すべてのデータ</td> </tr> <tr> <td data-bbox="407 417 609 480">0</td> <td data-bbox="609 417 808 480">Unicode 版</td> <td data-bbox="808 417 1409 480">ASCII データ</td> </tr> <tr> <td data-bbox="407 480 609 533">2</td> <td data-bbox="609 480 808 533">Unicode 版</td> <td data-bbox="808 480 1409 533">Unicode データ、UTF-16 LE エンコード</td> </tr> <tr> <td data-bbox="407 533 609 764">3 数値コード</td> <td data-bbox="609 533 808 764">Unicode 版</td> <td data-bbox="808 533 1409 764"> Unicode データ、UTF-16 LE エンコード ソースデータのエンコに適合する数値コードを決定するには、データ定義ウィザードを使用してインポートを実行し、エンコードされたテキスト オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、3、1 つのスペース、数値コードを順に指定します。 </td> </tr> </tbody> </table> <p data-bbox="495 806 537 831">メモ</p> <p data-bbox="495 842 1289 900">コードを指定しない場合は、非 Unicode 版の Analytics では 0 が、Unicode 版の Analytics では 2 が、それぞれ自動的に使用されます。</p>	コード	Analytics のエディション	ソースデータのエンコード	0	非 Unicode 版	すべてのデータ	0	Unicode 版	ASCII データ	2	Unicode 版	Unicode データ、UTF-16 LE エンコード	3 数値コード	Unicode 版	Unicode データ、UTF-16 LE エンコード ソースデータのエンコに適合する数値コードを決定するには、 データ定義ウィザード を使用してインポートを実行し、 エンコードされたテキスト オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、3、1 つのスペース、数値コードを順に指定します。
コード	Analytics のエディション	ソースデータのエンコード														
0	非 Unicode 版	すべてのデータ														
0	Unicode 版	ASCII データ														
2	Unicode 版	Unicode データ、UTF-16 LE エンコード														
3 数値コード	Unicode 版	Unicode データ、UTF-16 LE エンコード ソースデータのエンコに適合する数値コードを決定するには、 データ定義ウィザード を使用してインポートを実行し、 エンコードされたテキスト オプションを選択し、付属するドロップダウンリストを使って適合するエンコードを見つけます。 コードを指定するには、3、1 つのスペース、数値コードを順に指定します。														
SEPARATOR 文字 TAB SPACE	<p data-bbox="407 940 1409 999">ソースデータのフィールド間で使用される区切り文字。文字は引用符で囲まれた文字列として指定する必要があります。</p> <p data-bbox="407 1010 1409 1068">タブまたは空白を区切り文字として指定するには、それらを二重引用符で囲んで入力するか、またはそれらのキーワードを使用します。</p> <ul data-bbox="407 1079 837 1138" style="list-style-type: none"> ○ SEPARATOR " " または SEPARATOR SPACE ○ SEPARATOR " " または SEPARATOR SPACE 															
QUALIFIER 文字 NONE	<p data-bbox="407 1182 1409 1241">ソースデータ内のフィールド値を折り返すためと識別するために使用するテキスト修飾子文字。文字は引用符で囲まれた文字列として指定する必要があります。</p> <p data-bbox="407 1251 1341 1310">文字をテキスト修飾子として指定するには、二重引用符を一重引用符で囲む必要があります。 QUALIFIER ""</p> <p data-bbox="407 1320 1175 1350">テキスト修飾子がないことを指定するには、次のいずれかの方法を使用します。</p> <ul data-bbox="407 1360 605 1419" style="list-style-type: none"> ○ QUALIFIER "" ○ QUALIFIER NONE 															
CONSECUTIVE 省略可能	<p data-bbox="407 1465 972 1495">連続したテキスト修飾子を単一の修飾子として扱います。</p>															
STARTLINE 行番号	<p data-bbox="407 1570 591 1600">データが始まる行。</p> <p data-bbox="407 1610 1393 1669">たとえば、データの最初の 4 行にヘッダー情報が含まれており、それらをインポートの対象外とする場合は、行番号「5」を指定します。</p> <p data-bbox="495 1711 537 1736">メモ</p> <p data-bbox="495 1747 1321 1806">IMPORT MULTIDELIMITED を 1 回実行することでインポートするすべての区切り文字付きファイルにおいて、データの開始行を同じにすることをお勧めします。</p> <p data-bbox="495 1816 1326 1875">異なる開始行にする場合は、「同一のファイル構造が必要」ページ 1918 を参照してください。</p>															

名前	説明
KEEPTITLE 省略可能	<p>STARTLINE に指定する行番号をデータでなくフィールド名と見なします。KEEPTITLE を省略すると、汎用フィールド名が使用されます。</p> <p>メモ IMPORT MULTIDELIMITED を 1 回実行することでインポートするすべての区切り文字付きファイルにおいて、各フィールド名が同じ行番号になければなりません。各フィールド名の行番号が異なる場合は、"同一のファイル構造が必要" ページ 1918 を参照してください。</p>
CRCLEAR 省略可能	<p>テキスト修飾子間で発生するすべての CR 文字(キャリッジリターン)をスペース文字で置換します。CRCLEAR を使用するには、QUALIFIER と文字値を指定する必要があります。</p> <p>CRCLEAR と LFCLEAR の両方を使用する場合は、CRCLEAR を最初に記述する必要があります。</p>
LFCLEAR 省略可能	<p>テキスト修飾子間で発生するすべての LF 文字(ラインフィード)をスペース文字で置換します。LFCLEAR を使用するには、QUALIFIER と文字値を指定する必要があります。</p> <p>CRCLEAR と LFCLEAR の両方を使用する場合は、CRCLEAR を最初に記述する必要があります。</p>
REPLACENULL 省略可能	<p>スペースで区切られたファイルで発生するすべての NUL 文字を置換します。置き換えられたすべての NULL 文字の数がログに記録されます。</p>
ALLCHAR 省略可能	<p>インポートされたすべてのフィールドには、自動的に文字のデータ型が割り当てられます。</p> <p>ヒント インポートされたすべてのフィールドに文字のデータ型を割り当てると、区切り文字付きテキスト ファイルのインポート 処理が容易になります。Analytics にインポートされたデータのフィールドには、数値や日付時刻などのさまざまなデータ型を割り当て、書式の詳細を指定することができます。</p> <p>Analytics によって識別子のフィールドに数値のデータ型が自動的に割り当てられたテーブルをインポートする際、実際には文字のデータ型を使用する必要がある場合には、ALLCHAR を使用することができます。</p>

例

後続の例では、次のような 12 個の区切り文字付きファイルに保存された月次取引データがある場合を想定しています。

- [Transactions_Jan.csv](#) ~ [Transactions_Dec.csv](#)

メモ

インポートする区切り文字付きファイルごとに、独立した Analytics テーブルが作成されず。

すべての区切り文字付きファイルをインポートする

12個すべての区切り文字付きファイルをインポートするとします。各ファイル名に含まれる月名の位置で、ワイルドカード記号 (*) を使用します。

Analytics は各フィールドに適切なデータ型を割り当てようとします。

```
IMPORT MULTIDELIMITED FROM "Transactions_*.csv" @ SEPARATOR ","  
QUALIFIER ''' CONSECUTIVE STARTLINE 1 KEPTITLE
```

すべての区切り文字付きファイルを文字型データとしてインポートする

この例は上の例と似ていますが、相違点は、インポートされるすべてのフィールドに対して Analytics により自動的に文字のデータ型が割り当てられる点です。

```
IMPORT MULTIDELIMITED FROM "Transactions_*.csv" @ SEPARATOR ","  
QUALIFIER ''' CONSECUTIVE STARTLINE 1 KEPTITLE ALLCHAR
```

指定されたフォルダーからすべての区切り文字付きファイルをインポートする

C:\Point of sale audit\Data\Transaction master files フォルダーにあるすべての区切り文字付きファイルをインポートします。

```
IMPORT MULTIDELIMITED FROM "C:\Point of sale audit\Data\Transaction  
master files" @ SEPARATOR "," QUALIFIER ''' CONSECUTIVE STARTLINE 1  
KEPTITLE
```

指定されたフォルダーにあるすべての区切り文字付きファイルをインポートし、Analytics テーブルを別のフォルダーに保存

この例は上記の例と同じですが、Analytics テーブルをルート プロジェクト フォルダーに保存するのではなく、`C:\Point of sale audit\Data\Transaction working data` フォルダーに保存します。

```
IMPORT MULTIDELIMITED TO "C:\Point of sale audit\Data\Transaction
working data" FROM "C:\Point of sale audit\Data\Transaction master
files" 0 SEPARATOR "," QUALIFIER "'" CONSECUTIVE STARTLINE 1 KEPTITLE
```

備考

同一のファイル構造が必要

IMPORT MULTIDELIMITED を使って複数の区切り文字付きファイルを完全な形でインポートするには、それらすべてのファイルの構造が同一である必要があります。

異なる構造の区切り文字付きファイルをインポートした場合でも、後から Analytics でデータをクレンジングおよび正規化することもできます。ただし、このように行くと、より多くの作業が必要となる可能性があります。通常、これより簡単なのは、区切り文字付きファイルをインポートする前にそれらを同じ構造にしておくことです。

複数の区切り文字付きファイルを完全な形でインポートするには、以下の項目がすべてのファイルにおいて同じである必要があります。

項目	ACLScript のキーワード	問題	解決策
ソースデータの文字セットおよび文字エンコード	数値コード	(Analytics の Unicode 版のみ) ソースの区切り文字付きファイルに応じて、異なる文字エンコードが使用されています。たとえば、ASCII エンコードを使用しているファイルもあれば、Unicode エンコードを使用しているファイルもあります。	エンコード タイプ別にソース ファイルをグループ化し、各グループを別々にインポートします。
区切り文字	SEPARATOR	ソースの区切り文字付きファイルに応じて、フィールド間に異なる区切り文字が使用されています。	次のいずれかを実行します。 <ul style="list-style-type: none"> ソース ファイルをインポートする前に、それらの中の区切り文字を正規化します。

項目	ACLScript のキーワード	問題	解決策
			<ul style="list-style-type: none"> 区切り文字別にソース ファイルをグループ化し、各グループを別々にインポートします。
テキスト 修飾子文字	QUALIFIER	ソースの区切り文字付きファイルに応じて異なるテキスト 修飾子文字が使用されることで、フィールド値が折り返されるとともに識別されています。	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> ソース ファイルをインポートする前に、その中の修飾子文字を正規化します。 修飾子文字別にソース ファイルをグループ化し、各グループを別々にインポートします。
データの開始行	STARTLINE	ソースの区切り文字付きファイルごとにデータの開始行が異なります。	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> ソース ファイルをインポートする前に、それらの中の開始行を正規化します。 同じ開始行を持つソース ファイルをグループ化し、各グループを別々にインポートします。 すべてのファイルのうちで最も遅い開始行を行番号に指定してください。ファイルを Analytics テーブルにインポートしたら、"EXTRACT コマンド" ページ 1801を使用することで、任意のテーブルから、不要としたヘッダー情報が含まれたレコードだけを抽出できます。
フィールド名	KEEPTITLE	ソースの区切り文字付きファイルに応じて、各フィールド名が異なる行番号にあります。	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> ソース ファイルをインポートする前に、それらの中のフィールド名がある行番号を正規化します。 同じフィールド名が同じ行番号にあるソース ファイルをグループ化し、各グループを別々にインポートします。
フィールド名	KEEPTITLE	ソースの区切り文字付きファイルによって、フィールド名があるファイルと、フィールド名がないファイルがあります。	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> すべてのファイルをインポートする前に、フィールド名が必要なソース ファイルにフィールド名を追加します。 フィールド名のあるソース ファイルをグループ化するとともに、フィールド名のないソース ファイルもグループ化します。次いで、各グループを別々にインポートします。 汎用フィールド名を使ってすべてのファイルをインポートするには、KEEPTITLE を省略します。ファイルを Analytics テーブルにインポートしたら、"EXTRACT コマンド" ページ

項目	ACLScript のキーワード	問題	解決策
			1801を使用することで、任意のテーブルから必要なデータだけを抽出できます。

複数の IMPORT DELIMITED コマンド

IMPORT MULTIDELIMITED コマンドは、インポートされるファイルごとに実際に個別の IMPORT DELIMITED コマンドを 1 つずつ実行するものです。ログの IMPORT MULTIDELIMITED エントリをダブルクリックすると、個別の IMPORT DELIMITED コマンドが表示領域に表示されます。

複数の区切り文字付きファイルをインポート後に結合する

複数の区切り文字付きファイルを個々の Analytics テーブルにインポート後に、それらを 1 つの Analytics テーブルに結合できます。たとえば、12 個の月次テーブルのデータをすべてのデータを含む単一の年次テーブルに結合できます。

複数の Analytics テーブルを結合する方法については、「APPEND コマンド」ページ 1651 を参照してください。

IMPORT MULTIEXCEL コマンド

複数の Microsoft Excel ワークシートまたは名前付き範囲を定義およびインポートして、複数の Analytics テーブルを作成します。

構文

```
IMPORT MULTIEXCEL <TO インポート フォルダー> FROM {ソース ファイル名 | ソース フォルダー}
TABLE 入力ワークシートまたは名前付き範囲 <PREFIX> <KEEPTITLE> <CHARMAX 最大フィールド長>
```

メモ

IMPORT MULTIEXCEL パラメーターを上記と完全に同じ順序で、以下のテーブルで指定する必要があります。

ワークブックの保護ビューが有効な場合、analytics は Excel ワークブックからインポートできません。まず、ワークブックで編集を有効にし、ワークブックを保存して閉じてから、インポートを実行する必要があります。

パラメーター

名前	説明
TO インポート フォルダー 省略可能	<p>データをインポートするフォルダー。</p> <p>フォルダーを指定するには、絶対ファイルパス、または Analytics プロジェクトを含むフォルダーに相対的なファイルパスを使用します。インポート フォルダーは引用符で囲んで指定します。</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center; margin: 0;">例</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0; width: fit-content; margin-left: auto; margin-right: auto;"> <p>TO "C:\Point of sale audit\Data\Transaction working data"</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0; width: fit-content; margin-left: auto; margin-right: auto;"> <p>TO "Data\Transaction working data"</p> </div> </div> <p>TO を省略すると、データは Analytics プロジェクトを含むフォルダーにインポートされます。</p>

名前	説明
FROM ソース ファイル名 ソース フォル ダー	<p>ソース データ ファイル、またはソース データファイルを含むフォルダーの名前。 ソース ファイル名 またはソース フォルダーは引用符で囲んで指定します。</p> <p>ルートの Analytics プロジェクト フォルダーのソース データ ファイル</p> <ul style="list-style-type: none"> 単一の Excel ファイル Excel ファイルの完全なファイル名と拡張子を指定します。 <div data-bbox="410 543 1412 793" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center; font-weight: bold;">例</p> <div style="border: 1px solid #ccc; padding: 5px; text-align: center; margin: 10px auto; width: 80%;"> FROM "Transactions_FY18.xlsx" </div> </div> <ul style="list-style-type: none"> 複数の Excel ファイル 複数のファイルを指定するには、ファイル名の中の一意の文字の代わりに、ワイルドカード文字 (*) を使 用します。ワイルドカード文字は、任意の文字、数字、または特殊文字のゼロ (0) 回以上の出現を表 します。 <div data-bbox="410 995 1412 1398" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center; font-weight: bold;">例</p> <div style="border: 1px solid #ccc; padding: 5px; text-align: center; margin: 10px auto; width: 80%;"> FROM "Transactions_FY*.xlsx" </div> <p style="margin-top: 10px;">次のファイルが選択されます。</p> <p style="margin-left: 20px; color: red;">Transactions_FY18.xlsx</p> <p style="margin-left: 20px; color: red;">Transactions_FY17.xlsx</p> </div> <p>ワイルドカードは、ファイル名およびファイル拡張子における複数の位置にも使用できます。</p> <div data-bbox="410 1488 1412 1831" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p style="text-align: center; font-weight: bold;">例</p> <div style="border: 1px solid #ccc; padding: 5px; text-align: center; margin: 10px auto; width: 80%;"> FROM "Transactions_FY*.*" </div> <p style="margin-top: 10px;">次のファイルが選択されます。</p> <p style="margin-left: 20px; color: red;">Transactions_FY18.xlsx</p> </div>

名前	説明
	<div data-bbox="410 273 1414 342" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p style="text-align: center; color: #c00000;">Transactions_FY17.xls</p> </div> <p>ルートの Analytics プロジェクト フォルダのソース データ ファイル</p> <p>ソース データ ファイルまたはファイルが Analytics プロジェクトと同じディレクトリに位置しない場合、ファイルの位置を指定するために絶対ファイルパスまたはプロジェクトを含むフォルダへの相対ファイルパスを使用する必要があります。</p> <div data-bbox="376 567 1414 951" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p style="text-align: center; font-weight: bold; font-size: 1.2em;">例</p> <div data-bbox="467 726 1304 821" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>FROM "C:\Point of sale audit\Data\Transaction master files\Transactions_FY18.xlsx"</pre> </div> <div data-bbox="467 863 1304 930" style="border: 1px solid #ccc; padding: 5px;"> <pre>FROM "Data\Transaction master files\Transactions_FY*.xlsx"</pre> </div> </div> <p>ソース データ ファイルを含むフォルダ</p> <p>ファイル名を指定する代わりに、ソース データ ファイルを含むフォルダの名前を指定できます。ソース データ フォルダを指定するには、絶対ファイルパス、または Analytics プロジェクトを含むフォルダに相対的なファイルパスを使用します。</p> <div data-bbox="376 1199 1414 1556" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p style="text-align: center; font-weight: bold; font-size: 1.2em;">例</p> <div data-bbox="467 1360 1304 1428" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>FROM "C:\Point of sale audit\Data\Transaction master files"</pre> </div> <div data-bbox="467 1470 1304 1537" style="border: 1px solid #ccc; padding: 5px;"> <pre>FROM "Data\Transaction master files"</pre> </div> </div> <div data-bbox="423 1598 1312 1692" style="margin-top: 10px;"> <p>メモ</p> <p>フォルダを指定すると、ワークシート名が TABLE 値と一致する場合に、フォルダの任意の Excel ファイルのワークシートがインポートされます。</p> </div>
<p>TABLE 入力 ワークシート または名前 付き範囲</p>	<p>インポートするワークシートまたは名前付き範囲の名前。インポートされた各ワークシートまたは名前付き範囲の個別の Analytics テーブルが作成されます。</p> <p><i>input_worksheets_or_named_ranges</i> は引用符で囲んで指定します。</p> <p>ワークシートまたは範囲の名前で、固有の文字の代わりに、ワイルドカード (*) を使用します。</p>

名前	説明
	<p>たとえば、"Trans_*\$" は、次のワークシートを選択します。</p> <ul style="list-style-type: none"> ○ Trans_Jan ○ Trans_Feb ○ Trans_Mar ○ など <p>メモ</p> <p>ワイルドカード文字 (*) は、任意の文字、数字、または特殊文字のゼロ (0) 回以上の出現を表します。</p> <p>複数の場所でワイルドカードを使用できます。たとえば、*Trans*\$ は以下を選択します。</p> <ul style="list-style-type: none"> • Trans_Jan • Jan_Trans <p>ドル記号 (\$) の意味</p> <p>Excel ファイルでは、ワークシートは、ワークシート名 (Trans_Jan\$) に付加されているドル記号 (\$) で識別されます。ドル記号は Excel には表示されません。</p> <p>名前付き範囲は、ドル記号がないことによって特定されます (Trans_Jan_commercial)。</p> <p>IMPORT MULTIEXCEL を使用するときには、ドル記号の指定は不要です。ただし、次の状況では、それを含めるか、除外してください。</p> <ul style="list-style-type: none"> ○ "\$" を含める - ワークシートのみをインポートし、名前付き範囲をインポートしない場合は、ワークシート名の最後にドル記号 (\$) を入れます。 ○ "\$" を除外する - 名前付き範囲またはワークシートと名前付き範囲を 1 つのインポート処理でインポートする場合は、ドル記号を入れません。
<p>PREFIX 省略可能</p>	<p>Excel ファイルを Analytics テーブルの名前の前に追加します。</p> <p>ヒント</p> <p>別のファイルのワークシートの名前が同じ場合は、Excel ファイル名を前に付けると、テーブル名の競合を回避できます。</p>
<p>KEEPTITLE 省略可能</p>	<p>データの代わりに、フィールド名としてデータの最初の行を処理します。省略すると、汎用フィールド名が使用されます。</p> <p>メモ</p> <p>インポートするワークシートと名前付き範囲のすべての最初の行は、一貫性がある方法を使用してください。最初の行は、すべてのデータセットで、フィールド名、データのいずれかです。1 つのインポート処理で 2 つの内容を混在させないでください。</p> <p>データセットの最初の行へのアプローチに一貫性がない場合、2 つの別のインポート処理を使用してください。</p>
<p>CHARMAX 最大フィールド長 省略可能</p>	<p>Excel ソースデータファイル内の文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>Excel ファイル内の最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。</p> <p>フィールドを切り詰める機能では、時折出現する長い値が原因で、インポート処理でサポートされるレコード全体の長さを超えることがなくなります。</p> <ul style="list-style-type: none"> ○ 32,767 文字 (非 Unicode Analytics)

名前	説明
	<ul style="list-style-type: none"> 16,383 文字 (Unicode Analytics)

例

次の例は、Excel ファイルに保存された 3 年間分の月次取引データがあるとしてします。

- [Transactions_FY18.xlsx](#)
- [Transactions_FY17.xlsx](#)
- [Transactions_FY16.xlsx](#)

各 Excel ファイルには 12 ワークシート (年の各月に 1 つずつ)。また、ワークシートには、取引の各種サブセットを識別する一部の名前付き範囲もあります。

メモ

インポートする各ワークシートまたは名前付き範囲の個別の Analytics テーブルが作成されます。

ワークシートのインポート

すべての FY18 ワークシートのインポート

FY18 Excel ファイルから、すべての 12 ヶ月のワークシートをインポートします。

- 各ワークシート名で月が発生する場所で、ワイルドカード記号 (*) を使用します。
- ワークシート名の最後にドル記号 (\$) を入れ、ワークシートのみが選択され、名前付き範囲が選択されないようにします。

```
IMPORT MULTIEXCEL FROM "Transactions_FY18.xlsx" TABLE "Trans_*$"
```

すべての FY18 ワークシートをインポートし、フィールド名を保持し、最大文字フィールド長を指定します。

この例は上記の例と同じですが、Excel ファイルのフィールド名を保持し、文字フィールドの長さを制限します。

- KEeptitle を追加すると、Excel データの最初の行をフィールド名として使用できます。
- CHARMAX 50 を追加し、Excel ファイルの文字データとして発生するフィールドが、結果の Analytics テーブルで 50 文字に制限されます。

```
IMPORT MULTIEXCEL FROM "Transactions_FY18.xlsx" TABLE "Trans_*$"
KEeptitle CHARMAX 50
```

すべての3つのファイルからすべてのワークシートをインポート

3つのExcelファイルから、すべての36ヶ月のワークシートをインポートします。

- 各ワークシート名で月が発生する場所で、ワイルドカード記号 (*) を使用します。
- ワークシート名の最後にドル記号 (\$) を入れ、ワークシートのみが選択され、名前付き範囲が選択されないようにします。
- 各Excelファイル名で年が発生する場所で、ワイルドカード記号 (*) を使用します。
- 命名の競合の可能性を削減する方法として、PREFIX を使用して、ソースExcelファイルの名前を各Analyticsテーブル名の前に追加します。

```
IMPORT MULTIEXCEL FROM "Transactions_FY18.xlsx" TABLE "Trans_*$"
```

名前付き範囲のインポート

すべてのFY18 "Commercial_transaction" 名前付き範囲のインポート

すべての"Commercial_transaction" 名前付き範囲をFY18 Excelファイルからインポートし、ワークシートと他の名前付き範囲を無視します。

- 別の範囲の名前で一意の識別子が発生する場所でワイルドカード記号 (*) を使用します。
- ドル記号 (\$) を除外し、名前付き範囲が選択できるようにします。

```
IMPORT MULTIEXCEL FROM "Transactions_FY18.xlsx" TABLE "Commercial_
transaction_*
```

ワークシートと名前付き範囲のインポート

すべてのFY18 ワークシートと名前付き範囲のインポート

FY18 Excel ファイルから、すべての 12 ヶ月のワークシートと他の名前付き範囲をインポートします。

- TABLE でワイルドカード記号 (*) のみを使用して、ファイルのすべてのワークシートと名前付き範囲が選択されるようにします。
- ドル記号 (\$) を除外し、名前付き範囲が選択できるようにします。

```
IMPORT MULTIEXCEL FROM "Transactions_FY18.xlsx" TABLE "**"
```

ディレクトリの管理

指定されたフォルダーですべての Excel ファイルからすべてのワークシートをインポート

C:\Point of sale audit\Data\Transaction master files フォルダーで、すべての Excel ファイルからすべてのワークシートをインポートします。

- TABLE でワイルドカード記号 (*) のみを使用して、各ファイルのすべてのワークシートが選択されるようにします。また、ドル記号 (\$) を使用して、ワークシートのみが選択され、名前付き範囲が選択されないようにします。
- 命名の競合の可能性を削減する方法として、PREFIX を使用して、ソース Excel ファイルの名前を各 Analytics テーブル名の前に追加します。

```
IMPORT MULTIEXCEL FROM "C:\Point of sale audit\Data\Transaction master files" TABLE "$" PREFIX
```

指定されたフォルダーですべての Excel ファイルからすべてのワークシートをインポートし、Analytics テーブルを別のフォルダーに保存

この例は上記の例と同じですが、Analytics テーブルをルート プロジェクト フォルダーに保存するのではなく、`C:\Point of sale audit\Data\Transaction working data` フォルダーに保存します。

```
IMPORT MULTIEXCEL TO "C:\Point of sale audit\Data\Transaction working data" FROM "C:\Point of sale audit\Data\Transaction master files"
TABLE "*" PREFIX
```

備考

複数の IMPORT EXCEL コマンド

IMPORT MULTIEXCEL コマンドは、実際に、複数の個別の IMPORT EXCEL コマンドを実行します。インポートされる各ワークシートに1つあります。ログの IMPORT MULTIEXCEL エントリをダブルクリックすると、個別の IMPORT EXCEL コマンドが表示領域に表示されます。

インポートされた最後のテーブルは、自動的に開きます。

IMPORT MULTIEXCEL は OPEN キーワードをサポートしません。ただし、コマンドの実行後、インポートされた最後のテーブルが自動的に開きます。

インポート後に複数のワークシートを結合

複数のワークシートを個々の Analytics テーブルにインポート後に、それらを1つの Analytics テーブルに結合できます。たとえば、12 個の月次テーブルのデータをすべてのデータを含む単一の年次テーブルに結合できます。

複数の Analytics テーブルを結合する方法については、「APPEND コマンド」ページ 1651 を参照してください。

IMPORT ODBC コマンド

ODBC データソースからデータを定義およびインポートし、Analytics テーブルを作成します。

ODBC は Open Database Connectivity であり、データベースにアクセスするための標準的な方法です。

構文

```
IMPORT ODBC SOURCE ソース名 TABLE テーブル名 <QUALIFIER データ修飾子> <OWNER ユーザー名> <USERID ユーザー ID> <PASSWORD 数値> <WHERE WHERE句> <TO テーブル名> <WIDTH 最大フィールド幅> <MAXIMUM 最大フィールド長> <FIELDS フィールド <,...n>>
```

パラメーター

名前	説明
SOURCE ソース名	<p>接続する ODBC データソースのデータソース名 (DSN)。DSN は既に存在し、正しく構成されている必要があります。</p> <p>メモ コンピューターにインストールされている Windows ODBC ドライバーを使用するデータソースに制限されています。ACCESSDATA コマンドで使用できる Analytics ネイティブ データ コネクタは IMPORT ODBC で使用できない場合があります。</p>
TABLE テーブル名	<p>データをインポートする ODBC データソースのテーブル名。</p> <p>通常、テーブル名はソース データのデータベース テーブルを参照しますが、Analytics がインポートするものは何でもテーブルとして参照することができます。たとえば、Microsoft Text Driver を使用するならば、テーブル名は、データをインポートするテキスト ファイルを参照します。</p>
QUALIFIER データ修飾子 省略可能	<p>フィールド値を折り返すためと識別するためにテキスト 修飾子として使用する文字。文字は引用符で囲まれた文字列として指定する必要があります。</p> <p>引用符を使用して、二重引用符文字を指定します: ""。</p>
OWNER ユーザー名 省略可能	<p>接続するテーブルを所有するデータベース ユーザー アカウントの名前。</p>
USERID ユーザー ID 省略可能	<p>データソースにアクセスするために必要なユーザー名。</p>
PASSWORD 番号	<p>使用するパスワード定義。</p>

名前	説明
省略可能	<p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前にPASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code> により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ
WHERE WHERE 句 省略可能	<p>指定した条件に基づいて返されたレコードを制限する SQL WHERE 句。有効な SQL ステートメントである必要があるほか、引用符で囲んだ文字列として入力する必要があります。</p> <pre data-bbox="565 730 1344 793">WHERE "SALARY > 50000"</pre>
TO テーブル名 省略可能	<p>作成する Analytics テーブルの名前。</p> <p>テーブル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例：<code>TO "Invoices.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ◦ <code>TO "C:\data\Invoices.FIL"</code> ◦ <code>TO"data\Invoices.FIL"</code> <p>メモ</p> <p>Analytics のテーブル名は、最長で 64 文字の英数字(拡張子 .FIL を含まない)に制限されます。名前にはアンダースコア文字(<code>_</code>)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
WIDTH 最大フィールド長 省略可能	<p>インポートしているソースの文字データから発生する Analytics テーブルの任意のフィールドの文字の最大長。</p> <p>1 から 254 までの値を入力することができます。デフォルト値は 50 です。最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。</p>
MAXIMUM 最大フィールド長 省略可能	<p>インポートするテキスト、ノート、またはメモフィールドの文字の最大長。</p> <p>1 から 1100 までの値を入力することができます。デフォルト値は 100 です。最大フィールド長を超えるデータは、Analytics にインポートすると切り捨てられます。</p>
FIELDS <i>field</i> <,... <i>n</i> > 省略可能	<p>インポートするソースデータの個別のフィールド。名前を指定します。</p> <p>複数のフィールドを指定する場合は、カンマによってそれぞれのフィールドを区切る必要があります。FIELDS を省略すると、すべてのフィールドがインポートされます。</p> <p>フィールド名を引用符で囲むと、大文字と小文字が区別されるようになります。引用符を使用する場合は、フィールド名の大文字小文字は、FIELDS と ODBC データソースとの間で正</p>

名前	説明
	<p>確に一致する必要があります。引用符を使用した場合、フィールド名の大文字小文字が一致しないフィールドはインポートされません。</p> <p>メモ FIELDS は、IMPORT ODBC パラメーターの最後に置く必要があります。FIELDS が最後でない場合は、コマンドが失敗します。</p>

例

SQL Server からデータをインポートする

次の例では、SQL Server データベースから Trans_Dec11 という名前の Analytics テーブルにデータをインポートしています。

```
IMPORT ODBC SOURCE "SQLServerAudit" TABLE "Transactions" OWNER "audit"
TO "C:\ACL DATA\Trans_Dec11.FIL" WIDTH 100 MAXIMUM 200 FIELDS
"CARDNUM", "CREDLIM", "CUSTNO", "PASTDUEAMT"
```

備考

ODBC データソースに接続するための古い方法

IMPORT ODBC コマンドは、Analytics から ODBC 準拠のデータソースに接続するための古い方法です。ODBC データソースに接続するための新しい方法では、Data Access ウィンドウと ACCESSDATA コマンドを使用します。

IMPORT ODBC は引き続き Analytics で使用できます。ただし、この接続方法はスクリプトと Analytics コマンドラインでしか使用できなくなりました。この接続方法にデータ定義ウィザードからアクセスすることはできなくなりました。

日付時刻値の時刻部分を非表示にする

IMPORT ODBC コマンドを使用して Analytics テーブルを定義する場合は、このコマンドの前に SET SUPPRESSTIME ON コマンドを置くことにより、日付時刻値の時刻部分を非表示にすることができます。

この機能により、日付時刻値の時刻部分が自動的に切り捨てられていた、バージョン 10.0 より前の Analytics で作成された Analytics スクリプトを組み込むことができます。それらのスクリプトは、SET SUPPRESSTIME ON を追加しない場合には、日付時刻型対応バージョンの Analytics では動作しません。

詳細については、「SET コマンド」 ページ 2081の「SET SUPPRESSTIME」セクションを参照してください。

IMPORT PDF コマンド

Adobe PDF ファイルを定義およびインポートして、Analytics テーブルを作成します。

構文

```
IMPORT PDF TO table <PASSWORD 番号> インポート ファイル名 FROM ソース ファイル名
<SERVER プロファイル名> スキップ長さ <PARSER "VPDF"> <PAGES ページ範囲> {[レコード構文]
[フィールド構文] <...n>} <...n>
```

```
レコード構文 ::=
RECORD レコード名 レコード タイプ レコードの行 透明 [テスト構文] <...n>
```

```
テスト構文 ::=
TEST 含める/除外する 一致タイプ AT 開始行,開始位置,範囲 論理 テキスト
```

```
フィールド構文 ::=
FIELD 名前 型 AT 開始行,開始位置 SIZE 長さ,フィールドの行 DEC 値 WID バイト PIC 形式 AS 表示名
```

パラメーター

一般パラメーター

名前	説明
TO テーブル	データをインポートする Analytics テーブルの名前。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> <p>メモ</p> <p>テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> </div>
PASSWORD 番号	パスワードで保護された PDF ファイルのパスワードに使用します。

名前	説明
省略可能	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前にPASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、PASSWORD 2により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> ◦ "PASSWORD コマンド" ページ 2015 ◦ SET PASSWORD コマンド ◦ PASSWORD アナリティクス タグ
インポートファイル名	<p>作成する Analytics データファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL"</p> <p>デフォルトでは、データファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ◦ "C:\data\Invoices.FIL" ◦ "data\Invoices.FIL"
FROM ソースファイル名	<p>ソースデータファイルの名前。ソースファイル名は引用符で囲む必要があります。</p> <p>ソースデータファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> ◦ "C:\data\ソース ファイル名" ◦ "data\ソース ファイル名"
SERVER プロファイル名 省略可能	<p>インポートするデータを含むサーバーのプロファイル名。</p>
スキップする長さ 省略可能	<p>ファイルの先頭でスキップするバイト数。</p> <p>たとえば、先頭から 32 バイトにヘッダー情報が含まれている場合、この情報を除外するには、スキップする長さの値として 32 を指定します。</p> <p>メモ Unicode データでは、偶数バイトのみを指定します。奇数のバイトを指定すると、インポートされたデータの後續の処理で問題が発生する可能性があります。</p>
PARSER "VPDF" 省略可能	<p>ファイル定義の処理時に VeryPDF パーサーを使用して、PDF ファイルを解析します。</p> <p>PARSER を省略した場合は、デフォルトの Xpdf パーサーが使用されます。</p> <p>初めて PDF ファイルをインポートし、特に理由がない場合は、デフォルトの Xpdf パーサーを使用してください。Xpdf を使用して、既にデータの整列に問題が発生している場合は、VeryPDF パーサーを使用して解析結果が改善されるか確認してください。</p>

名前	説明
PAGES ページの範囲 省略可能	<p>PDF ファイル内のすべてのページをインポートしない場合の、含めるページの範囲。ページの範囲は、引用符で囲んだ文字列として指定する必要があります。</p> <p>以下を指定できます。</p> <ul style="list-style-type: none"> カンマで区切った個々のページ(1,3,5) ページの範囲(2-7) ページと範囲の組み合わせ(1, 3, 5-7, 11) <p>PAGES を省略する場合は、PDF ファイルのすべてのページがインポートされます。</p>

RECORD パラメーター

一般レコード定義情報。

メモ

一部のレコード定義情報は、データ定義ウィザードのオプションにマッピングされる数値コードを使用して指定されます。

スクリプトでは、オプション名ではなく、数値コードを指定します。

名前	説明
RECORD レコード名	<p>データ定義ウィザードのレコード名。</p> <p>IMPORT PDF コマンドでは、レコード名を指定する必要がありますが、レコード名値は結果の Analytics テーブルに表示されません。</p> <p>データ定義ウィザードでは、Analytics は、レコードのタイプに基づいて、デフォルト名を提供します。</p> <ul style="list-style-type: none"> 詳細 ヘッダーn フッターn <p>デフォルト名を指定するか、別の名前を指定できます。</p>
レコードのタイプ	<p>PDF ファイルを定義するときには、3つのレコードタイプがあります。</p> <ul style="list-style-type: none"> 0- 詳細 1- ヘッダー 2- フッター <p>メモ 単一の IMPORT PDF の実行では、複数のセットのヘッダーおよびフッターレコードを定義できますが、詳細レコードは1セットのみです。</p>
レコードの行	<p>PDF ファイルのレコードが占める行数。</p> <p>PDF ファイルのデータと一致する単一行または複数行のレコードを定義します。</p>
透明	<p>ヘッダーレコードの透明設定。</p>

名前	説明
	<p>メモ ヘッダーレコードにのみ適用されます。</p> <ul style="list-style-type: none"> 0 - 不透明 1 - 透明 <p>透明なヘッダーレコードは複数行の詳細レコードに分割されません。</p> <p>ヘッダーレコードがソース PDF ファイルの複数行の詳細レコードを分割する場合 (改ページで発生する場合があります)、 1 (透明) を指定すると、結果の Analytics テーブルの詳細レコードが統合されます。</p>

TEST パラメーター

PDF ファイルのレコードのセットを定義する条件。RECORD の各出現には、1 つ以上の TEST (8 つまで) があります。

メモ

一部の条件は、データ定義ウィザードのオプション (オプション名は以下で括弧内に示されています) にマッピングされる数値コードを使用して指定されます。

スクリプトでは、オプション名ではなく、数値コードを指定します。

名前	説明
TEST 含める/除外する	<p>一致するデータを処理する方法:</p> <ul style="list-style-type: none"> 0 - (含める) 条件を満たすデータがレコードのセットに含まれます 1 - (除外する) 条件を満たすデータがレコードのセットから除外されます
一致タイプ	<p>実行する一致のタイプ:</p> <ul style="list-style-type: none"> 0 - (正確に一致する) 一致するレコードには、指定された文字または文字列が、指定された位置で始まる、指定された開始行に含まれている必要があります。 2 - (英文字) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある 1 つ以上の英文字を含んでいる必要があります。 3 - (数字) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある 1 つ以上の数字を含んでいる必要があります。 4 - (空白) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある 1 つ以上の空白を含んでいる必要があります。 5 - (空白以外) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある 1 つ以上の空白以外の文字 (特殊文字を含む) を含んでいる必要があります。 7 - (行で検索) 一致するレコードには、指定された文字または文字列が、指定された位置で始まる、指定された開始行に含まれている必要があります。 8 - (範囲で検索) 一致するレコードには、[テキスト] フィールドにある文字または文字列が、指定された範囲の、レコードの指定された行の指定された範囲のどこかに含まれている必要があります。 10 - (独自のマップ) 一致するレコードには、指定された文字パターンと一致する文字が、指定された位置で始まる、指定された開始行に含まれている必要があります。

名前	説明				
AT 開始行, 開始位置, 範囲	<ul style="list-style-type: none"> ○ 開始行 -条件が適用されるレコードの行 たとえば、独自のマップを使用して郵便番号と一致させ、郵便番号が3行レコードの3行目に表示される場合は、開始行]で③を指定する必要があります。 <p>メモ 単一行のレコードの場合、開始行の値は常に①です。</p> <ul style="list-style-type: none"> ○ 開始位置 -条件と比較する PDF ファイルの開始バイト位置 ○ 範囲 -条件との比較で使用する PDF ファイルの開始バイト位置からのバイト数 範囲なしで開始バイト位置のみを使用している場合は、範囲に④を指定します。 <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
論理	<p>条件間の論理関係：</p> <ul style="list-style-type: none"> ○ ①- (And) 現在および次の条件は、論理 AND に関連しています ○ ②- (Or) 現在および次の条件は、論理 OR に関連しています ○ ④- (新しいグループ > And) 現在の条件は論理条件のグループの最後です。現在のグループと次のグループは、論理 AND で関連付けられます。 ○ ⑤- (新しいグループ > Or) 現在の条件は論理条件のグループの最後です。現在のグループと次のグループは、論理 OR で関連付けられます。 ○ ⑦- (終了) 現在の条件は論理条件のグループの最後です。 				
テキスト	<p>次に対して一致するリテラルまたはワイルドカード文字：</p> <ul style="list-style-type: none"> ○ 正確に一致する]、行で検索]、または 範囲で検索 -]の場合、PDF ファイルのレコードのセットを一意に識別する文字または文字列を指定します。 ○ 独自のマップ-]の場合、PDF ファイルのレコードのセットを一意に識別する文字パターンを指定します。 <p>カスタム マップオプションは、"MAP() 関数" ページ 2345と同じ構文を使用します。</p> <p>他の一致タイプについては、テキストは空の文字列 "" です。</p>				

FIELD パラメーター

フィールド定義情報。

名前	説明
FIELD 名前 型	<p>インポートするソース データファイル内の個別フィールドの名前およびデータ型。フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。</p> <p>型については、"フィールド データ型の識別子" ページ 1940を参照してください。</p>
AT 開始行, 開始位置	<ul style="list-style-type: none"> ○ 開始行 -PDF ファイルのレコードのフィールドの開始行

名前	説明				
	<p>PDF ファイルの複数行のレコードの場合、開始行はレコードの任意の行でフィールドを開始できます。レコードの行が①の場合、開始行は常に①です。</p> <ul style="list-style-type: none"> 開始位置 - PDF ファイルのフィールドの開始バイト位置 <p>メモ</p> <table border="1" data-bbox="638 449 1346 569"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
SIZE 長さ、フィールドの行	<ul style="list-style-type: none"> 長さ - Analytics テーブルレイアウトにおけるフィールドの長さ(バイト数) <p>メモ</p> <table border="1" data-bbox="638 810 1346 930"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p> <ul style="list-style-type: none"> 1フィールドの行 - PDF ファイルの単一フィールド値が占める行数 ファイルのデータと一致する単一行または複数行のフィールドを定義できます。 <p>メモ</p> <p>フィールドに指定された行数は、フィールドを含むレコードに指定された行数を超えることができません。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
DEC 値	<p>数値フィールドの小数点以下の桁数</p>				
WID バイト	<p>フィールドの表示幅(バイト数)。</p> <p>指定した値によって、Analytics のビューおよびレポートにおけるフィールドの表示幅が決まります。表示幅はデータを変更するものではありませんが、表示幅がフィールド長より短い場合にはデータが隠れる可能性があります。</p>				
PIC 書式	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソースデータの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソースデータの物理形式と正確に一致する必要があります。たとえば、ソースデータが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p>				

名前	説明
	書式は引用符で囲む必要があります。
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名 (代替列見出し) 。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン (;) を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にスペースがないことを確認してください。</p>

例

PDF ファイルの特定のページからデータをインポートする

次の例では、パスワードで保護された PDF ファイル `Vendors.pdf` の 1 ページからデータをインポートしています。

詳細レコードの 1 セットと 3 つのフィールドが、結果の Analytics テーブル `Vendor_List` に作成されます。

```
IMPORT PDF TO Vendor_List PASSWORD 1 "Vendor_List.FIL" FROM
"Vendors.pdf" 2 PAGES "1" RECORD "Detail" 0 1 0 TEST 0 3 AT 1,1,0 7 ""
FIELD "Vendor_Number" C AT 1,1 SIZE 10,1 DEC 0 WID 10 PIC "" AS ""
FIELD "Vendor_Name" C AT 1,33 SIZE 58,1 DEC 0 WID 58 PIC "" AS ""
FIELD "Last_Active_Date" D AT 1,277 SIZE 20,1 DEC 0 WID 20 PIC
"DD/MM/YYYY" AS ""
```

備考

このコマンドの動作の詳細については、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を参照してください。

Analytics Unicode 版 への PDF のインポートのトラブルシューティング

Analytics の Unicode 版を使って PDF ファイルをインポートする際に発生した問題は、長さの仕様に関連している可能性があります。

- 外国語の文字が突然表示されたり、結果の Analytics テーブルのレイアウトが歪んだりしている場合は、SIZE の長さを偶数に設定してください。

SIZE の長さに奇数のバイトを指定すると、インポートされたデータの処理で問題が発生する可能性があります。

- 作成する Analytics テーブルにレコードが含まれない場合、スキップするファイルの先頭にヘッダーデータがあるときは、`skip_length` を 2 またはその他の偶数に設定してみてください。

フィールド データ型の識別子

以下の表は、`FIELD` でデータ型を指定するときに使用する必要がある文字の一覧を示します。各文字は Analytics データ型の 1 つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、"C": `FIELD "Last_Name" C` と指定します。

詳細については、"Analytics のデータ型" ページ 811 を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C" (CHARACTER 型) が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL

コマンド

文字	Analytics データ型
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS
Z	ZONED

IMPORT PRINT コマンド

印刷イメージ(レポート) ファイルを定義およびインポートして、Analytics テーブルを作成します。

構文

```
IMPORT PRINT TO テーブル インポート ファイル名 FROM ソース ファイル名 <SERVER プロファイル名> 文字セット値 <コード ページ番号> {[レコード構文] [フィールド構文] <...n>} <...n>
```

```
レコード構文 ::=  
RECORD レコード名 レコード タイプ レコードの行 透明 [テスト構文] <...n>
```

```
テスト構文 ::=  
TEST 含める/除外する 一致タイプ AT 開始行,開始位置,範囲 論理 テキスト
```

```
フィールド構文 ::=  
FIELD 名前 型 AT 開始行,開始位置 SIZE 長さ,フィールドの行 DEC 値 WID バイト PIC 形式 AS 表示名
```

パラメーター

一般パラメーター

名前	説明
TO テーブル	データをインポートする Analytics テーブルの名前。 <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。 </div>
インポートファイル名	作成する Analytics データファイルの名前。 インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定しま

名前	説明
	<p>す。例: "Invoices.FIL"</p> <p>デフォルトでは、データファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ◦ "C:\data\Invoices.FIL" ◦ "data\Invoices.FIL"
FROM ソースファイル名	<p>ソース データ ファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データ ファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> ◦ "C:\data\ソース ファイル名" ◦ "data\ソース ファイル名"
SERVER プロファイル名 省略可能	<p>インポートするデータを含むサーバーのプロファイル名。</p>
文字セット値	<p>印刷イメージ(レポート)ファイルのエンコードに使用する文字セット。以下の値がサポートされています。</p> <ul style="list-style-type: none"> ◦ 0 - ASCII ◦ 1 - EBCDIC ◦ 2 - Unicode ◦ 3 - エンコードされたテキスト
コードページ番号 省略可能	<p>文字セット値に対して 3(エンコードされたテキスト)を指定した場合は、さらにコード ページ番号を入力する必要があります。</p>

RECORD パラメーター

一般レコード定義情報。

メモ

一部のレコード定義情報は、データ定義ウィザードのオプションにマッピングされる数値コードを使用して指定されます。

スクリプトでは、オプション名ではなく、数値コードを指定します。

名前	説明
RECORD レコード名	<p>データ定義ウィザードのレコード名。</p> <p>IMPORT PRINT コマンドでは、レコード名を指定する必要がありますが、レコード名値は結果の Analytics テーブルに表示されません。</p> <p>データ定義ウィザードでは、Analytics は、レコードのタイプに基づいて、デフォルト名を提供します。</p>

名前	説明
	<ul style="list-style-type: none"> ◦ 詳細 ◦ ヘッダーn ◦ フッターn <p>デフォルト名を指定するか、別の名前を指定できます。</p>
レコードのタイプ	<p>印刷イメージ ファイルを定義するときには、3つのレコード タイプがあります。</p> <ul style="list-style-type: none"> ◦ 0- 詳細 ◦ 1- ヘッダー ◦ 2- フッター <p>メモ 単一の IMPORT PRINT の実行では、複数のセットのヘッダーおよびフッターレコードを定義できますが、詳細レコードは1セットのみです。</p>
レコードの行	<p>印刷イメージ ファイルのレコードが占める行数。 ファイルのデータと一致する単一行または複数行のレコードを定義します。</p>
透明	<p>ヘッダーレコードの透明設定。</p> <p>メモ ヘッダーレコードにのみ適用されます。</p> <ul style="list-style-type: none"> ◦ 0- 不透明 ◦ 1- 透明 <p>透明なヘッダーレコードは複数行の詳細レコードに分割されません。 ヘッダーレコードがソースの印刷イメージ ファイルの複数行の詳細レコードを分割する場合(改ページで発生する場合があります)、1(透明)を指定すると、結果の Analytics テーブルの詳細レコードが統合されます。</p>

TEST パラメーター

印刷イメージ ファイルのレコードのセットを定義する条件。RECORD の各出現には、1つ以上の TEST (8つまで) があります。

メモ

一部の条件は、データ定義ウィザードのオプション(オプション名は以下で括弧内に示されています)にマッピングされる数値コードを使用して指定されます。

スクリプトでは、オプション名ではなく、数値コードを指定します。

名前	説明
TEST 含める/除外する	<p>一致するデータを処理する方法:</p> <ul style="list-style-type: none"> ◦ 0- (含める) 条件を満たすデータがレコードのセットに含まれます ◦ 1- (除外する) 条件を満たすデータがレコードのセットから除外されます

名前	説明						
一致タイプ	<p>実行する一致のタイプ:</p> <ul style="list-style-type: none"> ○ ①-(正確に一致する) 一致するレコードには、指定された文字または文字列が、指定された位置で始まる、指定された開始行に含まれている必要があります。 ○ ②-(英文字) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の英文字を含んでいる必要があります。 ○ ③-(数字) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の数字を含んでいる必要があります。 ○ ④-(空白) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の空白を含んでいる必要があります。 ○ ⑤-(空白以外) 一致するレコードには、指定された開始行の指定された開始位置にあるか、指定された範囲のすべての位置にある1つ以上の空白以外の文字(特殊文字を含む)を含んでいる必要があります。 ○ ⑦-(行で検索) 一致するレコードには、指定された文字または文字列が、指定された位置で始まる、指定された開始行に含まれている必要があります。 ○ ⑧-(範囲で検索) 一致するレコードには、[テキスト]フィールドにある文字または文字列が、指定された範囲の、レコードの指定された行の指定された範囲のどこかに含まれている必要があります。 ○ ⑩-(独自のマップ) 一致するレコードには、指定された文字パターンと一致する文字が、指定された位置で始まる、指定された開始行に含まれている必要があります。 						
AT 開始行, 開始位置, 範囲	<ul style="list-style-type: none"> ○ 開始行 -条件が適用されるレコードの行 たとえば、独自のマップを使用して郵便番号と一致させ、郵便番号が3行レコードの3行目に表示される場合は、開始行]で③を指定する必要があります。 <p>メモ 単一行のレコードの場合、開始行の値は常に①です。</p> <ul style="list-style-type: none"> ○ 開始位置 -条件と比較する印刷イメージファイルの開始バイト位置 ○ 範囲 -条件との比較で使用する印刷イメージファイルの開始バイト位置からのバイト数 範囲なしで開始バイト位置のみを使用している場合は、範囲に①を指定します。 <p>メモ</p> <table border="1" data-bbox="639 1325 1346 1528"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データの場合、範囲は、偶数バイト数である必要があります。たとえば、⑤①, ⑤⑨ (10 バイト) です。奇数バイト数を指定すると、条件に対する正しい一致が行われなくなる可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						
論理	<p>条件間の論理関係:</p> <ul style="list-style-type: none"> ○ ①-(And) 現在および次の条件は、論理 AND に関連しています ○ ②-(Or) 現在および次の条件は、論理 OR に関連しています ○ ④-(新しいグループ > And) 現在の条件は論理条件のグループの最後です。現在のグループと次のグループは、論理 AND で関連付けられます。 ○ ⑤-(新しいグループ > Or) 現在の条件は論理条件のグループの最後です。現在のグループと次のグループは、論理 OR で関連付けられます。 						

名前	説明
	<ul style="list-style-type: none"> 7-(終了) 現在の条件は論理条件のグループの最後です。
テキスト	<p>次に対して一致するリテラルまたはワイルドカード文字：</p> <ul style="list-style-type: none"> 正確に一致する]、行で検索]、または範囲で検索]-の場合、印刷イメージファイルのレコードのセットを一意に識別する文字または文字列を指定します。 独自のマップ]-の場合、印刷イメージファイルのレコードのセットを一意に識別する文字パターンを指定します。 <p>カスタム マップオプションは、"MAP() 関数" ページ 2345と同じ構文を使用します。</p> <p>他の一致タイプについては、テキストは空の文字列 "" です。</p>

FIELD パラメーター

フィールド定義情報。

名前	説明						
FIELD 名前 型	<p>インポートするソース データファイル内の個別フィールドの名前およびデータ型。フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。</p> <p>型については、"フィールド データ型の識別子" ページ 1948を参照してください。</p>						
AT 開始行, 開始位置	<ul style="list-style-type: none"> 開始行 -印刷イメージファイルのレコードのフィールドの開始行 <p>印刷イメージファイルの複数行のレコードの場合、開始行はレコードの任意の行でフィールドを開始できます。レコードの行が1の場合、開始行は常に1です。</p> <ul style="list-style-type: none"> 開始位置 -印刷イメージファイルのフィールドの開始バイト位置 <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						
SIZE 長さ, フィールドの行	<ul style="list-style-type: none"> 長さ -Analytics テーブルレイアウトにおけるフィールドの長さ(バイト数) <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データでは、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						

名前	説明
	<ul style="list-style-type: none"> ◦ フィールドの行 -印刷イメージ ファイルの単一フィールド値が占める行数 ファイルのデータと一致する単一行または複数行のフィールドを定義できます。 <p>メモ フィールドに指定された行数は、フィールドを含むレコードに指定された行数を超えることができません。</p>
DEC 値	数値フィールドの小数点以下の桁数
WID バイト	<p>フィールドの表示幅(バイト数)。</p> <p>指定した値によって、Analytics のビューおよびレポートにおけるフィールドの表示幅が決まります。表示幅はデータを変更するものではありませんが、表示幅がフィールド長より短い場合にはデータが隠れる可能性があります。</p>
PIC 書式	<p>メモ 数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> ◦ 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 ◦ 日付時刻フィールド -ソース データの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ 日付時刻フィールドの場合、形式はソース データの物理形式と正確に一致する必要があります。たとえば、ソース データが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にスペースがないことを確認してください。</p>

例

印刷イメージ(レポート) ファイルからデータをインポートする

印刷イメージ(レポート) ファイル **Report.txt** からデータをインポートします。

1 件のヘッダーレコード、詳細レコードの1 セット、および5 つのフィールドが、結果の Analytics テーブル `Inventory_report` に作成されます。

```
IMPORT PRINT TO Inventory_report "Inventory_report.FIL" FROM
"Report.txt" 0 RECORD "Header1" 1 1 0 TEST 0 0 AT 1,17,0 7 ":" FIELD
"Field_1" C AT 1,19 SIZE 2,1 DEC 0 WID 2 PIC "" AS "Prod Class" FIELD
"Field_2" C AT 1,24 SIZE 31,1 DEC 0 WID 31 PIC "" AS "Prod
Description" RECORD "Detail" 0 1 0 TEST 0 0 AT 1,59,59 7 "."FIELD
"Field_3" X AT 1,6 SIZE 9,1 DEC 0 WID 9 PIC "" AS "Item ID" FIELD
"Field_4" C AT 1,16 SIZE 24,1 DEC 0 WID 24 PIC "" AS "Item Desc."FIELD
"Field_5" N AT 1,40 SIZE 10,1 DEC 0 WID 10 PIC "" AS "On Hand" FIELD
"Field_6" N AT 1,50 SIZE 12,1 DEC 2 WID 12 PIC "" AS "Cost" FIELD
"Field_7" N AT 1,62 SIZE 12,1 DEC 2 WID 12 PIC "" AS "Total"
```

備考

このコマンドの動作の詳細については、「印刷イメージ(レポート)ファイルとPDFファイルの定義とインポート」ページ 279を参照してください。

フィールド データ型の識別子

以下の表は、`FIELD` でデータ型を指定するときに使用する必要がある文字の一覧を示します。各文字は Analytics データ型の1 つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、"C": `FIELD "Last_Name" C` と指定します。

詳細については、「Analytics のデータ型」ページ 811を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C"(CHARACTER 型) が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY

文字	Analytics データ型
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS
Z	ZONED

IMPORT SAP コマンド

Direct Link を使用して、SAP システムからデータをインポートし、Analytics テーブルを作成します。

メモ

IMPORT SAP コマンドは、Direct Link がローカルコンピュータと組織の SAP システムにインストールおよび構成されている場合にのみサポートされます。

構文

```
IMPORT SAP PASSWORD 番号 TO テーブル名 SAP SOURCE "SAP AGENT" インポートの詳細
```

パラメーター

名前	説明
PASSWORD 番号	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前にPASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none"> "PASSWORD コマンド" ページ 2015 SET PASSWORD コマンド PASSWORD アナリティクスタグ <p>メモ このパスワードは、SAP システムにアクセスするのに使用します。</p>
TO テーブル名	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
SAP SOURCE "SAP AGENT"	<p>SAP データをインポートするために必要です。"SAP AGENT" が選択可能な唯一の選択肢です。</p>

名前	説明
インポートの詳細	クエリの詳細。<q></q> タグで囲む必要があります。この文字列では、"Direct Link クエリタグ" ページ 1953に記載されているタグを使用してクエリを定義します。 この値の物理的なサイズは最大 16 KB です。

例

複数のテーブルに対するクエリーを実行する

この例では、IMPORT SAP コマンドを使用して、複数のテーブルに対するクエリーを実行します。

有効なクエリ文字列を作成するには、タグの順序と入れ子が正確であることが必要です。例内のタグの順序と入れ子は、正確です。この例は、必要とされる IMPORT SAP クエリタグの順序と入れ子を判断するのに使用してください。

メモ

この例は、読みやすくなるように複数の行に分けて書かれています。ただし、スクリプトでは、コマンドとクエリ文字列は改行せずに入力する必要があります。

ヒント

IMPORT SAP クエリ文字列の構文は通常、複雑です。IMPORT SAP コマンドとクエリ文字列をスクリプトに追加する最良の方法は、Analytics の **ログ** タブから既存の IMPORT SAP コマンドをコピーして、必要に応じてクエリタグを編集することです。

```
IMPORT SAP PASSWORD 1 TO Purchasing_doc SAP SOURCE "SAP AGENT"
<q version="6.0">
  <s>0</s>
  <d>IDES</d>
  <u>mzunini</u>
  <c>800</c>
  <lg>en</lg>
  <cf>C:\ACL Data\Purchasing_doc.fil</cf>
  <sf>E:\Data\DL_JSMITH111107.DAT</sf>
  <jcount>11110701</jcount>
  <jname>DL_JSMITH111107.DAT</jname>
  <d1>75</d1>
  <m>2</m>
  <dt>20140321</dt>
  <tm>033000</tm>
```

```
<r>500</r>
<ar>0</ar>
<e>500</e>
<ts>
  <t>
    <n>EKKO</n>
    <a>T00001</a>
    <td>Purchasing Document Header</td>
    <fs>
      <f>EBELN</f>
      <f>BUKRS</f>
      <f>BSTYP</f>
      <f>BSART</f>
      <f>STATU</f>
      <f>WKURS</f>
    </fs>
    <wc>
      <w>
        <f>BUKRS</f>
        <o>0</o>
        <l>1000</l>
        <h></h>
      </w>
    </wc>
  </t>
  <t>
    <n>EKPO</n>
    <a>T00002</a>
    <td>Purchasing Document Item</td>
    <fs>
      <f>EBELP</f>
      <f>WERKS</f>
      <f>MENGE</f>
      <f>BRTWR</f>
    </fs>
    <wc></wc>
  </t>
</ts>
<js>
  <jc>
    <pt>
      <pa>T00001</pa>
      <pf>EBELN</pf>
    </pt>
    <ct>
      <ca>T00002</ca>
```

```

        <cf>EBELN</cf>
    </ct>
</jc>
</js>
</q>

```

備考

"Direct Link クエリタグ" 下に記載されているテーブルには、*import_details* パラメーターに指定できるタグの一覧が示されています。必須列では、以下の値を使用して、各タグの指定がどのような場合に必須であるかが示されています。

- Y - 必須
- N - オプション
- M - 複数テーブルのクエリの場合のみ必須
- B - 必須。ただし、値を渡すことはできません
- W - フィルターを利用する場合のオプション
- S - スケジュールモードが指定されるときに必須

Direct Link クエリタグ

名前	タグ	必須	説明
テーブル別名	<a>	M	クエリ内で一意にテーブルを識別する別名。これにより、同じテーブルを2回以上使用できるようになります。 6文字を上限とします。
すべての行	<ar>	Y	すべての一致する行がクエリの結果セットの一部として返されることを示します。 有効な値は次のとおりです。 1 - <r> タグで指定されたレコード数を上書きします(最大行) 0 - <r> タグで指定されたレコード数を返します(最大行) このタグは、常に <r></r> タグの後に指定します。
クライアント	<c>	N	SAP システム内のクライアント。
子テーブル別名	<ca>	M	子テーブルの別名。
子テーブルフィールド	<cf>	M	結合条件に基づく子テーブル内のフィールド。
クライアント ファイル名	<cf>	Y	クエリの結果が格納されるクライアント システム上のターゲット ファイルを識別します。

名前	タグ	必須	説明
子テーブル	<ct>	M	結合条件の子テーブル。
宛先	<d>	N	SAP システムを特定するために使用する、SAP RFC ライブラリファイル (<code>saprfc.ini</code>) 内の宛先を指定します。
データ長	<dl>	B	各行内の文字の数。これには、レコードの終わりを示すキャリッジ リターン文字とライン フィード文字 (CR+LF、つまり 16 進数の文字 0D+0A) が含まれます。
日付	<dt>	S	スケジュールモードを使用するときに必須です。SAP ジョブを実行する時間を指定します。 YYYYMMDD の形式である必要があります。たとえば、2014 年 12 月 31 日は、20140131 と指定する必要があります。
予想される行数	<e>	B	クエリが返すと予想される行数。
フィールド名	<f>	Y	ネイティブフィールド名。
フィルターフィールド	<f>	W	フィルターを適用するネイティブフィールド名。
フィールド	<fs>	Y	クエリ結果の一部として返される、テーブル内のフィールドの一覧。
上限値	<h>	W	Between 演算子を使用するときには上限値を格納します。他の演算子を使用している場合は無視されます。
結合条件	<jc>	M	結合条件。
ジョブ数	<jcount>	B	バックグラウンド モードのクエリを識別するために、SAP によって内部的に使用されます。
ジョブ名	<jname>	B	バックグラウンド モードのクエリを識別するために、SAP によって内部的に使用されます。
結合関係	<js>	Y	クエリ内でテーブルをリンクするための結合条件の一覧。
結合切り替え	<jw>	N	結合切り替え列挙型に等価な数値。 有効な値は次のとおりです。 0 - 内部結合 1 - 左外部結合
下限値	<l>	W	Between 演算子を使用するときには下限値、他の演算子を使用するときには値を格納します。
言語	<lg>	Y	SAP データベースのフィールドのロケールを決定するために使用される言語識別子。
モード	<m>	Y	サブミッション モード列挙型に等価な数値。 有効な値は次のとおりです。

コマンド

名前	タグ	必須	説明
			0 -即時に抽出 1 -バックグラウンド 2 -スケジュール済み
テーブル名	<n>	Y	ネイティブテーブル名。
演算子	<o>	W	演算子列挙型に等価な数値。 有効な値は次のとおりです。 0 --等しい(=) 1 --等しくない(<>) 2 --未満(<) 3 --以下(<=) 4 --より大きい(>) 5 --以上(>=) 6 --間 7 --含む
親テーブル別名	<pa>	M	親テーブルの別名。
親テーブルフィールド	<pf>	M	結合条件に基づく親テーブル内のフィールド。
親テーブル	<pt>	M	結合条件の親テーブル。
クエリ	<q>	Y	クエリを囲みます。
最大行数	<r>	Y	クエリが返す行の最大数。
選択済み	<s>	Y	<s> タグが <f> タグの下に現れる場合、それはフィールドがクエリの結果セットの一部として返されるかどうかを示します。
システム	<s>	Y	<s> タグが <q> タグの下に現れる場合は、このクエリが使用されるシステムの種類を識別します(現在のところ、SAPのみがサポートされます)。
サーバーファイル名	<sf>	B	バックグラウンドモードクエリの結果を保持するサーバー上のファイルを識別します。
サーバーグループ名	<sg>	N	サーバーグループの名前。最大 20 文字。
サーバー名	<sn>	N	サーバーの名前。最大 20 文字。
テーブル	<t>	Y	テーブル。
テーブル記述	<td>	Y	SAP データ ディクショナリーのテーブル記述。常に <a> タグの下に現れます。

名前	タグ	必須	説明
時間	<tm>	S	スケジュールモードを使用するときに必須です。SAP ジョブを実行する時間を指定します。 hhmmss の形式である必要があります。たとえば、2:30 pm は 143000 と指定する必要があります。
テーブル	<ts>	Y	クエリがデータを抽出するテーブルの一覧。
テーブルタイプ	<tt>	Y	SAP テーブルのタイプ。 有効な値は次のとおりです。 0 - クラスター 1 - 透明 2 - プール 3 - ビュー
ユーザー名	<u>	N	ユーザーのログオン名。
フィルター	<w>	W	テーブルのデータに適用されるフィルター。
フィルター	<wc>	W	テーブル内に含まれるデータに適用されるフィルターの一覧。
フィルター切り替え	<ws>	N	フィルター切り替え列挙型に等価な数値。 有効な値は次のとおりです。 0 - (Or) および (Or) 1 - (And) または (And)

IMPORT XBRL コマンド

XBRL ファイルを定義およびインポートして、Analytics テーブルを作成します。

構文

```
IMPORT XBRL TO テーブル インポート ファイル名 FROM ソース ファイル名 CONTEXT コンテキスト名 <...n> [フィールド構文] <...n> <IGNORE フィールド番号> <...n>
```

```
フィールド構文 ::=
FIELD 名前 型 AT 開始位置 DEC 値 WID バイト PIC 書式 AS 表示名
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
インポートファイル名	<p>作成する Analytics データ ファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL".</p> <p>デフォルトでは、データファイル (.FIL) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ◦ "C:\data\Invoices.FIL" ◦ "data\Invoices.FIL"
FROM ソースファイル名	<p>ソース データ ファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データ ファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> ◦ "C:\data\ソース ファイル名"

名前	説明				
	<ul style="list-style-type: none"> "data\ソース ファイル名" 				
CONTEXT コンテキスト名	テーブルを定義する XBRL コンテキスト。1 つ以上のコンテキストを指定する場合、コンテキストはすべて同じタイプ(Instant、Period、または Forever) である必要があります。				
FIELD 名前型	<p>インポートするソース データ ファイル内の個別フィールドの名前およびデータ型。フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。</p> <p>型については、"フィールド データ型の識別子" 次のページを参照してください。</p>				
AT 開始位置	<p>Analytics データ ファイル内のフィールドの開始バイトを指定します。</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
DEC 値	数値フィールドの小数点以下の桁数				
WID バイト	<p>Analytics テーブルレイアウトにおけるフィールドの長さ(バイト数)</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
PIC 書式	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソース データの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソース データの物理形式と正確に一致する必要があります。たとえば、ソース データが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>				
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミ</p>				

名前	説明
	<p>コロン(:)を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にスペースがないことを確認してください。</p>
IGNORE フィールド番号 省略可能	<p>テーブルレイアウトからフィールドを除外します。</p> <p>フィールド番号は、ソースデータにおけるフィールドの位置を指定します。たとえば、IGNORE 5 は、ソースデータの5番目のフィールドを Analytics テーブルレイアウトから除外します。</p>

例

XBRL ファイルを Analytics テーブルにインポートする

次の例は、XBRL ファイルの **Current_AsOf** コンテキストから **Financials** という Analytics テーブルにデータをインポートしています。

```
IMPORT XBRL TO Financials "Financials.fil" FROM
"FinancialStatemenXBRL.xml" CONTEXT "Current_AsOf" FIELD "Item" C AT 1
DEC 0 WID 57 PIC "" AS "" FIELD "Value" X AT 58 DEC 0 WID 7 PIC "" AS
"" IGNORE 1 IGNORE 3
```

備考

このコマンドの動作の詳細については、「XBRL ファイルのインポート」ページ 353を参照してください。

フィールド データ型の識別子

以下の表は、FIELD でデータ型を指定するときに使用する必要がある文字の一覧を示します。各文字は Analytics データ型の1つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、「C」: FIELD "Last_Name" C と指定します。

詳細については、「Analytics のデータ型」ページ 811を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C"(CHARACTER 型)が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS

コマンド

文字	Analytics データ型
Z	ZONED

IMPORT XML コマンド

XML ファイルを定義およびインポートして、Analytics テーブルを作成します。

構文

```
IMPORT XML TO テーブル インポートファイル名 FROM ソースファイル名 [フィールド構文] <...n>
```

```
フィールド構文 ::=
FIELD 名前 型 AT 開始位置 DEC value WID バイト PIC 書式 AS 表示名 RULE xpath
式
```

パラメーター

名前	説明
TO テーブル	<p>データをインポートする Analytics テーブルの名前。</p> <p>メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
インポートファイル名	<p>作成する Analytics データ ファイルの名前。</p> <p>インポートファイル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例: "Invoices.FIL".</p> <p>デフォルトでは、データ ファイル (.FIL) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータ ファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> "C:\data\Invoices.FIL" "data\Invoices.FIL"
FROM ソースファイル名	<p>ソース データ ファイルの名前。ソース ファイル名は引用符で囲む必要があります。</p> <p>ソース データ ファイルが Analytics プロジェクトと同じフォルダーに位置しない場合、ファイルの位置を指定するために絶対パスまたは相対パスを使用する必要があります。</p> <ul style="list-style-type: none"> "C:\data\ソース ファイル名"

名前	説明				
	<ul style="list-style-type: none"> "data\ソース ファイル名" 				
FIELD 名前 型	<p>インポートするソース データ ファイル内の個別フィールドの名前およびデータ型。フィールドをインポート対象から除外する場合は、そのフィールドを指定しないでください。</p> <p>型については、"フィールド データ型の識別子" 見開きページを参照してください。</p>				
AT 開始位置	<p>Analytics データ ファイル内のフィールドの開始バイトを指定します。</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、一般的に、奇数で開始するバイト位置を指定してください。偶数の開始位置を指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
DEC 値	<p>数値フィールドの小数点以下の桁数</p>				
WID バイト	<p>Analytics テーブルレイアウトにおけるフィールドの長さ(バイト数)</p> <p>メモ</p> <table border="1"> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics</td> <td>2 バイト = 1 文字</td> </tr> </table> <p>Unicode 版 Analytics では、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字				
Unicode 版 Analytics	2 バイト = 1 文字				
PIC 書式	<p>メモ</p> <p>数値フィールドまたは日付時刻フィールドにのみ適用されます。</p> <ul style="list-style-type: none"> 数値フィールド - Analytics のビューとレポートに含まれる数値の表示形式。 日付時刻フィールド - ソース データの日付時刻値の物理形式(日付時刻文字、区切り文字の順など) <p>メモ</p> <p>日付時刻フィールドの場合、形式はソース データの物理形式と正確に一致する必要があります。たとえば、ソース データが 12/31/2014 である場合は、書式を "MM/DD/YYYY" として入力します。</p> <p>書式は引用符で囲む必要があります。</p>				
AS 表示名	<p>新しい Analytics テーブルのビューにおけるフィールドの表示名(代替列見出し)。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン(;)を入れます。</p> <p>フィールドの定義時には、AS は必須です。表示名をフィールド名と同じにしたい場合は、空の表示名を入力します。つまり、次の構文を使用します。AS ""2 つの二重引用符の間にス</p>				

名前	説明
	スペースがないことを確認してください。
RULE <i>xpath</i> 式	xpath 式はXML ファイルからフィールド内容を選択するために使用されます。 xpath はXML ファイルのデータにアクセスする標準的な方法です。たとえば、 <code>acct/title/text()</code> はXML ファイルの <code><title></code> タグ内のテキストを取得します。

例

XML ファイルのデータを Analytics テーブルにインポートする

次の例では、XML ファイルのデータを **Employees** という名前の Analytics テーブルにインポートしています。

```
IMPORT XML TO Employees "Employees.fil" FROM "emp.XML" FIELD "Empno" C
AT 1 DEC 0 WID 6 PIC "" AS "" RULE "/RECORDS/RECORD/Empno/text()"
FIELD "First" C AT 7 DEC 0 WID 13 PIC "" AS "" RULE
"/RECORDS/RECORD/First/text()" FIELD "Last" C AT 20 DEC 0 WID 20 PIC
"" AS "" RULE "/RECORDS/RECORD/Last/text()" FIELD "HireDate" D AT 40
DEC 0 WID 10 PIC "YYYY-MM-DD" AS "" RULE
"/RECORDS/RECORD/HireDate/text()" FIELD "Salary" N AT 50 DEC 2 WID 8
PIC "" AS "" RULE "/RECORDS/RECORD/Salary/text()"
```

備考

このコマンドの動作の詳細については、「XML ファイルのインポート」ページ 344を参照してください。

フィールド データ型の識別子

以下の表は、`FIELD` でデータ型を指定するときに使用する必要がある文字の一覧を示します。各文字は Analytics データ型の 1 つに対応します。

たとえば、文字データ型が必要な姓フィールドを定義する場合は、「C」: `FIELD "Last_Name" C` と指定します。

詳細については、「Analytics のデータ型」ページ 811を参照してください。

メモ

データ定義ウィザードを使用して EBCDIC、Unicode、または ASCII フィールドを含むテーブルを定義する場合、それらのフィールドには自動的に文字 "C"(CHARACTER 型) が割り当てられます。

IMPORT ステートメントを手作業で入力するか、既存の IMPORT ステートメントを編集する場合、EBCDIC または Unicode フィールドに対して、より特有の文字 "E" または "U" に置き換えることができます。

文字	Analytics データ型
A	ACL
B	BINARY
C	CHARACTER
D	DATETIME
E	EBCDIC
F	FLOAT
G	ACCPAC
I	IBMFLOAT
K	UNSIGNED
L	LOGICAL
N	PRINT
P	PACKED
Q	BASIC
R	MICRO
S	CUSTOM
T	PCASCII
U	UNICODE
V	VAXFLOAT
X	NUMERIC
Y	UNISYS

文字	Analytics データ型
Z	ZONED

INDEX コマンド

物理的ではなく連続的な順番でレコードへのアクセスを可能にする、Analytics テーブルのインデックスを作成します。

構文

```
INDEX {<ON> キー フィールド <D> <...n>|<ON> ALL <EXCLUDE フィールド名 <...n>>} TO
ファイル名 <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <OPEN> <ISOLocale ロケールコード>
```

パラメーター

名前	説明
ON キーフィールド D <...n> ON ALL	<p>インデックスで使用するキーフィールドまたはフィールド、または式。</p> <p>演算フィールドや一時的に作成した式など、データ型に関係なく、あらゆる種類のフィールドのインデックスを作成することができます。</p> <ul style="list-style-type: none"> ON キーフィールド-では、指定した1つまたは複数のフィールドが使用されます。 複数のフィールドでインデックス作成を行う場合は、テーブルは入れ子でインデックスが作成されます。入れ子でのフィールド間の順序は、フィールドを指定した順になります。 キーフィールドに降順にインデックスを作成するDを含めます。デフォルトのインデックス順は昇順です。 ON ALL-では、テーブル内のすべてのフィールドが使用されます。 テーブル内のすべてのフィールドを基準にしてインデックス作成を行うと、入れ子でインデックスが作成されます。入れ子でのフィールド間の順序は、テーブルレイアウトで指定した順になります。 ON ALL では昇順でしかインデックスを作成することができません。
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用して、インデックスを作成するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p> <pre>ON ALL EXCLUDE field_1 field_2</pre>
TO ファイル名	<p>インデックスと関連付けられたインデックスファイルの名前。インデックスファイルは .INX 拡張子</p>

名前	説明
	<p>で作成されます。</p> <p>メモ Analytics のユーザー インターフェイスでは、インデックスの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
OPEN 省略可能	<p>テーブルを開き、インデックスをテーブルに適用します。</p>
ISOLOCALE ロケールコード 省略可能	<p>メモ Analytics の Unicode 版にのみ適用されます。</p> <p>システム ロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ○ 言語 - ISO 639 標準言語コード ○ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

例

インデックスを作成し、テーブルを開く

業者テーブルで、業者市区郡フィールドにインデックスを作成し、テーブルを開きます。

```
OPEN Vendor
INDEX ON Vendor_City to "CityIndex" OPEN
```

インデックスを作成し、テーブルに適用する

業者テーブルで、業者市区郡フィールドにインデックスを作成します。後から、インデックスをテーブルに適用します。

```
OPEN Vendor
INDEX ON Vendor_City to "CityIndex"
.
.
.
SET INDEX TO "CityIndex"
```

備考

このコマンドの動作の詳細については、"レコードのインデックス付け" ページ 1196を参照してください。

INDEX コマンドの並べ替えシーケンス

INDEX コマンドは、**並べ替え順序** オプション(`ソール > オプション > テーブル`)で指定された並べ順を使用します。デフォルトの並べ替え順序は次のとおりです。

詳細については、"**並べ替え順** オプションとソート順序" ページ 1184を参照してください。

Analytics のエディション	並べ替え順]のデフォルト	関連付けられているソート順序
非 Unicode	システム デフォルト	数字、大文字、小文字、小文字:

Analytics のエディション	並べ替え順]のデフォルト	関連付けられているソート順序
	(ASCII)	<div style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;">0, 1, 2... A, B, C... a, b, c...</div> <p>たとえば、"Z" は "a" の前にソートされます。</p>
Unicode	混合言語 (UCA) (Unicode 照合アルゴリズム)	<p>数字、小文字と大文字の混合：</p> <div style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;">0, 1, 2... a, A, b, B, c, C...</div> <p>たとえば、"a" は "Z" の前にソートされます。</p>

大文字と小文字の区別

INDEX では大文字と小文字は区別されません。ご使用の Analytics のエディション(非 Unicode または Unicode)によって、文字列の大文字と小文字がインデックス作成に影響する場合があります。

大文字と小文字がインデックスに影響しないようにしたい場合は、INDEX と一緒に UPPER() 関数を使用できます。

```
INDEX ON UPPER(キー フィールド) TO "Index_file"
```

JOIN コマンド

2 つの Analytics テーブルのフィールドを連結して 1 つの新しい Analytics テーブルにします。

メモ

曖昧一致を使用して、テーブルを結合するには、"FUZZYJOIN コマンド" ページ 1817を参照してください。

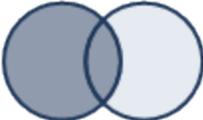
構文

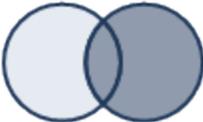
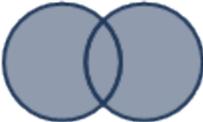
```
JOIN {PKEY 主キー フィールド|PKEY ALL <EXCLUDE フィールド名 <...n>>} {FIELDS 主フィールド|FIELDS ALL <EXCLUDE フィールド名 <...n>>} {SKEY 副キー フィールド|SKEY ALL <EXCLUDE フィールド名 <...n>>} <WITH 副フィールド|WITH ALL <EXCLUDE フィールド名 <...n>>> {キーワードなし|MANY|UNMATCHED|PRIMARY|SECONDARY|PRIMARY SECONDARY} <IF テスト> TO テーブル名 <LOCAL> <OPEN> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <APPEND> <PRESORT> <SECSORT> <ISOLocale ロケール コード>
```

パラメーター

名前	説明
PKEY 主テーブルのキー フィールド PKEY ALL	主テーブルのキーフィールド(複数可)または式。 <ul style="list-style-type: none"> PKEY 主キー フィールド - 指定されたフィールドを使用します フィールドはリストする順序で使用されます。 PKEY ALL - テーブルのすべてのフィールドを使用します フィールドは、テーブルレイアウトに表示される順序で使用されます。
EXCLUDE フィールド名 省略可能	PKEY ALL を使用して、結合を実行するときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、PKEY ALL を微調整できます。 EXCLUDE は PKEY ALL の直後に置く必要があります。例： <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>PKEY ALL EXCLUDE field_1 field_2</pre> </div>
FIELDS 主フィールド FIELDS ALL	結合先の出カテーブルに含める、主テーブル内のフィールドまたは式。 <ul style="list-style-type: none"> FIELDS 主フィールド - 指定されたフィールドが含まれます

名前	説明
	<p>フィールドはリストする順序で含まれます。</p> <ul style="list-style-type: none"> ◦ FIELDS ALL - はテーブルのすべてのフィールドに含まれます <p>フィールドは、テーブルレイアウトに表示される順序で含まれます。</p> <p>メモ 結合テーブルに含める場合は、主キーフィールドを明示的に指定する必要があります。FIELDS ALL を指定しても含まれます。</p>
<p>EXCLUDE 主フィールド 省略可能</p>	<p>FIELDS ALL を使用して、結合を実行するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre data-bbox="565 737 1344 800">FIELDS ALL EXCLUDE field_1 field_2</pre>
<p>SKEY 副テーブルのキー フィールド SKEY ALL</p>	<p>副テーブルのキーフィールド(複数可)または式。</p> <ul style="list-style-type: none"> ◦ SKEY 副キーフィールド - 指定されたフィールドを使用します <p>フィールドはリストする順序で使用されます。</p> <ul style="list-style-type: none"> ◦ SKEY ALL - テーブルのすべてのフィールドを使用します <p>フィールドは、テーブルレイアウトに表示される順序で使用されます。</p>
<p>EXCLUDE フィールド名 省略可能</p>	<p>SKEY ALL を使用して、結合を実行するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、SKEY ALL を微調整できます。</p> <p>EXCLUDE は SKEY ALL の直後に置く必要があります。例：</p> <pre data-bbox="565 1262 1344 1325">SKEY ALL EXCLUDE field_1 field_2</pre>
<p>WITH 副フィールド WITH ALL 省略可能</p>	<p>結合先の出カテーブルに含める、副テーブル内のフィールドまたは式。</p> <ul style="list-style-type: none"> ◦ WITH 副フィールド - 指定されたフィールドが含まれます <p>フィールドはリストする順序で含まれます。</p> <ul style="list-style-type: none"> ◦ WITH ALL - はテーブルのすべてのフィールドに含まれます <p>フィールドは、テーブルレイアウトに表示される順序で含まれます。</p> <p>メモ 結合テーブルに含める場合は、副キーフィールドを明示的に指定する必要があります。WITH ALL を指定すると含まれます。 結合タイプ "UNMATCHED" を使用する場合には、"WITH" は指定できません。</p>
<p>EXCLUDE フィールド名 省略可能</p>	<p>WITH ALL を使用して、結合を実行するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、WITH ALL</p>

名前	説明												
	<p>を微調整できます。</p> <p>EXCLUDE はWITH ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>WITH ALL EXCLUDE <i>field_1 field_2</i></p> </div>												
<p>キーワードなし MANY UNMATCHED PRIMARY SECONDARY PRIMARY SECONDARY</p>	<p>実行する結合の種類。</p> <p>キーワードなし(結合タイプのどのキーワードも指定しない)</p>  <table border="1" data-bbox="516 716 1414 873"> <thead> <tr> <th>結合された出カテータブルの内容：</th> <th>結合]ダイアログボックスの対応するオプション</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> すべての一致した主レコードおよび最初に一致した副レコード </td> <td>一致した主レコードおよび副レコード (一致する1件目の副レコード)</td> </tr> </tbody> </table> <p>MANY</p>  <table border="1" data-bbox="516 1083 1414 1272"> <thead> <tr> <th>結合された出カテータブルの内容：</th> <th>結合]ダイアログボックスの対応するオプション</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> すべての一致した主レコードおよびすべての一致した副レコード 主および副テーブル間の各一致に1つのレコード </td> <td>一致した主レコードおよび副レコード (重複する副レコードのすべての一致)</td> </tr> </tbody> </table> <p>UNMATCHED</p>  <table border="1" data-bbox="516 1482 1414 1602"> <thead> <tr> <th>結合された出カテータブルの内容：</th> <th>結合]ダイアログボックスの対応するオプション</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> 不一致の主レコード </td> <td>不一致の主レコード</td> </tr> </tbody> </table> <p>PRIMARY</p> 	結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション	<ul style="list-style-type: none"> すべての一致した主レコードおよび最初に一致した副レコード 	一致した主レコードおよび副レコード (一致する1件目の副レコード)	結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション	<ul style="list-style-type: none"> すべての一致した主レコードおよびすべての一致した副レコード 主および副テーブル間の各一致に1つのレコード 	一致した主レコードおよび副レコード (重複する副レコードのすべての一致)	結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション	<ul style="list-style-type: none"> 不一致の主レコード 	不一致の主レコード
結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション												
<ul style="list-style-type: none"> すべての一致した主レコードおよび最初に一致した副レコード 	一致した主レコードおよび副レコード (一致する1件目の副レコード)												
結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション												
<ul style="list-style-type: none"> すべての一致した主レコードおよびすべての一致した副レコード 主および副テーブル間の各一致に1つのレコード 	一致した主レコードおよび副レコード (重複する副レコードのすべての一致)												
結合された出カテータブルの内容：	結合]ダイアログボックスの対応するオプション												
<ul style="list-style-type: none"> 不一致の主レコード 	不一致の主レコード												

名前	説明												
	<table border="1" data-bbox="516 270 1414 415"> <tr> <td data-bbox="516 270 966 327">結合された出カテーブルの内容:</td> <td data-bbox="966 270 1414 327">結合]ダイアログ ボックスの対応するオプション</td> </tr> <tr> <td data-bbox="516 327 966 415"> <ul style="list-style-type: none"> すべての主レコード(一致と不一致の両方)、最初に一致した副レコード </td> <td data-bbox="966 327 1414 415">すべての主レコードと、キーに一致する副レコード</td> </tr> </table> <p data-bbox="565 453 1198 520">メモ キーワード、BOTH は、PRIMARY を指定することと同じです。</p> <p data-bbox="516 558 667 583">SECONDARY</p>  <table border="1" data-bbox="516 751 1414 957"> <tr> <td data-bbox="516 751 966 808">結合された出カテーブルの内容:</td> <td data-bbox="966 751 1414 808">結合]ダイアログ ボックスの対応するオプション</td> </tr> <tr> <td data-bbox="516 808 966 957"> <ul style="list-style-type: none"> すべての副レコード(一致と不一致)およびすべての一致する主レコード 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 </td> <td data-bbox="966 808 1414 957">すべての副レコードと、キーに一致する主レコード</td> </tr> </table> <p data-bbox="516 974 781 999">PRIMARY SECONDARY</p>  <table border="1" data-bbox="516 1167 1414 1352"> <tr> <td data-bbox="516 1167 966 1224">結合された出カテーブルの内容:</td> <td data-bbox="966 1167 1414 1224">結合]ダイアログ ボックスの対応するオプション</td> </tr> <tr> <td data-bbox="516 1224 966 1352"> <ul style="list-style-type: none"> すべての主および副レコード(一致と不一致) 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 </td> <td data-bbox="966 1224 1414 1352">すべての主レコードおよび副レコード</td> </tr> </table>	結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション	<ul style="list-style-type: none"> すべての主レコード(一致と不一致の両方)、最初に一致した副レコード 	すべての主レコードと、キーに一致する副レコード	結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション	<ul style="list-style-type: none"> すべての副レコード(一致と不一致)およびすべての一致する主レコード 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 	すべての副レコードと、キーに一致する主レコード	結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション	<ul style="list-style-type: none"> すべての主および副レコード(一致と不一致) 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 	すべての主レコードおよび副レコード
結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション												
<ul style="list-style-type: none"> すべての主レコード(一致と不一致の両方)、最初に一致した副レコード 	すべての主レコードと、キーに一致する副レコード												
結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション												
<ul style="list-style-type: none"> すべての副レコード(一致と不一致)およびすべての一致する主レコード 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 	すべての副レコードと、キーに一致する主レコード												
結合された出カテーブルの内容:	結合]ダイアログ ボックスの対応するオプション												
<ul style="list-style-type: none"> すべての主および副レコード(一致と不一致) 重複する副一致の最初のインスタンスのみが主レコードに結合されます。 	すべての主レコードおよび副レコード												
IF テスト 省略可能	<p data-bbox="516 1388 1393 1444">各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p data-bbox="565 1482 1333 1581">メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>												

名前	説明
	<p>メモ</p> <p>ほとんどの結合タイプで、IF 条件が主テーブルにのみ適用されます。例外は多対多結合で、IF 条件が副テーブルも参照できます。副テーブルを参照するには、完全修飾フィールド名(テーブル名.フィールド名)を指定する必要があります。例：</p> <pre data-bbox="656 464 1276 533">IF Customer.State="NY"</pre>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字()を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出カファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します

名前	説明
	<p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>APPEND</p> <p>省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
<p>PRESORT</p> <p>省略可能</p>	<p>コマンドを実行する前に主キーフィールドで主テーブルを並べ替えます。</p> <p>メモ</p> <p>GROUPコマンドの内部ではPRESORTを使用することができません。</p> <p>並べ替えの代わりにインデックス</p> <p>主テーブルは、並べ替える代わりにインデックスを付けることができます。大きなテーブルでは並べ替えよりインデックス作成を行った方が、テーブルの結合に必要な時間を短くすることができます。</p> <p>また、インデックス付きの共通のキーフィールドを使用して2つのテーブルを結合する場合は、PRESORTとSECSORTを省略します。</p>
<p>SECSORT</p> <p>省略可能</p>	<p>コマンドを実行する前に副テーブルのキーフィールドで副テーブルを並べ替えます。</p> <p>メモ</p> <p>SECSORTはGROUPコマンド内では使用できません。</p> <p>並べ替えの代わりにインデックス</p> <p>副テーブルは、並べ替える代わりにインデックスを付けることができます。大きなテーブルでは並べ替えよりインデックス作成を行った方が、テーブルの結合に必要な時間を短くすることができます。</p> <p>また、インデックス付きの共通のキーフィールドを使用して2つのテーブルを結合する場合は、PRESORTとSECSORTを省略します。</p>
<p>ISOLOCALE ロケールコード</p> <p>省略可能</p>	<p>メモ</p> <p>AnalyticsのUnicode版にのみ適用されます。</p> <p>システムロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p>

名前	説明
	<p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

例

業者である可能性がある従業員を検索するための方法として、2つのテーブルを結合する

次の例は、共通キーフィールドとして住所 (Address および Vendor_Street フィールド) を使用し、Empmast および Vendor テーブルを結合します。

JOIN コマンドは、新しいテーブルと、一致する主および副レコードを作成します。これにより、同じ住所の従業員と業者のリストが作成されます。

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
JOIN PKEY Address FIELDS Empno First Last Address SKEY Vendor_Street
WITH Vendor_No Vendor_Name Vendor_Street TO "Employee_Vendor_Match"
OPEN PRESORT SECSORT
```

このバージョンの JOIN コマンドは、主テーブルと副テーブルのすべてのフィールドを結合された出力テーブルに含めます。

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
JOIN PKEY Address FIELDS ALL SKEY Vendor_Street WITH ALL TO "Employee_
Vendor_Match" OPEN PRESORT SECSORT
```

このバージョンの JOIN コマンドは IF 条件を使用して、結合された出力テーブルを、カリフォルニアに所在地がある従業員と業者に制限しています。

結合タイプは `MANY` です。IF 条件で副テーブルを参照する場合には、このタイプが必要です。副テーブルのフィールド名は完全修飾名でなければなりません (`Vendor.Vendor_State`)。

```
OPEN Empmast PRIMARY
OPEN Vendor SECONDARY
JOIN PKEY Address FIELDS ALL SKEY Vendor_Street WITH ALL IF State="CA"
AND Vendor.Vendor_State="CA" TO "Employee_Vendor_Match" OPEN PRESORT
MANY SECSORT
```

一致する顧客がない売掛金レコードを検出する方法として、2つのテーブルを結合します。

次の例では、ArテーブルとCustomerテーブルを、顧客番号(CustNo)を共通のキーフィールドとして用いて結合します。

JOIN コマンドは、結合タイプ UNMATCHED を使用して、主テーブルの不一致レコードを含む新しいテーブルを作成しています。その結果、どの Customer レコードとも関連付けられていない Ar レコードの一覧が生成されます。

```
OPEN Ar PRIMARY
OPEN Customer SECONDARY
JOIN PKEY CustNo FIELDS CustNo Due Amount SKEY CustNo UNMATCHED TO
"CustomerNotFound.fil" OPEN PRESORT SECSORT
```

備考

このコマンドの動作の詳細については、「テーブルの結合」ページ 960を参照してください。

LIST コマンド

Analytics テーブルの 1 つ以上のフィールドのデータを、列で書式設定されたディスプレイに出力します。

構文

```
LIST {FIELDS フィールド名 <AS 表示名> <...n>|FIELDS ALL} <LINE 番号 フィールドリスト
> <TO {SCREEN|ファイル名|PRINT}> <UNFORMATTED> <IF テスト> <WHILE テスト> <FIRST
範囲|NEXT 範囲> <HEADER ヘッダーテキスト> <FOOTER フッターテキスト> <SKIP 行数> <EOF>
<APPEND>
```

パラメーター

名前	説明
FIELDS フィールド名 <...n> FIELDS ALL	出力に含めるフィールド。 <ul style="list-style-type: none"> FIELDS フィールド名 - 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。 FIELDS ALL - はテーブルのすべてのフィールドに含まれます フィールドは、テーブルレイアウトに表示される順序で含まれます。
AS 表示名 省略可能	FIELDS フィールド名を使ってデータを一覧表示する場合にのみ使用します。 出力におけるフィールドの表示名(代替列見出し)。表示名をフィールド名、またはソース テーブル内の既存の表示名と同じにしたい場合は、AS を使用しないでください。 表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミ コロン(;)を入れます。
LINE 番号 フィールドリスト 省略可能	複数の行が各レコードの出力に使用されます。 <ul style="list-style-type: none"> 番号 - の値は 2 から 60 までの行番号(両端を含む)のいずれかでなければなりません。 フィールド リスト - その行に含めるフィールド
TO SCREEN ファイル名 PRINT 省略可能	コマンドの結果を送信する場所： <ul style="list-style-type: none"> SCREEN -- Analytics の表示領域に結果を表示します <div style="border-left: 2px solid green; padding-left: 10px; margin: 10px 0;"> <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けら れているソーステーブル内の 1 つまたは複数のレコードにドリルダウンする ことができます。</p> </div> <ul style="list-style-type: none"> ファイル名 - は結果の保存先となるファイルです。

名前	説明
	<p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> <p>◦ 印刷 - 通常使うプリンターに結果を送信します</p>
UNFORMATTED 省略可能	<p>出力は書式設定されていないテキストとして表示されます。出力は EXPORT ASCII コマンドで作成される出力と同じです。書式なしのデータは、ほかのソフトウェアプログラムで処理する目的でファイルに出力できます。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数:</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。 ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。 フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
SKIP 行数 省略可能	<p>リストの各レコード間に、指定した数の空白行が挿入されます。たとえば「<code>LIST ALL SKIP 1</code>」と指定すると、1 行おきの行間、つまり各レコード間に空白行が 1 行あるリストが作成されます。</p>

名前	説明
EOF 省略可能	ファイルの終わりに達した後、コマンドをもう一度実行します。 これにより、GROUP コマンド内でテーブルの最後のレコードが処理されることが保証されます。すべてのフィールドが以前のレコードを参照する演算フィールドである場合にのみ使用してください。
APPEND 省略可能	コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。

例

例外をリストアップしてテキスト ファイルに保存する

LIST を使用して、在庫テーブル内で発見された例外をリストアップするレポートを作成するとします。レポートはテキスト ファイルとして保存されます。以上を行うコマンドの例を次に示します。

```
LIST Product_number Description Quantity Unit_cost Value IF Quantity <
0 OR Unit_cost < 0 HEADER "Negative Values" TO "Exceptions.txt"
```

備考

LIST を使用する場面

LIST を使用する場面は、データの印刷、画面へのデータの表示、テキスト ファイルへのデータのエクスポートを行うときです。

書式設定と合計

UNFORMATTED を指定する場合を除いて、次の情報が自動的に含まれます。

- ページ番号
- 日付
- 時刻
- ユーザー ID
- 列見出し

また、数値列が自動的に合計されます。

LOCATE コマンド

指定された値または条件に一致する最初のレコードを検索するか、または指定されたレコード番号に移動します。

構文

```
LOCATE {IF テスト <WHILE テスト> <FIRST 範囲|NEXT 範囲>|RECORD 番号}
```

パラメーター

名前	説明
IF テスト	検索する値または条件。文字リテラル値は引用符で囲み、日付時刻値は逆引用符で囲む必要があります。
WHILE テスト 省略可能	各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。 メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。
FIRST 範囲 NEXT 範囲 省略可能	処理するレコード数： <ul style="list-style-type: none">○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します 範囲は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。
RECORD 番号	位置を示すレコード番号。

例

指定した値と一致する最初のレコードを検索する

次の例では、LOCATE コマンドを使用して、特定の値が最初に現れるテーブルを見つける方法を示します。

```
LOCATE IF Vendor_Name = "United Equipment"
```

```
LOCATE IF Vendor_Name = "Uni"
```

```
LOCATE IF Invoice_Amount > 1000
```

```
LOCATE IF Invoice_Date = `20141231`
```

指定した条件または式と一致する最初のレコードを検索する

次の例では、LOCATE コマンドを使用して、特定の条件または式が最初に現れるテーブルを見つける方法を示します。

```
LOCATE IF Vendor_Name = "United Equipment" AND Invoice_Amount > 1000  
AND Invoice_Date > `20140930`
```

```
LOCATE IF Vendor_City = v_city
```

レコード番号によってレコードを検索する

次の例では、LOCATE コマンドを使用して、テーブル内の特定のレコードへ移動する方法を示します。

```
LOCATE RECORD 50
```

備考

このコマンドの動作の詳細については、「一致する最初のレコードの選択」ページ 1229を参照してください。

機能の仕組み

LOCATE コマンドは、指定した値または条件と一致する、テーブル内の最初のレコードに直接移動する場合に使用します。

指定した値または条件が見つかった場合、テーブル内で最初に一致するレコードが選択されます。指定した値または条件が見つからなかった場合、テーブルは最初のレコードに位置付けられます。

また、LOCATE を使用すると、特定のレコード番号に直接移動できます。

LOCATE と FIND および SEEK の比較

FIND コマンドや SEEK コマンドとは違い、LOCATE コマンドでは、検索対象はインデックス付きのテーブルや単一の文字フィールドに制限されません。LOCATE を使用すると、あらゆる型のリテラル、任意のデータ型を使用した式、データ型が混在した式を検索することができます。

インデックス付きでないテーブルの検索に使用した場合、LOCATE コマンドはテーブル内の各レコードを順次処理する必要があるため、FIND コマンドや SEEK よりも処理速度が著しく低下する可能性があります。必要な処理時間は、テーブルのサイズ、一致するレコードの場所、および WHILE、FIRST、または NEXT を使用して検索範囲を狭めているかどうかによって異なります。

部分一致がサポートされる場合とは

文字検索では、部分一致がサポートされます。フィールドまたは検索対象フィールドに含まれる長い値の一部を検索値に指定できるのです。ただし、検索値は、一致を成すフィールドの先頭に現れる必要があります。

部分一致を有効または無効にする

SET コマンドを使用するか、**オプション**]ダイアログボックスを使用して、部分一致を有効または無効にできます。

部分一致を有効にする	部分一致を無効にする
<p>指定: <code>SET EXACT OFF</code></p> <p>または</p> <p>選択解除: [オプション]ダイアログボックス(ツール>オプション>テーブル)の 正確な文字比較]</p> <p>結果: フィールドまたは検索対象フィールドに含まれる長い値の一部を検索値に指定できます。検索値は、一致を成すフィールドの先頭に現れる必要があります。</p>	<p>指定: <code>SET EXACT ON</code></p> <p>または</p> <p>選択: [オプション]ダイアログボックス(ツール>オプション>テーブル)の 正確な文字比較]</p> <p>結果: 検索値は、一致を成すフィールドの値と正確に一致しなければなりません。</p>

SET EXACT の詳細については、"SET コマンド" ページ 2081を参照してください。

正確な文字比較を行う]オプションの詳細については、"テーブルオプション" ページ 125を参照してください。

LOOP コマンド

指定された条件が真と評価されている間、一連の ACLScript コマンドをレコードで繰り返し実行します。

メモ

LOOP コマンドは GROUP コマンドで囲む必要があります。

構文

```
LOOP WHILE テスト  
  コマンド  
  <...n>  
END
```

パラメーター

名前	説明
WHILE テスト	LOOP コマンド内のコマンドを実行するために、True と評価される必要があるテスト。そのテストが True と評価された場合、それが False と評価されるまで対象コマンドが繰り返し実行されます。
コマンド <...n>	実行対象となる 1 つまたは複数のコマンド。 LOOP コマンド内には複数のコマンドを入力することができます。各コマンドは別々の行で入力する必要があります。
END	LOOP コマンドの終わり。

例

カンマ区切りのフィールドを分割する

請求データを含むテーブルがあり、請求金額の特定の情報を部署ごとに抽出する必要があるとします。1 つの請求書は複数の部署に関連付けることができ、部署コードがカンマ区切りでテーブルに保存されます。

請求金額を部署ごとに抽出するには

1. GROUP コマンドを使用して、テーブルの各レコードを処理します。
2. 各レコードに関連付けられた部署数 (n) を計算します。
3. LOOP コマンドを使用して、レコードに関連付けられた各部署のデータを抽出する操作を n 回繰り返します。

```
COMMENT
GROUP を使用すると、各部署コード フィールドのコメントをカウントし、レコードに関連付けら
れている部署数を特定します
LOOP でフィールドの各コードをレコードごとに「ループ」します。ループを繰り返すたびに、単一
のコードのレコードが result1 テーブルに抽出されます
END

GROUP
v_department_count = OCCURS(Dept_Code, ',')
v_counter = 0
LOOP WHILE v_counter <= v_department_count
v_dept = SPLIT(Dept_Code, ',', (v_counter + 1))
EXTRACT FIELDS Invoice_Number, Amount, v_dept AS "Department" TO
result1
v_counter = v_counter + 1
END
END
```

備考

ヒント

LOOP コマンドおよび GROUP コマンドの詳細なチュートリアルについては、"統制構造" ページ 1467および"グループ化とループ処理" ページ 1470を参照してください。

LOOP の用途

ループは、処理するデータのセグメントがレコード内に繰り返し含まれている場合に多く用いられます。

機能の仕組み

各 LOOP コマンドでは、テストする WHILE 条件を指定し、END ステートメントで閉じる必要があります。指定したテストが True と評価される間は、LOOP と END の間にあるコマンドが現在のレコードに対して繰り返し実行されます。

テストが False になると、その時点でコマンドは実行されなくなります。

無限ループを回避する

無限ループを回避するために、必ず最終的にFalseが返されるテストを指定してください。また、[SET LOOP コマンド](#)を使用すると、無限ループを防ぐことができます。

MERGE コマンド

同一の構造を持つ2つの並べ替え済み Analytics テーブルのレコードを結合して、元のテーブルと同じ並べ替え順になっている新しい Analytics テーブルに出力します。

構文

```
MERGE {{ON キーフィールド|ON ALL <EXCLUDE フィールド名 <...n>>}}|{PKEY 主キー フィールド|PKEY ALL <EXCLUDE フィールド名 <...n>>}} {SKEY 副キー フィールド|SKEY ALL <EXCLUDE フィールド名 <...n>>}} <IF テスト> TO テーブル名 <LOCAL> <OPEN> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <APPEND> <PRESORT> <ISOLocale ロケール コード>
```

メモ

MERGE では、文字型のフィールドまたは文字型の演算フィールドのみ、キーフィールドとして使用できます。

主テーブルおよび副テーブルのキーフィールドは、どちらも昇順で並べ替えられている必要があります。一方または両方のキーフィールドが並べ替えられていない、あるいは降順で並べ替えられている場合には、MERGE コマンドは失敗します。

PRESORT を使用すると、主テーブルのキーフィールドを並べ替えることができます。副テーブルのキーフィールドが並べ替えられていない場合は、マージを実行する前に、まず別個の並べ替え操作でフィールドの並べ替えを行ってください。

主テーブルと副テーブルは、並べ替える代わりにインデックスを付けることができます。大きなテーブルでは並べ替えよりインデックス作成を行った方が、テーブルのマージに必要な時間を短くすることができます。

パラメーター

名前	説明
ON キーフィールド ON ALL	<p>メモ</p> <p>主および副テーブルの対応するキーフィールドの名前が同じ場合にのみ ON を使用できます。フィールド名が異なる場合や実際の物理フィールドでなく式の場合は、PKEY と SKEY を使用する必要があります。</p> <p>主テーブルと副テーブルの両方のキーフィールド。</p> <ul style="list-style-type: none"> ON キーフィールド - 指定されたフィールドを使用します。フィールドはリストする順序で使用されます。

名前	説明
	<ul style="list-style-type: none"> ○ ON ALL - はテーブルのすべてのフィールドを使用します フィールドは、テーブルレイアウトに表示される順序で使用されます。
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用してマージするときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>ON ALL EXCLUDE field_1 field_2</pre> </div>
PKEY 主テーブルのキー フィールド PKEY ALL	<p>主テーブルのキー フィールド(複数可)または式。</p> <ul style="list-style-type: none"> ○ PKEY 主キー フィールド - 指定されたフィールドを使用します フィールドはリストする順序で使用されます。 ○ PKEY ALL - テーブルのすべてのフィールドを使用します フィールドは、テーブルレイアウトに表示される順序で使用されます。
EXCLUDE フィールド名 省略可能	<p>PKEY ALL を使用してマージするときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、PKEY ALL を微調整できます。</p> <p>EXCLUDE は PKEY ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>PKEY ALL EXCLUDE field_1 field_2</pre> </div>
SKEY 副テーブルのキー フィールド SKEY ALL	<p>副テーブルのキー フィールド(複数可)または式。</p> <ul style="list-style-type: none"> ○ SKEY 副キー フィールド - 指定されたフィールドを使用します フィールドはリストする順序で使用されます。 ○ SKEY ALL - テーブルのすべてのフィールドを使用します フィールドは、テーブルレイアウトに表示される順序で使用されます。
EXCLUDE フィールド名 省略可能	<p>SKEY ALL を使用してマージするときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、SKEY ALL を微調整できます。</p> <p>EXCLUDE は SKEY ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>SKEY ALL EXCLUDE field_1 field_2</pre> </div>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p>

名前	説明
	<p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字()を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出力ファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
APPEND	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p>

名前	説明
省略可能	<p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
PRESORT 省略可能	<p>コマンドを実行する前に主キーフィールドで主テーブルを並べ替えます。</p> <p>メモ</p> <p>GROUP コマンドの内部では PRESORT を使用することができません。</p> <p>以下の場合、PRESORT を省略します。</p> <ul style="list-style-type: none"> ◦ 主テーブルのキーフィールドが既に並べ替えられている場合 ◦ インデックス付きの共通のキーフィールドを使用して2つのテーブルをマージする場合
ISOLOCALE ロケールコード 省略可能	<p>メモ</p> <p>Analytics の Unicode 版にのみ適用されます。</p> <p>システムロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルトシステムロケールが使用されます。</p>

例

キーフィールド名が同一のテーブルをマージする

次の例では、キーフィールド名が同一である2つのテーブルをマージします。

```
OPEN Employees_Location_1 PRIMARY
OPEN Employees_Location_2 SECONDARY
MERGE ON Last_Name TO "AllEmployees" PRESORT
```

キーフィールド名が異なるテーブルをマージする

次の例では、キーフィールド名が異なる2つのテーブルをマージします。

```
OPEN Employees_Location_1 PRIMARY
OPEN Employees_Location_2 SECONDARY
MERGE PKEY Last_Name SKEY Surname TO "AllEmployees" PRESORT
```

備考

このコマンドの動作の詳細については、「テーブルのマージ」ページ 951を参照してください。

マージの代替手段

マージを正常に実行するには、特別な方法があります。追加してから並べ替える、あるいは抽出または追加してから並べ替えるのでも、同じ結果を得ることができます。

詳細については、「APPEND コマンド」ページ 1651と「EXTRACT コマンド」ページ 1801を参照してください。

2つのソーステーブルが既に並べ替えられている場合は、マージはより効率的で、より迅速に実行できます。

NOTES コマンド

Analytics テーブル内の個々のレコードに関連付けられている注釈(ノート)を作成、編集、または削除します。

構文

```
NOTES <IF テスト> <TEXT ノート テキスト> <APPEND> <CLEAR>
```

パラメーター

名前	説明
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p> <ul style="list-style-type: none"> IF テストを指定しなければ、ノートテキストはテーブルの各レコードに追加されます。 IF テストとCLEAR を指定すると、条件を満たすレコードのノートが削除されます。
TEXT ノートテキスト 省略可能	<p>テキストはノートとして追加されます。ノートテキストは、引用符で囲まれた文字列、または文字式である必要があります。</p>
APPEND 省略可能	<p>ノート テキストは既存ノートの末尾に追加されます。省略した場合は、既存のノートがすべて上書きされます。</p>
クリア 省略可能	<p>ノートは削除されます。テーブル内のすべてのレコード ノートを削除しても、自動生成された RecordNote フィールドはテーブルレイアウトから削除されません。</p>

例

複数のレコードに同じノートを追加する

次の例では、指定したレコードの既存のノートがすべて上書きされます。

```
NOTES IF MATCH(RECNO()),1,3,5,7) TEXT "ノート本文"
```

複数のレコードに同じノートを追加または付加する

次の例では、指定したレコードの既存のノートに、新しいノートテキストが付加されます。

```
NOTES IF MATCH(RECNO()),1,3,5,7) TEXT "ノート本文" APPEND
```

複数のレコードからノートを削除する

次の例では、テーブルのすべてのレコードメモが削除されます。

```
NOTES CLEAR
```

次の例では、指定したレコードのノートが削除されます。

```
NOTES IF MATCH(RECNO()),1,3,5,7) CLEAR
```

次の例では、レコード 1 ~ 100 のノートが削除されます。

```
NOTES IF RECNO() <= 100 CLEAR
```

備考

RecordNote フィールドを削除する

テーブルレイアウトから **RecordNote** フィールドを削除することでテーブル内のすべてのノートを削除するには、`DELETE NOTES` コマンドをオプションの指定なしで使用します。

NOTIFY コマンド

受信者に中継する SMTP メールサーバーに電子メール通知メッセージを送信します。

構文

認証されているコマンド構文

```
NOTIFY SMTP SMTPUSER SMTPアカウント名 PORT ポート番号 <ENABLESSL> USER 送信元の電子メールアドレス PASSWORD 暗号化されたパスワード MAILBOX SMTPサーバー名 ADDRESS 受信者の電子メールアドレス <CC cc 受信者の電子メールアドレス> <BCC bcc 受信者の電子メールアドレス> SUBJECT 件名 MESSAGE メッセージテキスト <ATTACHMENT ファイルパス>
```

認証されていないコマンド構文

メモ

この以前のバージョンの NOTIFY コマンドは、ローカルメールシステム、または認証を必要としない SMTP メールサーバーで使用するためのものです。通常認証が必要な最新の電子メールシステムでは動作しません。

```
NOTIFY USER送信元の電子メールアドレス <PASSWORD暗号化されたパスワード> MAILBOXSMTPサーバー名 ADDRESS受信者の電子メールアドレス <CCcc 受信者の電子メールアドレス> <BCCbcc 受信者の電子メールアドレス> SUBJECTメッセージの件名 MESSAGE メッセージの本文テキスト <ATTACHMENT ファイルのパス>
```

パラメーター

名前	説明
SMTP	SMTP サーバー(Simple Mail Transfer Protocol サーバー) に対して認証された送信を行うことを指定します。 <ul style="list-style-type: none"> SMTP- サーバーが認証された接続を必要とする SMTP を指定します

名前	説明
	<p>オープンな SMTP サーバーを使用し、ポート 25 を使用して接続している場合、または認証を必要としないローカルメールシステムを使用している場合は、SMTP- を指定しないでください</p> <p>メモ SMTP を使用する事実上すべての新しい電子メールシステムは、認証が必要です。</p> <p>以下のパラメーターを有効にするには、SMTP を指定する必要があります。</p> <ul style="list-style-type: none"> ○ SMTPUSER ○ PORT ○ ENABLESSL
SMTPUSER SMTPアカウント名	<p>SMTP サーバーに対して認証を行い、アクセスするためのユーザーアカウント名。</p> <p>SMTPUSER を省略すると、USER によって指定された送信元電子メールアドレスを使用して、ローカルメールシステム、または認証を必要としない SMTP メールサーバーにアクセスします。</p>
PORT ポート番号	<p>SMTP サーバーへのアクセスに使用するポート番号。</p> <p>IT 部門に依頼して、使用するポートを調査しなければならない場合があります。SMTP サーバーの一般的なポート番号は次のとおりです。</p> <ul style="list-style-type: none"> ○ 587 ○ 2525 ○ 465 <p>SMTPUSER を使用する場合、PORT を使用する必要があります。SMTPUSER を省略した場合、PORT は省略でき、デフォルトのポート 25 が使用されます。</p>
ENABLESSL 省略可能	<p>セキュリティ保護された SMTP/SSL 接続を使用して SMTP サーバーにアクセスすることを指定します。</p> <p>ご使用の SMTP サーバーが SSL 接続を必要とするかどうかについては、IT 部門に問い合わせることが必要になる場合があります。または、電子メールで ENABLESSL を指定したものとしないものの両方をテストで送信してみます。</p>
USER 送信元の電子メールアドレス	<p>SMTP サーバー経由で送信する電子メール通知の送信元の電子メールアドレスを定義します。</p> <p>メモ SMTP サーバーの設定によっては、USER と SMTPUSER が同じアカウントである場合とそうでない場合があります。</p>
PASSWORD 暗号化されたパスワード	<p>SMTP サーバーのユーザーアカウントパスワード。パスワードは暗号化された文字列として指定する必要があります。</p> <p>詳細については、「暗号化されたパスワードの生成」ページ 2000 を参照してください。</p> <p>ローカルメールシステムまたは認証を必要としない SMTP メールサーバーにアクセスするには、パスワードは不要な場合があります。</p>
MAILBOX SMTPサーバー	<p>電子メールメッセージを送信するために使用する SMTP サーバーのドメイン名。例：</p>

名前	説明
名	<pre>MAILBOX "smtp.example.com"</pre>
ADDRESS 受信者の電子メールアドレス	1人または複数人の受信者の電子メールアドレス。 複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。最大 1020 文字を入力します。
CC cc 受信者の電子メールアドレス 省略可能	1人または複数人のカーボンコピー受信者の電子メールアドレス。 複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。最大 1000 文字を入力します。
BCC bcc 受信者の電子メールアドレス 省略可能	1人または複数人のブラインドカーボンコピー受信者の電子メールアドレス。 複数人の受信者の電子メールアドレスの場合は、カンマで区切ります。
SUBJECT メッセージの件名	電子メールメッセージの件名。
MESSAGE メッセージの本文テキスト	電子メールメッセージの本文テキスト。 メッセージはプレーンテキストのため、HTML はサポートされていません。メッセージに改行を挿入したい場合は、2つのカラット文字 (^) を使用します。
ATTACHMENT ファイルのパス 省略可能	1つ以上の添付ファイルのパスとファイル名。 ファイルが Analytics プロジェクトと同じフォルダーにある場合は、パスを指定する必要はありません。 パスとファイル名を引用符で囲みます。ファイルパスにカンマで区切ったファイルのリストを入力して、複数の添付ファイルを指定します。 <pre>ATTACHMENT "result1.csv,result2.csv"</pre> メモ カンマ区切りのリストでは、カンマの後にスペースがないようにしてください。

例

エラー報告の電子メールを送信する

メモ

認証されたコマンド構文。

スクリプトを実行し、失敗した場合には電子メール通知を送信したいとします。NOTIFY を使って電子メールメッセージを定義し、以下の2つの添付ファイルを添付します。

- ログファイル
- エラーが記録された .fil ファイル

```
NOTIFY SMTP SMTPUSER "mail_admin@example.com" PORT 587 ENABLESSL USER
"support@company.com" PASSWORD "9QZC2524830864..."MAILBOX
"smtp.company.com" ADDRESS "script_admin@company.com" SUBJECT "エラーレ
ポート" MESSAGE "スクリプトの処理に失敗しました。詳細を添付しました。"ATTACHMENT
"errors.fil,ACL_Demo.log"
```

エラー報告の電子メールを送信する

メモ

認証されていないコマンド構文。

スクリプトを実行し、失敗した場合には電子メール通知を送信したいとします。NOTIFY を使って電子メールメッセージを定義し、以下の2つの添付ファイルを添付します。

- ログファイル
- エラーが記録された .fil ファイル

```
NOTIFY USER "support@company.com" MAILBOX "mail.company.com" ADDRESS
"script_admin@company.com" SUBJECT "エラーレポート" MESSAGE "スクリプトの処理
に失敗しました。詳細を添付しました。"ATTACHMENT "errors.fil,ACL_Demo.log"
```

備考

受信者と添付ファイル

NOTIFY コマンドを使用すると、1人以上の受信者に電子メール通知メッセージを送信することができます。メッセージには、データファイルや Analytics プロジェクトを添付して含めることができます。

NOTIFY コマンドの一般的な使用法は、スクリプトが予期せず失敗したときに適切な担当者に警告することです。

互換性のある SMTP サーバー

NOTIFY コマンドは、SMTP (Simple Mail Transfer Protocol) をサポートするメールサーバーで使用でき、認証接続を必要とする幅広い SMTP サーバーで動作するはずですが、ただし、すべての SMTP サーバーやサービスで動作するわけではありません。たとえば、NOTIFY は Gmail では動作しません。サードパーティアプリを Google アカウントに接続するには、NOTIFY がサポートするユーザー名とパスワード以上のセキュリティ構成が必要です。

一般に、NOTIFY を使用して SMTP サーバーに電子メールを正常に送信するには、Analytics 側だけでなく SMTP サーバー側にも互換性のある構成が必要であることに注意してください。NOTIFY が接続に失敗し、電子メールを正常に送信できない場合は、IT 部門と連携して、Analytics とサーバーの間に互換性のある接続構成があることを確認してください。

暗号化されたパスワードの生成

メモ

Analytics は、サーバープロファイルのパスワード、データベースプロファイルのパスワード、またはメールサーバーのパスワードを指定または入力するときに最大 30 文字サポートしています。この長さを超えるパスワード値を指定すると、パスワードが切り詰められ、接続エラーになります。

NOTIFY コマンドの PASSWORD パラメーターとともに暗号化されたパスワード値を指定する必要があります。NOTIFY で使用するパスワードを適切に暗号化するには、Analytics のユーザーインターフェイスの **通知** ダイアログボックスにパスワードを入力する必要があります。

1. 省略可能。Analytics のコマンドラインで、`SET NOTIFYRETRYATTEMPTS TO 0` と入力します
NOTIFY の再試行を一時的にオフにすることで、暗号化されたパスワードの生成が高速化されます。
2. Analytics のメインメニューから **ツール > 電子メールによる通知** を選択します。
3. **通知** ダイアログボックスで、必須のフィールドをすべて記入し、必要に応じて任意のフィールドに記入します。

送信者	送信元の電子メールアドレス	必須
パスワード	SMTP サーバーのユーザーアカウントパスワード (クリアテキスト)。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> メモ パスワードが 30 文字を超えると、NOTIFY コマンドは失敗します。 </div>	必須
メールボックス パス	SMTP サーバー名	必須
変換先	受信者の電子メールアドレス	必須

Cc	cc 受信者の電子メールアドレス	省略可能
Bcc	bcc 受信者の電子メールアドレス	省略可能
件名	メッセージの件名	必須
テキスト	メッセージテキスト	必須
添付ファイル	ファイルのパス	省略可能

4. **[OK]**をクリックします。

認証を必要とするSMTP サーバーに接続しようとする、NOTIFY コマンドが成功したように見えても、接続の試行は(予想どおり)失敗します。

5. NOTIFY コマンドをログからコピーしてスクリプトに貼り付けます。

NOTIFY コマンドにPASSWORD の暗号化された値が格納されます。

6. スクリプトで、コピーされた NOTIFY コマンドに次のパラメーターを追加し、適切なパラメーター値を指定します。

パラメーターは、**NOTIFY** キーワードの直後に挿入できます。

- SMTP
- SMTPUSER SMTPアカウント名
- PORT ポート番号
- ENABLESSL(必要な場合)

7. 以前に NOTIFY 再試行回数を **0** に指定した場合は、コマンドラインに **SET NOTIFYRETRYATTEMPTS TO 5** と入力します。

再試行回数を **5** ではなく、別の回数に設定していた場合は、その該当する回数を入力します。

8. スクリプトを実行して NOTIFY コマンドをテストします。

SMTP メールサーバーで動作するようにすべての必須の値を正しく構成すると、NOTIFY コマンドは成功するようになります。

接続の試行に失敗した場合の Analytics の応答の仕方

Analytics をメールサーバーに接続できない場合は、接続の試行が 10 秒おきに 5 回繰り返されます。この自動再試行機能は、メールサーバーが一時的に利用できなくなった場合の問題に対応します。

すべての接続試行に失敗した場合、NOTIFY コマンドはキャンセルされ、ログにメッセージが書き込まれますが、コマンドを含むスクリプトは処理を続行します。

SET コマンドを使用すると、デフォルトの再試行動作を変更できます。以下を指定できます。

- 異なる接続試行回数
- 追加の接続試行なし

- 異なる試行間隔
- NOTIFY コマンドが接続に失敗し、キャンセルされた場合のスキプトの失敗

SET コマンドの詳細については、"SET コマンド" ページ 2081を参照してください。

SET コマンド構文	例および備考
SET NOTIFYRETRYATTEMPTS <TO> 数	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px; text-align: center;"> SET NOTIFYRETRYATTEMPTS TO 10 </div> <p>デフォルト 設定 : 5 回</p> <p>NOTIFY コマンドが最初の試行に失敗した後、電子メールの送信を試行する回数を指定します。0 から 255 までの数値を入力します。0 を入力すると、最初の失敗後に追加の試行は行われません。</p> <p>NOTIFY コマンドが電子メールの送信に失敗することについて考えられる理由の 1 つは、電子メールサーバーを利用できないということです。</p>
SET NOTIFYRETRYINTERVAL <TO> 秒数	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px; text-align: center;"> SET NOTIFYRETRYINTERVAL TO 30 </div> <p>デフォルト 設定 : 10 秒</p> <p>NOTIFYRETRYATTEMPTS 間の秒数を指定します。1 から 255 までの数値を入力します。</p>
SET NOTIFYFAILSTOP {ON OFF}	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px; text-align: center;"> SET NOTIFYFAILSTOP ON </div> <p>デフォルト 設定 : OFF</p> <ul style="list-style-type: none"> ◦ NOTIFYFAILSTOP が OFF - の場合 スクリプト内の NOTIFY コマンドが失敗しても、スクリプトの続行が Analytics によって許可されます。 ◦ NOTIFYFAILSTOP が ON - の場合 スクリプト内の NOTIFY コマンドが失敗すると、Analytics により、スクリプトの処理が停止されると共にメッセージがログに書き込まれます。スクリプトは、最初の失敗後に停止するか、または NOTIFYRETRYATTEMPTS に指定された試行回数後、どの試行も成功しなかった場合に停止します。 <p>メモ</p> <p>電子メールの受信者が無効である場合は、NOTIFY コマンドの失敗とは見なされず、NOTIFYFAILSTOP の設定にかかわらず、スクリプトが停止することはありません。</p>

OPEN コマンド

Analytics テーブルおよび関連するデータ ファイルを開きます。

構文

```
OPEN {テーブル名 | データ ファイル <FORMAT レイアウト名>} <BUFFERLENGTH 長さ> <CRLF>
<DBASE> <INDEX インデックス ファイル> <PRIMARY|SECONDARY> <SKIP バイト> <RELATION
キー フィールド>
```

パラメーター

名前	説明
テーブル名	開く Analytics テーブルの名前。
データファイル	FORMAT レイアウト名で指定されたテーブルに関連付けるデータファイル。 拡張子が指定されない場合、Analytics はファイル拡張子を .fil と仮定します。拡張子のないファイルを開くには、ファイル名の最後にピリオド(.)を入れてください。
FORMAT レイアウト名 省略可能	開くデータファイルに適用する Analytics テーブルレイアウト。
BUFFERLENGTH 長さ 省略可能	テーブルに割り当てる入力バッファー領域の長さ(バイト数)。デフォルト値は 33,000 バイトです。 バッファー領域を増やすと処理速度は向上しますが、Analytics コマンドを格納するために使用可能な RAM が使われます。 バッファー長を越す IBM 可変長ブロックを読み取ると、Analytics はエラーメッセージを表示して処理を停止します。デフォルト値は、 オプション]ダイアログボックスの テーブル]タブにある [バッファー サイズ] フィールドで設定します。 BUFFERLENGTH <i>n</i> のデフォルト値は、ほとんどの状況を処理できるような設定になっているため、パラメーターの変更が必要になることはほとんどありません。
CRLF 省略可能	可変長 ASCII ファイルが読み取られることを指定します。変化するレコード長は Analytics によって自動的に調整されます。 デフォルトで、ファイルは固定長ファイルであると仮定されます。
DBASE 省略可能	データソースが dBASE ファイルであることを指定します。Analytics によって dBASE ファイルの種類が認識され、そのファイル記述を基に自動的にテーブルが作成されます。.dbf ファイル拡張子の付いた dBASE ファイルでは省略することができます。
INDEX インデックスファイル	テーブルを開くとき、そのテーブルに適用されるインデックスファイル。

名前	説明						
省略可能	インデックスファイル名に拡張子を指定しないと、.inx と見なされます。INDEX は、主テーブルまたは副テーブルのいずれかに指定することができます。						
PRIMARY SECONDARY 省略可能	テーブルを主テーブルまたは副テーブルのいずれとして開くかを指定します。省略した場合、テーブルは主テーブルとして開かれます。						
SKIP バイト 省略可能	<p>テーブルの物理的な先頭部分でバイパスするバイトの数。</p> <p>テーブルのヘッダーレコード、あるいはテーブル先頭部分でテーブルのレイアウトに従っていない部分を見捨てる場合に SKIP を使用できます。省略した場合、テーブルは最初のバイトから読み取られます。</p> <p>メモ</p> <table border="1"> <tbody> <tr> <td>非 Unicode 版 Analytics</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、拡張 ASCII (ANSI) データ</td> <td>1 バイト = 1 文字</td> </tr> <tr> <td>Unicode 版 Analytics、Unicode データ</td> <td>2 バイト = 1 文字</td> </tr> </tbody> </table> <p>Unicode データでは、偶数バイトのみを指定します。奇数バイトを指定すると、文字が正しく表示されない可能性があります。</p>	非 Unicode 版 Analytics	1 バイト = 1 文字	Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字	Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字
非 Unicode 版 Analytics	1 バイト = 1 文字						
Unicode 版 Analytics、拡張 ASCII (ANSI) データ	1 バイト = 1 文字						
Unicode 版 Analytics、Unicode データ	2 バイト = 1 文字						
RELATION キーフィールド 省略可能	<p>テーブルを一時的に関連付けられたテーブルとして開くことを指定します。Analytics は、テーブルを閉じるときにこの関係を保存しません。</p> <p>RELATION を使用する際は、INDEX パラメーターも指定する必要があります。キーフィールドは、ACL が 2 テーブル間の関係を作成するためのキーフィールドまたは式です。</p>						

例

テーブルレイアウトを指定してテーブルを開く

次の例では、**March_2012** テーブルレイアウトを使って **April_2012** テーブルを開きます。

```
OPEN April_2012 FORMAT March_2012
```

dBASE ファイルを開く

次の例では、既存のテーブルが存在しない **Inventory.dbf** という名前の dBASE ファイルを開きます。

```
OPEN Inventory
```

テーブルを開き、既存のインデックスを適用する

主または副いずれかのテーブルを開いて、そのテーブルに既存のインデックスを適用するには、次の構文を使用します。

```
OPEN Accounts_receivable INDEX Customer_number_AR
```

```
OPEN Customer SECONDARY INDEX Customer_number
```

テーブルを開き、別のテーブルとの間の一時的な関連付けを作成する

Customers テーブル(主テーブル)を開き、このテーブルと **Accounts_receivable** テーブル(副テーブル)の間に一時的な関連付けを作成する必要があります。

Customer_index というインデックスと、主テーブル内の **Last_name** というキーフィールドを使用する場合、コマンドは次のようになります。

```
OPEN Accounts_receivable INDEX Customer_index RELATION Last_name
```

OUTLIERS コマンド

数値型フィールドにおける統計上の異常値を検出します。異常値の検出対象は、数値型フィールド全体であるか、あるいは1つまたは複数の文字型、数値型、または日付時刻型キーフィールドの値に基づく複数のグループです。

構文

```
OUTLIERS {AVERAGE|MEDIAN} {PKEY キー フィールド <...n>|PKEY ALL <EXCLUDE フィールド名 <...n>>|NOKEY} ON 数値フィールド <OTHER フィールド <...n>|OTHER ALL <EXCLUDE フィールド名 <...n>> NUMSTDEV 標準偏差数 <IF テスト> <TO {SCREEN|テーブル名}> <PRESORT> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <OPEN>
```

メモ

サーバーのテーブルに対してローカルで OUTLIERS コマンドを実行することはできません。OUTLIERS コマンドはその全体を指定する必要があります。簡略化することはできません。

パラメーター

名前	説明
AVERAGE MEDIAN	<p>数値型フィールド(異常値フィールド)の値の中心点を計算する方法。</p> <ul style="list-style-type: none"> ○ AVERAGE - を指定すると、値の平均(平均値)が計算されます。 ○ MEDIAN - を指定すると、値の中央値が計算されます。 <p>中心点は、以下のいずれかについて計算されます。</p> <ul style="list-style-type: none"> ○ 数値型フィールド全体 ○ 各キーフィールド グループの数値 <p>これらの中心点は、後で数値型フィールドまたは各グループの標準偏差を計算する際に使用されます。</p> <p>メモ MEDIAN を指定する場合は、数値型フィールドを基準にして並べ替えを行っておく必要があります。数値型フィールドを基準にした並べ替えがまだ行われていない場合は、PRESORT を使用します。</p>

名前	説明
	<p>ヒント</p> <p>異常値がないかどうかを調べるデータに大きな偏りがある場合は、MEDIANを指定した方が、データの大勢をより正しく表す結果を生成することができます。</p>
<p>PKEY キー フィールド <...n> PKEY ALL NOKEY</p>	<p>テーブルのデータをグループ化するために使用する1つ以上の文字、数値、日付時刻フィールド。</p> <p>NOKEYを指定した場合は、データがグループ化されず、フィールドレベルの異常値が検出されます。</p> <p>メモ</p> <p>キーフィールドを並べ替える必要があります。1つまたは複数のフィールドを基準にした並べ替えがまだ行われていない場合は、PRESORTを使用します。</p> <ul style="list-style-type: none"> <p>PKEY キー フィールド - 指定したフィールドを使用して、テーブルのデータをグループ化します。複数のフィールドはスペースで区切る必要があります。また、異なるデータ型を指定できます。</p> <p>複数のフィールドを指定すると、出力テーブルで入れ子のグループが作成されます。入れ子でのフィールド間の順序は、フィールドを指定した順になります。</p> <p>グループごとに、数値型フィールドにおけるそのグループの数値に関する標準偏差が計算されます。このグループの標準偏差は、グループの異常値を検出するための基準値として使用されます。</p> <p>PKEY ALL - では、テーブルのデータをグループ化するために、テーブル内のすべてのフィールドが使用されます。</p> <p>すべてのフィールドを指定した場合、出力テーブルにネストされたグループが作成されます。ネストは、テーブルレイアウトで指定した順序に従います。</p> <p>グループごとに、数値型フィールドにおけるそのグループの数値に関する標準偏差が計算されます。このグループの標準偏差は、グループの異常値を検出するための基準値として使用されます。</p> <p>メモ</p> <p>すべてのフィールドでグループ化すると、数値フィールドが含まれますが、意味がない場合があります。EXCLUDEを使用すると、数値フィールドをグループ化から除外できます。</p> <p>NOKEY - は、テーブルのデータをグループ化しません。</p> <p>数値型フィールド全体に対する標準偏差が計算されます。このフィールドの標準偏差は、フィールドの異常値を検出するための基準値として使用されます。</p>
<p>EXCLUDE フィールド名 省略可能</p>	<p>PKEY ALL を使用してテーブルデータをグループ化するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、PKEY ALL を微調整できます。</p> <p>EXCLUDE は PKEY ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>PKEY ALL EXCLUDE <i>field_1 field_2</i></p> </div>

名前	説明
ON 数値フィールド	<p>異常値がないかどうかを調べる数値型フィールド。一度に1つのフィールドしか調べることができません。</p> <p>異常値とは、フィールドまたは主キーグループの標準偏差またはそのような標準偏差の指定倍数によって設定される上限と下限の範囲に含まれない値のことです。</p>
OTHER フィールド <...n> OTHER ALL 省略可能	<p>出力に含める1つ以上の追加フィールド。</p> <ul style="list-style-type: none"> OTHER フィールド<...n>- 指定されたフィールドが含まれます フィールドはリストする順序で含まれます。 OTHER ALL -キーフィールドまたは異常値フィールドに指定されていない、テーブルのすべてのフィールドが含まれます フィールドはテーブルレイアウトの順序で含まれます。 <p>メモ キーフィールドと異常値フィールドは、自動的に出力テーブルに追加されるため、OTHERに指定する必要はありません。</p>
EXCLUDE フィールド名 省略可能	<p>OTHER ALL を使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、OTHER ALL を微調整できます。</p> <p>EXCLUDE は OTHER ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>OTHER ALL EXCLUDE field_1 field_2</pre> </div>
NUMSTDEV 標準偏差数	<p>数値型フィールドにおいて、平均または中央値から異常値の上限および下限までに含まれる標準偏差の数。任意の正の整数または10進数(0.5、1、1.5、2、...)を指定できます。</p> <p>異常値の境界を作成するための式：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> $\text{平均/中央値} \pm (\text{標準偏差数} * \text{標準偏差})$ </div> <p>メモ 標準偏差はデータセットの分布の測定です。つまり、値の拡散方法を測定します。異常値計算は母集団標準偏差を使用します。</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <h3 style="text-align: center;">異常値の境界の例</h3> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>NUMSTDEV 2</pre> </div> <p>数値型フィールド全体または各キーフィールドグループに対し、以下が設定されます。</p> <ul style="list-style-type: none"> 平均または中央値より標準偏差の2倍だけ大きい、異常値の上限 <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> $\text{平均/中央値} + (2 * \text{SD})$ </div> </div>

名前	説明
	<p>• 平均または中央値より標準偏差の2倍だけ小さい、異常値の下限</p> <p><code>平均/中央値 - (2 * SD)</code></p> <p>上限より大きいか下限より小さい任意の値が、異常値として出力結果に追加されます。</p> <p>メモ 同じデータセットに対し、標準偏差数の値を大きくすると、返される異常値の数が減る可能性があります。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>TO SCREEN テーブル名 省略可能</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
<p>PRESORT 省略可能</p>	<p>当該のコマンドを実行する前に並べ替え操作を実行します。</p> <p>ヒント 入力テーブル内の1つまたは複数の該当フィールドを基準にした並べ替えが既に行われている場合は、PRESORT を指定しないことで処理時間を短縮できます。</p>

名前	説明										
	<table border="1"> <thead> <tr> <th>PRESORT と以下を指定した場合:</th> <th>並べ替えの基準:</th> </tr> </thead> <tbody> <tr> <td>PKEY, AVERAGE</td> <td> <ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールドまたはフィールド、次に数値フィールド(数値フィールドが計算される場合) <p>メモ 計算された数値フィールドの並べ替えは内部的な Analytics の技術要件です。</p> </td> </tr> <tr> <td>PKEY, MEDIAN</td> <td>キーフィールドまたはフィールド、次に数値フィールド</td> </tr> <tr> <td>NOKEY, AVERAGE</td> <td>並べ替えなし</td> </tr> <tr> <td>NOKEY, MEDIAN</td> <td>数値フィールド</td> </tr> </tbody> </table>	PRESORT と以下を指定した場合:	並べ替えの基準:	PKEY, AVERAGE	<ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールドまたはフィールド、次に数値フィールド(数値フィールドが計算される場合) <p>メモ 計算された数値フィールドの並べ替えは内部的な Analytics の技術要件です。</p>	PKEY, MEDIAN	キーフィールドまたはフィールド、次に数値フィールド	NOKEY, AVERAGE	並べ替えなし	NOKEY, MEDIAN	数値フィールド
PRESORT と以下を指定した場合:	並べ替えの基準:										
PKEY, AVERAGE	<ul style="list-style-type: none"> キーフィールドまたはフィールド キーフィールドまたはフィールド、次に数値フィールド(数値フィールドが計算される場合) <p>メモ 計算された数値フィールドの並べ替えは内部的な Analytics の技術要件です。</p>										
PKEY, MEDIAN	キーフィールドまたはフィールド、次に数値フィールド										
NOKEY, AVERAGE	並べ替えなし										
NOKEY, MEDIAN	数値フィールド										
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>										
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数:</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>										
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。</p>										

例

異常な取引金額の検出

Sample Project.acl 内の Ar テーブル全体から、異常な取引金額を検出したいとします。

異常値の上下限を「Amount(金額)フィールドから標準偏差の±3 倍」に設定します。次のテストにより、772 個のレコードから成るテーブルから、16 個の異常値が返されます。

```
OPEN Ar
OUTLIERS AVERAGE NOKEY ON Amount NUMSTDEV 3 PRESORT TO "Outliers_
AR.fil" OPEN
```

「標準偏差の±3.5倍」にしてテストを繰り返します。異常値の上下限がAmountフィールドの値群の中心点から遠くなったため、6個の異常値がこのテストによって返されます。

```
OPEN Ar
OUTLIERS AVERAGE NOKEY ON Amount NUMSTDEV 3.5 PRESORT TO "Outliers_
AR.fil" OPEN
```

顧客別の異常な取引金額の検出

Sample Project.acl内のArテーブルから、顧客別に異常な取引金額を検出したいとします。異常値の上下限を「各顧客の取引グループから標準偏差の±3倍」に設定します。

```
OPEN Ar
OUTLIERS AVERAGE PKEY No ON Amount NUMSTDEV 3 PRESORT TO "Outliers_
Customer_AR.fil" OPEN
```

このテストにより、7個の異常値が返されます。各顧客の取引グループについて、標準偏差と平均が報告されます。

	顧客番号(No.)	取引金額	STDEV	AVERAGE	グループ番号
1	065003	4,954.64	1015.58	833.83	1
2	262001	3,567.34	772.44	438.81	2
3	262001	(2,044.82)	772.44	438.81	2
4	376005	(931.55)	411.18	484.57	3
5	501657	5,549.19	1332.80	441.14	4
6	811002	3,409.82	634.20	672.10	5
7	925007	3,393.87	736.48	906.16	6

顧客 262001 に関する異常値の特定方法

Ar テーブルには顧客 262001 の取引が 101 個あり、そのうち 2 つが異常値として報告されたとします。これらは、その顧客の異常値境界を超えるためです。

異常値	下限	上限	異常値
(2,044.82)	(1,878.51)	2,756.13	3,567.34

顧客 262001 に関する異常値境界の計算方法

異常値の上下限は、顧客 262001 の全取引の平均に、取引の標準偏差の指定倍数をプラスマイナスしたものです。

顧客 262001 の全取引の平均	438.81
標準偏差の指定倍数	3
取引の標準偏差	772.44
	$438.81 \pm (3 * 772.44)$ $= 438.81 \pm 2,317.32$ $= (1,878.51) \text{ (下限)}$ $= 2,756.13 \text{ (上限)}$

MEDIAN を使用した、顧客別の異常な取引金額の検出

AVERAGE の代わりに MEDIAN を使用した場合でも、上記の例で行ったのと同じ異常値テストを行うことができます。

```
OPEN Ar
OUTLIERS MEDIAN PKEY No ON Amount NUMSTDEV 3 PRESORT TO "Outliers_
Customer_AR_Median.fil" OPEN
```

上のテストでは、直前のテストで返された 7 個の異常値の代わりに、10 個の異常値が返されます。MEDIAN と AVERAGE では、データの性質に応じて多少異なる結果が返されます。

	顧客番号 (No.)	取引金額	STDEV	MEDIAN	グループ番号
1	065003	4,954.64	1015.58	663.68	1
2	262001	(2,044.82)	772.44	450.67	2
3	262001	3,567.34	772.44	450.67	2
4	376005	(931.55)	411.18	517.16	3
5	501657	4,426.14	1332.80	146.80	4
6	501657	5,549.19	1332.80	146.80	4
7	811002	3,409.82	634.20	624.53	5
8	925007	2,972.78	736.48	717.88	6
9	925007	3,030.71	736.48	717.88	6
10	925007	3,393.87	736.48	717.88	6

各顧客に関する異常値境界の計算方法

異常値の上下限は、各顧客の取引の中央値に、取引の標準偏差の指定倍数をプラスマイナスしたものです。

顧客 262001 の例 : $450.67 \pm (3 * 772.44)$

備考

このコマンドの動作の詳細については、「異常値の特定」ページ 1159を参照してください。

異常値境界フィールドを結果テーブルに追加する

Analytics は、自動的に、**STDEV** および **AVERAGE** または **MEDIAN** 演算フィールドを異常値結果テーブルに追加します。結果テーブルで異常値を特定するために使用される異常値境界を示す 2 つの演算フィールドも追加すると役立つ場合があります。

1. 異常値結果テーブルを開きます。
2. この式を Analytics コマンドラインに貼り付けて、必要に応じて編集し、Enter を押します。

```
DEFINE FIELD 下限 COMPUTED AVERAGE - (標準偏差数 * STDEV)
```

- 標準偏差数については、使用した実際の標準偏差乗数を代替します。
 - 平均ではなく中央点として中央値を使用した場合は、AVERAGEにMEDIANを使用します。
3. この式を Analytics コマンドラインに貼り付けて、必要に応じて編集し、Enter を押します。

```
DEFINE FIELD 上限 COMPUTED AVERAGE + (標準偏差数 * STDEV)
```

- 標準偏差数については、使用した実際の標準偏差乗数を代替します。
 - 平均ではなく中央点として中央値を使用した場合は、AVERAGEにMEDIANを使用します。
4. ビュー内で右クリックしてから **列の追加**]を選択します。
 5. **使用可能なフィールド**]リストから、**下限**および**上限**をダブルクリックし、**選択したフィールド**]リストに追加します。
 6. **OK**]をクリックします。
 7. 省略可能。列ヘッダーをドラッグして、追加したフィールドを再配置します。

PASSWORD コマンド

スクリプトの実行中にパスワードの入力をユーザーに求めるために、パスワード値を含まないパスワード定義を作成します。

構文

```
PASSWORD 番号 <プロンプト>
```

パラメーター

名前	説明
数字	パスワード定義を一意に識別する1から10までの値。
プロンプト 省略可能	パスワードの入力を求めるために使用されるダイアログボックスに表示される文字型の有効な式。この場合、リテラル文字列は引用符で囲みます。 プロンプトを省略した場合は、メッセージのないデフォルトダイアログボックスが表示されます。

例

パスワード情報の入力を求める

PASSWORD コマンドを使用して、必要な3つのパスワードを入力するようスクリプトでユーザーに求めるとします。ユーザーが必要なパスワードを入力したら、スクリプトは中断することなく、残りの処理を完了することができます。

```
PASSWORD 1 "売掛金データベースのパスワードを入力してください"  
PASSWORD 2 "買掛金データベースのパスワードを入力してください"  
PASSWORD 3 "顧客データベースのパスワードを入力してください"
```

Analytics テーブルを更新するときパスワードを指定する

パスワードで保護されたデータファイルを更新するために、REFRESH コマンドと PASSWORD コマンドを組み合わせた例を次に示します。

```
PASSWORD 1 "パスワード:"  
REFRESH Abc PASSWORD 1
```

サーバー テーブルを定義するためにパスワードを指定する

AX Connector 経由でサーバー テーブルを定義するとき、データベース プロファイルと関連するサーバー プロファイルのパスワードを必要とする場合には、DEFINE TABLE DB コマンドと共に PASSWORD コマンドを使用できます。

```
DEFINE TABLE DB SOURCE Inventory_DBProfile PASSWORD 9 PASSWORD 3
```

備考

PASSWORD の用途

PASSWORD コマンドを使用すると、スクリプトがパスワード保護されたデータにアクセスするか、パスワード保護されたデータをインポートまたは構成する前に、ユーザーのパスワード入力を要求します。

スクリプトでは最大 10 個の異なるパスワード定義を作成できます。

PASSWORD は次の場合に有用です。

- スクリプトに実際のパスワードを入力する (SET PASSWORD コマンドで要求される) ことを避けたい場合
- 各ユーザーが異なるパスワードを入力する必要がある場合

パスワードの保存方法

ユーザーが入力したパスワードは一時的に安全にメモリに格納されます。

ユーザーがプロンプト ダイアログボックスにパスワードを入力するときには、アスタリスク (*) を使用して、文字がマスクされます。パスワードは、スクリプトにもログにも表示されません。

サーバーに基づくアナリティクスのパスワードを保存する

PASSWORD コマンドは、ロボット、またはレガシーサーバースクリプトのアナリティクススクリプト実行ではサポートされません。

ユーザーがロボットでアナリティクススクリプトをスケジュールするときには、パスワードを確認するために [PASSWORD アナリティクスタグ](#)を使用できます。

[SET PASSWORD コマンド](#)を使用すると、レガシーサーバースクリプトでパスワードを指定できます。

PAUSE コマンド

スクリプトを一時停止し、ユーザーのダイアログボックスに情報を表示します。

構文

```
PAUSE メッセージ <IF テスト>
```

パラメーター

名前	説明
メッセージ	ダイアログボックスに表示するメッセージ。199文字を上限とします。 メッセージは二重引用符で囲む必要があります。メッセージ内に二重引用符がある場合は、二重引用符を一重引用符で囲んでください。
IF テスト 省略可能	各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。 メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。

例

エラーメッセージを表示する

特定の要件を満たすユーザー入力が必要であるとします。入力が要件を満たさないことを検出した場合には、PAUSE コマンドを使ってエラーメッセージをダイアログボックスに表示します。次のようになります。

```
PAUSE "この製品クラスの値は 2 桁にしてください。"
```

備考

PAUSE の用途

PAUSE の用途は、スクリプトの実行中に読み取り専用のメッセージを画面に表示することです。エラーメッセージや分析操作の結果などの情報を表示できます。

機能の仕組み

メッセージダイアログボックスが表示されている間、スクリプトの実行は停止され、ユーザーが [OK] をクリックしてメッセージダイアログボックスを閉じた後にのみ再開されます。このため、無人で実行する必要があるスクリプトやアナリティクスで PAUSE を使用することはできません。

制限

PAUSE には次の制限があります。

- GROUP コマンド内に含めることができない
- ロボットで実行されるアナリティクスで使用できない

PREDICT コマンド

予測モデルをラベルがないデータセットに適用し、個別のレコードに関連付けられたクラスまたは数値を予測します。

メモ

Analytics を 32 ビット コンピューターで実行する場合は、PREDICT コマンドはサポートされません。このコマンドに必要な演算は、プロセッサの負荷が高いため、64 ビット コンピューターの方が適しています。

構文

```
PREDICT MODEL モデル名 TO テーブル名 <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
```

パラメーター

名前	説明
MODEL モデル名	<p>クラスまたは値を予測するために使用するモデルファイルの名前。以前に TRAIN コマンドで生成されたモデルファイルを使用します。</p> <p>ファイル拡張子 <code>*.model</code> を指定する必要があります。例：</p> <pre>MODEL "Loan_default_prediction.model"</pre> <p>メモ モデルファイルは、ラベルのないデータセットと同じフィールドまたはほぼ同じフィールドのデータセットで学習されている必要があります。 バージョン 14.1 の Analytics で学習されたモデルファイルは使用できません。 バージョン 14.1 のモデルファイルは、それ以降のバージョンの Analytics と互換性がありません。PREDICT コマンドと使用して新しい予測モデルを学習してください。</p>
TO テーブル名	<p>予測プロセスで生成された Analytics テーブル</p> <p>この出力テーブルには、学習プロセス中に指定したキーフィールドと、予測プロセスで生成された 1 つまたは 2 つのフィールドが含まれます。</p> <ul style="list-style-type: none"> 予測 - ラベルがないデータセットの各レコードに関連付けられた予測されたクラスまたは数値。

名前	説明
	<ul style="list-style-type: none"> ○ 確率 - (分類のみ) 予測されたクラスが正しい確率 <p>テーブル名、.FIL 拡張子を持つ引用符で囲まれた文字列として指定します。例: <code>TO "Loan_applicants_default_predicted.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ○ <code>TO "C:\Loan_applicants_default_predicted.FIL"</code> ○ <code>TO "ML Predict output\Loan_applicants_predicted_default.FIL"</code> <p>メモ テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数:</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>

例

分類モデルを使用して、予測を行う

分類モデルを PREDICT コマンドを入力し、融資が実行された場合に、現在の融資申請者が債務不履行になる予測を行います。

以前に、融資のデフォルト情報を含む履歴融資データのセットに対して TRAIN コマンドを使用して、分類モデルを生成しました。

```
OPEN "Loan_applicants_current"  
PREDICT MODEL "Loan_default_prediction.model" TO "Loan_applicants_  
default_predicted.FIL"
```

回帰モデルを使用して予測を行う

回帰モデルを PREDICT コマンドに入力することで、将来の住宅販売価格を予測します。

あなたは以前に、販売価格を含む最近の住宅販売データのセットに対して TRAIN コマンドを使用して、回帰モデルを作成しました。

```
OPEN "House_price_evaluation"  
PREDICT MODEL "House_price_prediction.model" TO "House_prices_  
predicted.FIL"
```

備考

このコマンドの動作の詳細については、「クラスと数値の予測」ページ 1350を参照してください。

PRINT コマンド

テキスト ファイル、Analytics ログ ファイル、その他外部ファイルとしてエクスポートされた Analytics プロジェクト項目 (スクリプト (.aclscript)、テーブル (.layout)、ワークスペース (.wsp) など) を印刷します。コマンドで生成されたグラフも印刷できます。

構文

```
PRINT {ファイル名 | GRAPH}
```

パラメーター

名前	説明
ファイル名 GRAPH	印刷する項目： <ul style="list-style-type: none">◦ ファイル名 - 印刷するファイルの相対または絶対パスとファイル名 例: "C:\ACL Data\Sample Data Files\ACL_Demo.log" または "Sample Data Files\ACL_Demo.log"。 パスまたはファイル名にスペースが含まれている場合は、ファイル名を引用符で囲む必要があります。◦ GRAPH - コマンドの結果として以前に出力されたグラフを印刷します

例

ログ ファイルを印刷する

ACL_Demo.acl プロジェクト用のログ ファイルを印刷するには、次のコマンドを指定します。

```
PRINT "C:\ACL Data\Sample Data Files\ACL_Demo.log"
```

グラフを印刷する

BENFORD コマンドから生成されたグラフを印刷するには、次のコマンドを指定します。

```
OPEN Metaphor_APTrans_2002  
BENFORD ON Invoice_Amount LEADING 1 TO GRAPH  
PRINT GRAPH
```

備考

プリンターを選択する

使用されるプリンターは、Microsoft Windows で設定された通常使うプリンターです。プリンターを変更するには Windows の通常使うプリンターを変更する必要があります。

関連コマンド

プロジェクト内の Analytics テーブルの内容を印刷するには、DO REPORT コマンドを使用します。

PROFILE コマンド

Analytics テーブルで 1 つまたは複数の数値フィールドまたは数値式について要約統計値を生成します。

構文

```
PROFILE {<FIELDS> 数値フィールド <...n>|<FIELDS> ALL <EXCLUDE 数値フィールド
<...n>>} <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
```

パラメーター

名前	説明
FIELDS 数値フィールド <...n> FIELDS ALL	個別のフィールドを指定してプロファイルするか、ALL を指定して Analytics テーブルのすべての数値フィールドをプロファイルします。
EXCLUDE 数値フィールド 省略可能	FIELDS ALL を使用したプロファイリングでのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。 EXCLUDE はただちに FIELDS ALL に従う必要があります。例： <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
IF テスト 省略可能	各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。 メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。
WHILE テスト 省略可能	各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。 メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。
FIRST 範囲 NEXT 範囲	処理するレコード数：

名前	説明
省略可能	<ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>

例

単一フィールドのプロファイルを作成する

Salary フィールドのプロファイルを作成するコマンドを次に示します。

```
OPEN Employee_Payroll
PROFILE FIELDS Salary
```

このコマンドにより、次の出力が生成されます。

フィールド名	合計値	絶対値	最小	最大
給与 (Salary)	1,152,525	1,152,525	15,340	52,750

備考

出力に表示される統計

コマンドに指定された数値フィールドまたは数値式ごとに次の統計が表示されます。

- 合計値
- 絶対値
- 最小値
- 最大値

QUIT コマンド

現在のセッションを終了し、Analytics を終了します。

メモ

Analytics を終了せずに、スクリプトを終了するには、"ESCAPE コマンド" ページ 1769を参照してください。

構文

```
QUIT
```

例

ファイルが存在するかどうかを確認して、存在しない場合に Analytics を終了させる例を考えます。

他のユーザーに実行させるスクリプトを作成しましたが、必要なファイルが存在しない場合に Analytics を終了させる必要があるとします。

次の例では、必要な **Inventory.csv** ファイルが存在するかどうかを確認して、存在しない場合に Analytics を終了させています。

```
IF FILESIZE("Inventory.csv") = -1 QUIT
```

スクリプトが完了したら自動的に Analytics を終了させる

次のスクリプトは、Inventory テーブルを要約し、出力結果を生成して、自動的に Analytics を終了させています。

```
OPEN Inventory
SUMMARIZE ON Location ProdCls SUBTOTAL Value TO "Inventory_value_by_
location_class.FIL" PRESORT CPERCENT
QUIT
```

備考

変更は保存される

QUIT を実行すると、Analytics が終了させられる前に、開いているすべての Analytics テーブルが保存されて閉じられます。

アクティブなビューやスクリプトの変更を保存していない場合、Analytics は終了前にその変更を保存するように指示するメッセージを表示します。

RANDOM コマンド

一連の乱数を生成します。

構文

```
RANDOM NUMBER n <SEED シード値> MINIMUM 最小値 MAXIMUM 最大値 <COLUMNS n>
<UNIQUE> <SORTED> <TO {SCREEN|ファイル名}> <APPEND>
```

パラメーター

名前	説明
NUMBER <i>n</i>	生成される乱数一式のサイズ。 最大 32767 の数字を生成できます。
SEED シード値 省略可能	乱数ジェネレーターを初期化するために使用される値。 シード値を指定する場合、任意の数字を指定することができます。シード値がそれぞれ一意であると、異なる乱数一式になります。同じシード値を指定すると、同じ乱数一式が生成されます。分析を複製する場合は、同じ乱数一式を再生成しなければならないことがあります。 <ul style="list-style-type: none"> シード値 -には、特定の乱数一式を複製したい場合のシード値を明示的に指定し、保存します。 シード値なし -には、シード値を Analytics にランダムに選択させたい場合に、'0' またはブランクのシード値を入力します。
MINIMUM 最小値	乱数一式に使用可能な最小の数値。有効な任意の数値または数式が使用できます。
MAXIMUM 最大値	乱数一式に使用可能な最大の数値。有効な任意の数値または数式が使用できます。
COLUMNS <i>n</i> 省略可能	乱数一式の表示に使用する列数。 COLUMNS を省略した場合のデフォルトは 6 列です。
UNIQUE 省略可能	一意の数値のみが乱数一式に含まれます。 UNIQUE を省略した場合は、乱数一式内で重複する値が生成される可能性があります。

名前	説明
	<p>メモ</p> <p>乱数一式の指定サイズが最小値と最大値の差異の75%を超える場合は、UNIQUEを選択しないでください。選択すると、乱数の選択肢が破棄される数が多くなりすぎます。</p>
SORTED 省略可能	<p>乱数一式が昇順で表示されます。</p> <p>SORTEDを省略した場合は、数字が無作為な順序で選択されて表示されます。</p>
TO SCREEN ファイル名 省略可能	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ ファイル名 - は結果の保存先となるファイルです。 <p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> <p>TOを省略した場合は、乱数一式が画面に出力されます。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>

例

100 個の乱数が含まれるテキスト ファイルを生成する

10,000 ~ 20,000 の番号が付いた一連のファイルから、ランダムに 100 個のハード コピー ファイルを取り出したいとします。

10,000 ~ 20,000 の範囲に含まれる 100 個の乱数が含まれたテキスト ファイルを生成するには、RANDOM コマンドを使用することができます。その後で、それらの乱数に一致する番号の付いたハード コピー ファイルを取り出します。これらの番号は、一意であり、10 個の列に昇順で並べられます。

```
RANDOM NUMBER 100 SEED 45387 MINIMUM 10000 MAXIMUM 20000 COLUMNS 10  
UNIQUE SORTED TO "Random_Numbers.txt"
```

備考

このコマンドの動作の詳細については、"乱数の生成" ページ 234を参照してください。

乱数アルゴリズム

RANDOM コマンドでは、Analytics のデフォルトの乱数アルゴリズムが使用されます。SAMPLE コマンドと異なり、RANDOM コマンドではメルセンヌツイスター乱数アルゴリズムは使用できません。

RCOMMAND コマンド

Analytics テーブルを外部 R スクリプトにデータフレームとして渡し、外部 R スクリプトからの出力を使用して、Analytics プロジェクトで新しいテーブルを作成します。

構文

```
RCOMMAND FIELDS フィールド <...n> RSCRIPT スクリプトへのパス TO テーブル名 <IF テスト>
<WHILE テスト> <FIRST 範囲|NEXT 範囲> <KEEPTITLE> <SEPARATOR 文字> <QUALIFIER
文字> <OPEN>
```

パラメーター

名前	説明
FIELDS フィールド名 <...n>	<p>R スクリプトに送信されるデータフレームに追加する、ソースの Analytics テーブル内のフィールド、あるいは式。</p> <p>使用する Analytics のエディションによっては、次の特殊文字を含むデータを R スクリプトに送信するときにエラーが発生する可能性があります。</p> <ul style="list-style-type: none"> 非 Unicode - "\" Unicode - "y" または "S" 両方 - ボックスは、ブロック、黒い四角、および垂直の破損した棒などの文字を描画します <p>メモ 日本語の文字と中国語の文字が両方とも含まれているテーブルなど、複数の言語が混在するデータも、サポートされていません。</p>
RSCRIPT スクリプトのパス	R スクリプトの、ファイルシステムにおける絶対パスまたは相対パス。スクリプトへのパスは引用符で囲んでください。
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> テーブル名 - は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p>

名前	説明
	<ul style="list-style-type: none"> TO "C:\Output.FIL" TO "Results\Output.FIL" <p>メモ テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <p>R スクリプトによって返すデータフレームまたはマトリクスから作成する出力テーブルの名前。</p>
IF テスト 省略可能	現在のレコードを処理するのに満たす必要がある条件。R スクリプトに渡されるデータフレームには、この条件を満たすレコードのみが含まれます。
WHILE テスト 省略可能	各レコードを処理するために真である必要がある条件式。条件がfalseと評価するか、テーブルの最後に達したら、コマンドは実行を中止します。
FIRST 範囲 NEXT 範囲 省略可能	処理するレコード数： <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します 範囲は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。
KEEPTITLE 省略可能	データの代わりに、フィールド名としてデータの最初の行を処理します。省略すると、汎用フィールド名が使用されます。 このオプションは、R スクリプトの列名を使ってデータを取得したい場合に必要です。
SEPARATOR 文字 省略可能	フィールド間の区切りとして使用する文字。文字は引用符で囲まれた文字列として指定する必要があります。 デフォルトの<文字>はカンマです。
QUALIFIER 文字	フィールド値を折り返すためと識別するためにテキスト修飾子として使用する文字。文字は

名前	説明
省略可能	引用符で囲まれた文字列として指定する必要があります。 デフォルトの<文字>は二重引用符です。 <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 20px;"> メモ 入力フィールドに表示される文字を使用しないでください。QUALIFIER 文字が入力データに表示される場合は、結果に影響する可能性があります。 </div>
OPEN 省略可能	コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。

例

R スクリプト (Hello world) を起動し、稼働する

Hello world スクリプトを作成し、Analytics と R スクリプト の接続をテストする場合があります。

Analytics コマンド

```
RCOMMAND FIELDS "Hello", ", world!" TO "r_result" RSCRIPT
"C:\scripts\r_scripts\analysis.r"
```

R スクリプト (analysis.r)

```
srcTable<-acl.readData()

# ACL に返送するテーブルを作成
output<-data.frame(
  c(srcTable[1,1]),
  c(srcTable[1,2])
)

# 列名を追加して ACL にテーブルを返送
colnames(output) <- c("Greeting","Subject")
acl.output<-output
```

行と列の座標を使用してフィールド データにアクセスする

いくつかの請求書関連フィールドを Analytics 外での分析のため R スクリプトに送信するとします。

Analytics コマンド

```
RCOMMAND FIELDS Department_Code Invoice_Amount Invoice_Date Invoice_
Number Vendor_Number TO "r_result" RSCRIPT "C:\scripts\r_
scripts\analysis.r"
```

R スクリプト (analysis.r)

```
# R スクリプトでデータ フレームの 2 行目から請求書番号を取り出す
srcTable<-ac1.readData()
srcTable[2,4]
```

列名を使用してフィールド データにアクセスする

いくつかの請求書関連フィールドを Analytics 外での分析のため R スクリプトに送信するとします。列が R スクリプト内で名前呼び出せるように、KEEPTITLE オプションを使用します。

Analytics コマンド

```
RCOMMAND FIELDS Department_Code Invoice_Amount Invoice_Number TO "r_
result" RSCRIPT "C:\scripts\r_scripts\analysis.r" KEEPTITLE
```

R スクリプト (analysis.r)

```
# R スクリプトでデータ フレームの 2 行目から請求書番号を取り出す
srcTable<-ac1.readData()
srcTable["2","Invoice_Number"]
```

1000.00 の金額を超える請求書レコードを R スクリプトに送信する

いくつかの請求書関連フィールドを Analytics 外での分析のため R スクリプトに送信するとします。R スクリプトに送信するレコード数を制限するには、IF を使用します。1000.00 を超える請求金額が含まれるレコードのみが送信されます。

Analytics コマンド

```
RCOMMAND FIELDS Department_Code Invoice_Amount Invoice_Number TO "r_result" IF Invoice_Amount > 1000.00 RSCRIPT "C:\scripts\r_scripts\analysis.r" KEEPTITLE
```

R スクリプト (analysis.r)

```
# R スクリプトでデータ フレームの 2 行目から請求書番号を取り出す  
srcTable<-ac1.readData()  
srcTable["2","Invoice_Number"]
```

請求書レコードを送信し、乗算した請求書金額を返す

いくつかの請求書関連フィールドを Analytics 外での分析のため R スクリプトに送信するとします。この R スクリプトでは、指定した列のすべてのセルに対して単一のアクションが実行されます。

Analytics コマンド

```
RCOMMAND FIELDS Department_Code Invoice_Amount Invoice_Number TO "r_result" RSCRIPT "C:\scripts\r_scripts\analysis.r" KEEPTITLE
```

R スクリプト (analysis.r)

```
# 2 倍の値の ACL テーブルのスライスを返す
srcTable<-acl.readData()
acl.output<-srcTable["Invoice_Amount"] * 2
```

備考

このコマンドの動作の詳細については、"R スクリプトの実行" ページ 1379を参照してください。

R スクリプトでの Analytics データの参照

Analytics テーブルは R データフレームとしてスクリプトに渡されます。データフレームはタブ形式のデータオブジェクトであり、データの異なるモードまたは型の列を含むことがあります。

R スクリプトで Analytics によって作成されたデータ型を操作するには、`acl.readData()` 関数を呼び出し、返されたデータフレームを変数に格納します。

```
# Analytics テーブルをデータ フレーム myTable に格納します。これはスクリプトを使用して参照
できます
myTable<-acl.readData()
```

データフレームのセルからデータを取得するには、次のアプローチのうちの 1 つを使用できます。

- 行と列の座標の使用：

```
# データ フレームの 1 行 2 列の値を取得します。
myTable[1,2]
```

メモ

座標はコマンドで指定されたフィールドの順序に基づいているのであり、テーブルレイアウトや、現在開いているビューに基づいているものではありません。

- 行と列の名前の使用：

```
# データ フレームの 1 行 "myColumnTitle" 列の値を取得します。
myTable["1", "myColumnTitle"]
```

列の名前を使用するには、コマンドの `KEEPTITLE` オプションを指定する必要があります。

行は、"1"、"2"、"3"と命名されており、そのように増加します。名前と座標の組み合わせを使用することもできます。

Analytics へのデータの返送

データフレームまたはマトリクスを Analytics に戻し、新しいテーブルを作成するには、次の構文を使用します。

```
# myNewTable データ フレームを Analytics に戻し、新しいテーブルを作成します
aql.output<-myNewTable
```

メモ

R スクリプトが終了するときに、データフレームまたはマトリクスを Analytics に戻す必要があります。データフレームまたはマトリクスの列には原子値のみが含まれ、リスト、マトリクス、配列、非原子オブジェクトがないことを確認します。値を Analytics データ型に変換できない場合は、コマンドが失敗します。

データ型のマッピング

Analytics データ型は、Analytics プロジェクトと R スクリプトの間の翻訳プロセスを使用して、R データ型に変換されます。

Analytics データ型	R データ型
論理	論理
数値	数値
文字	文字
日付時刻	日付、POSIXct、POSIXlt

パフォーマンスとファイル サイズ制限

R スクリプトの実行と返されるデータの処理にかかる時間は、入力データが 1 GB を超える場合には長くなります。R では、2 GB 以上の入力ファイルはサポートされていません。

また、R に送信されるレコード数もパフォーマンスに影響します。ファイルサイズは同じだがレコード数が異なる 2 つのテーブルでは、レコード数が少ないテーブルの方が処理が高速になります。

複数バイト文字データの処理

中国語のような、複数バイトの文字セットでデータを R に送信する場合、R スクリプトにシステムロケールを適切に設定する必要があります。複数バイトのデータのテーブルを R に正常に送信するには、R スクリ

コマンド

プトの第 1 行に、次の関数が含まれている必要があります。

```
# ロケールを中国語に設定する例  
Sys.setlocale("LC_ALL", "Chinese")
```

Sys.setlocale() の詳細については、R ドキュメントを参照してください。

R のログファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダの `ac1rlang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

REFRESH コマンド

関連するデータソースを基に、Analytics テーブル内のデータを更新します。

構文

```
REFRESH <テーブル名> <PASSWORD >番号>
```

パラメーター

名前	説明
テーブル名 省略可能	更新する Analytics テーブルの名前。テーブル名を指定しない場合、開いているテーブルが更新されます。
PASSWORD 番号 省略可能	<p>使用するパスワード定義。</p> <p>実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。</p> <p>番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、<code>PASSWORD 2</code>により、2番目のパスワードを使用することを指定します。</p> <p>パスワードの入力または設定の詳細については、次を参照してください。</p> <ul style="list-style-type: none">◦ "PASSWORD コマンド" ページ 2015◦ SET PASSWORD コマンド◦ PASSWORD アナリティクスタグ <p>メモ このパスワードは、元のソースデータシステムにアクセスするのに使用します。PDFを除き、ファイルベースのデータソースでパスワードでは、REFRESHを使用できません。</p>

例

パスワードを要求せずにテーブルを更新する

データソースにパスワードが必要でない場合は、REFRESH コマンドと、更新する Analytics テーブルの名前を入力します。

```
REFRESH Invoices
```

対話型のスクリプト内でユーザーに入力するよう求めたパスワードを使ってテーブルを更新する

対話型のスクリプトを作成している場合は、ユーザーにパスワードの入力を求めることができます。

```
PASSWORD 1 "パスワードを入力してください:"  
REFRESH Invoices PASSWORD 1
```

ACCESSDATA コマンドを使用して、パスワード保護されたデータソースから最初にインポートされたテーブルを更新する場合、パスワード プロンプトは自動であり、個別に指定する必要はありません。

```
REFRESH Invoices
```

非対話型のスクリプト内に設定したパスワードを使ってテーブルを更新する

パスワード値の入力をユーザーに求めないようにするには、スクリプト内にパスワードを設定します。

```
SET PASSWORD 1 TO "password"  
REFRESH Invoices PASSWORD 1
```

この方法の欠点は、スクリプト内でパスワードが文字としてはっきりと表示されることです。

備考

このコマンドの動作の詳細については、「Analytics テーブルのデータの更新」ページ 782を参照してください。

機能の仕組み

REFRESH コマンドは、テーブルを定義およびインポートするときに最初に使用される IMPORT コマンドまたは ACCESSDATA コマンドを再実行して、テーブルの内容を更新します。

REFRESH でテーブル内容のみが更新される

REFRESH コマンドでは、Analytics テーブルにおける既存のフィールドの内容のみが更新されます。Analytics テーブルレイアウトは更新されません。

ソースデータの構造が変更された場合 (フィールドが追加または削除された場合など) は、REFRESH を使用できません。データを再インポートする必要があります。

更新がサポートされるデータソース

REFRESH コマンドを使用すると、以下のコマンドのいずれかを使用して作成された Analytics テーブルの内容を更新できます。

- IMPORT ACCESS
- IMPORT DELIMITED
- IMPORT EXCEL
- IMPORT ODBC(レガシー ODBC コマンド)
- IMPORT PDF
- IMPORT PRINT
- IMPORT SAP
- IMPORT XBRL
- IMPORT XML
- ACCESSDATA(ODBC データソース)

メモ

REFRESH コマンドで古い Excel ファイル (*.xls) や Microsoft Access ファイル (*.mdb) を扱うには、32ビット Microsoft Access データベースエンジンがインストールされている必要があります。詳細については、「Microsoft Access データベースエンジンを除外するオプション」ページ 2664を参照してください。

REFRESH と ACCESSDATA

次のガイドラインは、ACCESSDATA コマンドを使用して ODBC データソースからインポートされたテーブルを更新するときに適用されます。

- **開いているテーブル**- 更新するときにテーブルが開いている場合は、一時的に、テーブルのサイズの 2 倍に相当するディスク領域が必要です。ディスク領域が限られている場合は、更新する前にテーブルを閉じてください。
- **Analytics 12** - Analytics バージョン 12 で ACCESSDATA コマンドを使用してインポートされたテーブルは、新しいバージョンの Analytics を使用している場合でも更新できません。
これらのテーブルを更新するには、Analytics 12.5 以降を使用して再インポートしてください。

REFRESH とパスワード

データベースまたはクラウド データ サービスに存在するパスワードで保護されたデータソースで、REFRESH コマンドを使用できます。

Excel ファイルなどのパスワードで保護されたファイルベースのデータソースでは、REFRESH コマンドを使用できません。ただし、パスワードで保護された PDF は例外です。

RENAME コマンド

Analytics プロジェクト 項目またはファイルの名前を変更します。

構文

```
RENAME 項目の種類 名前 <AS|TO> 新しい名前 <OK>
```

パラメーター

名前	説明
項目の種類 名前	<p>名前の変更元となるプロジェクト 項目またはファイルの種類および名前。</p> <p>メモ ほとんどの場合、アクティブであるか、開いているか、使用中の場合には、項目またはファイル名を変更できません。</p> <p>次に示す有効な種類のうち、いずれかを指定します。</p> <ul style="list-style-type: none"> ○ FIELD - 物理的なデータフィールド、演算フィールド、または変数 <ul style="list-style-type: none"> • フィールドを含むテーブルが開いている必要があります。ただし、アクティブなビューにはフィールドを含めることができません。 • ただし、演算フィールドが参照しているフィールドの名前を変更することはできません。 ○ FORMAT - Analytics テーブル ○ INDEX - インデックス ○ REPORT - レポートまたはビュー ○ WORKSPACE - ワークスペース ○ SCRIPT(またはBATCH) - スクリプト ○ DATA - Analytics データ ファイル (*.FIL) ○ FILE - ファイル システムのデータ ファイル ○ LOG - Analytics ログ ファイル (.log) ○ TEXT - テキスト ファイル
AS TO 新しい名前	<p>プロジェクト 項目またはファイルの新しい名前</p> <p>メモ 長さ制限はほとんどの Analytics プロジェクト 項目名に適用されます。詳細については、"Analytics における文字およびサイズの制限" ページ 1414を参照してください。</p>
OK 省略可能	<p>アクションを確認せずに、項目を削除または上書きします。</p>

例

フィールド名の変更

ProdNo フィールドの名前を **ProdNum** に変更する場合があります。追加の確認を行わずにこのアクションを実行する場合は、「OK」を使用します。以上を行うコマンドを次に示します。

```
OPEN Inventory  
RENAME FIELD ProdNo AS ProdNum OK
```

REPORT コマンド

開いている Analytics テーブルに基づき、レポートを書式設定し、生成します。

構文

```
REPORT <ON 内訳フィールド> <PAGE> <NODUPS> <WIDTH 文字数> <AS 表示名>> <...n>
FIELD その他のフィールド <WIDTH 文字数> <AS 表示名> <...n> <SUPPRESS> <NOZEROS>
<LINE n その他のフィールド> <PRESORT <並べ替えフィールド>> <...n> <SUMMARIZED> <SKIP
n> <EOF> <TO {SCREEN|PRINT|ファイル名 <HTML>}> テスト <WHILE テスト> <FIRST 範囲
|NEXT 範囲> <HEADER ヘッダー テキスト> <FOOTER フッター テキスト> <APPEND>
```

パラメーター

名前	説明
ON 内訳フィールド PAGE NODUPS WIDTH 文字 AS 表示名 <...n> 省略可能	<p>レポートをセクションに区切るために使用される文字フィールド。</p> <p>内訳フィールドの値が変わるたびに、新しいレポート セクションと小計が作成されます。レポートをセクションに分割することで、レポートがスキャンしやすくなります。</p> <ul style="list-style-type: none"> ブレイクフィールド-ブレイクフィールドとして使用するフィールド <p>ビューに基づいたレポートを実行 (DO REPORT) するには、ビューにおける左端の 1 つまたは複数の文字型フィールドを内訳フィールドとする必要があります。</p> <ul style="list-style-type: none"> PAGE -は、内訳フィールド値が変化するたびに改ページを挿入します。 NODUPS -ブレイクフィールドから重複表示値を抑制します <p>たとえば、請求書レコードごとに顧客名が記載されている場合、各顧客名の最初のインスタンスだけを記載すれば、レポートが読みやすくなるかもしれません。</p> <ul style="list-style-type: none"> WIDTH 文字 -フィールドの出力長さ(文字数) AS 表示名 -レポートのフィールドの表示名(代替列タイトル) <p>表示名を文字列として指定します。列見出しを改行したい場合は、語句の間にセミコロン (;) を入れます。表示名をフィールド名、またはソース テーブル内の既存の表示名と同じにしたい場合は、AS を使用しないでください。</p> <p>メモ 内訳フィールド、PAGE、NODUPS、または PRESORT を使用するには、ON を指定する必要があります。</p>
FIELD 他のフィールド WIDTH 文字 AS 表示名 <...n>	<p>レポートに含めるフィールド。</p> <ul style="list-style-type: none"> WIDTH 文字 -フィールドの出力長さ(文字数)

名前	説明
	<ul style="list-style-type: none"> ○ AS 表示名 -レポートのフィールドの表示名 (代替列タイトル) <p>表示名を文字列として指定します。列見出しを改行したい場合は、語句の間にセミコロン (;) を入れます。表示名をフィールド名、またはソーステーブル内の既存の表示名と同じにしたい場合は、AS を使用しないでください。</p> <p>SUBTOTAL および ACCUMULATE キーワードは、廃止され、FIELD に取って代わられました。すべての数値フィールドの小計は、自動で計算されます。</p> <p>メモ 内訳フィールドは、自動的にレポートに追加されるため、その他のフィールドに指定する必要はありません。</p>
SUPPRESS 省略可能	レポートから空白の明細行を除外します。
NOZEROS 省略可能	フィールドのゼロ値に空白値を使用します。 たとえば、レポートのフィールド大量のゼロ値がある場合、ゼロ以外の値のみを表示すると読みやすくなります。
LINE <i>n</i> その他のフィールド 省略可能	行番号 <i>n</i> に表示される列とフィールドの出力行数を指定します。 値が指定されない場合、列は単一行をデフォルトとします。 <i>n</i> には、2 ~ 60 に含まれる値 (両端を含む) を指定する必要があります。 レポートの列見出しは、最初の行にあるフィールドによって決まります。その他のフィールドは、レポートに使用するフィールドや式を指定します。
PRESORT 並べ替えフィールド <... <i>n</i> > 省略可能	<ul style="list-style-type: none"> ○ 並べ替え内訳フィールド、1つ以上の内訳フィールドが指定されている場合。 ○ 並べ替え並べ替えフィールド、1つ以上の並べ替えフィールドが指定されている場合。 <p>PRESORT は、並べ替えフィールドとしてリストされていない限り、その他のフィールドとしてリストされたフィールドを並べ替えません。</p>
SUMMARIZED 省略可能	詳細行のない、小計と合計のみのレポートを作成します。 小計は、一意の内訳フィールドの値に対して生成されます。SUMMARIZED を指定しない場合には、KEY で指定した各内訳フィールドの小計だけでなく詳細行も含んだレポートが作成されます。
SKIP <i>n</i> 省略可能	レポート内の詳細行の間に空白行を挿入します。 <i>n</i> には、挿入する行数を示す整数を指定する必要があります。たとえば、SKIP 1 は 1 行おきの行間のレポートを生成します。
EOF 省略可能	ファイルの終わりに達した後、コマンドをもう一度実行します。 これにより、GROUP コマンド内でテーブルの最後のレコードが処理されることが保証されます。すべてのフィールドが以前のレコードを参照する演算フィールドである場合にのみ使用してください。
TO SCREEN PRINT ファイル名 <HTML> 省略可能	コマンドの結果を送信する場所： <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します

名前	説明
	<p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code> 印刷 - 通常使うプリンターに結果を送信します デフォルトでは、レポートはファイルに出力され、ASCII テキスト ファイルとして保存されます。レポートを HTML ファイル(.htm)として出力したい場合は、HTML を指定します。 TO を省略した場合は、レポートが画面に出力されます。
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲 省略可能</p>	<p>処理するレコード数:</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。 FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>HEADER ヘッダーテキスト 省略可能</p>	<p>レポートの各ページの最上部に挿入されるテキスト。 ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
<p>FOOTER フッターテキスト 省略可能</p>	<p>レポートの各ページの最下部に挿入されるテキスト。 フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、</p>

名前	説明
	Analytics のシステム変数である FOOTER の値よりも優先されます。
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>

例

HTML レポートを生成する

次の例では、Ar テーブルからレポートを生成し、書式付きの HTML ファイルに出力します。

```
OPEN Ar
REPORT ON No FIELD Due Type Amount TO "C:\Reports\AR.htm" HTML
```

RETRIEVE コマンド

バックグラウンド処理に送信された Direct Link クエリの結果を取得します。

メモ

RETRIEVE コマンドは、Direct Link がローカルコンピューターと組織の SAP システムにインストールおよび構成されている場合にのみサポートされます。

構文

```
RETRIEVE テーブル名 PASSWORD 番号
```

パラメーター

名前	説明
テーブル名	Direct Link クエリによって Analytics 内に最初に作成されるテーブルの名前。 RETRIEVE を使用する前に、テーブルが存在する必要があります。
PASSWORD 番号	使用するパスワード定義。 実際のパスワードを入力することを求めたり指定したりするには、PASSWORD 番号の構文は使用しません。パスワード定義とは、以前に PASSWORD コマンドか SET PASSWORD コマンドを使用して入力または設定されたパスワードのことです。 番号はパスワード定義の番号です。たとえば、以前に2つのパスワードをスクリプトで設定したり入力したりしている場合、またはアナリティクススクリプトをスケジュールしている場合、 <code>PASSWORD 2</code> により、2番目のパスワードを使用することを指定します。 パスワードの入力または設定の詳細については、次を参照してください。 <ul style="list-style-type: none">"PASSWORD コマンド" ページ 2015SET PASSWORD コマンドPASSWORD アナリティクス タグ <p>メモ このパスワードは、SAP システムにアクセスするのに使用します。</p>

例

バックグラウンド モードでのクエリの結果を取得する

次の例では、パスワードを設定した後、**DD02T_Data** という名前の Analytics テーブルに対するバックグラウンド モードでのクエリの結果を取得します。

```
SET PASSWORD 1 TO "pwd"  
RETRIEVE DD02T_Data PASSWORD 1
```

SAMPLE コマンド

レコード サンプルングまたは金額単位 サンプルング方法を使用して、レコードのサンプルを抽出します。
レコードのサンプルング 金額単位のサンプルング

構文

メモ

フィルターリング(IF ステートメント) や範囲 パラメーターを適用するとサンプルの妥当性を低下させてしまうため、それらのオプションはこの構文に含まれていません。

固定間隔選択方法

```
SAMPLE <ON> RECORD INTERVAL 間隔値 <FIXED 初期値> {RECORD|FIELDS フィールド名  
<...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL> <OPEN>  
<APPEND>
```

セル選択方法

```
SAMPLE <ON> RECORD CELL INTERVAL 間隔値 <RANDOM シード値> {RECORD|FIELDS フィー  
ルド名 <...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL>  
<OPEN> <APPEND> <MERSENNE_TWISTER>
```

ランダム選択方法

```
SAMPLE <ON> RECORD NUMBER サンプルサイズ <RANDOM シード値> <ORDER> {RECORD|FIELDS  
フィールド名 <...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL>  
<OPEN> <APPEND> <MERSENNE_TWISTER>
```

パラメーター

メモ

値を指定する際、3桁の区切り記号は含めないでください。

名前	説明
ON RECORD	レコード サンプリングを使用します。
INTERVAL 間隔値 FIXED 最初の値 CELL INTERVAL 間隔値 NUMBER サンプルサイズ	<p>INTERVAL 間隔値 FIXED 最初の値 固定間隔選択方法を使用します。</p> <p>最初のレコードを選択すると、そのレコードから固定した間隔、つまり固定した距離だけ離れたレコードが続いて選択されます。たとえば、最初に選択したレコードの後には、20件に1件のレコードが選択されます。</p> <ul style="list-style-type: none"> INTERVAL の間隔値 -には、サンプルサイズの計算によって出力された間隔値を指定します。 FIXED の最初の値 -には、選択する最初のレコード番号を指定します。 <p>最初の値にゼロ('0')を指定するか、FIXED を省略した場合は、Analytics により最初のレコードがランダムに選択されます。</p> <p>CELL INTERVAL 間隔値 セル選択方法を使用します。</p> <p>データセットが複数の均等なセルまたはグループに分割され、各セルから1つのレコードがランダムに選択されます。</p> <p>各セルのサイズは間隔値によって指定します。サンプル数の計算によって生成された間隔の値を入力します。</p> <p>NUMBER サンプルサイズ ランダム選択方法を使用します。</p> <p>すべてのレコードがデータセット全体からランダムに選択されます。</p> <p>サンプルサイズの計算によって生成されたサンプルサイズを入力します。</p>
RANDOM シード値 省略可能	<p>メモ セルおよびランダム選択方法の場合のみ。</p> <p>Analytics の乱数ジェネレーターを初期化するために使用するシード値。</p> <p>ゼロ('0')の値を指定するか、またはRANDOM を省略した場合は、シード値がランダムに選択されます。</p>
ORDER 省略可能	<p>メモ ランダム選択方法の場合のみ。</p> <p>FIELDS を指定する場合には、ORDER のみを使用できます。</p> <p>出力結果にORDER フィールドを追加します。</p> <p>このフィールドには、各レコードの選択順序がランダムであることが表示されます。</p>

名前	説明
RECORD FIELDS フィールド名 <...n> FIELDS ALL	<ul style="list-style-type: none"> ◦ RECORD - レコード全体が出力テーブルに含まれます フィールドは、ソーステーブルレイアウトの順序で含まれます。 ◦ FIELDS フィールド名 - レコード全体でなく、個々のフィールドが出力テーブルに含まれます 含めるフィールドまたは式を指定します。複数のフィールドを指定する場合、空白によってそれぞれのフィールドを区切る必要があります。 フィールドは一覧の順序で含まれます。 ◦ FIELDS ALL - すべてのフィールドが出力テーブルに含まれます フィールドは、ソーステーブルレイアウトに表示される順序で含まれます。
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用してサンプリングするときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 - は、結果の保存先となる Analytics テーブルのことです。 テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code> <p>デフォルトでは、テーブルデータファイル (.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ テーブル名は 64 文字の英数字 (.FIL 拡張子を含まない) に制限されています。名前にはアンダースコア文字 (_) を使用できませんが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。 LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
APPEND	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p>

名前	説明
省略可能	<p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
MERSENNE_TWISTER 省略可能	<p>メモ</p> <p>セルおよびランダム選択方法の場合のみ。</p> <p>Analytics の乱数ジェネレーターでは、メルセンヌツイスター アルゴリズムが使用されます。このパラメーターを省略した場合は、Analytics のデフォルトのアルゴリズムが使用されます。</p> <p>メモ</p> <p>バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、単純にデフォルトの Analytics アルゴリズムを使用してください。</p>

例

レコード サンプルの抽出

レコード サンプルを使用して、請求書を含む勘定の規定された統制からの逸脱率を推定します。統計的に有効なサンプルサイズを計算した後、サンプルを抽出する準備ができています。ランダム選択方法を使用します。

以下の例：

- 開いている Analytics テーブルからサンプルを抽出する
- シード値 123456 を指定してランダム選択方法を使用する
- サンプルサイズとして 95 レコードを指定する
- 指定したフィールドのみを出力テーブルに含める
- Analytics の乱数ジェネレーターでメルセンヌツイスター アルゴリズムを使用することを指定する

```
SAMPLE ON RECORD RANDOM 123456 NUMBER 95 FIELDS RefNum CustNum Amount
Date Type TO "Ar_record_sample" OPEN MERSENNE_TWISTER
```

備考

このコマンドの動作の詳細については、"レコード サンプルングの実行" ページ 1051を参照してください。

構文

メモ

フィルターリング(IF ステートメント) や範囲パラメーターを適用するとサンプルの妥当性を低下させてしまうため、それらのオプションはこの構文に含まれていません。

固定間隔選択方法

```
SAMPLE <ON> MUS 数値フィールド INTERVAL 間隔値 <FIXED 最初の値> <CUTOFF 最上層の  
カットオフ値> <SUBSAMPLE> <NOREPLACEMENT> {RECORD|FIELDS フィールド名  
<...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL> <OPEN>  
<APPEND>
```

セル選択方法

```
SAMPLE <ON> MUS 数値フィールド CELL INTERVAL 間隔値 <CUTOFF 最上層のカットオフ値>  
<RANDOM シード値> <SUBSAMPLE> <NOREPLACEMENT> {RECORD|FIELDS フィールド名  
<...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL> <OPEN>  
<APPEND> <MERSENNE_TWISTER>
```

ランダム選択方法

```
SAMPLE <ON> MUS 数値フィールド NUMBER サンプル サイズ POPULATION 絶対値 <RANDOM  
シード値> <SUBSAMPLE> <NOREPLACEMENT> <ORDER> {RECORD|FIELDS フィールド名  
<...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>} TO テーブル名 <LOCAL> <OPEN>  
<APPEND> <MERSENNE_TWISTER>
```

パラメーター

メモ

値を指定する際、3桁の区切り記号は含めないでください。

名前	説明
ON MUS数値フィールド	金額単位サンプリング(monetary unit sampling: MUS)を使用します。 MUS数値フィールドは、サンプリングのベースとして使用する数値フィールドまたは式です。
INTERVAL 間隔値 FIXED 最初の値 CELL INTERVAL 間隔値 NUMBER サンプルサイズ POPULATION 絶対値	<p>INTERVAL 間隔値 FIXED 最初の値 固定間隔選択方法を使用します。</p> <p>最初の金額単位を選択すると、その金額単位から固定した間隔、つまり固定した距離だけ離れた金額単位が続いて選択されます。たとえば、最初に選択した金額単位の後は、5000個おきの金額単位が選択されます。</p> <ul style="list-style-type: none"> INTERVAL の間隔値 -には、サンプルサイズの計算によって出力された間隔値を指定します。 FIXED の最初の値 -には、選択する最初の金額単位を指定します。 <p>最初の値にゼロ('0')を指定するか、FIXED を省略した場合は、Analytics により最初の金額単位がランダムに選択されます。</p> <p>CELL INTERVAL 間隔値 セル選択方法を使用します。</p> <p>データセットが複数の均等なセルまたはグループに分割され、各セルから1つの金額単位がランダムに選択されます。</p> <p>各セルのサイズは間隔値によって指定します。サンプル数の計算によって生成された間隔の値を入力します。</p> <p>NUMBER サンプルサイズ POPULATION 絶対値 ランダム選択方法を使用します。</p> <p>すべての金額単位がデータセット全体からランダムに選択されます。</p> <ul style="list-style-type: none"> NUMBER - のサンプルサイズには、計算によって出力されたサンプルサイズを入力します。 POPULATION の絶対値 -には、MUS数値フィールドの絶対値合計額を指定します。この合計額が、サンプルの選択元の母集団になります。
CUTOFF 最上層のカットオフ値 省略可能	<p>メモ 固定間隔選択方法およびセル選択方法の場合のみ。</p> <p>最上層のカットオフ値。</p> <p>カットオフ値以上のMUS(Monetary Unit Sampling: 金額単位サンプリング)数値フィールドの金額が自動的に選択され、サンプルに取り込まれます。</p> <p>CUTOFF を省略した場合は、間隔値に等しいデフォルトのカットオフ値が使用されます。</p>

名前	説明
RANDOM シード値 省略可能	<p>メモ セルおよびランダム選択方法の場合のみ。</p> <p>Analytics の乱数ジェネレーターを初期化するために使用するシード値。 ゼロ('0')の値を指定するか、またはRANDOMを省略した場合は、シード値がランダムに選択されます。</p>
SUBSAMPLE 省略可能	<p>メモ FIELDSを指定する場合には、SUBSAMPLEのみを使用できます。</p> <p>出力結果にSUBSAMPLEフィールドを追加します。 サンプルフィールドの各金額が複数の独立した取引の合計を表す場合に、サンプリングした各合計金額のうち1つの取引に対してのみ監査手続きを実行したいときには、SUBSAMPLEフィールドの値を使って任意に個々の取引を選択できます。 詳細については、"金額単位 サンプリングの実行" ページ 1078を参照してください。</p>
NOREPLACEMENT 省略可能	<p>同じレコードが複数回選択されることはありません。このため、サンプルにはSIZEコマンドで計算されたレコード数よりも少ないレコードが含まれるようになる可能性があります。 NOREPLACEMENTを省略した場合、またはREPLACEMENTを指定した場合は、レコードを複数回選択できます。</p>
ORDER 省略可能	<p>メモ ランダム選択方法の場合のみ。 FIELDSを指定する場合には、ORDERのみを使用できます。</p> <p>出力結果にORDERフィールドを追加します。 このフィールドには、各レコードの選択順序がランダムであることが表示されます。</p>
RECORD FIELDS フィールド名 <...n> FIELDS ALL	<ul style="list-style-type: none"> ◦ RECORD - レコード全体が出力テーブルに含まれます フィールドは、ソーステーブルレイアウトの順序で含まれます。 ◦ FIELDS フィールド名 - レコード全体でなく、個々のフィールドが出力テーブルに含まれます 含めるフィールドまたは式を指定します。複数のフィールドを指定する場合、空白によってそれぞれのフィールドを区切る必要があります。 フィールドは一覧の順序で含まれます。 ◦ FIELDS ALL - すべてのフィールドが出力テーブルに含まれます フィールドは、ソーステーブルレイアウトに表示される順序で含まれます。
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用してサンプリングするときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。 EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>

名前	説明
TO テーブル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
LOCAL 省略可能	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出カファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出カテーブルを作成する場合にのみ有効です。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
MERSENNE_TWISTER 省略可能	<p>メモ</p> <p>セルおよびランダム選択方法の場合のみ。</p> <p>Analytics の乱数ジェネレーターでは、メルセンヌ ツイスター アルゴリズムが使用されます。このパラメーターを省略した場合は、Analytics のデフォルトのアルゴリズムが使用されます。</p>

名前	説明
	<p>メモ</p> <p>バージョン 12 より前の Analytics で作成された Analytics スクリプトまたはサンプリングの結果との後方互換性が必要な場合は、単純にデフォルトの Analytics アルゴリズムを使用してください。</p>

例

金額単位サンプルの抽出

金額単位サンプリングを使用して、請求書を含む勘定の金額虚偽表示の合計金額を推定します。

統計的に有効なサンプルサイズを計算した後、サンプルを抽出する準備ができています。固定間隔選択方法を使用します。

以下の例：

- 取引金額フィールドに基づいて、開いている Analytics テーブルからサンプルを抽出する
- 間隔値 \$6,283.33 で固定間隔選択方法を使用する
- 選択した最初のレコードに 100,000 (\$1,000 に含まれるセント数) 番目の金額単位を指定する
- 最上層のカットオフとして \$5,000 を使用する
- レコード全体を出力テーブルに含める

```
SAMPLE ON Amount INTERVAL 6283.33 FIXED 1000.00 CUTOFF 5000.00 RECORD
TO "Ar_monetary_unit_sample" OPEN
```

備考

このコマンドの動作の詳細については、「金額単位サンプリングの実行」ページ 1078を参照してください。

SAVE コマンド

Analytics テーブルのコピーを作成してそれを別の名前でも保存したり、Analytics プロジェクトを保存します。

構文

Analytics テーブルのコピーを作成し、それを別の名前でも保存するには

```
SAVE 新規テーブル FORMAT ACLテーブル
```

現在のプロジェクトへの変更を保存するには

```
SAVE
```

パラメーター

名前	説明
新規テーブル	作成または保存する新しい Analytics テーブルの名前。 メモ テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。
FORMAT ACLテーブル	既存の Analytics テーブルの名前。テーブルレイアウトの名前を使用します。関連するデータファイルの名前ではありません。

例

既存のテーブルを基に新しいテーブルを作成する

下記の例では、既存のテーブル `Payables_master` に基づいて、`Payables_March` という名前の新しいテーブルを作成しています。その後で、`Payables_March` を3月の買掛金データファイルに

リンクすることもできます。

```
SAVE Payables_March FORMAT Payables_master
```

備考

機能の仕組み

SAVE FORMAT は、ナビゲーターの **総覧** タブで Analytics テーブルをコピーして貼り付けるのと同様の結果を生成します。新しい Analytics テーブルが作成され、元のテーブルと同じデータ ファイルまたはデータソースに関連付けられます。

必要に応じて、新しく作成されたテーブルを別のデータソースにリンクすることができます。

SAVE を使用して入力要求を回避する

特定の時点において、現在のプロジェクトへの変更を保存するように、Analytics によって要求されます。スクリプトの実行が中断されないようにするために、Analytics から変更の保存を要求される前に SAVE コマンドを使用して変更を保存しておくことができます。

SAVE LAYOUT コマンド

Analytics テーブルレイアウトを外部のテーブルレイアウト ファイル(.layout)に保存するため、またはテーブルレイアウトのメタデータを Analytics テーブルに保存します。

メモ

バージョン 11 より前の Analytics では、外部テーブルレイアウト ファイルのファイル拡張子には .fmt を使用していました。拡張子を手動で指定すれば、拡張子 .fmt のテーブルレイアウト ファイルを今後も保存することができます。

構文

```
SAVE LAYOUT {FILE|TABLE} TO {ファイル名|テーブル名}
```

パラメーター

名前	説明
FILE TABLE	<ul style="list-style-type: none"> FILE - Analytics テーブルレイアウトを外部テーブルレイアウト ファイル(.layout)に保存します。 TABLE - テーブルレイアウトのメタデータを ACLテーブルに保存する場合は、TABLEを指定します。
TO ファイル名 テーブル名	<p>出力ファイルの名前、出力場所:</p> <ul style="list-style-type: none"> ファイル名 - .layout ファイルの名前 ファイル名を引用された文字列として指定します。例: TO "Ap_Trans.layout". .layout ファイル拡張子はデフォルトで使用されるため、指定は省略可能です。 デフォルトでは、出力ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> TO "C:\Ap_Trans.layout" TO "Table Layouts\Ap_Trans.layout" <p>メモ テーブルレイアウト名を 64 文字の英数字(.layout 拡張子を含まない)に制限し、テーブルレイアウトが Analytics にインポートされるときに名前が切り捨てられないようにします。 名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

名前	説明
	<ul style="list-style-type: none"> テーブル名 -Analytics テーブルと .fil ファイルの名前 テーブル名を引用された文字列として指定します。例: <code>TO "Ap_Trans_layout_metadata.fil"</code>。 .fil ファイル拡張子はデフォルトで使用されるため、指定は省略可能です。 デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Ap_Trans_layout_metadata.fil"</code> <code>TO "Layout Metadata\Ap_Trans_layout_metadata.fil"</code> <p>メモ テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>

例

テーブルレイアウトを外部テーブルレイアウトファイル(.layout)に保存する

次の例では、開いているテーブルで使用されているテーブルレイアウトを、**Ap_Trans.layout** という名前の外部テーブルレイアウトファイルに保存しています。

この場合、テーブルレイアウトファイルは Analytics プロジェクト フォルダーに保存されます。

```
SAVE LAYOUT FILE TO Ap_Trans.layout
```

次の例では、テーブルレイアウトファイルは指定したフォルダーに保存されます。

```
SAVE LAYOUT FILE TO "C:\ACL_DATA\AP Audit 2013\Ap_Trans.layout"
```

テーブルレイアウトのメタデータのコピーを新しい Analytics テーブルに保存する

次の例では、開いているテーブルで使用されているテーブルレイアウトのメタデータのコピーを、**Ap_Trans_layout_metadata** という名前の新しい Analytics テーブルに保存しています。

次の例では、新しい Analytics テーブルは Analytics プロジェクト フォルダーに保存されます。

```
SAVE LAYOUT TABLE TO Ap_Trans_layout_metadata
```

次の例では、Analytics テーブルを指定したフォルダーに保存しています。

```
SAVE LAYOUT TABLE TO "C:\ACL_DATA\AP Audit 2013\Ap_Trans_layout_metadata"
```

備考

SAVE LAYOUT FILE と SAVE LAYOUT TABLE の比較

SAVE LAYOUT コマンドは、次の2つの用途に使用されます。

- **FILE** - 開いている Analytics テーブルのテーブルレイアウトを、拡張子 **.layout** の付いた外部テーブルレイアウト ファイルに保存します。
- **TABLE** - 開いている Analytics テーブルのテーブルレイアウトからメタデータを抽出し、それを新しい Analytics テーブルに保存します。

SAVE LAYOUT FILE

機能の仕組み

SAVE LAYOUT FILE は、開いている Analytics テーブルのテーブルレイアウトを外部のテーブルレイアウト ファイルに、拡張子 **.layout** で保存します。

テーブルレイアウトにはメタデータが含まれており、これにより、関連付けられたソース データ ファイルの生データの構造化された解釈が提供されます。テーブルレイアウトにソース データ自体は一切含まれていません。

SAVE LAYOUT FILE(テーブルレイアウトの .layout ファイルとしての保存) の用途

テーブルレイアウトを .layout ファイルとして保存すると、テーブルレイアウトとそのメタデータが移動可能かつ再利用可能になります。

.layout ファイルは、Analytics プロジェクトにインポートして、対応するソース データ ファイルに関連付けることができます。つまり、ソース データ ファイル内のデータ要素は、テーブルレイアウトのメタデータによって規定されるフィールド定義に対応している必要があります。

たとえば、3 月と4 月のソース データ ファイル内のデータの構造は同一であると仮定して、取引ファイルの3 月のテーブルレイアウトを保存し、それを4 月の取引を含んでいるソース データ ファイルに関連付けることができます。.layout ファイルをこのように使用すると、新しいテーブルレイアウトを一から作成する手間を省くことができます。

Analytics テーブルの構造の詳細については、「テーブルレイアウトを使用したデータの構造化」ページ 768を参照してください。

SAVE LAYOUT TABLE

機能の仕組み

SAVE LAYOUT TABLE は、開いている Analytics テーブルのテーブルレイアウトからメタデータを抽出し、それを新しい Analytics テーブルに保存します。

新しいテーブルは、テーブルレイアウトそのものではなくむしろ標準の Analytics テーブルで、元のテーブルのテーブルレイアウト メタデータの要約を含んでいます。Analytics スクリプトでこの要約にアクセスすることで、その情報に基づいてスクリプト内で意思決定することが可能になります。

元のテーブル内の各フィールドについて、テーブルレイアウト メタデータのうち以下の部分が新しいテーブルに抽出されます。

メモ

新しいテーブル内のこれらのフィールド名は、どのローカライズ版の Analytics を使用しているかに関係なく、常に英語で生成されます。

新しいテーブルのフィールド名	テーブルレイアウト メタデータ
フィールド名	フィールドの名前
データ型	フィールドのデータ型
カテゴリ	フィールドのデータ カテゴリ
開始位置	フィールドの開始位置
フィールド長	フィールドの長さ

コマンド

新しいテーブルのフィールド名	テーブルレイアウト メタデータ
小数位	フィールドの小数点以下の桁数(数値フィールドのみ)
書式	フィールドの書式(日付時刻フィールドと数値フィールドのみ)
代替タイトル	フィールドの代替列見出し
列幅	ビュー内での列の幅

追加の詳細

演算フィールド	抽出されたメタデータに演算フィールドは含まれますが、演算フィールドで使用される式や、条件はどれも記録されません。また、演算フィールドには、開始位置、フィールド長、および小数点以下の桁数も記録されません。
関連付けられたフィールド	関連フィールドは、テーブルレイアウトの一部ではないので含まれません。
フィールド レベルのフィルター フィールド ノート	フィールド レベルのフィルターおよびフィールド ノートは含まれません。
代替列見出し 列幅	代替列見出しおよび列の幅に記録される値は、テーブルレイアウトで指定された値です。列に対して指定できる、ビューレベルの値ではありません。

SAVE LOG コマンド

コマンド ログ全体、または現在の Analytics セッションのログ エントリを、外部ファイルに保存します。

構文

```
SAVE LOG <SESSION> AS ファイル名 {<ASCII>|HTML} <OK>
```

パラメーター

名前	説明
SESSION 省略可能	現在の Analytics セッションのログ エントリだけが保存されます。
AS ファイル名	出力ファイルの名前。 ファイル名の値は引用符で囲んだ文字列として指定します。例: AS "コマンド ログ"ファイル拡張子 (.txt、.htm または .html) を指定することはできますが、必須ではありません。 デフォルトでは、出力ファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none">AS "C:\Command Log.TXT"AS "Results\Command Log.TXT"
ASCII HTML	出力結果の書式: <ul style="list-style-type: none">ASCII(またはキーワードなし) - ASCII プレーンテキスト ファイルHTML - HTML ファイル
OK 省略可能	ファイル名と同じ名前のファイルが既に存在している場合でも、確認なしで上書きします。

例

買掛金分析のコマンドのログを保存する

コマンド

3月の買掛金ファイルに対してデータ分析を実行して、対応するコマンドのログを調書の一部として保存したいとします。

次の例では現在のAnalyticsセッションの項目をHTMLファイルに保存します。同じ名前のファイルが存在している場合でも、確認なしで上書きします。

```
SAVE LOG SESSION AS "C:\Payables_March_Log.htm" HTML OK
```

SAVE TABLELIST コマンド

Analytics プロジェクトのすべてのテーブルのリストを Analytics テーブルまたは CSV ファイルに保存します。

構文

```
SAVE TABLELIST {FILE|TABLE} TO {テーブル名 | ファイル名}
```

パラメーター

名前	説明
FILE TABLE	<ul style="list-style-type: none"> FILE - テーブルリストを CSV ファイル (.csv) に保存します。 TABLE - Analytics テーブルにテーブルリストを保存します。
TO テーブル名 ファイル名	<p>テーブルリストを保存する場所：</p> <ul style="list-style-type: none"> テーブル名 - TABLE を使用するときに出力される Analytics テーブルおよびそれに関連付けられた .fil ファイルの名前 <p>.fil ファイル拡張子はデフォルトで使用され、指定する必要はありません。このテーブルは、作業中の Analytics プロジェクトと同じフォルダーに保存され、別のフォルダーには保存できません。</p> <p>メモ</p> <p>Analytics テーブルの名前は、64 文字までの英数字に制限されます。名前にはアンダースコア文字 (_) を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <ul style="list-style-type: none"> ファイル名 - FILE を使用するときの .csv ファイルの名前 <p>.csv ファイル拡張子はデフォルトで使用されるので、指定する必要はありません。絶対パスまたは相対パスを指定して、Analytics プロジェクトが入っているフォルダーとは別の既存のフォルダーに CSV ファイルを保存することができます。相対パスを指定する場合は、Analytics の作業ディレクトリに対して相対です。</p> <p>スペースを含む場合は、引用された文字列として値を指定する必要があります。</p>

例

新しいテーブルを作成する

次の例では、Analytics プロジェクトに **Table_list_complete** という新しいテーブルを作成しています。

```
SAVE TABLELIST TABLE TO Table_list_complete
```

CSV ファイルを作成する

次の例では、**C:\ACL Data** フォルダに **Table_list_complete.csv** という新しい CSV ファイルを作成しています。

```
SAVE TABLELIST FILE TO "C:\ACL Data\Table_list_complete"
```

備考

出力の列

出力される Analytics テーブルまたは CSV ファイルには 3 つの列があります。

- **テーブル名** - Analytics テーブルレイアウト の名前。
- **タイプ** - Analytics テーブルがローカルテーブルとサーバー テーブルのどちらを表すかを示します。
- **データ ファイルパス** - ソース データ ファイルへの完全パス。

SAVE WORKSPACE コマンド

ワークスペースを作成して保存します。

構文

```
SAVE WORKSPACE ワークスペース名 {フィールド名 <...n>}
```

パラメーター

名前	説明
ワークスペース名	現在の Analytics プロジェクト で作成および追加するワークスペースの名前。
フィールド名 <...n>	ワークスペースに追加するフィールドの名前。複数のフィールド名をスペースで区切って指定することができます。

例

ワークスペースのアクティブ化

Metaphor_Inventory_2002 テーブルにある 2 つの演算フィールドを用いて **Inventory_margin** というワークスペースを作成します。次に、そのワークスペースをアクティブにして、演算フィールドが **Inventory** テーブルで利用できるようにします。

```
OPEN Metaphor_Inventory_2002
SAVE WORKSPACE Inventory_margin Gross_unit_margin Percent_unit_margin
OPEN Inventory
ACTIVATE WORKSPACE Inventory_margin OK
```

備考

演算フィールドの作成に使用するフィールド名の一致要件

ワークスペースに保存する演算フィールドを作成する式の中で使用するあらゆるフィールド名は、そのワークスペースを使用するテーブルのフィールド名と一致していなければなりません。

たとえば、ワークスペース内に `Value = Sale_price * Quantity` という演算フィールドがある場合は、アクティブテーブルにも **Sale_price** フィールドと **Quantity** フィールドが存在している必要があります。

SEEK コマンド

インデックス付き文字フィールドで、指定した文字式または文字列と一致する最初の値を検索します。

構文

```
SEEK 検索式
```

パラメーター

名前	説明
検索式	検索する文字式。 任意の有効な文字式、文字変数、あるいは引用符で囲まれた文字列として指定することができます。検索式は、大文字と小文字が区別されます。先頭にスペースを含むことができ、文字のように扱われます。

例

文字変数と一致するフィールドの最初の値を見つける

Card_Number フィールドは文字フィールドとして定義されており、昇順でインデックス付けされています。

次の例は、v_card_num 変数の値と完全に一致するか、先頭が一致するフィールドの最初の値を検索します。

```
INDEX ON Card_Number TO "CardNum" OPEN  
SET INDEX TO "CardNum"  
SEEK v_card_num
```

文字文字列と一致するフィールドの最初の値を見つける

Card_Number フィールドは文字フィールドとして定義されており、昇順でインデックス付けされています。

次の例は、文字リテラル"AB-123" 値と完全に一致するか、先頭が一致するフィールドの最初の値を検索します。

```
INDEX ON Card_Number TO "CardNum" OPEN
SET INDEX TO "CardNum"
SEEK "AB-123"
```

備考

このコマンドの動作の詳細については、「一致する最初のレコードの選択」 ページ 1229を参照してください。

機能の仕組み

SEEK コマンドを使用すると、テーブル内で、インデックス付きの文字フィールドに指定した検索式を含んでいる最初のレコードに直接移動できます。

- **検索式が見つかった場合** -、テーブル内で最初の一一致するレコードが選択されます。
- **検索式が見つからない場合**、-"キーと一致するインデックスがありません" というメッセージが表示され、テーブルは、検索式よりも大きい値を持つ最初のレコードに位置付けられます。

検索式よりも大きい値がインデックス付きフィールドに存在しない場合は、テーブルは最初のレコードに位置付けられます。

インデックスの必要性

SEEK を使用して、文字フィールドを検索するには、最初に昇順でフィールドにインデックスを作成する必要があります。複数の昇順の文字フィールドを基にインデックスが作成されている場合は、そのインデックスに指定されている最初のフィールドだけが検索されます。

SEEK で、文字以外のインデックスフィールドや、降順でインデックス付けされた文字フィールドを検索することはできません。

部分一致がサポートされる場合とは

部分一致がサポートされます。インデックス付きフィールドに含まれる長い値の一部を検索式に指定できるのです。ただし、検索式は、一致を成すフィールドの先頭に現れる必要があります。

SEEK コマンドは、**正確な文字比較を行う**]オプション(SET EXACT ON/OFF)の影響を受けません。

SEQUENCE コマンド

Analytics テーブル内の 1 つ以上のフィールドが順番どおりに整列されているかどうかを確認したり、順序が正しくない項目を識別したりします。

構文

```
SEQUENCE <ON> {<FIELDS> キー フィールド <D> <...n>|<FIELDS> ALL <EXCLUDE フィールド名 <...n>>} <UNFORMATTED> <ERRORLIMIT n> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT range> <TO {<SCREEN|ファイル名|PRINT}>} <APPEND> <HEADER ヘッダー テキスト> <FOOTER フッター テキスト> <PRESORT> <ISOLocale ロケール コード>
```

パラメーター

名前	説明
ON FIELDS キー フィールド D <...n> FIELDS ALL	<p>連続順序をテストする 1 つ以上の文字、数値、日付時刻フィールド。</p> <ul style="list-style-type: none"> FIELDS キー フィールド - 指定したフィールドをテストします 複数のフィールドはスペースで区切る必要があります。また、異なるデータ型を指定できません。 複数のフィールドでテストする場合、フィールドはリストの順でテストされます。 キー フィールド 値を降順でテストする D を含めます。デフォルトのテスト並べ順は昇順です。 FIELDS ALL - テーブルのすべてのフィールドをテストします すべてのフィールドでテストする場合、フィールドはテーブルレイアウトに出現する順序でテストされます。 FIELDS ALL の唯一のオプションは、昇順でキー フィールド 値をテストすることです。 <p>メモ 複数のフィールドでテストするときには、ソース テーブルでネストされた連続順序をテストしています。SEQUENCE の有効な使用には、ソース テーブルの既存のネストされた連続順序と同じ順序でテスト フィールドを指定する必要があります。複数のテスト フィールドはネストされたグループとしてテストされません。このようなフィールドは相互に独立してテストされることはありません。</p>
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用して連続順序のテストをするときにのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p>

名前	説明
	<p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
UNFORMATTED 省略可能	結果をファイルに出力する場合、ページ見出しや改ページは除去されます。
ERRORLIMIT <i>n</i> 省略可能	コマンドが停止するまでに許容されるエラー数。デフォルト値は 10 です。
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
TO SCREEN ファイル名 PRINT 省略可能	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ○ ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定し</p>

名前	説明
	<p>ます。</p> <ul style="list-style-type: none"> • TO "C:\Output.TXT" • TO "Results\Output.TXT" <p>◦ 印刷 – 通常使うプリンターに結果を送信します</p>
<p>APPEND 省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
<p>HEADER ヘッダーテキスト 省略可能</p>	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analyticsのシステム変数であるHEADERの値よりも優先されます。</p>
<p>FOOTER フッターテキスト 省略可能</p>	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analyticsのシステム変数であるFOOTERの値よりも優先されます。</p>
<p>PRESORT 省略可能</p>	<p>コマンドを実行する前にキーフィールドでテーブルを並べ替えます。</p> <p>メモ GROUPコマンドの内部ではPRESORTを使用することができません。</p>
<p>ISOLocale ロケールコード 省略可能</p>	<p>メモ AnalyticsのUnicode版にのみ適用されます。</p> <p>システムロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLocaleを使用しない場合は、デフォルトシステムロケールが使用されます。</p>

Analytics の出力変数

名前	含む
WRITE n	コマンドによって検出された順番検査エラーの合計数。

例

順番誤りの従業員 ID のテスト

次の例は、EmployeeID フィールドで特定されたすべての順番検査エラーをテキスト ファイルに書き込んでいます。

```
SEQUENCE ON EmployeeID ERRORLIMIT 10 TO "SequenceErrors.txt"
```

備考

このコマンドの動作の詳細については、「順番検査」ページ 1249を参照してください。

SEQUENCE を GROUP 内で使用する

SEQUENCE コマンドは、GROUP コマンド内で実行しても、グループの処理に影響を及ぼさなくなっています。ただし、GROUP コマンド以降のデータ順番検査エラーを報告することはありません。

SET コマンド

構成可能な Analytics オプションを設定します。

メモ

SET コマンドは、Analytics セッション中のみの Analytics オプションを設定します。この動作は、SET コマンドの使用場所が Analytics コマンド ラインであろうと Analytics スクリプトであろうと適用されます。

Analytics オプションが Analytics セッション間で存続されるようにするには、**オプション**ダイアログボックスを使用する必要があります。**オプション**ダイアログボックスでは、オプションのサブセットのみが使用できます。詳細については、「Analytics のオプションの設定」ページ 121を参照してください。

SET BEEP	SET LEARN	SET READAHEAD
SET CENTURY	SET LOCKAUTOSAVEFILE	SET RETRY
SET CLEAN	SET LOG	SET RETRYIMPORT
SET DATE	SET LOOP	SET SAFETY
SET DELETE_FILE	SET MARGIN	SET SEPARATORS
SET DESIGNATION	SET MATH	SET SESSION
SET ECHO	SET MONTHS	SET SORTMEMORY
SET EXACT	SET NOTIFYFAILSTOP	SET SUPPRESSTIME
SET FILTER	SET NOTIFYRETRYATTEMPTS	SET SUPPRESSXML
SET FOLDER	SET NOTIFYRETRYINTERVAL	SET TEST
SET FORMAT	SET ORDER	SET TIME
SET FUZZYGROUPSIZE	SET OVERFLOW	SET UTCZONE
SET GRAPH	SET PASSWORDSET PASSWORD	SET VERIFY
SET HISTORY	SET PERIODS	SET WIDTH
SET INDEX	SET PICTURE	

構文

構文	例および備考
SET BEEP 値	<pre>SET BEEP 2</pre> <p>デフォルト設定: 0</p> <p>コマンド処理が完了したときにビープ音を鳴らす回数を指定します。</p> <p>value パラメーターは 0 ~ 255 の範囲で指定する必要があります。0 を指定するとビープ音はオフになります。</p> <p>オプション] ダイアログ ボックス: "インターフェイス オプション" ページ 123</p>
SET CENTURY 値	<pre>SET CENTURY 40</pre> <p>デフォルト設定: 40</p> <p>世紀を解釈する開始年を 2 桁の年で指定します。</p> <p>値 パラメーターは 0 から 99 までである必要があります。</p> <p>世紀解釈の開始年の値を 40 に設定すると、2 桁の年の 40 から 99 までは 1940 年から 1999 年と解釈され、2 桁の年の 00 から 39 までは 2000 年から 2039 年として解釈されます。</p> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
SET CLEAN {ON OFF}	<pre>SET CLEAN ON</pre> <p>このオプションをオンにすると、Analytics は無効な文字データは空白に、無効な数値データはゼロに置き換えます。</p>
SET DATE <TO> {0 1 2 文字列}	<pre>SET DATE "YYYY/MM/DD"</pre> <p>デフォルト設定: MM/DD/YYYY</p> <p>ビュー、レポート、エクスポートされたファイルにおける日付、および日付時刻の日付部分の表示形式を指定します。</p> <ul style="list-style-type: none"> 0 - 日付を MM/DD/YYYYYY 形式に設定します。 1 - 日付を MM/DD/YYYY 形式に設定します。 2 - 日付を MM/DD/YYYYYY 形式に設定します。 string - 日付を指定のカスタム形式に設定します。 <p>オプション] ダイアログ ボックスで日、月、年にそれぞれ 'D'、'M'、'Y' 以外を指定している場合でも、SET DATE コマンドを使ってカスタム日付書式を指定するときには、日、月、年に 'D'、'M'、'Y' を使用する必要があります。例:</p>

構文	例および備考
	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET DATE "DD MMM YYYY"</pre> </div> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
<p>SET DELETE_FILE {ON OFF}</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET DELETE_FILE ON</pre> </div> <p>デフォルト設定: OFF</p> <p>テーブルレイアウトを削除するときに、テーブルレイアウトに関連付けられたデータファイル (*.fil) を自動的に削除するかどうかを制御します。</p> <ul style="list-style-type: none"> ○ ON - 関連するデータフィールドは自動的に削除されます。 ○ OFF - 関連付けられたデータファイルは削除されません。 <p>注意</p> <p>このオプションをオンにするときには注意してください。テーブルと共に元のデータファイルが削除される場合もあります。データファイルはすぐに削除されます。Windows のごみ箱には送られません。</p> <p>アンダースコア() を DELETE_FILE に含める必要があります。</p> <p>コマンドラインで、SET DELETE_FILE をパラメーターなしで指定すると、DELETE_FILE が現在オンであるかオフであるかが表示されます。</p> <p>オプション] ダイアログ ボックス: "テーブルオプション" ページ 125</p>
<p>SET DESIGNATION 値</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET DESIGNATION "Produced by ABC Corporation"</pre> </div> <p>値/パラメーターは、印刷された各ページの一番上に表示されるラベルを指定する、引用符で囲まれた文字列です。</p> <p>オプション] ダイアログ ボックス: "ビュー オプション" ページ 130</p>
<p>SET ECHO {ON NONE}</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET ECHO NONE COMMENT スクリプトのコマンドと結果をログから除外します。 SET ECHO ON</pre> </div> <p>スクリプト内のコマンドと結果を Analytics コマンドログに保存するかどうかを制御します。</p> <ul style="list-style-type: none"> ○ ON - スクリプト内のコマンドと結果をコマンドログに保存します。 ○ NONE - スクリプト内のコマンドと結果をコマンドログに保存しません。

構文	例および備考				
	<p>メモ</p> <p>SET ECHO コマンドは、スクリプト内のコマンドと結果のログ記録に対してのみ適用されます。Analytics ユーザーインターフェイスから実行されたコマンド、またはコマンドラインから発行されたコマンドと、それによる結果は、ECHO の設定に関係なく常にログに記録されます。</p> <p>コマンドラインで、SET ECHO をパラメーターなしで指定すると、スクリプト内のコマンドと結果のログ記録が現在オンであるかオフであるかが表示されます。</p>				
<p>SET EXACT {ON OFF}</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px; text-align: center;"> SET EXACT ON </div> <p>デフォルト設定: OFF</p> <p>文字型フィールド、識別子、リテラル値を Analytics で比較する方法を制御できます。</p> <p>メモ</p> <p>スペースは文字のように扱われます。</p> <ul style="list-style-type: none"> ○ ON - 比較文字列を一致と判定するには、完全に同一である必要があります。異なる長さの2つの文字列を比較する場合、Analytics により、長い方の文字列の長さに一致するよう、短い方の文字列の末尾にスペースが埋め込まれます。 たとえば、"AB" は "AB" と等しいですが、"ABC" と等しいとは見なされません。 ○ OFF - Analytics は、長さが等しくない2つの文字列を比較する場合、短い方の文字列を使用します。比較は左から右へ行われます。 たとえば、"AB" は、"AB" と一致するほか、"ABC" と一致すると見なされます。 <p>SET EXACT のその他の例については、「テーブルオプション」ページ 125の [正確な文字比較を行う] オプションのヘルプを参照してください。</p> <p>先頭および末尾のスペースを削除することで、文字列と文字列内部のスペースのみが比較されるようにするには、ALLTRIM() 関数を使用します。</p> <p>たとえば、値に対して ALLTRIM() を使用した ALLTRIM(" AB") = ALLTRIM("AB") は True になりますが、この関数を使用しない場合の比較の結果は False になります。</p> <p>Analytics のコマンドや関数は、SET EXACT の影響を受けるものもあれば、受けないものもあります。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">影響を受ける</th> <th style="text-align: center;">影響を受けない</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;"> <ul style="list-style-type: none"> ○ LOCATE コマンド ○ MATCH() 関数 ○ BETWEEN() 関数 </td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> ○ JOIN コマンド ○ DEFINE RELATION コマンド ○ FIND() 関数 ○ FINDMULTI() 関数 </td> </tr> </tbody> </table> <p>オプション] ダイアログ ボックス: 「テーブルオプション」 ページ 125</p>	影響を受ける	影響を受けない	<ul style="list-style-type: none"> ○ LOCATE コマンド ○ MATCH() 関数 ○ BETWEEN() 関数 	<ul style="list-style-type: none"> ○ JOIN コマンド ○ DEFINE RELATION コマンド ○ FIND() 関数 ○ FINDMULTI() 関数
影響を受ける	影響を受けない				
<ul style="list-style-type: none"> ○ LOCATE コマンド ○ MATCH() 関数 ○ BETWEEN() 関数 	<ul style="list-style-type: none"> ○ JOIN コマンド ○ DEFINE RELATION コマンド ○ FIND() 関数 ○ FINDMULTI() 関数 				
<p>SET FILTER <TO> {テスト フィルター名}</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px; text-align: center;"> SET FILTER TO ProdNo = "070104347" </div>				

構文	例および備考
	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET FILTER TO ProdNoFilter</pre> </div> <p>開いているテーブルでグローバルフィルター(ビュー フィルター)を作成し、論理テストを指定するか、または保存されている既存のフィルターの名前を指定します。</p> <p>SET FILTER をパラメーターなしで指定し、開いているテーブルからフィルターを削除します。</p>
<p>SET FOLDER フォルダーパス</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET FOLDER /Tables/Results</pre> </div> <p>コマンド出力のために、総覧]タブ内の Analytics プロジェクト フォルダを指定します。デフォルトの出力フォルダは、アクティブなテーブルを含んでいるフォルダです。</p> <p>これは DOS 形式のパスで、「/フォルダ名/サブフォルダ名」の形式を用います。最初のスラッシュ(/)は 総覧]タブのルート階層を表します。ファイルの絶対パスを指定する必要があります。</p> <ul style="list-style-type: none"> ◦ <code>SET FOLDER /Tables/Results</code> は、出力フォルダを Results サブフォルダに設定します。Results サブフォルダが存在しなければ、作成されます。 ◦ <code>SET FOLDER /</code> は、出力フォルダを 総覧]タブのルート階層に設定します。 ◦ <code>SET FOLDER</code> は、出力フォルダをデフォルト設定(アクティブなテーブルを含んでいるフォルダ)にします。 <p>出力フォルダは、リセットするか、プロジェクトを閉じない限り、設定したままの状態であり続けます。プロジェクトを開くときに、出力フォルダはデフォルトのアクティブなテーブルフォルダに戻ります。</p>
<p>SET FORMAT {ON OFF}</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET FORMAT ON</pre> </div> <p>デフォルト設定 : OFF</p> <p>新しいテーブルを開いたときに、現在のテーブルレイアウトと演算フィールド定義を Analytics で自動的に表示するかどうかを制御します。</p> <ul style="list-style-type: none"> ◦ ON - 現在のテーブルレイアウトと演算フィールドの定義が自動的に表示されます。また、その結果はコマンドログに保存されます。 ◦ OFF - 現在のテーブルレイアウトと演算フィールド定義は表示されません。 <p>オプション] ダイアログ ボックス: "テーブルオプション" ページ 125</p>
<p>SET FUZZYGROUPSIZE <TO> 数</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET FUZZYGROUPSIZE TO 10</pre> </div> <p>デフォルト設定 : 20 項目</p> <p>出力結果のあいまい重複グループに表示できる項目の最大数を指定します。数パラメーターは 2 未満にすることや、100 を超えることはできません。指定されたサイズは、Analytics セッションの間は有効となります。</p>

構文	例および備考
SET GRAPH グラフの種類	<div data-bbox="607 270 1344 338" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET GRAPH LINE</pre> </div> <p>以降に生成されるすべてのグラフで使用するグラフの種類を指定します。コマンドの実行は、指定されたグラフの種類と互換性がある必要があります。たとえば、BENFORD コマンドで PIE2D または PIE3D チャートを作成することはできません。互換性がないグラフの種類が指定された場合は、デフォルトのグラフの種類が使用されます(BAR3D)。</p> <p>グラフの種類パラメーターは下記のうちの1つである必要があります。</p> <ul style="list-style-type: none"> ○ PIE2D ○ PIE3D ○ BAR2D ○ BAR3D - これがデフォルトのグラフの種類です。 ○ STACKED2D ○ STACKED3D ○ LAYERED ○ LINE ○ BENFORD - 2D 棒グラフと 2D 折れ線グラフを組み合わせます。
SET HISTORY <TO> 値	<div data-bbox="607 972 1344 1039" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET HISTORY TO 50</pre> </div> <p>保持するテーブル履歴エントリの最大数を指定します。値パラメーターは 1 ~ 100 である必要があります。</p>
SET INDEX <TO> 値	<div data-bbox="607 1178 1344 1245" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET INDEX TO "CustomerCode.INX"</pre> </div> <p>アクティブなテーブルに適用するインデックスを指定します。</p>
SET LEARN <TO> スクリプト	<div data-bbox="607 1354 1344 1421" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET LEARN TO InventoryRec</pre> </div> <p>スクリプト レコーダーでコマンドを記録するために使用されるスクリプト ファイルの名前を指定します。</p>
SET LOCKAUTOSAVEFILE {ON OFF}	<div data-bbox="607 1560 1344 1627" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET LOCKAUTOSAVEFILE ON</pre> </div> <p>デフォルト設定 : OFF</p> <p>データを Analytics ログファイル(*.LOG)に保存するメカニズムを指定します。</p> <ul style="list-style-type: none"> ○ ON - ログデータを、書き込みバッファを使用せず、直接ディスクに保存します。 <p>書き込みバッファを使用せずにログデータが直接ディスクに保存されると、複雑なスクリプトが実行され、ウイルス対策監視や自動バックアップなどのコンピューターの他の</p>

構文	例および備考
	<p>プロセスと競合する可能性があるときにログファイルの破損を防止できます。</p> <p>メモ ONを指定すると、Analyticsの実行が低速になる場合があります。ログファイルの破損が発生した場合にのみ、このオプションを使用してください。</p> <ul style="list-style-type: none"> OFF - ログデータをディスクに保存する前に、書き込みバッファに保存します。書き込みバッファは一時データ保存場所であり、ハードディスクドライブよりもアクセスが高速であるため、Analytics スクリプトの全体的な実行が高速化します。
<p>SET LOG <TO> {ファイル OFF}</p>	<div data-bbox="607 642 1346 709" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET LOG TO "analysis.log"</pre> </div> <div data-bbox="607 751 1346 819" style="border: 1px solid #ccc; padding: 5px;"> <pre>SET LOG OFF</pre> </div> <p>最初のコマンドはログインを指定したログに切り替えます。指定したログが存在していない場合は、作成されます。</p> <p>2番目のコマンドはログを元の Analytics コマンド ログに復元します。</p> <p>メモ Analytics プロジェクト パスとログ名の最大長は 259 文字です。これには、ファイルパス、ログ名、ファイル拡張子 (.log) が含まれます。</p>
<p>SET LOOP <TO> 数</p>	<div data-bbox="607 1136 1346 1203" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET LOOP TO 100</pre> </div> <p>LOOP コマンドで処理されるレコードについて、1つのレコードに対して実行可能なコマンドの最大数を指定します。</p> <p>numの範囲は0 ~ 32767です。0を指定すると、1レコードあたりのコマンド実行数の制限は無効になります。</p> <p>注意 0を指定すると、スクリプトが無限ループに入る危険性があります。ベストプラクティスは、常にSET LOOP 制限を指定することです。</p> <p>レコードあたりのコマンド実行回数の計算方法</p> <p>特定のレコードについて、SET LOOP 最大値との比較に使用される数値は、そのレコードを処理する GROUP-LOOP ブロック内の全コマンドの合計数です。コマンドの内訳は以下のとおりです。</p>

構文	例および備考														
	<p>ヒント</p> <p>レコードのコマンドの総数の計算方法を理解するために、GROUP-LOOP ブロック内のコマンドを一行ずつ手動で数えることができます。GROUP コマンドで開始し、GROUP コマンドを終了させる END コマンドで終了します。</p> <p>LOOP コマンドまできたら、特定のレコードに対してループが反復する回数を考慮する必要があります。詳細は、以下のコマンドの内訳を参照してください。</p> <p>変数の割り当てはコマンドであることを忘れないでください。オプションの ASSIGN コマンドキーワードは、たいていの場合、省略されます。コメントもコマンドです (COMMENT)。</p> <table border="1" data-bbox="557 667 1414 1388"> <thead> <tr> <th>GROUP-LOOP ブロック内のコマンド</th> <th>カウント</th> </tr> </thead> <tbody> <tr> <td>GROUP コマンド、END コマンドで終了</td> <td>2</td> </tr> <tr> <td>ELSE IF 文と ELSE 文</td> <td>レコードが処理される ELSE IF または ELSE ブロックの前の各オカレンスにつき、1 レコードが処理される ELSE IF または ELSE ブロックにつき、1</td> </tr> <tr> <td>GROUP ... END 内の各コマンド、またはレコードが処理される ELSE IF または ELSE ブロックの中の各コマンド LOOP ... END、またはループに含まれるコマンドはいずれも含まれません。</td> <td>ループを含まない、グループ内、あるいは ELSE IF または ELSE ブロック内のコマンドの数に等しくなります。</td> </tr> <tr> <td>(LOOP コマンド + LOOP ... END 内のコマンドの数) × (そのレコードのループ反復回数) LOOP コマンドを終了させる END コマンドを含みません。</td> <td>LOOP コマンドの積 × レコードのループ反復回数に等しくなります</td> </tr> <tr> <td>LOOP コマンドの最終実行で、False と評価されるもの</td> <td>1</td> </tr> <tr> <td>LOOP コマンドを終了させる END コマンド</td> <td>1</td> </tr> </tbody> </table>	GROUP-LOOP ブロック内のコマンド	カウント	GROUP コマンド、END コマンドで終了	2	ELSE IF 文と ELSE 文	レコードが処理される ELSE IF または ELSE ブロックの前の各オカレンスにつき、1 レコードが処理される ELSE IF または ELSE ブロックにつき、1	GROUP ... END 内の各コマンド、またはレコードが処理される ELSE IF または ELSE ブロックの中の各コマンド LOOP ... END、またはループに含まれるコマンドはいずれも含まれません。	ループを含まない、グループ内、あるいは ELSE IF または ELSE ブロック内のコマンドの数に等しくなります。	(LOOP コマンド + LOOP ... END 内のコマンドの数) × (そのレコードのループ反復回数) LOOP コマンドを終了させる END コマンドを含みません。	LOOP コマンドの積 × レコードのループ反復回数に等しくなります	LOOP コマンドの最終実行で、False と評価されるもの	1	LOOP コマンドを終了させる END コマンド	1
GROUP-LOOP ブロック内のコマンド	カウント														
GROUP コマンド、END コマンドで終了	2														
ELSE IF 文と ELSE 文	レコードが処理される ELSE IF または ELSE ブロックの前の各オカレンスにつき、1 レコードが処理される ELSE IF または ELSE ブロックにつき、1														
GROUP ... END 内の各コマンド、またはレコードが処理される ELSE IF または ELSE ブロックの中の各コマンド LOOP ... END、またはループに含まれるコマンドはいずれも含まれません。	ループを含まない、グループ内、あるいは ELSE IF または ELSE ブロック内のコマンドの数に等しくなります。														
(LOOP コマンド + LOOP ... END 内のコマンドの数) × (そのレコードのループ反復回数) LOOP コマンドを終了させる END コマンドを含みません。	LOOP コマンドの積 × レコードのループ反復回数に等しくなります														
LOOP コマンドの最終実行で、False と評価されるもの	1														
LOOP コマンドを終了させる END コマンド	1														
<p>SET MARGIN 側面 <TO> 値</p>	<div style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;"> <p>SET MARGIN TOP TO 100</p> </div> <p>デフォルト設定 : 51 (51/100 インチ)</p> <p>側面パラメーターには LEFT、RIGHT、TOP、または BOTTOM を指定します。四方の余白を変更する場合は、個別の SET MARGIN コマンドで各余白を指定する必要があります。値に 100 を指定すると、1 インチの余白が作られます。</p> <p>オプション] ダイアログ ボックス: "印刷オプション" ページ 146</p>														
<p>SET MATH <TO> {FIRST LAST MIN MAX}</p>	<div style="border: 1px solid gray; padding: 5px;"> <p>SET MATH TO MIN</p> </div>														

構文	例および備考
	<p>デフォルト設定: MAX</p> <p>2つのオペランドが数式で評価されるときに小数点精度が動作する方法を指定します。</p> <ul style="list-style-type: none"> ◦ FIRST - オペランドのペアの最初のオペランドの小数点桁数を使用します ◦ LAST - オペランドのペアの最後のオペランドの小数点桁数を使用します ◦ MIN - オペランドのペアで最も少ない小数点桁数を使用します ◦ MAX - オペランドのペアで最も多い小数点桁数を使用します <p>複数のオペランドの式では、SET MATH 設定はペアで動作します。標準の演算順序 (BOMDAS) で評価されるときに、指定された設定を各オペランドのペアに適用し、必要に応じて端数処理します。</p> <p>SET MATH 設定で、結果の小数点桁数が減る場合、結果は切り捨てられず、端数処理されます。</p> <p>詳細については、"数式の端数処理と小数点精度の制御" ページ 874を参照してください。</p> <p>メモ Analytics テーブルが開いている間には、SET MATH を使用できません。</p>
<p>SET MONTHS <TO> 文字列</p>	<div data-bbox="607 863 1344 932" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET MONTHS "Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec"</p> </div> <p>3文字形式のデフォルトの月名を指定します。文字列パラメーターは、カンマで区切られた月名の省略形のリストです。</p> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
<p>SET NOTIFYFAILSTOP {ON OFF}</p>	<div data-bbox="607 1115 1344 1184" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET NOTIFYFAILSTOP ON</p> </div> <p>デフォルト設定: OFF</p> <p>NOTIFY コマンドが失敗した場合のスクリプトの動作を制御します。</p> <ul style="list-style-type: none"> ◦ ON - スクリプト内の NOTIFY コマンドが失敗した場合、スクリプト処理を停止し、ログにメッセージを書き込みます。スクリプトは、最初の失敗後に停止するか、または NOTIFYRETRYATTEMPTS に指定された試行回数後、どの試行も成功しなかった場合に停止します。 ◦ OFF - スクリプトの NOTIFY コマンドが失敗しても、スクリプト処理を続行します。 <p>オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p>
<p>SET NOTIFYRETRYATTEMPTS <TO> 数</p>	<div data-bbox="607 1535 1344 1604" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET NOTIFYRETRYATTEMPTS TO 10</p> </div> <p>デフォルト設定: 5回</p> <p>NOTIFY コマンドが最初の試行に失敗した後、電子メールの送信を試行する回数を指定します。0 から 255 までの数値を入力します。0 を入力すると、最初の失敗後に追加の試行は行われません。</p> <p>NOTIFY コマンドが電子メールの送信に失敗することについて考えられる理由の1つは、電子メールサーバーを利用できないということです。</p>

構文	例および備考
SET NOTIFYRETRYINTERVAL <TO> 秒数	<p>オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p> <div data-bbox="607 338 1344 405" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET NOTIFYRETRYINTERVAL TO 30</pre> </div> <p>デフォルト設定: 10 秒</p> <p>NOTIFYRETRYATTEMPTS 間の秒数を指定します。1 から 255 までの数値を入力します。</p> <p>オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p>
SET ORDER <TO> 値	<div data-bbox="607 632 1344 699" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET ORDER TO aAbBcC...</pre> </div> <div data-bbox="607 741 1344 808" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET ORDER</pre> </div> <p>デフォルト設定: システムデフォルト</p> <p>文字フィールドの並べ替え順を指定します。values パラメーターは、指定の並べ替え順のすべての文字を列挙します。</p> <p>2 番目のコマンドは、並べ替えの順序をシステムのデフォルト設定に戻します。</p> <p>詳細については、"並べ替え順] オプションとソート順序" ページ 1184を参照してください。</p>
SET OVERFLOW {ON OFF}	<div data-bbox="607 1106 1344 1173" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET OVERFLOW OFF</pre> </div> <p>デフォルト設定: ON</p> <p>数値のオーバーフローが発生した場合のスクリプトの動作を制御します。</p> <ul style="list-style-type: none"> ○ ON - 数値のオーバーフローが発生した場合、スクリプトの処理を停止し、メッセージをログに書き込みます。 ○ OFF - 数値のオーバーフローが発生しても、スクリプト処理を続行します。 <p>オプション] ダイアログ ボックス: "数値オプション" ページ 143</p>
SET PASSWORD 番号 <TO> 文字列	<div data-bbox="607 1488 1344 1556" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET PASSWORD 1 TO "password123"</pre> </div> <p>パスワード定義を作成し、スクリプトの無人実行用にパスワード値を指定します。</p> <p>番号パラメーターはパスワード定義を一意に識別するものであり、1 から 10 までの値である必要があります。パスワード値は、引用符で囲んだ値として指定します。</p>

構文	例および備考
	<p>注意</p> <p>SET PASSWORD コマンドは、スクリプトに実際のパスワードをクリアテキストで表示するため、この状況には適していません。より安全な代替手段として、"PASSWORD コマンド" ページ 2015を使用してスクリプト実行の最初にパスワードの入力を求め、メモリパスワード値を一時的に安全に格納できます。</p>
<p>SET PERIODS <TO> 値 <...n></p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET PERIODS TO "0,30,90,180,10000"</p> </div> <p>デフォルト設定 : 0,30,60,90,120,10000</p> <p>AGE コマンドによって使用される年齢調べ間隔のデフォルトを指定します。</p> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
<p>SET PICTURE 書式</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET PICTURE "(9,999,999.99)"</p> </div> <p>デフォルト設定 : -999999.99</p> <p>数値のデフォルト書式を指定します。</p> <p>オプション] ダイアログ ボックス: "数値オプション" ページ 143</p>
<p>SET READAHEAD <TO> バッファのサイズ</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET READAHEAD TO 100</p> </div> <p>デフォルト設定 : 33 キロバイト</p> <p>読み込むデータブロック(データバッファ)の大きさをキロバイト単位で指定します。5 から 255 までの数値を入力します。この設定の変更は、サポートからアドバイスされていない限り、行わないでください。</p> <p>オプション] ダイアログ ボックス: "テーブルオプション" ページ 125</p>
<p>SET RETRY <TO> 数 SET RETRYIMPORT <TO> num</p>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <p>SET RETRY TO 20</p> </div> <p>デフォルト設定 : 0 回</p> <p>最初の試行が失敗した場合、Analytics がコマンドの実行を試行する回数を指定します。0 から 255 までの数値を入力します。0 を入力すると、最初の失敗後に追加の試行は行われません。</p> <p>再試行の間の待ち時間は 1 秒です。</p> <p>再試行を指定する機能は、データベースまたはクラウド データ サービスに接続する場合に便利ですが、一時的に利用できない可能性があります。データのディスクへの書き込みまたはデータのディスクからの読み取りで問題が発生した場合にもコマンドを再試行できます。</p> <p>SET RETRY は以下のコマンドに適用されます。</p>

構文	例および備考																																								
	<table border="1" data-bbox="557 270 1414 865"> <tr><td>ACCESSDATA</td><td>DIRECTORY</td><td>INDEX</td><td>SAMPLE</td></tr> <tr><td>AGE</td><td>DUPLICATES</td><td>JOIN</td><td>SAVE</td></tr> <tr><td>APPEND</td><td>EVALUATE</td><td>LIST</td><td>SEQUENCE</td></tr> <tr><td>BENFORD</td><td>EXPORT</td><td>MERGE</td><td>SIZE</td></tr> <tr><td>CLASSIFY</td><td>EXTRACT</td><td>OUTLIER</td><td>SORT</td></tr> <tr><td>CLUSTER</td><td>FUZZYDUP</td><td>PREDICT</td><td>STATISTICS</td></tr> <tr><td>CROSTAB</td><td>FUZZYJOIN</td><td>RANDOM</td><td>STRATIFY</td></tr> <tr><td>CVSPREPARE</td><td>HISTOGRAM</td><td>RCOMMAND</td><td>SUMMARIZE</td></tr> <tr><td>CVSSAMPLE</td><td>GAPS</td><td>REFRESH</td><td>TRAIN</td></tr> <tr><td>CVSEVALUATE</td><td>IMPORT</td><td>REPORT</td><td>VERIFY</td></tr> </table> <p data-bbox="646 909 683 930">メモ</p> <p data-bbox="646 947 1344 1031">SET RETRYIMPORT は旧バージョンとの互換性を保つために引き続きサポートされます。SET RETRYIMPORT と SET RETRY では、同じアクションが実行されます。</p> <p data-bbox="565 1073 1159 1100">オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p>	ACCESSDATA	DIRECTORY	INDEX	SAMPLE	AGE	DUPLICATES	JOIN	SAVE	APPEND	EVALUATE	LIST	SEQUENCE	BENFORD	EXPORT	MERGE	SIZE	CLASSIFY	EXTRACT	OUTLIER	SORT	CLUSTER	FUZZYDUP	PREDICT	STATISTICS	CROSTAB	FUZZYJOIN	RANDOM	STRATIFY	CVSPREPARE	HISTOGRAM	RCOMMAND	SUMMARIZE	CVSSAMPLE	GAPS	REFRESH	TRAIN	CVSEVALUATE	IMPORT	REPORT	VERIFY
ACCESSDATA	DIRECTORY	INDEX	SAMPLE																																						
AGE	DUPLICATES	JOIN	SAVE																																						
APPEND	EVALUATE	LIST	SEQUENCE																																						
BENFORD	EXPORT	MERGE	SIZE																																						
CLASSIFY	EXTRACT	OUTLIER	SORT																																						
CLUSTER	FUZZYDUP	PREDICT	STATISTICS																																						
CROSTAB	FUZZYJOIN	RANDOM	STRATIFY																																						
CVSPREPARE	HISTOGRAM	RCOMMAND	SUMMARIZE																																						
CVSSAMPLE	GAPS	REFRESH	TRAIN																																						
CVSEVALUATE	IMPORT	REPORT	VERIFY																																						
<p>SET SAFETY {ON OFF}</p>	<div data-bbox="607 1140 1344 1207" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> SET SAFETY OFF </div> <p data-bbox="557 1251 748 1272">デフォルト 設定 : ON</p> <p data-bbox="557 1293 1409 1346">以下のいずれかの項目を上書きする前に確認ダイアログボックスを表示するかどうかを制御します。</p> <ul data-bbox="557 1367 1333 1566" style="list-style-type: none"> ◦ テーブルレイアウトにおけるフィールド ◦ Analytics テーブル ◦ Analytics データファイル(*.fil) を含むファイル ◦ ON - 確認ダイアログボックスが表示され、上書きを確認する必要があります。 ◦ OFF - 確認ダイアログボックスは表示されず、自動的に上書きが行われます。 <p data-bbox="557 1587 1403 1640">コマンド ラインで、SET SAFETY をパラメーターなしで指定すると、SAFETY が現在オンであるかオフであるかが表示されます。</p> <p data-bbox="565 1661 1230 1688">オプション] ダイアログ ボックス: "インターフェイス オプション" ページ 123</p>																																								
<p>SET SEPARATORS <TO> 値</p>	<div data-bbox="607 1724 1344 1791" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> SET SEPARATORS TO ". , ," </div> <p data-bbox="557 1835 1409 1862">Analytics で使用されるデフォルト の小数点の記号、桁区切り文字、およびリストの区切</p>																																								

構文	例および備考
	<p>リ文字を指定します。SET SEPARATORS の値は、3つの有効な区切り文字を次の順序で指定する必要があります。</p> <ul style="list-style-type: none"> ○ 小数点の記号(ピリオド、カンマ、またはスペース) ○ 桁区切り記号(ピリオド、カンマ、またはスペース) ○ リストの区切り文字(セミコロン、カンマ、またはスペース) <p>3つの区切り文字のうち、小数点の記号は一意である必要があります。このコマンドを使用する場合は、3つの区切り文字をすべて指定してください。リストの区切り文字は主に、関数のパラメーターを区切るために使用されます。</p> <p>オプション] ダイアログ ボックス: "数値オプション" ページ 143</p>
<p>SET SESSION <セッション名></p>	<div data-bbox="607 646 1344 709" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET SESSION</pre> </div> <div data-bbox="607 751 1344 814" style="border: 1px solid #ccc; padding: 5px;"> <pre>SET SESSION "Analysis"</pre> </div> <p>Analytics コマンド ログ内に新しいセッションを出力します。セッションは、現在の自国スタンプによって識別されます。</p> <p>オプションの セッション名 では、30文字までの追加識別情報を追加できます。引用符は許可されていますが、必要ありません。</p>
<p>SET SORTMEMORY 数字</p>	<div data-bbox="607 1031 1344 1094" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET SORTMEMORY 800</pre> </div> <p>デフォルト設定: 0</p> <p>ソート処理とインデックス処理に割り当てるメモリの最大量を指定します。数字パラメーターは0から2000メガバイト(MB)で、20MB単位で入力します。ソートメモリに0を設定した場合、Analyticsにより、現在空いているメモリが使用されます。</p> <p>オプション] ダイアログ ボックス: "テーブルオプション" ページ 125</p>
<p>SET SUPPRESSTIME {ON OFF}</p>	<div data-bbox="607 1354 1344 1417" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <pre>SET SUPPRESSTIME ON</pre> </div> <p>デフォルト設定: OFF</p> <p>Analytics が日付時刻ソースデータの時間部分の表示を抑制するかどうかを制御します。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>この設定は、ODBC データソースを使用する(IMPORT ODBC コマンド)、またはデータベースに直接アクセスする(DEFINE TABLE DB コマンド) Analytics テーブルを定義する場合にのみ使用します。</p> </div> <ul style="list-style-type: none"> ○ ON - テーブルを定義するときに、Analytics により日付時刻値の時間部分の表示が抑制されます。たとえば、20141231 235959 を読み取って、ビューに表示したら、その後は 20141231 として処理されます。

構文	例および備考
	<p>以前の Analytics スクリプト (v.10.0 より前) の日付時刻では、日付時刻データの時刻部分は切り捨てられることが前提となっていたが、このコマンドを含めると、日付時刻型対応バージョンの Analytics でそのスクリプトを実行できるようになります。</p> <p>Analytics は、日付時刻書式の日付部分のみを使用するので、時刻部分を出力しません。時間データは、*.fil ファイルまたはデータベーステーブルにまだ存在します。必要であれば、データの時刻部分を含めるように、フィールドを定義し直したり、新しいフィールドを定義したりすることができます。</p> <ul style="list-style-type: none"> OFF - ODBC または直接データベースアクセスを使用して定義された Analytics テーブルには、完全な日付時刻値が含まれます。 <p>SET SUPPRESSTIME ON/OFF コマンドは、スクリプト内またはコマンドラインから発行できます。</p> <p>コマンドラインで、SET SUPPRESSTIME をパラメーターなしで指定すると、日付時刻データの時刻部分の出力抑制が現在オンであるかオフであるかが表示されます。</p>
<p>SET SUPPRESSXML {ON OFF}</p>	<div data-bbox="607 751 1344 821" style="border: 1px solid gray; padding: 5px; text-align: center;"> <p>SET SUPPRESSXML ON</p> </div> <p>デフォルト設定: OFF</p> <p>画面へのコマンド出力をプレーンテキストと書式設定されたテキストのどちらで表示するかを制御します。</p> <ul style="list-style-type: none"> ON - 画面へのコマンド出力をプレーンテキストとして表示します。 OFF - 画面へのコマンド出力を書式設定されたテキストとして表示します。 <p>コマンドラインでパラメーターを指定せずに SET SUPPRESSXML を指定すると、現在の設定がオン/オフのどちらであるかが表示されます。</p> <p>オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p>
<p>SET TEST {ON OFF}</p>	<div data-bbox="607 1203 1344 1272" style="border: 1px solid gray; padding: 5px; text-align: center;"> <p>SET TEST ON</p> </div> <p>デフォルト設定: ON</p> <p>GROUP コマンドに関連する IF、WHILE、FOR、NEXT テストの結果をログに記録するかどうかを制御します。</p> <ul style="list-style-type: none"> ON - ログに結果を記録します。 OFF - ログに結果を記録しません。 <p>オプション] ダイアログ ボックス: "コマンド オプション" ページ 132</p>
<p>SET TIME <TO> 文字列</p>	<div data-bbox="607 1583 1344 1652" style="border: 1px solid gray; padding: 5px; text-align: center;"> <p>SET TIME "hh:mm:ss PM"</p> </div> <p>デフォルト設定: hh:mm:ss PM</p> <p>ビュー、レポート、およびエクスポートされたファイルでの、日付時刻の時間部分と、独立した時間値の表示形式を指定します。</p> <p>オプション] ダイアログ ボックスで別の時刻書式文字を指定した場合でも、SET TIME コ</p>

構文	例および備考
	<p>マンドを使用して時刻書式を指定するときは、時間には'h'、分には'm'、秒には's'を使用する必要があります。例：</p> <pre data-bbox="607 359 1344 428">SET TIME TO "hh:mm"</pre> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
<p>SET UTCZONE {ON OFF}</p>	<pre data-bbox="607 533 1344 602">SET UTCZONE OFF</pre> <p>デフォルト設定: ON</p> <p>UTC オフセットを含む時刻データの表示を制御します。UTC は協定世界時で、経度 0 度地点の時刻です。</p> <ul data-bbox="558 758 1409 814" style="list-style-type: none"> ○ ON - UTC オフセットのある時刻データは、UTC に相当する時刻として表示されます。 ○ OFF - UTC オフセットのある時刻データは、UTC に変換されずに表示されます。 <p>例：</p> <ul data-bbox="558 877 1149 934" style="list-style-type: none"> ○ 2015 年 1 月 01 日 04:59:59(<code>SET UTCZONE ON</code>) ○ 2014 年 12 月 31 日 23:59:59-05:00(<code>SET UTCZONE OFF</code>) <p>メモ</p> <p>UTC への時間の変換は表示上のことのみを目的としており、ソースデータには影響しません。この 2 つの異なる表示モードは、いつでも変更して切り替えることができます。</p> <p>オプション] ダイアログ ボックス: "日付と時刻オプション" ページ 135</p>
<p>SET VERIFY {ON OFF BLANK}</p>	<pre data-bbox="607 1199 1344 1268">SET VERIFY BLANK</pre> <p>デフォルト設定: OFF</p> <p>テーブルが開かれるとき、またはテーブルのフィールドがコマンドによって処理されるときに、データの自動検証を制御します。</p> <ul data-bbox="558 1430 1409 1696" style="list-style-type: none"> ○ ON - Analytics は、データフィールドの内容がテーブルレイアウト内のフィールドのデータ型に対応しているかどうかを自動的に検証します。テーブルビューでは、影響を受けたフィールドには <code>### ERR ###</code> と表示されます。エラーが発生した場合、データ処理は停止します。 ○ OFF - 自動データ検証を行いません。 ○ BLANK - Analytics は ON パラメーターで説明した検証に加えて、無効な文字データを空白に、無効な数値データをゼロに置き換えます。エラーが発生してもデータ処理は続行され、Analytics プロジェクトフォルダーの <code>ERROR.LOG</code> にエラーが記録されます。 <p>オプション] ダイアログ ボックス: "数値オプション" ページ 143</p>
<p>SET WIDTH <TO> 文字</p>	<pre data-bbox="607 1776 1344 1845">SET WIDTH TO 20</pre>

構文	例および備考
	<p>デフォルト設定: 12</p> <p>数値演算フィールドや一時的に作成した数式の表示幅のデフォルト値(文字)を指定します。Analytics は最大幅を指定することはできません。</p> <p>オプション] ダイアログ ボックス: "数値オプション" ページ 143</p>

SIZE コマンド

金額単位のサンプリングまたはレコードのサンプリングに対し、統計的に有効なサンプルサイズおよびサンプル間隔を計算します。

レコードのサンプリング 金額単位のサンプリング

構文

```
SIZE RECORD CONFIDENCE 信頼レベル POPULATION 母集団の大きさ PRECISION 許容逸脱率 <ERRORLIMIT 予想誤謬率> <TO {SCREEN|ファイル名}>
```

パラメーター

メモ

値を指定する際、3桁の区切り記号やパーセント記号は含めないでください。

名前	説明
RECORD	レコード サンプルのサンプルサイズを計算する ATTRIBUTE は、RECORDと同じ操作を行う、廃止されたパラメーターです。
CONFIDENCE 信頼度	必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。 たとえば、95を指定した場合は、サンプルが実際に95%の確率で母集団を代表しているとお客様が信頼したいということを意味します。信頼度は"サンプリングリスク"の補数です。信頼度が95%ということはサンプリングリスクが5%ということと同じです。
POPULATION 母集団の大きさ	テーブルからサンプリングするレコード数
PRECISION 許容逸脱率	許容逸脱率。許容逸脱率とは、発生し得るが、発生した場合でも統制が有効であると見なすことのできる、規定した統制からの逸脱率の上限のことです。 たとえば、5を指定するとは、逸脱率が5%を超えたときに統制を無効と見なすことができるということです。
ERRORLIMIT 予想誤謬率 省略可能	母集団における推定逸脱率。検出する規定の統制からの逸脱率です。 たとえば、1を指定するとは、この逸脱率が1%以内であることを指定することです。 このパラメーターを省略した場合は、母集団における推定逸脱率として0%が使用されます。

名前	説明
TO SCREEN ファイル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ◦ ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code>

Analytics の出力変数

名前	含む
SAMPINT n	コマンドによって計算された、必須のサンプル間隔。
SAMPSIZE n	コマンドによって計算された、必須のサンプルサイズ。

例

レコード サンプルのサンプルサイズおよび間隔を計算する

レコード サンプルを使用して、請求書を含む勘定の規定された統制からの逸脱率を推定することを決定しました。

サンプルを抽出する前に、各層の統計的に有効なサンプルサイズを計算します。

Analytics によって抽出されるサンプルの 95% の時間が全体として母集団を表す信頼度が必要です。

指定された信頼度を使用し、下記の例では、レコード サンプルを抽出する場合に使用するためのサンプルサイズおよびサンプル間隔値がそれぞれ 95 および 8.12 として計算されます。

```
SIZE RECORD CONFIDENCE 95 POPULATION 40000 PRECISION 5 ERRORLIMIT 2
```

備考

このコマンドの動作の詳細については、「レコード サンプルのサンプルサイズを計算」ページ 1045を参照してください。

構文

```
SIZE MONETARY CONFIDENCE 信頼度 POPULATION 母集団の大きさ MATERIALITY 許容虚偽表示 <ERRORLIMIT 推定虚偽表示額> <TO {SCREEN|ファイル名}>
```

パラメーター

メモ

値を指定する際、3 桁の区切り記号やパーセント記号は含めないでください。

名前	説明
MONETARY	金額単位 サンプルのサンプルサイズを計算する
CONFIDENCE 信頼度	必要な信頼度。この信頼度で、結果のサンプルが母集団全体を表します。 たとえば、95 を指定した場合は、サンプルが実際に 95% の確率で母集団を代表しているとお客様が信頼したいということを意味します。信頼度は"サンプリングリスク"の補数です。信頼度が 95% ということはサンプリングリスクが 5% ということと同じです。
POPULATION 母集団の大きさ	数値 サンプルフィールドの合計絶対値
MATERIALITY 許容虚偽表示	許容虚偽表示。許容虚偽表示とは、サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。重大な虚偽表示額とまでは見なされません。 たとえば、29000 を指定するとは、虚偽表示合計金額が \$29,000 を上回る場合に重大な虚偽表示額と見なすということです。
ERRORLIMIT 推定虚偽表示額	推定虚偽表示額。サンプルフィールドの値として許容する最大の虚偽表示合計金額のことです。

名前	説明
省略可能	たとえば、5800を指定するとは、虚偽表示合計金額として\$5,800までを許容することです。 このパラメーターを省略した場合は、推定予想虚偽表示額として\$0.00が使用されます。
TO SCREEN ファイル名	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント 表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 -は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例：<code>TO "Output.TXT"</code> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code>

Analytics の出力変数

名前	含む
SAMPINT n	コマンドによって計算された、必須のサンプル間隔。
SAMPSIZE n	コマンドによって計算された、必須のサンプルサイズ。

例

金額単位サンプルのサンプルサイズおよび間隔を計算する

金額単位サンプリングを使用して、請求書を含む勘定の金額虚偽表示の合計金額を推定することを決定しました。

サンプルを抽出する前に、各層の統計的に有効なサンプルサイズを計算します。

Analytics によって抽出されるサンプルの 95% の時間が全体として母集団を表す信頼度が必要です。

指定された信頼度を使用し、下記の例では、金額単位サンプルを抽出する場合に使用するためのサンプルサイズおよびサンプル間隔値がそれぞれ 93 および 6,283.33 として計算されます。

```
SIZE MONETARY CONFIDENCE 95 POPULATION 585674.41 MATERIALITY 29000  
ERRORLIMIT 5800 TO SCREEN
```

備考

このコマンドの動作の詳細については、「金額単位サンプルのサンプルサイズを計算」ページ 1071を参照してください。

SORT コマンド

Analytics テーブルのレコードを、指定されたキーに基づいて昇順または降順に並べ替えます。結果は新しい物理的に並べ替えられた Analytics テーブルに出力されます。

構文

```
SORT {<ON> キー フィールド <D> <...n>|<ON> ALL <EXCLUDE フィールド名 <...n>>}
<FIELDS フィールド名 <AS 表示名> <...n>|FIELDS ALL <EXCLUDE フィールド名 <...n>>}
TO テーブル名 <LOCAL> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <APPEND>
<OPEN> <ISOLOCALE ロケール コード>
```

パラメーター

名前	説明
ON キーフィールド D <...n> ON ALL	<p>並べ替えで使用するキーフィールドまたはフィールド、または式。</p> <p>演算フィールドや一時的に作成した式など、データ型に関係なく、あらゆる種類のフィールドを並べ替えることができます。</p> <ul style="list-style-type: none"> ON キーフィールド -では、指定した1つまたは複数のフィールドが使用されます。複数のフィールドで並べ替えを行う場合は、出力テーブルは入れ子で並べ替えられます。入れ子でのフィールド間の順序は、フィールドを指定した順になります。キーフィールドを降順に並べ替える D を含めます。デフォルトのソート順は昇順です。 ON ALL -では、テーブル内のすべてのフィールドが使用されます。テーブル内のすべてのフィールドを基準にして並べ替えを行うと、出力テーブルが入れ子で並べ替えられます。入れ子でのフィールド間の順序は、テーブルレイアウトで指定した順になります。ON ALL では昇順でしか並べ替えることができません。
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用して並べ替えるときにのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p> <pre>ON ALL EXCLUDE field_1 field_2</pre>

名前	説明
<p>FIELDS フィールド名 <...n> FIELDS ALL 省略可能</p>	<p>メモ キーフィールドは、自動的に出力テーブルに追加されるため、FIELDS に指定する必要はありません。</p> <p>出力に含めるフィールド：</p> <ul style="list-style-type: none"> ○ FIELDS フィールド名 - 指定されたフィールドを使用します フィールドは一覧の順序で使用されます。 演算フィールドを出力先テーブル内の適切なデータ型 (ASCII 型または Unicode 型 (Analytics のエディションによる)、ACL 型 (ネイティブの数値データ型)、日付時刻型、あるいは論理型) の物理フィールドに変換します。演算された実際の値を物理フィールドに設定します。 ○ FIELDS ALL - テーブルのすべてのフィールドを使用します。 フィールドは、テーブルレイアウトに現れる順序と同じ並びで使用されます。 演算フィールドを出力先テーブル内の適切なデータ型 (ASCII 型または Unicode 型 (Analytics のエディションによる)、ACL 型 (ネイティブの数値データ型)、日付時刻型、あるいは論理型) の物理フィールドに変換します。演算された実際の値を物理フィールドに設定します。 ○ FIELDS を省略する場合 - レコード全体、つまりすべてのフィールドとレコードの未定義部分すべてが、並べ替え後の出力テーブルに追加されます。 フィールドは、テーブルレイアウトに定義されているとおりの順に使用されます。 演算フィールドは保持されます。 <p>ヒント レコードに含まれるデータの一部のみが必要な場合は、並べ替え後の出力テーブルにすべてのフィールド、つまりレコード全体が含まれないようにしてください。必要なフィールドのみを選択します。多くの場合、これにより、並べ替え処理が高速化します。</p>
<p>AS 表示名 省略可能</p>	<p>FIELDS を使用するときのみ有効です。</p> <p>新しい Analytics テーブルのビューにおけるフィールドの表示名 (代替列見出し)。表示名をフィールド名、またはソーステーブル内の既存の表示名と同じにしたい場合は、AS を使用しないでください。</p> <p>表示名の値は引用符で囲まれた文字列。列見出しを改行したい場合は、語句の間にセミコロン (;) を入れます。</p> <p>メモ AS は、新しいテーブルへの出力時にのみ使用します。既存のテーブルに追加している場合は、既存テーブルの代替列見出しが優先されます。</p>
<p>EXCLUDE フィールド名 省略可能</p>	<p>FIELDS ALL を使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p>

名前	説明
	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 0 auto;"> FIELDS ALL EXCLUDE <i>field_1 field_2</i> </div>
<p>TO テーブル名</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.FIL"</code> • <code>TO "Results\Output.FIL"</code> <p>メモ</p> <p>テーブル名は 64 文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
<p>LOCAL 省略可能</p>	<p>Analytics プロジェクトと同じ場所に出力ファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲 省略可能</p>	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p>

名前	説明
	FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	テーブルを開き、インデックスをテーブルに適用します。
ISOLOCALE ロケールコード 省略可能	<p>メモ Analytics の Unicode 版にのみ適用されます。</p> <p>システム ロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

例

単一のフィールドおで並べ替え、レコード全体を出力する

サンプル Inventory テーブルのレコードを製品番号で並べ替えます。並べ替えられたレコードは、新しい Analytics テーブル Inventory_Product_Number に抽出されます。

レコード全体が出力テーブルに含まれます。

```
SORT ON ProdNo TO "Inventory_Product_Number"
```

デフォルトの昇順を降順に変更するには、キーフィールド名の後にDを追加します。

```
SORT ON ProdNo D TO "Inventory_Product_Number"
```

単一のフィールドで並べ替え、フィールドのサブセットを出力する

サンプル **Inventory** テーブルのレコードを製品番号で並べ替えるとします。キーフィールドと指定された非キーフィールドのみが新しい Analytics テーブル **Inventory_Quantity_on_Hand** に抽出されます。

3 番目の非キーフィールド **QtyOH** に対し、出力テーブルでの表示名として **Qty on Hand**(在庫数量) が付けられます。

```
SORT ON ProdNo FIELDS ProdDesc ProdStat QtyOH AS "Qty on Hand" TO  
"Inventory_Quantity_on_Hand"
```

単一のフィールドで並べ替え、すべてのフィールドを出力する

サンプル **Inventory** テーブルのレコードを製品番号で並べ替えるとします。すべてのフィールドは、新しい Analytics テーブル **Inventory_Product_Number** に抽出されます。

FIELDS ALL がレコード全体の出力と異なる点は、**FIELDS ALL** では、ソーステーブルのすべての演算フィールドが出力テーブルの物理フィールドに変換され、フィールドに実際の演算された値が入力されることです。

```
SORT ON ProdNo FIELDS ALL TO "Inventory_Product_Number"
```

複数のフィールドで並べ替える(ネストされた並べ替え)

サンプル **Inventory** テーブルのレコードをロケーション、製品クラス、製品番号の順で並べ替えるとします。並べ替えられたレコードを、新しい Analytics テーブル **Inventory_Location_Class_Number** に抽出します。以上を行うコマンドの例は次のようになります。

```
SORT ON Location ProdCls ProdNo TO "Inventory_Location_Class_Number"
```

関連するフィールドを使用して並べ替える

サンプル **Ap_Trans** テーブルのレコードを次のフィールドで並べ替えます。

- Vendor_State(業者が拠点とする州。関連:Vendor テーブル)
- Vendor_City(業者が拠点とする市区町村。関連:Vendor テーブル)
- Vendor_No(業者番号。Ap_Trans テーブル)

すべての3つのキーフィールド、および関連するフィールド **Vendor.Vendor_Name** を含む指定された非キーフィールドを、新しい Analytics テーブル **Ap_Trans_State_City** に抽出する例を次に示します。

```
SORT ON Vendor.Vendor_State Vendor.Vendor_City Vendor_No FIELDS
Vendor.Vendor_Name Invoice_No Invoice_Date Invoice_Amount Prodno
Quantity Unit_Cost TO "Ap_Trans_State_City"
```

備考

このコマンドの動作の詳細については、「レコードの並べ替え」ページ 1188を参照してください。

SORT コマンドの並べ替えシーケンス

SORT コマンドは、**並べ替え順序** オプション(**ツール > オプション > テーブル**)で指定された並べ順を使用します。デフォルトの並べ替え順序は次のとおりです。

詳細については、「**並べ替え順序** オプションとソート順序」ページ 1184を参照してください。

Analytics のエディション	並べ替え順序]のデフォルト	関連付けられているソート順序
非 Unicode	システム デフォルト (ASCII)	数字、大文字、小文字、小文字: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;">0, 1, 2... A, B, C... a, b, c...</div> たとえば、「Z」は「a」の前にソートされます。
Unicode	混合言語 (UCA) (Unicode 照合アル)	数字、小文字と大文字の混合:

Analytics のエディション	並べ替え順]のデフォルト	関連付けられているソート順序
	ゴリズム)	<div style="border: 1px solid black; padding: 5px; display: inline-block;"> 0, 1, 2... a, A, b, B, c, C... </div> <p>たとえば、"a" は "Z" の前にソートされます。</p>

大文字と小文字の区別

SORT は大文字と小文字を区別します。ご使用の Analytics のエディション(非 Unicode または Unicode)によって、文字列の大文字と小文字が並べ替えに影響する場合があります。

大文字と小文字が並べ替えに影響しないようにしたい場合は、SORT と一緒に UPPER() 関数を使用できます。

```
SORT ON UPPER(キー フィールド) TO "Sorted_Table"
```

関連するフィールドで並べ替える

関連するフィールドで並べ替え、並べ替えられた出力テーブルに、非キーフィールドとして関連するフィールドを含めることができます。SORT コマンドで関連するフィールドを参照するには、子テーブル名.フィールド名を指定します。

固定長データファイルと可変長データファイルの比較

SORT コマンドは固定長、可変長いずれのデータファイルでも使用することができます。

STATISTICS コマンド

Analytics テーブルで、1 つ以上の数値または日付時刻フィールドについて統計計算します。

構文

```
STATISTICS {<ON> フィールド名 <...n> | <ON> ALL <EXCLUDE フィールド名 <...n>>} <STD>
<MODMEDQ> <NUMBER n> <TO {SCREEN | ファイル名 | PRINT}> <IF テスト> <WHILE テスト>
<FIRST 範囲 | NEXT 範囲> <APPEND>
```

パラメーター

名前	説明
ON フィールド名 <...n> ON ALL	合計する 1 つ以上の数値または日付時刻フィールドを指定するか、あるいは、Analytics テーブル内の数値フィールドと日付時刻フィールドすべてについて統計を生成する場合は ON ALL を指定します。
EXCLUDE フィールド名 省略可能	ON ALL を使用して統計情報を生成するときのみ有効です。 コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。 EXCLUDE は ON ALL の直後に置く必要があります。例： <pre>ON ALL EXCLUDE field_1 field_2</pre>
STD 省略可能	STD を指定すると、指定したフィールドの標準偏差もほかの統計情報と共に計算されます。
MODMEDQ 省略可能	指定したフィールド群の最頻値、中央値、上位四分位数、および 3 番目の四分位数値を他の統計のほかにも計算します。
NUMBER <i>n</i> 省略可能	処理の間に保持しておく高値と低値の数。デフォルト値は 5 です。
TO SCREEN ファイル名 PRINT 省略可能	コマンドの結果を送信する場所： <ul style="list-style-type: none"> SCREEN -- Analytics の表示領域に結果を表示します

名前	説明
	<p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: <code>TO "Output.TXT"</code> デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 <ul style="list-style-type: none"> <code>TO "C:\Output.TXT"</code> <code>TO "Results\Output.TXT"</code> 印刷 - 通常使うプリンターに結果を送信します
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲 省略可能</p>	<p>処理するレコード数:</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>APPEND 省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p>

名前	説明
	<p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analyticsによって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>

Analytics の出力変数

メモ

テーブル内の複数のフィールドを対象として統計を生成した場合、システムで生成される出力変数には、最初に列挙したフィールドの値が含まれています。

名前	含む
ABS n	コマンドによって計算された絶対値。
AVERAGE n	コマンドによって計算された平均値。
COUNT n	<p>コマンドによって計算されたレコード数。</p> <ul style="list-style-type: none"> ◦ 変数名がCOUNT1の場合、それは最近実行されたコマンドのレコード数を格納しています。 ◦ 変数名COUNTnのnが1より大きい場合、変数はGROUPコマンドの内で実行されたコマンドのレコード数を格納しています。 <p>nの値はGROUP内のコマンドの行番号に基づいて割り当てられます。たとえば、コマンドがGROUPコマンドの1行下にある場合、値はCOUNT2が割り当てられます。コマンドがGROUPコマンドの4行下にある場合、値はCOUNT5が割り当てられます。</p>
HIGH n	<p>このコマンドによって確認された5番目に高い値。</p> <p>5番目に高い値はデフォルト設定です。この設定を変更するにはNUMBERパラメーターを使用します。たとえば、NUMBER 3は、3番目に高い値を格納することを指定するものです。</p> <p>メモ</p> <p>Analyticsで最も高い値が確認される際、重複値は除外されません。たとえば、降順の値リストが100、100、99、98の場合、3番目に高い値は98でなく99になります。</p>
LOW n	<p>このコマンドによって確認された5番目に低い値。</p> <p>5番目に低い値はデフォルト設定です。この設定を変更するにはNUMBERパラメーターを使用します。たとえば、NUMBER 3は、3番目に低い値を格納することを指定するものです。</p>

名前	含む
	<p>メモ</p> <p>Analytics で最も低い値が確認される際、重複値は除外されません。たとえば、昇順の値リストが 1、1、2、3 の場合、3 番目に低い値は 3 でなく 2 になります。</p>
MAX n	コマンドによって確認された最大値。
MEDIAN n	コマンドによって確認された中央値。
MIN n	コマンドによって確認された最小値。
MODE n	コマンドによって確認された最頻値。
Q25 n	コマンドによって計算された最初の四分位数值(下位四分位数值)。
Q75 n	コマンドによって計算された 3 番目の四分位数值(上位四分位数值)。
RANGE n	コマンドによって計算された最大値と最小値の差。
STDDEV n	コマンドによって計算された標準偏差値。
TOTAL n	<p>コマンドによって計算された最初の合計値。</p> <p>TOTAL コマンドが GROUP コマンド内になければ、n の値は 1 です。n の値は GROUP コマンド内の TOTAL コマンドの行番号に相当します。</p> <p>詳細については、"GROUP コマンド" ページ 1829を参照してください。</p>

例

条件付き統計を生成する

次の例では、製品クラス ID(ProdCls) が '01' であるレコードの、数量(Qty) フィールドの統計を生成しています。

```
STATISTICS ON Quantity IF ProdCls = "01"
```

STRATIFY コマンド

数値フィールドの値に基づいて、レコードを数値間隔でグループ化します。各間隔のレコード数をカウントし、指定した数値フィールドの小計を間隔ごとに求めます。

構文

```
STRATIFY <ON> 数値フィールド MINIMUM 値 MAXIMUM 値 {<INTERVALS 数値>|FREE 間隔
値 <...n> 最後の間隔} <SUPPRESS> <SUBTOTAL 数値フィールド <...n>|SUBTOTAL ALL
<EXCLUDE 数値フィールド <...n>>> <KEY 内訳フィールド> <TO {SCREEN|テーブル名|ファイル
名|GRAPH|PRINT}> <LOCAL> <IF テスト> <FIRST 範囲|NEXT 範囲> <WHILE テスト>
<APPEND> <OPEN> <HEADER ヘッダー テキスト> <FOOTER フッター テキスト> <STATISTICS>
```

パラメーター

名前	説明
ON 数値フィールド	階層化する数値フィールドまたは式。
MINIMUM 値	数値型のフィールドにのみ適用されます。最初の数値間隔の最小値。 FREE を使用している場合、MINIMUM は省略可能です。それ以外の場合は必要になります。
MAXIMUM 値	数値型のフィールドにのみ適用されます。最後の数値間隔の最大値。 FREE を使用している場合、MAXIMUM は省略可能です。それ以外の場合は必要になります。
INTERVALS 数 省略可能	数値型のフィールドにのみ適用されます。 MINIMUM 値と MAXIMUM 値によって指定された範囲の間に Analytics が生成する、均等な間隔の数。間隔の数を指定しない場合は、デフォルトの数が使用されます。 デフォルトは、 オプション] ダイアログボックスの コマンド] タブの 間隔の数] によって決定されます。
FREE 間隔値 <...n> 最終 間隔 省略可能	数値型のフィールドにのみ適用されます。 各間隔の開始点と最後の間隔の終了点を指定することにより、カスタム サイズの間隔を作成することができます。 MINIMUM 値と MAXIMUM 値を指定した場合は、これらの値がそれぞれ最初の間隔の開始点と最後の間隔の終了点となり、各 間隔値 が範囲内に追加の間隔を生成します。指定する間隔値は、MINIMUM 値より大きく、かつ MAXIMUM 値以下である必要があります。

名前	説明
	<p>間隔値は、数値順でなければならず、重複値を含めることはできません。次に例を示します。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、FREE -1000, 0, 1000, 2000, 3000 のように指定します。</p> </div> <p>FREE と INTERVALS の両方を指定する場合は、INTERVALS が無視されます。</p>
<p>SUPPRESS 省略可能</p>	<p>MAXIMUM 値より大きい値と MINIMUM 値より小さい値をコマンド出力から除外します。</p>
<p>SUBTOTAL 数値フィールド <...n> SUBTOTAL ALL 省略可能</p>	<p>グループごとに小計を計算する 1 つ以上の数値フィールドまたは式。</p> <p>複数のフィールドはスペースで区切る必要があります。テーブル内のすべての数値フィールドについて小計を求める場合は ALL を指定します。</p> <p>小計フィールドを選択しないと、階層化の対象とするフィールドの小計が自動的に計算されます。</p> <p>階層化するフィールドと、それ以外の 1 つ以上のフィールドの小計を出したい場合、または小計された階層化対象フィールドの統計を含めたい場合は、階層化するフィールドを明示的に指定する必要があります。</p>
<p>EXCLUDE 数値フィールド 省略可能</p>	<p>SUBTOTAL ALL を使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、SUBTOTAL ALL を微調整できます。</p> <p>EXCLUDE は SUBTOTAL ALL の直後に置く必要があります。例：</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>SUBTOTAL ALL EXCLUDE field_1 field_2</p> </div>
<p>KEY ブレークフィールド 省略可能</p>	<p>小計計算をグループ化するフィールドまたは式。ブレークフィールドの値が変わるたびに、小計が計算されます。</p> <p>ブレークフィールドは、文字フィールドか式である必要があります。指定できるフィールドは 1 つですが、1 つ以上のフィールドを含んでいる式を使用することができます。</p>
<p>TO SCREEN テーブル名 ファイル名 GRAPH PRINT</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ◦ SCREEN -- Analytics の表示領域に結果を表示します <div style="border-left: 3px solid #008000; padding-left: 10px; margin: 10px 0;"> <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。</p> </div> <ul style="list-style-type: none"> ◦ テーブル名 -は、結果の保存先となる Analytics テーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル (.fil) は、Analytics プロジェクトが入っているフォルダーに保存されます。</p>

名前	説明
	<p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> TO "C:\Output.FIL" TO "Results\Output.FIL" <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <ul style="list-style-type: none"> ファイル名 - は結果の保存先となるファイルです。 ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。 例: TO "Output.TXT" デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。 既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。 TO "C:\Output.TXT" TO "Results\Output.TXT" GRAPH - は結果をグラフに表示し、それを Analytics の表示領域に表示します 印刷 - 通常使うプリンターに結果を送信します
<p>LOCAL 省略可能</p>	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>FIRST 範囲 NEXT 範囲 省略可能</p>	<p>処理するレコード数:</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p>

名前	説明
	<p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
APPEND 省略可能	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
OPEN 省略可能	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
HEADER ヘッダーテキスト 省略可能	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
FOOTER フッターテキスト 省略可能	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>
STATISTICS 省略可能	<p>メモ</p> <p>SUBTOTAL も指定されていない場合は使用できません。</p> <p>すべての SUBTOTAL フィールドの平均値、最小値、および最大値を計算します。</p>

例

請求金額で階層化する

Ar(売掛金)テーブルを Invoice_Amount(請求金額) フィールドで階層化する必要があるとします。請求金額の小計は自動的に求められます。

出力は以下のような \$1000 間隔でグループ化されます。

- \$0 ~ \$999.99
- \$1,000 ~ \$1,999.99
- といった具合です。

各間隔の請求金額合計も求められます。

```
OPEN Ar
STRATIFY ON Invoice_Amount MINIMUM 0 MAXIMUM 10000 INTERVALS 10 TO
"Stratified_invoices.FIL"
```

備考

このコマンドの動作の詳細については、「データの階層化」ページ 1304を参照してください。

機能の仕組み

STRATIFY は、数値フィールドの値に基づいて、レコードを均等な数値間隔またはカスタム サイズの数値間隔にグループ化します。

出力には間隔ごとに、ソーステーブルのうち、その間隔に属するレコードの数が記録された特別なレコードが含まれます。

MINIMUM および MAXIMUM 値を自動的に入力する

STRATIFY コマンドを実行する前に、階層化の対象フィールドで STATISTICS コマンドまたは PROFILE コマンドを実行して、フィールド内の最小値と最大値を MINIMUM パラメーターと MAXIMUM パラメーターに自動的に設定することができます。

自動生成された小計と統計フィールドの名前

STATISTICS を使用して、1 つ以上の SUBTOTAL フィールドで統計演算を実行し、結果を Analytics テーブルに出力する場合は、パラメーターによって自動生成されたフィールドの名前は次のようになります。

自動生成されたフィールドの説明	出力テーブルのフィールド名	出力テーブルの列見出し(表示名)
小計フィールド	集計対象となる、ソーステーブルのフィールド名	Total + 集計対象となる、ソーステーブルの代替列見出し
平均フィールド	a_ 集計対象となる、ソーステーブルのフィールド名	Average + 集計対象となる、ソーステーブルの代替列見出し

自動生成されたフィールドの説明	出力テーブルのフィールド名	出力テーブルの列見出し(表示名)
最小フィールド	m_ 集計対象となる、ソーステーブルのフィールド名	Minimum + 集計対象となる、ソーステーブルの代替列見出し
最大フィールド	x_ 集計対象となる、ソーステーブルのフィールド名	Maximum + 集計対象となる、ソーステーブルの代替列見出し

SUMMARIZE コマンド

1つ以上の文字フィールド、数値フィールド、または日付時刻フィールドの同じ値に基づいて、レコードをグループ分けします。各グループのレコード数をカウントし、指定した数値フィールドの小計をグループごとに求めます。

構文

```
SUMMARIZE {ON キー フィールド <...n>|ON ALL <EXCLUDE フィールド名 <...n>>}
<SUBTOTAL 数値フィールド <...n>|SUBTOTAL ALL <EXCLUDE 数値フィールド <...n>>}
<OTHER フィールド <...n>|OTHER ALL <EXCLUDE フィールド名 <...n>>} <TO {SCREEN|テ-
ブル名|PRINT}> <LOCAL> <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
<PRESORT> <APPEND> <OPEN> <HEADER ヘッダー テキスト> <FOOTER フッター テキスト>
<STATISTICS> <MODMEDQ> <STDEV> <CPERCENT> <ISOLocale ロケール コード>
```

パラメーター

名前	説明
ON キーフィールド <...n> ON ALL	<p>要約する1つ以上の文字、数値、日付時刻フィールド。</p> <ul style="list-style-type: none"> ON キーフィールド - 指定したフィールドを使用します 複数のフィールドはスペースで区切る必要があります。また、異なるデータ型を指定できません。 1つ以上のフィールドで要約する場合、フィールドはリストの順序で要約されます。PRESORTを指定する場合、出力テーブルのネストされた並べ替えが同じ順序に従います。 ON ALL - テーブルのすべてのフィールドを使用します すべてのフィールドで要約する場合、フィールドはテーブルレイアウトに出現する順序で要約されます。PRESORTを指定する場合、出力テーブルのネストされた並べ替えが同じ順序に従います。
EXCLUDE フィールド名 省略可能	<p>ON ALL を使用して要約するときのみに有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、ON ALL を微調整できます。</p> <p>EXCLUDE は ON ALL の直後に置く必要があります。例：</p> <pre>ON ALL EXCLUDE field_1 field_2</pre>

名前	説明
SUBTOTAL 数値フィールド <...n> SUBTOTAL ALL 省略可能	<p>グループごとに小計を計算する1つ以上の数値フィールドまたは式。</p> <p>複数のフィールドはスペースで区切る必要があります。テーブル内のすべての数値フィールドについて小計を求める場合はALLを指定します。</p>
EXCLUDE 数値フィールド 省略可能	<p>SUBTOTAL ALLを使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDEでは、指定したフィールドを除外して、SUBTOTAL ALLを微調整できます。</p> <p>EXCLUDEはSUBTOTAL ALLの直後に置く必要があります。例：</p> <pre>SUBTOTAL ALL EXCLUDE field_1 field_2</pre>
OTHER フィールド <...n> OTHER ALL 省略可能	<p>出力に含める1つ以上の追加フィールド。</p> <ul style="list-style-type: none"> OTHER フィールド <...n> - 指定した1つまたは複数のフィールドが含まれます。 OTHER ALL - キーフィールドまたは小計フィールドとして指定しなかった、テーブル内のすべてのフィールドが含まれます。 <p>OTHERは、各要約グループ内のすべてのレコードが同じ値を含んでいるフィールドに対してのみ使用してください。要約されるグループに対し異なる値を含んでいるフィールドを指定すると、グループ内の最初のレコードの値しか表示されず、これでは意味がありません。</p> <p>例：</p> <ul style="list-style-type: none"> テーブルを顧客番号で要約する - 場合には、適切な「OTHER(その他の)フィールド」の例としては顧客名を指定できます。通常、顧客名は、同じ顧客番号を持つすべてのレコードで同じです。 州で業者テーブルを要約する - 不適切「OTHER(その他の)フィールド」は都市です。各州の一覧の最初の都市のみが出力に表示されます。このような場合、州と市の両方をキーフィールドとして(州、市の順番で)要約を行うことをお勧めします。
EXCLUDE フィールド名 省略可能	<p>OTHER ALLを使用するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDEでは、指定したフィールドを除外して、OTHER ALLを微調整できます。</p> <p>EXCLUDEはOTHER ALLの直後に置く必要があります。例：</p> <pre>OTHER ALL EXCLUDE field_1 field_2</pre>
TO SCREEN テーブル名 PRINT	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> SCREEN -- Analyticsの表示領域に結果を表示します <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の1つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> テーブル名 - は、結果の保存先となるAnalyticsテーブルのことです。 <p>テーブル名には、ファイル拡張子 .FIL を付けた文字列を引用符で囲んで指定する必要があります。例：<code>TO "Output.FIL"</code></p>

名前	説明
	<p>デフォルトでは、テーブルデータファイル(.fil)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> TO "C:\Output.FIL" TO "Results\Output.FIL" <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p> <p>○ 印刷 – 通常使うプリンターに結果を送信します</p>
<p>LOCAL 省略可能</p>	<p>Analytics プロジェクトと同じ場所に出カファイルを保存します。</p> <p>メモ</p> <p>Analytics テーブルである出力ファイルを含むサーバーテーブルに対してコマンドを実行するときのみ適用されます。</p> <p>LOCAL パラメーターは TO パラメーターの直後に置く必要があります。</p>
<p>IF テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
<p>WHILE テスト 省略可能</p>	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲 省略可能</p>	<p>処理するレコード数：</p> <ul style="list-style-type: none"> FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>PRESORT 省略可能</p>	<p>コマンドを実行する前にキーフィールドでテーブルを並べ替えます。</p> <p>メモ</p> <p>GROUP コマンドの内部では PRESORT を使用することができません。</p>

名前	説明
	<p>PRESORT を使用する場合</p> <p>PRESORT を使用すると、出力は並べ替えられ、キーフィールド内の同一値セットごと、または値の等しい組み合わせごとに1つの一意のグループが含まれます。</p> <p>ヒント 入力テーブルがすでに並べ替えられている場合は、PRESORT を指定しないことで処理時間を短縮できます。</p> <p>Presort を使用しない場合</p> <p>PRESORT を使用しない場合、出力結果は、入力テーブルの並べ替え順序を使用します。</p> <p>キーフィールドに入っている同一の値が連続して並んでいない場合、出力結果には、同一値セットごと、または値の等しいの組み合わせごとに2つ以上のグループが含まれることとなります。</p> <p>メモ コンテキストによっては、同一値セットごと、または値の等しい組み合わせに2つ以上のグループが含まれることが、要約の目的よりも優先されることがあります。</p>
<p>APPEND 省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>
<p>OPEN 省略可能</p>	<p>コマンドを実行した後、コマンドによって作成されたテーブルを開きます。コマンドが出力テーブルを作成する場合にのみ有効です。</p>
<p>HEADER ヘッダーテキスト 省略可能</p>	<p>レポートの各ページの最上部に挿入されるテキスト。</p> <p>ヘッダーテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である HEADER の値よりも優先されます。</p>
<p>FOOTER フッターテキスト 省略可能</p>	<p>レポートの各ページの最下部に挿入されるテキスト。</p> <p>フッターテキストは引用符で囲んだ文字列として指定する必要があります。この値は、Analytics のシステム変数である FOOTER の値よりも優先されます。</p>

名前	説明
STATISTICS 省略可能	<p>メモ SUBTOTAL も指定されていない場合は使用できません。</p> <p>すべての SUBTOTAL フィールドの平均値、最小値、および最大値を計算します。</p>
MODMEDQ 省略可能	<p>メモ SUBTOTAL も指定されていない場合は使用できません。</p> <p>最頻値、中央値、最初の四分位数、および3番目の四分位数値がすべての SUBTOTAL フィールドに対して計算されます。</p>
STDEV 省略可能	<p>メモ SUBTOTAL も指定されていない場合は使用できません。</p> <p>標準偏差と合計の割合がすべての SUBTOTAL フィールドに対して計算されます。</p>
CPERCENT 省略可能	レコード数における各グループの割合を計算します。
ISOLOCALE 省略可能	<p>メモ Analytics の Unicode 版にのみ適用されます。</p> <p>システム ロケールは「言語-国」の形式で入力します。たとえば、カナダフランス語はコード「fr-ca」を入力します。</p> <p>次のコードを使用します。</p> <ul style="list-style-type: none"> ◦ 言語 - ISO 639 標準言語コード ◦ 国 - ISO 3166 標準国コード <p>国コードを指定しない場合は、言語のデフォルト国が使用されます。</p> <p>ISOLOCALE を使用しない場合は、デフォルト システム ロケールが使用されます。</p>

例

顧客ごとの総取引額

Customer_Number(顧客番号)フィールドに基づいて Ar(売掛金)テーブルを要約し、Trans_Amount(取引額)フィールドの小計を求めたいとします。出力は、顧客ごとでグループ化され、各顧客の総取引額が含まれます。

```
OPEN Ar
SUMMARIZE ON Customer_Number SUBTOTAL Trans_Amount TO "Customer_
total.FIL" PRESORT
```

顧客ごとの取引日別の総取引額

Customer_Number(顧客番号)と**Trans_Date**(取引日)フィールドに基づいてAr(売掛金)テーブルを要約したいと思います。**Trans_Amount**フィールドの小計を求めます。

出力は、顧客ごとと、顧客内の取引日ごとにグループ化され、各顧客の取引日別に総取引額が含まれます。

```
OPEN Ar
SUMMARIZE ON Customer_Number Trans_Date SUBTOTAL Trans_Amount TO
"Customer_total_by_date.FIL" PRESORT
```

顧客ごと、取引日別の取引額の合計、平均、最小、および最大

直前の例にSTATISTICSを追加します。

顧客ごとに、その顧客が取引をした日付別の取引額の小計に加え、日付別の取引額の平均値、最小値、および最大値も計算することができます。次のように指定します。

```
OPEN Ar
SUMMARIZE ON Customer_Number Trans_Date SUBTOTAL Trans_Amount TO
"Customer_stats_by_date.FIL" PRESORT STATISTICS
```

同じ取引額、同じ日付

Trans_Date(取引日)フィールドと**Trans_Amount**(取引金額)フィールドに基づいて、クレジットカード取引テーブルを集計する場合を考えます。

出力は日付別にグループ化され、その日付内で金額別にグループ化されます。関連付けられたカウントを使用して、同じ金額と同じ日付の取引を特定できます。以上を行うコマンドの例を次に示します。

```
OPEN CC_Trans
SUMMARIZE ON Trans_Date Trans_Amount TO "Transactions_by_date_
amount.FIL" OPEN PRESORT
SET FILTER TO COUNT > 1
```

備考

このコマンドの動作の詳細については、「データの要約」ページ 1326を参照してください。

機能の仕組み

SUMMARIZE はフィールドに同じ値があるレコード、または複数のフィールドで値の組み合わせが同じであるレコードをグループ化します。出力結果には、グループごとに1つのレコードが含まれ、各レコードには、ソーステーブル内でそのグループに属するレコードの数が含まれます。

小計と統計：出力結果の計算とフィールド名

1つ以上の任意のパラメーターを使用して、指定する SUBTOTAL フィールドで統計演算を実行することもできます。統計計算は、出力においてグループ別に内訳が示されます。

省略可能なパラメーター	出力テーブルの列見出し (表示名)	出力テーブルのフィールド名	小計フィールドで実行された計算
SUBTOTAL	Total + 小計対象となる代替列見出し	小計対象となるフィールド名	各グループの小計された値
STATISTICS	Average + 小計対象となる代替列見出し	a_ 小計対象となるフィールド名	各グループの平均値
	Minimum + 小計対象となる代替列見出し	m_ 小計対象となるフィールド名	各グループの最小値
	Maximum + 小計対象となる代替列見出し	x_ 小計対象となるフィールド名	各グループの最大値
MODMEDQ	Median + 小計対象となる代替列見出し	c_ 小計対象となるフィールド名	各グループの中央値 <ul style="list-style-type: none"> ○ 奇数の値セット：中央値 ○ 偶数の値セット：中央にある2つの値の平均

省略可能なパラメーター	出力テーブルの列見出し (表示名)	出力テーブルのフィールド名	小計フィールドで実行された計算
	Mode + 小計対象となる代替列見出し	o_ 小計対象となるフィールド名	各グループの最も頻繁に発生する値 <ul style="list-style-type: none"> 2回以上出現する値がない場合は、"N/A"が表示される 関連付けの場合は、最も低い値が表示される
	Q25 + 小計対象となる代替列見出し	q_ 小計対象となるフィールド名	各グループの最初の四分位数値(下四分位数値) <ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
	Q75 + 小計対象となる代替列見出し	p_ 小計対象となるフィールド名	各グループの3番目の四分位数値(下四分位数) <ul style="list-style-type: none"> 結果は Analytics のアルゴリズムによって計算された補間値である Microsoft Excel の QUARTILE および QUARTILE.INC 関数と同じ結果を生成する
STDEV	STDDEV + 小計対象となる代替列見出し	d_ 小計対象となるフィールド名	各グループの標準偏差
	% Field + 小計対象となる代替列見出し	f_ 小計対象となるフィールド名	フィールドの合計に対する割合として表される各グループの小計
CPERCENT	パーセントカウント	COUNT_PERCENTAGE	各グループに属するソーステーブルレコードの割合 <div style="border-left: 2px solid blue; padding-left: 10px; margin-left: 20px;"> メモ 小計フィールドは必要ありません。 </div>

TOP コマンド

Analytics テーブルの最初のレコードに移動します。

構文

```
TOP
```

パラメーター

このコマンドにはパラメーターがありません。

備考

TOP を使用する場面

直前に実行されたコマンド (FIND など) により、テーブル内で先頭以外のレコードが選択されていた場合に先頭レコードに移動するのに、TOP を使用することができます。

TOTAL コマンド

Analytics テーブルの 1 つ以上のフィールドの合計値を計算します。

構文

```
TOTAL {<FIELDS> 数値フィールド <...n>|<FIELDS> ALL <EXCLUDE 数値フィールド
<...n>>} <IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲>
```

パラメーター

名前	説明
FIELDS 数値フィールド <...n> FIELDS ALL	<p>値の合計を求める数値フィールド。</p> <p>テーブル内の数値フィールドの各々について合計を求める場合は ALL を指定します。</p>
EXCLUDE 数値フィールド 省略可能	<p>FIELDS ALL を使用して合計するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p> <p>メモ WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲	処理するレコード数：

名前	説明
省略可能	<ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRSTとNEXTを省略すると、すべてのレコードがデフォルトで処理されます。</p>

Analytics の出力変数

メモ

テーブル内の複数のフィールドを合計した場合、システムで生成される出力変数には、最初に列挙したフィールドの合計が含まれています。

名前	含む
TOTAL n	<p>コマンドによって計算された最初の合計値。</p> <p>TOTALコマンドがGROUPコマンド内になれば、nの値は1です。nの値はGROUPコマンド内のTOTALコマンドの行番号に相当します。</p> <p>詳細については、「GROUPコマンド」ページ 1829を参照してください。</p>

例

最初の25レコードの合計を求める

次の例は、テーブル内の最初の25レコードのMKTVALフィールドの総量を計算しています。

```
TOTAL FIELDS MKTVAL FIRST 25
```

備考

TOTAL の用途

TOTALは、コントロール合計を作成して、ソースデータの完全性や正確性を検証する場合に使用できます。このコマンドにより、指定されたフィールドあるいは式の算術合計が計算されます。

TRAIN コマンド

自動的な機械学習を使用して、トレーニング データ セット に対する最適の予測モデルを作成します。

メモ

Analytics を 32 ビット コンピューターで実行する場合は、TRAIN コマンドはサポートされません。このコマンドに必要な演算は、プロセッサの負荷が高いため、64 ビット コンピューターの方が適しています。

構文

```
TRAIN {CLASSIFIER|REGRESSOR} <ON> キー フィールド <...n> TARGET ラベル付きフィールド
SCORER {ACCURACY|AUC|F1|LOGLOSS|PRECISION|RECALL|MAE|MSE|R2} SEARCHTIME 分
MAXEVALTIME 分 MODEL モデル名 TO テーブル名 <IF テスト> <WHILE テスト> <FIRST 範囲
|NEXT 範囲> FOLDS 分割数 <SEED シード値> <LINEAR> <NOFP>
```

メモ

TRAIN コマンドで使用されるデータセットのサポートされる最大サイズは 1 GB です。

パラメーター

名前	説明
CLASSIFIER REGRESSOR	予測モデルを学習するときに使用する予測タイプ。 <ul style="list-style-type: none"> ○ CLASSIFIER - 分類アルゴリズムを使用して、モデルを学習します レコードが属するクラスまたはカテゴリを予測する場合は分類を使用します。 ○ REGRESSOR - 回帰アルゴリズムを使用して、モデルを学習します レコードに関連付けられた数値を予測する場合は回帰を使用します。
ON キーフィールド <...n>	1 つ以上学習入力フィールド。 フィールドは、文字、数値、または論理型を使用できます。複数のフィールドはスペースで区切る必要があります。

名前	説明				
	<p>メモ 文字フィールドは「分類的」である必要があります。つまり、カテゴリまたはクラスを特定し、最大数の一意の値を含む必要があります。 最大値は 最大カテゴリ オプション(ツール > オプション > コマンド) で指定されます。</p>				
TARGET ラベル付きフィールド	<p>学習入力フィールドに基づいて予測するようにモデルが学習されているフィールド。 別の予測タイプ(分類または回帰)は別のフィールド データ型で動作します。</p> <table border="1" data-bbox="514 558 1414 678"> <tr> <td data-bbox="514 558 810 617">CLASSIFIER で有効</td> <td data-bbox="810 558 1414 617">文字または論理対象フィールド</td> </tr> <tr> <td data-bbox="514 617 810 678">REGRESSOR で有効</td> <td data-bbox="810 617 1414 678">数値対象フィールド</td> </tr> </table>	CLASSIFIER で有効	文字または論理対象フィールド	REGRESSOR で有効	数値対象フィールド
CLASSIFIER で有効	文字または論理対象フィールド				
REGRESSOR で有効	数値対象フィールド				
SCORER ACCURACY AUC F1 LOGLOSS PRECISION RECALL MAE MSE R2	<p>生成されたモデルのスコアを決定(調整およびランク付け)するときに使用するメトリクス。 生成されたモデルのうち、このメトリクスの最善値を有するモデルが保持され、そうでないモデルは破棄されます。 使用している予測タイプ(分類または回帰)に応じて、メトリクスの別のサブセットが有効となります。</p> <table border="1" data-bbox="514 909 1414 1052"> <tr> <td data-bbox="514 909 810 991">CLASSIFIER で有効</td> <td data-bbox="810 909 1414 991">SCORER ACCURACY AUC F1 LOGLOSS PRECISION RECALL MAE MSE R2</td> </tr> <tr> <td data-bbox="514 991 810 1052">REGRESSOR で有効</td> <td data-bbox="810 991 1414 1052">MAE MSE R2</td> </tr> </table> <p>メモ 分類メトリクス AUC は、ラベル付きフィールドにバイナリデータ(Yes/No、True/False などの2つのクラス)が含まれるときにのみ有効です。</p>	CLASSIFIER で有効	SCORER ACCURACY AUC F1 LOGLOSS PRECISION RECALL MAE MSE R2	REGRESSOR で有効	MAE MSE R2
CLASSIFIER で有効	SCORER ACCURACY AUC F1 LOGLOSS PRECISION RECALL MAE MSE R2				
REGRESSOR で有効	MAE MSE R2				
SEARCHTIME 分	<p>予測モデルの学習および最適化にかかる合計時間(分)。 学習および最適化は、異なるパイプライン構成の検索を伴います(異なるモデル、プリプロセッサ、およびハイパーパラメーターの組み合わせ)。</p> <p>メモ TRAIN コマンドの合計実行時間は SEARCHTIME と最大で MAXEVALTIME の2倍です。</p> <p>ヒント MAXEVALTIME の10倍以上である SEARCHTIME を指定します。 この時間割り当ては、処理時間と多様なモデルタイプの評価を可能にすることの間で、合理的なバランスを取っています。</p>				
MAXEVALTIME 分	<p>最大実行時間は、モデル評価ごとの分数です。</p> <p>ヒント 100 MB の学習データごとに45分を割り当てます。 この時間割り当ては、処理時間と多様なモデルタイプの評価を可能にすることの間で、合理的なバランスを取っています。</p>				

名前	説明
MODEL モデル名	<p>学習プロセスで生成されたモデルファイル。</p> <p>モデルファイルには、学習データセットに最適なモデルが含まれます。モデルを PREDICT コマンドに入力し、新しい未確認のデータセットに関する予測を生成します。</p> <p>引用された文字列としてモデル名を指定します。例：<code>TO "Loan_default_prediction"</code></p> <p>ファイル拡張子 <code>*.model</code> を指定するか、それを Analytics で自動で指定させます。</p> <p>デフォルトでは、モデルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにモデルファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ○ <code>TO "C:\Loan_default_prediction"</code> ○ <code>TO "ML Train output\Loan_default_prediction.model"</code>
TO テーブル名	<p>学習プロセスで生成されたモデル評価テーブルの名前。</p> <p>モデル評価テーブルには、以下の2つの異なるタイプの情報が格納されています。</p> <ul style="list-style-type: none"> ○ スコアラー/メトリクス。これらは、学習プロセスによって生成されたモデルファイルの予測パフォーマンスの定量的な推定、分類メトリクスまたは回帰メトリクスを意味します。 <p>メトリクスが異なれば、提供される推定のタイプも異なります。スコアラーには、SCORER に指定したメトリクスが表示されます。メトリクスには、指定しなかったメトリクスが表示されます。</p> <ul style="list-style-type: none"> ○ 重要性/係数-(降順)：モデルによって生成された予測に対する各機能(予測印子)の寄与度を示す値。 <p>テーブル名、.FIL ファイル拡張子を持つ引用符で囲まれた文字列として指定します。例：<code>TO "Model_evaluation.FIL"</code></p> <p>デフォルトでは、テーブルデータファイル(.FIL)は、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにデータファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> ○ <code>TO "C:\Model_evaluation.FIL"</code> ○ <code>TO "ML Train output\Model_evaluation.FIL"</code> <p>メモ</p> <p>テーブル名は64文字の英数字(.FIL 拡張子を含まない)に制限されています。名前にはアンダースコア文字(_)を使用できますが、他の特殊文字やスペースは使用できません。名前の先頭を数字にすることはできません。</p>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ</p> <p>IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT)が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p>

名前	説明
	<p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
FIRST 範囲 NEXT 範囲 省略可能	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ◦ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ◦ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
FOLDS 分割数	<p>モデルを評価および最適化するとき使用する交差検証分割数。</p> <p>分割は学習データセットの下位分割であり、交差検証プロセスで使用されます。</p> <p>一般的に、モデルの学習時に、5 ~ 10 分割を使用すると、適切な結果が得られます。許可された最小分割数は2です。最大数は10です。</p> <p>ヒント</p> <p>分割数が増えると、モデルの予測性能の推定値が改善されることがありますが、全体的な実行時間も長くなります。</p>
SEED シード値 省略可能	<p>Analytics の乱数ジェネレーターを初期化するために使用するシード値。</p> <p>SEED を省略した場合は、シード値がランダムに選択されます。</p> <p>明示的にシード値を指定し、将来に学習プロセスを同じデータセットで複製したい場合は、それを記録します。</p>
LINEAR 省略可能	<p>学習して、線形モデルのみのスコアを決定します。</p> <p>LINEAR が省略される場合、分類または再帰に関連するすべてのモデルタイプが評価されます。</p> <p>メモ</p> <p>データセットが大きくなると、一般的に、線形モデルのみが含まれている場合に、学習プロセスがより短い時間で完了します。</p> <p>線形モデルのみを含めると、出力の係数が保証されます。</p>
NOFP 省略可能	<p>機能選択とデータ前処理を学習プロセスから除外します。</p> <p>機能選択は、予測モデルを最適化する際に最も有用な学習データセットで、自動化されたフィールドの選択です。自動化された選択は予測性能を改善することがありますが、モデル最適化に関連するデータ量が減ります。</p> <p>データ前処理は、学習データセットでの調整や標準化などの変換を実行し、学習アルゴリズムにより適したものにします。</p> <p>注意</p> <p>理由がある場合にかぎり、機能選択とデータ前処理を除外してください。</p>

例

分類モデルの学習

債務不履行になる融資申請者を予測するために後続のプロセスで使用できる分類モデルに学習させる必要があります。

顧客が債務不履行になったかどうかなど各融資の確認済みの結果が含まれる、融資データ履歴セットに関する学習をモデルに行わせます。

直後の予測プロセスで、TRAIN コマンドで生成されたモデルを使用して、現在の融資申請者データを処理します。

```
OPEN "Loan_applicants_historical"  
TRAIN CLASSIFIER ON Age Job_Category Salary Account_Balance Loan_  
Amount Loan_Period Refinanced Credit_Score TARGET Default SCORER  
LOGLOSS SEARCHTIME 960 MAXEVALTIME 90 MODEL "Loan_default_  
prediction.model" TO "Model_evaluation.FIL" FOLDS 5
```

回帰モデルに学習させる

将来の住宅販売価格を予測するために後続のプロセス内で使用できる回帰モデルに学習させる必要があります。

販売価格を含む最近の住宅販売データのセットを当該のモデルに学習させます。

直後の予測プロセスで、TRAIN コマンドで生成されたモデルを使って住宅価格評価を生成します。

```
OPEN "House_sales"  
TRAIN REGRESSOR ON Lot_Size Bedrooms Bathrooms Stories Driveway  
Recroom Full_Basement Gas_HW Air_conditioning Garage_Places Preferred_  
Area TARGET Price SCORER MSE SEARCHTIME 960 MAXEVALTIME 90 MODEL  
"House_price_prediction.model" TO "Model_evaluation.FIL" FOLDS 5
```

備考

このコマンドの動作の詳細については、「クラスと数値の予測」ページ 1350を参照してください。

VERIFY コマンド

Analytics テーブル中のデータがテーブルレイアウトのフィールド定義と一致していることを確認することにより、1 つ以上の Analytics テーブルのデータの妥当性エラーをチェックします。

構文

```
VERIFY {<FIELDS> フィールド名 <...n>|<FIELDS> ALL <EXCLUDE フィールド名 <...n>>}
<IF テスト> <WHILE テスト> <FIRST 範囲|NEXT 範囲> <ERRORLIMIT n> <TO {SCREEN|ファイル名|PRINT}> <APPEND>
```

パラメーター

名前	説明
FIELDS フィールド名 <...n> FIELDS ALL	<p>検証するフィールドまたは式。ALL を指定すると、テーブルのすべてのフィールドを検証します。</p> <p>メモ 定義では、演算フィールド、アドホック式、およびバイナリフィールドは常に有効です。</p>
EXCLUDE フィールド名 省略可能	<p>FIELDS ALL を使用して検証するときのみ有効です。</p> <p>コマンドから除外するフィールド。EXCLUDE では、指定したフィールドを除外して、FIELDS ALL を微調整できます。</p> <p>EXCLUDE はただちに FIELDS ALL に従う必要があります。例：</p> <pre>FIELDS ALL EXCLUDE field_1 field_2</pre>
IF テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。コマンドは、その条件を満たすレコードに対してのみ実行されます。</p> <p>メモ IF パラメーターは、任意の範囲パラメーター(WHILE、FIRST、NEXT) が適用された後に、テーブルに残るレコードに対してのみ評価されます。</p>
WHILE テスト 省略可能	<p>各レコードを処理するために真である必要がある条件式。条件が false と評価するか、テーブルの最後に達したら、コマンドは実行を中止します。</p>

名前	説明
	<p>メモ</p> <p>WHILE を FIRST または NEXT とともに使用する場合は、1 つの制限に達するとすぐに、レコードの処理が停止します。</p>
<p>FIRST 範囲 NEXT 範囲</p> <p>省略可能</p>	<p>処理するレコード数：</p> <ul style="list-style-type: none"> ○ FIRST - 指定されたレコード数に達するまで、最初のレコードから処理を開始します ○ NEXT - 指定されたレコード数に達するまで、現在選択されているレコードから処理を開始します <p>範囲は処理するレコード数を指定します。</p> <p>FIRST と NEXT を省略すると、すべてのレコードがデフォルトで処理されます。</p>
<p>ERRORLIMIT <i>n</i></p> <p>省略可能</p>	<p>コマンドが停止するまでに許容されるエラー数。デフォルト値は 10 です。</p>
<p>TO SCREEN ファイル名 PRINT</p> <p>省略可能</p>	<p>コマンドの結果を送信する場所：</p> <ul style="list-style-type: none"> ○ SCREEN -- Analytics の表示領域に結果を表示します <p>ヒント</p> <p>表示領域内で、リンク付きの任意の結果値をクリックすると、関連付けられているソーステーブル内の 1 つまたは複数のレコードにドリルダウンすることができます。</p> <ul style="list-style-type: none"> ○ ファイル名 - は結果の保存先となるファイルです。 <p>ファイル名には、適切なファイル拡張子を付けた文字列を引用符で囲んで指定します。</p> <p>例：<code>TO "Output.TXT"</code></p> <p>デフォルトでは、テーブルファイルは、Analytics プロジェクトが入っているフォルダーに保存されます。</p> <p>既存の異なるフォルダーにファイルを保存するには、絶対または相対ファイルパスを指定します。</p> <ul style="list-style-type: none"> • <code>TO "C:\Output.TXT"</code> • <code>TO "Results\Output.TXT"</code> <ul style="list-style-type: none"> ○ 印刷 - 通常使うプリンターに結果を送信します
<p>APPEND</p> <p>省略可能</p>	<p>コマンドの出力を既存ファイルに上書きしないで、そのファイルの末尾に追加します。</p> <p>メモ</p> <p>コマンドの出力と既存のファイルの構造が同一であるようにする必要があります。</p> <ul style="list-style-type: none"> • 同じフィールド • 同じフィールド順序 • 一致するフィールドが同じ長さ • 一致するフィールドが同じデータ型 <p>出力は、既存ファイルとの間でファイル構造が違っている場合でも、Analytics によって既存ファイルに追加されます。出力と既存のファイルの構造が一致しない場合は、データが混在、不足、不正確になります。</p>

Analytics の出力変数

名前	含む
WRITE n	コマンドによって検証されたすべてのフィールドのデータ検証エラーの合計数。

例

データの検証と誤謬限度の指定

テーブル内のすべての列を検証し、誤謬上限を 10 に設定するとします。次のコマンドは、10 個のデータ検証エラーが検出されると処理を停止します。

```
VERIFY ALL ERRORLIMIT 10 TO "ImportErrors.txt"
```

備考

機能の仕組み

VERIFY は、1 つ以上のフィールドの値をテーブルレイアウトの各フィールドに指定されているデータ型と比較し、エラーを報告します。このコマンドは、次のことを確認します。

- **文字フィールド** -が有効な文字だけを含んでおり、印刷できない文字を含んでいない。
- **数値フィールド** -が有効な数字データだけを含んでいる。数値フィールドは、数字のほかにも、数字の前に 1 つのプラス記号またはマイナス記号を含んでいたたり、1 つの小数点を含んでいたたりしても構いません。
- **日付時刻フィールド** -が、有効な日付、日付時刻、または時刻を含んでいる。

識別される各エラーについては、レコード番号とフィールド名が、16 進形式の無効な値と共に出力されません。

関数の概要

ACLScript 関数は、範囲が狭い Analytics のカスタマイズされたルーチンです。特定のタスクまたは演算を実行し、値を返します。

たとえば、ALLTRIM() 関数はすべての先頭または末尾のスペースをフィールドのテキスト値から削除します。

Analytics で使用可能な関数を網羅したリストを以下にアルファベット順またはカテゴリ別に整理して示します。

- "検索、置換" ページ 2142
- "比較" ページ 2143
- "変換" ページ 2144
- "テキスト" ページ 2145
- "数学" ページ 2147
- "日付/時刻" ページ 2148
- "財務" ページ 2150
- "フィールドとレコード" ページ 2151
- "テーブル、ファイル、およびプロジェクト" ページ 2152
- "変数テスト" ページ 2152
- "Python" ページ 2153
- "R" ページ 2153
- "ビット エンコードと文字エンコード" ページ 2154

表記規則と使用方法

関数構文規則

表記	説明
かっこ ()	<ul style="list-style-type: none"> ○ 関数への入力値は、かっこで囲む必要があります。 <div style="border: 1px solid gray; padding: 5px; margin: 10px 0; text-align: center;">ALLTRIM(業者名)</div> <ul style="list-style-type: none"> ○ 開かっこは関数名の直後に置き、間にスペースを入れません。 <div style="border: 1px solid gray; padding: 5px; margin: 10px 0; text-align: center;">ALLTRIM(業者名)</div> <p>間違った例:</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0; text-align: center;">ALLTRIM (業者名)</div>

表記	説明
	<ul style="list-style-type: none"> 入力値を指定しない場合でも、かっこを使用する必要があります。 <div data-bbox="599 327 1344 396" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>RECNO()</pre> </div>
区切り文字	<ul style="list-style-type: none"> 関数への複数の入力値は、区切り文字で区切る必要があります。 <div data-bbox="599 491 1344 560" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>SUBSTRING(<製品 ID>,5,12)</pre> </div> <ul style="list-style-type: none"> 有効な区切り文字は、空白スペース、カンマ、セミicolonです。 オプション]ダイアログボックスの数値]タブのリストの区切り文字]では、カンマまたはセミcolonの区切り文字を指定する必要があります。 <div style="margin-top: 10px;"> <p>ヒント</p> <p>読みやすくするために、空白スペースと、カンマまたはセミcolonのどちらかを両方とも使用することもできます。</p> </div> <div data-bbox="654 869 1274 938" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>SUBSTRING(<製品 ID>, 5, 12)</pre> </div>
修飾子あり	<ul style="list-style-type: none"> リテラル文字値の前後には、一重引用符または二重引用符が必要です。 <div data-bbox="599 1031 1344 1100" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>EXCLUDE(<製品 ID>, "#-")</pre> </div> <ul style="list-style-type: none"> リテラル日付時刻値の前後にはバッククォートが必要です。 <div data-bbox="599 1199 1344 1268" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>AGE(期日, `20141231`)</pre> </div> <ul style="list-style-type: none"> 数値には修飾子は使用しません。 <div data-bbox="599 1367 1344 1436" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>ABS(-7.2)</pre> </div> <ul style="list-style-type: none"> 論理値(T/F)には修飾子は使用しません。 <div data-bbox="599 1535 1344 1604" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>LEVDIST(業者名, 業者名_2, F)</pre> </div> <ul style="list-style-type: none"> フィールド名には修飾子は使用しません。 <div data-bbox="599 1703 1344 1772" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>ALLTRIM(業者名)</pre> </div>
リテラル日付時刻書式	<ul style="list-style-type: none"> リテラル日付値はYYYYMMDDまたはYYMMDD書式で入力する必要があります。 <ul style="list-style-type: none"> <div data-bbox="578 1829 695 1864" style="border: 1px solid #ccc; padding: 2px; display: inline-block;"> <pre>`20141231`</pre> </div>

表記	説明
	<ul style="list-style-type: none"> • `141231` ○ リテラル時刻値は、前に`T`または`t`を付けたうえで`hhmmss`または`hhmm`書式で入力する必要があります。 • `t235959` • `20141231 235959`

関数名の省略

注意

ACLは、演算フィールド、式、スクリプトの関数名を省略せず、各名前の完全なバージョンを使用することをお勧めします。

省略により、演算フィールド、式、スクリプトが読み取りにくく、理解が困難になります。完全な関数名なしでは、オンラインヘルプでの関数の検索が難しくなります。

AnalyticsプロジェクトまたはAnalyticsスクリプトが、省略に精通していない人によって修正または継承される場合、特に省略(短縮)が問題になります。

演算フィールド、式、スクリプトの関数を指定するときには、名前を省略できます。関数名から十分な先頭文字を含め、すべてのAnalytics関数で関数を一意に特定する必要があります。

例：

- `MAX` は `MAXIMUM` 関数を一意に特定し、有効な略語にします。
- `MA` は `MAXIMUM` 関数を一意に特定しないため、エラーメッセージを生成します。



関数が一意に特定される場合は、必要に応じて略語を短くすることができます。

たとえば、次のすべての略語は `ALLTRIM` 関数で有効です。

- `ALLTR`
- `ALLT`
- `ALL`
- `AL`

メモ

略語が短くなると、他のユーザーが認識することが難しくなります。

関数ドキュメントで使用される規則

表記	使用目的
大文字	ACLScript 関数の名前です。 メモ Analytics のどのドキュメントでも、関数名は大文字で記載されていますが、これはあくまでも書式設定の表記規則上、そう記載されているにすぎません。Analytics 自体では、関数を大文字で入力する必要はありません。
斜体	ユーザーが指定する関数パラメーター。
 (縦棒)	山かっこまたは波かっこで囲まれた構文項目の選択肢を区切ります。項目のうち 1 つだけを使用できます。
< > (山かっこ)	オプションの構文項目。山かっこを入力しないでください。
{ } (波かっこ)	必須の構文項目。波かっこを入力しないでください。
<, ... n >	前の項目が n 回繰り返されることを示します。出現はカンマによって区切られます。

関数ドキュメントで使用されるデータ型用語

次の用語は、関数パラメーター引数のデータ型を特定し、値を返すために使用されます。

用語	次のものを使用できます。
文字	文字リテラルや、Analytics の文字 (C) カテゴリに属するフィールド名、式、または変数。
数値	数値や、Analytics 数値 (N) データ カテゴリに属するフィールド名、式、または変数。
日付時刻	日付時刻リテラルや、Analytics の日付時刻 (D) カテゴリに属するフィールド名、式、または変数。
論理	論理値や、Analytics の論理 (L) カテゴリに属するフィールド名、式、または変数。
フィールド	任意の Analytics データ カテゴリに属するフィールドの名前。

検索、置換

検索関数では、さまざまなタイプの検索を実行できます。データの文字、範囲内の値、空の値、およびパターンに一致する値の特定の単語または順序を検索できます。

関数

置換関数は、データを検索および置換するためのさまざまなオプションを提供します。

ヒント

関数を使用して、テーブルのデータに対して、強力で効果的な検索とフィルタリングを実行するさまざまな例については、"[Analytics関数を使って検索とフィルタリングを行う](#)" ページ 1235を参照してください。

関数の説明

関数	説明
AT()	文字値における部分文字列の特定の出現の開始位置を示す数値を返します。
BETWEEN()	指定された値が範囲内にあるかどうかを示す論理値を返します。
CLEAN()	文字列内で、最初の無効な文字とそれ以降のすべての文字を空白に置き換えます。
FIND()	指定された文字列が、特定のフィールド中またはレコード全体のどこかに存在するかを示す論理値を返します。
FINDMULTI()	指定された1つまたは複数の文字列が、特定のフィールド中、またはレコード全体のどこかに存在するかを示す論理値を返します。
ISBLANK()	入力値が空白かどうかを示す論理値を返します。
MAP()	文字列が、ワイルドカード文字やリテラル文字を含んでいる指定された書式文字列と一致するかどうかを示す論理値を返します。
MATCH()	指定された値が比較対象の値のうちのどれかと一致するかどうかを示す論理値を返します。
OCCURS()	部分文字列が指定された文字値内に現れる回数を数えて返します。
REGEXFIND()	正規表現で指定されたパターンが文字列内に現れるかどうかを示す論理値を返します。
REGEXREPLACE()	正規表現と一致する文字列のすべてのインスタンスを新しい文字列で置き換えます。
REPLACE()	指定された文字列のすべてのインスタンスを新しい文字列で置き換えます。
TEST()	指定された文字列がレコード内の特定のバイト位置に現れるかどうかを示す論理値を返します。

比較

比較関数は、指定された値に対して、同一またはほぼ同一のテキスト値を検出するさまざまな方法を提供します。

ヒント

同一のテキスト値のみを検出する場合は、このサンプル方法も使用できます。

```
field_name = "テキスト値"
```

```
例: last_name = "Smith"
```

テキスト値は大文字と小文字を区別します。

関数の説明

関数	説明
DICECOEFFICIENT()	2つの指定された文字列のダイス係数を返します。これは、2つの文字列間の類似度を測定したものです。
ISFUZZYDUP()	文字列が、比較文字列のあいまい重複であるかどうかを示す論理値を返します。
LEVDIST()	指定された2つの文字列間のレーベンシュタイン距離を返します。これは、2つの文字列がどの程度異なっているかを測る数値です。
SOUNDEX()	ほかの文字列との発音の比較に使用できる、指定された文字列の soundex コードを返します。
SOUNDSLIKE()	文字列が比較文字列と発音学的に一致しているかどうかを示す論理値を返します。

変換

変換機能では、データ型を変換できます。これらの機能の1つの重要な使用方法は、特定のデータ型の入力を求める Analytics コマンドのフィールドを準備することです。

関数の説明

関数	説明
BINTOSTR()	ZONED または EBCDIC 文字データから変換された Unicode 文字データを返します。"Binary to String" の省略形です。
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
DTOU()	Analytics 日付値を指定された言語およびロケール書式の Unicode 文字列に変換します。"Date to Unicode" の省略形です。
EBCDIC()	EBCDIC 文字エンコードに変換された文字列を返します。
HASH()	入力値に基づいてソルト付き暗号化ハッシュ値を返します。
LEADINGZEROS()	文字列または数値の先頭にゼロを追加します。
PACKED()	パックデータ型に変換された数値データを返します。
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。
STRING()	数値を文字列に変換します。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。
UNSIGNED()	符号なしデータ型に変換された数値データを返します。
UTOD()	書式設定された日付が含まれる Unicode 文字列を Analytics 日付値に変換します。"Unicode to Date" の省略形です。
VALUE()	文字列を数値に変換します。
ZONED()	数値データを文字データに変換し、出力の先頭にゼロを追加します。

テキスト

テキスト関数では、文字データでさまざまなタスクを実行できます。

たとえば、先頭または末尾のスペースを削除したり、特定の文字を除外または含めたり、文字列の一部のみを分離したり、大文字または小文字を標準したりできます。

関数の説明

関数	説明
ALLTRIM()	入力文字列から先頭と末尾のスペースを除去した文字列を返します。
BINTOSTR()	ZONED または EBCDIC 文字データから変換された Unicode 文字データを返します。"Binary to String" の省略形です。
BLANKS()	指定された数の空白スペースを含んでいる文字列を返します。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
DTOU()	Analytics 日付値を指定された言語およびロケール書式の Unicode 文字列に変換します。"Date to Unicode" の省略形です。
EBCDIC()	EBCDIC 文字エンコードに変換された文字列を返します。
EXCLUDE()	指定した文字を除外する文字列を返します。
INCLUDE()	指定した文字のみを含む文字列を返します。
INSERT()	元の文字列の指定のバイト位置に、指定した文字列が挿入された内容が返されます。
LAST()	文字列の末尾から指定された数の文字を返します。
LEADINGZEROS()	文字列または数値の先頭にゼロを追加します。
LENGTH()	文字列に含まれている文字数を返します。
LOWER()	アルファベット文字を小文字に変換した文字列を返します。
LTRIM()	入力文字列から先頭のスペースを除去した文字列を返します。
OMIT()	指定した 1 つ以上の部分文字列が削除された文字列を返します。
PROPER()	各単語の最初の文字を大文字に、残りの文字を小文字に設定した文字列を返します。
REMOVE()	指定した文字のみを含む文字列を返します。
REPEAT()	指定された回数だけ部分文字列を繰り返す文字列を返します。
REVERSE()	文字の順番を逆にした文字列を返します。
RJUSTIFY()	文字列の末尾の空白をすべて先頭に移動させ、指定された文字列と同じ長さの右寄せした文字列を返します。
SORTWORDS()	連続する順序で並べ替えられた個別の単語を含む文字列を返します。

関数

関数	説明
SPLIT()	文字列のうちの指定された部分を返します。
STRING()	数値を文字列に変換します。
SUBSTR()	文字列のうちの指定された部分文字列を返します。
TRANSFORM()	指定された文字列内の双方向テキストの表示方向を逆にします。
TRIM()	入力文字列から末尾のスペースを除去した文字列を返します。
UPPER()	アルファベット文字を大文字に変換した文字列を返します。
ZONED()	数値データを文字データに変換し、出力の先頭にゼロを追加します。

数学

数学関数はさまざまな数学的計算を実行します。

関数の説明

関数	説明
ABS()	数式の絶対値を返します。数の絶対値は符号のない数です。
COS()	ラジアン単位で表された角度のコサインを、小数点以下 15 桁の精度で返します。
DEC()	指定された小数点以下の桁数で値または数式の結果を返します。
EXP()	数式の指数値(底 10)を指定された小数点以下の桁数で返します。
FREQUENCY()	連続する先頭の正の数値桁について、ベンフォードの法則で期待される頻度を小数点以下 8 桁で返します。
INT()	数式またはフィールド値の整数値を返します。
LEADING()	指定された数の先頭桁を含んでいる文字列を返します。
LOG()	数式またはフィールド値の対数(底 10)を、指定された小数点以下の桁数で返します。
MAXIMUM()	一連の数値中の最大値、または一連の日付時刻値中の最も新しい値を返します。
MINIMUM()	一連の数値中の最小値、または一連の日付時刻値中の最も古い値を返します。
MOD()	2つの数値を除算して余りを返します。

関数	説明
NORMDIST()	正規分布データセットのランダム変数が指定した値以下になるか、指定した値とまったく同じになる確率を返します。
NORMSINV()	標準正規分布で指定された確率に関連付けられたzスコアを返します。zスコアは、標準正規分布の平均を基準にした、値の標準偏差数です。
PI()	円周率 π の値を、小数点以下 15 桁で返します。
RAND()	指定した境界内にあるランダム数を返します。
ROOT()	数式の平方根を返します。
ROUND()	数値を丸めた整数を返します。
SIN()	ラジアン単位で表された角度のサインを、小数点以下 15 桁の精度で返します。
TAN()	ラジアン単位で表された角度のタンジェントを、小数点以下 15 桁の精度で返します。
VALUE()	文字列を数値に変換します。
ZONED()	数値データを文字データに変換し、出力の先頭にゼロを追加します。
ZSTAT()	標準 Z 統計量を返します。

日付/時刻

日付および時刻関数では、日付、日付時刻、または時刻データのさまざまなタスクを実行できます。たとえば、2つの日付間の日数を計算し、月のみなどの日付の一部を抽出したり、各日付に対応する曜日を検索できます。

日時関数の詳細

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

関数の説明

関数	説明
AGE()	指定した日付と指定した締切日間の経過日数(年齢)、現在のオペレーティングシステム日付、または任意の2つの日付間の経過日数を返します。
CDOW()	指定された日付または日付時刻の曜日を返します。"Character Day of Week"の省略形です。
CMOY()	指定された日付または日付時刻の月の名前を返します。"Character Month of Year"の省略形です。
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date"の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime"の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time"の省略形です。
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
DAY()	指定された日付または日付時刻から日にちを抽出し、それを数値(1 ~ 31)として返します。
DOW()	指定された日付または日付時刻の曜日を表す数値(1 ~ 7)を返します。"Day of Week"の省略形です。
EOMONTH()	指定された日付から起算して、指定された月数だけ前または後の月の末日の日付を返します。
GOMONTH()	指定された日付から、指定された月数前または月数後の日付を返します。
HOUR()	指定された時刻または日付時刻から時間を抽出し、それを24時間制の数値として返します。
MAXIMUM()	一連の数値中の最大値、または一連の日付時刻値中の最も新しい値を返します。
MINIMUM()	一連の数値中の最小値、または一連の日付時刻値中の最も古い値を返します。
MINUTE()	指定された時刻または日付時刻から分数を抽出し、それを数値として返します。
MONTH()	指定された日付または日付時刻から月を抽出し、それを数値(1 ~ 12)として返します。
NOW()	現在のオペレーティングシステム時刻を日付時刻データ型で返します。
SECOND()	指定された時刻または日付時刻から秒数を抽出し、それを数値として返します。

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。
TODAY()	現在のオペレーティングシステム日付を日付時刻データ型で返します。
UTOD()	書式設定された日付が含まれる Unicode 文字列を Analytics 日付値に変換します。"Unicode to Date" の省略形です。
WORKDAY()	2つの日付の間の就業(稼働)日数を返します。
YEAR()	指定された日付または日付時刻から年を抽出し、それを YYYY 書式の数値として返します。

財務

財務関数は、年金、貸付金、投資、元金、利息、支払いに関連するさまざまな計算を実行します。

メモ

Analytics 12.0 から、Analytics の前提条件である Visual C++ 再頒布可能パッケージに Microsoft が行った変更により、一部の Analytics 財務関数の結果が Analytics の以前のバージョンの結果と多少異なるようになりました。

Visual C++ の変更は、演算精度を向上させるために Microsoft によって行われました。このため、PMT() や FVSCCHEDULE() などの Analytics 関数の丸め処理が、以前とは異なる方法で動作するようになっています。

関数の説明

関数	説明
CUMIPMT()	期間の範囲中に貸付金に対して支払う累積利息を返します。
CUMPRINC()	期間の範囲中に貸付金に対して支払う累積元金を返します。
EFFECTIVE()	貸付金に適用される実効年利率を返します。

関数

関数	説明
FVANNUITY()	一定の利率を使って計算した一連の預金額の将来価値を返します。将来価値は、毎月の預金額と複利利息の合計です。
FVLUMPSUM()	一定の利率を使って計算した現在の総額の将来価値を返します。
FVSCHEDULE()	一連の利率を使って計算した現在の総額の将来価値を返します。
IPMT()	単一の期間で貸付金に対して支払われた利息を返します。
NOMINAL()	貸付金の名目年利率を返します。
NPER()	貸付金を完済するのに必要な期間の数値を返します。
PMT()	貸付金を完済するのに必要な定期払込金額(元金 + 利息)を返します。
PPMT()	単一の期間で貸付金に対して支払われた元金を返します。
PVANNUITY()	一定の利率を使って計算した一連の将来価値の現在価値を返します。現在価値は現在の総額です。
PVLUMPSUM()	一定の利率を使って計算する特定の将来総額を生成するのに必要な現在価値を返します。現在価値は現在の総額です。
RATE()	期間ごとの利率を返します。

フィールドとレコード

フィールドおよびレコード関数は、Analytics テーブルを構成する基本コンポーネントに対してさまざまなタスクを実行します。

たとえば、フィールドが存在するかどうかをテストし、フィールドのデータ型を検出し、レコード番号を取り込むことができます。フィールドおよびレコード関数は、Analytics スクリプトでのデータ分析中に有用なヘルパータスクを実行できます。

関数の説明

関数	説明
FTYPE()	フィールドまたは変数のデータカテゴリや、Analytics プロジェクト項目の種類を識別する文字を返します。
HASH()	入力値に基づいてソルト付き暗号化ハッシュ値を返します。
ISDEFINED()	指定されたフィールドが定義されている場合は T(true)、そうでない場合は F(false) を返します。

関数	説明
OFFSET()	指定されたバイト数によって補正された開始位置からフィールドの値を返します。
RECLLEN()	現在のレコードの長さを返します。
RECNO()	現在のレコード番号を返します。
RECOFFSET()	現在のレコードを基準にして、指定された n 番目にあるレコードのフィールド値を返します。
VERIFY()	物理データフィールドのデータが有効かどうかを示す論理値を返します。

テーブル、ファイル、およびプロジェクト

テーブル、ファイル、およびプロジェクト関数は、Analytics スクリプトでのデータ分析中に役立つヘルパータスクを実行できます。

たとえば、FTYPE() 関数を使用して、フィールドのデータカテゴリを特定できます。これは、他の関数またはコマンドをフィールドに正しく適用するために知る必要がある場合があります。

関数の説明

関数	説明
FILESIZE()	指定されたファイルのサイズをバイト数で返します。ファイルが存在しない場合は -1 を返します。
FTYPE()	フィールドまたは変数のデータカテゴリや、Analytics プロジェクト項目の種類を識別する文字を返します。
GETOPTIONS()	指定された Analytics オプションの現在の設定 (オプション] ダイアログ ボックスの設定) を返します。
PROPERTIES()	指定された Analytics プロジェクト項目のプロパティ情報を返します。

変数テスト

変数テスト関数では、変数のデータ型および変数が存在するかどうかを確認できます。

関数の説明

関数	説明
FTYPE()	フィールドまたは変数のデータカテゴリや、Analytics プロジェクト項目の種類を識別する文字を返します。

関数

関数	説明
ISDEFINED()	指定されたフィールドが定義されている場合は T(true)、そうでない場合は F(false) を返します。

Python

ACLScript の Python 関数は、Python プログラミング言語を使用して実行された計算の結果を Analytics スクリプトに取り込みます。

ACLScript の Python 関数を使用するには、Analytics スクリプトが実行されるコンピューターに互換性があるバージョンの Python をインストールして構成する必要があります。詳細については、"Python の Analytics 連携用設定" ページ 2697を参照してください。

関数の説明

関数	説明
PYDATE()	外部の Python スクリプトの関数によって計算された日付値を返します。Python によるデータ処理は Analytics の外部で行われます。
PYDATETIME()	外部の Python スクリプトの関数によって計算された日付時刻値を返します。Python によるデータ処理は Analytics の外部で行われます。
PYLOGICAL()	外部の Python スクリプトの関数によって計算された論理値を返します。Python によるデータ処理は Analytics の外部で行われます。
PYNUMERIC()	外部の Python スクリプトの関数によって計算された数値を返します。Python によるデータ処理は Analytics の外部で行われます。
PYSTRING()	外部の Python スクリプトの関数によって計算された文字値を返します。Python によるデータ処理は Analytics の外部で行われます。
PYTIME()	外部の Python スクリプトの関数によって計算された時刻値を返します。Python によるデータ処理は Analytics の外部で行われます。

R

ACLScript の R 関数は、R プログラミング言語を使用して実行された計算の結果を Analytics スクリプトに取り込みます。

ACLScript の R 関数を使用するには、Analytics スクリプトが実行されるコンピューターに互換性があるバージョンの R をインストールして構成する必要があります。詳細については、"ACL for Windows のシステム要件" ページ 2707を参照してください。

関数の説明

関数	説明
RDATE()	R の関数またはスクリプトによって計算された日付値を返します。R によるデータ処理は Analytics の外部で行われます。
RDATETIME()	R の関数またはスクリプトによって計算された日付時刻値を返します。R によるデータ処理は Analytics の外部で行われます。
RLOGICAL()	R の関数またはスクリプトによって計算された論理値を返します。R によるデータ処理は Analytics の外部で行われます。
RNUMERIC()	R の関数またはスクリプトによって計算された数値を返します。R によるデータ処理は Analytics の外部で行われます。
RSTRING()	R の関数またはスクリプトによって計算された文字列の値を返します。R によるデータ処理は Analytics の外部で行われます。
RTIME()	R の関数またはスクリプトによって計算された時刻値を返します。R によるデータ処理は Analytics の外部で行われます。

ビット エンコードと文字エンコード

ビット エンコード関数と文字エンコード関数により、ビット、バイト、文字のエンコードのレベルでデータを認識、操作する一連のツールを実現できます。

関数の説明

関数	説明
ASCII()	指定された文字の ASCII コードを返します。
BIT()	現在のレコードにおける指定されたバイト位置のバイナリ表記を 8 文字の文字列として返します。
BYTE()	現在のレコード内で指定されたバイト位置に格納されている文字を返します。
CHR()	指定された ASCII コードに対応する文字を返します。
DBYTE()	レコード内の指定されたバイト位置にある Unicode 文字を返します。
DHEX()	Unicode 文字列を 16 進数の文字列に変換します。
DIGIT()	指定された位置にある Packed データ型のバイトから、上位または下位の数字を返します。

関数

関数	説明
HEX()	ASCII 文字列を 16 進数の文字列に変換します。
HTOU()	16 進数の文字列を Unicode 文字列に変換します。"Hexadecimal to Unicode" の省略形です。
MASK()	2 つの文字列の最初の各バイトに対してビットごとの AND 演算を実行します。
SHIFT()	入力値の最初の文字のビットを左または右に移動させた 1 文字の文字列を返します。

ABS() 関数

数式の絶対値を返します。数の絶対値は符号のない数です。

構文

```
ABS(数値)
```

パラメーター

名前	型	説明
数値	数値	絶対値を確認する値。

出力

数値。

例

基本的な例

7.2 が返されます。

```
ABS(7.2)
```

7.2 が返されます。

```
ABS(-7.2)
```

AGE() 関数

指定した日付と指定した締切日の間の経過日数(年齢)、現在のオペレーティングシステム日付、または任意の2つの日付間の経過日数を返します。

構文

```
AGE(日付/日付時刻/文字列 <, 締切日>)
```

パラメーター

名前	型	説明
日付/日付時刻/文字列	文字 日付時刻	年齢を調べるフィールド、式、またはリテラル値。
締切日 省略可能	文字 日付時刻	日付/日付時刻/文字列と比較するフィールド、式、またはリテラル値。これを省略した場合は、締切日として、現在のオペレーティングシステム日付が使用されます。

メモ

日付/日付時刻/文字列と締切日は日付時刻値を受け入れることはできません。時刻値だけでAGE()を使用することはできません。

詳細については、「日付時刻データでAGE()を使用する」ページ 2160を参照してください。

出力

数値。

例

基本的な例

締切日がない

2014年12月31日と現在の日付の間の日数を返します。

- 正の値が返された場合、値は、過去2014年12月31日からこれまでの日数に相当します。
- 負の値が返された場合、値は、将来2014年12月31日になるまでの日数に相当します。
- '0'が返された場合、2014年12月31日は現在の日付です。

```
AGE(`20141231`)
```

期日フィールドの日付と現在の日付の間の日数が返されます。

```
AGE(期日)
```

データ型の混在

以下のいずれによっても、指定した2つの日付間の日数である518が返されます。

```
AGE(`20130731`, `20141231`)
```

```
AGE("20130731", "20141231")
```

```
AGE(`20130731`, "20141231")
```

```
AGE(`20130731 235959`, `20141231`)
```

締切日とフィールドの使用

期日フィールドの日付と、締切日である2014年12月31日の間の日数が返されます。

- 締切日より前の日付を指定した場合は、その日付から締切日までの日数に等しい正の値が返されます。

関数

- 締切日より後の日付を指定した場合は、締切日からその日付が発生するまでの日数に等しい負の値が返されます。

```
AGE(期日, `20141231`)
```

2014年12月31日と期日フィールドの日付の間の日数が返されます。結果の絶対値はすぐ上の例と同じですが、戻り値の符号(正または負)が逆になります。

```
AGE(`20141231`, 期日)
```

フィールド同士の日付の比較

支払日フィールドの日付と、期日フィールドの日付の間の日数が返されます。

- 期日前の支払日を指定した場合は、期日までの支払いを示す正の値が返されます。
- 期日後の支払日は、支払遅延を示す負の値を返します。

```
AGE(支払日, 期日)
```

支払日フィールドの日付と、期日フィールドの日付の間の日数に猶予期間の15日を加算した数値が返されます。

- 期日前、または期日後15日までの支払日を指定した場合は、正の値が返されます。
- 期日後15日を超えている支払日を指定した場合は、猶予期間外の支払遅延を示す負の値が返されます。

```
AGE(<支払日>, <期日>+15)
```

高度な例

期限が過ぎた支払いを抽出する

締切日を2014年12月31日として、請求書(Invoice_Date)の経過日数が180日を超えている各レコードについて、名前、金額、請求日付を抽出します。

```
EXTRACT FIELDS Name Amount 請求日 TO "Overdue" IF AGE(請求日, `20141231`) > 180
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

機能の仕組み

AGE() 関数は、2 つの日付間の日数を計算します。

AGE() の使用に適する場面

AGE() を使用できる場面は、2 つの日付を比較して期日を過ぎた取引先を特定する場合や、残高の経過日数を分析する場合など、2 つの日付の間の経過日数が必要な任意の作業を実行する場合です。

負の戻り値

日付/日付時刻/文字列に指定した値が締切日として指定された日付より後の日付であるか、オペレーティングシステム日付(締切日が指定されていない場合)より後の日付である場合は、負数が返されません。

-518 を返します。

```
AGE(`20141231`, `20130731`)
```

どちらの日付が新しいのかに関係なく、2 つの日付間の経過日数を常に正数にする場合、ABS() 関数内の AGE() 関数をネストします。

518 を返します。

```
ABS(AGE(`20141231`, `20130731`))
```

日付時刻データで AGE() を使用する

AGE() 関数では、片方または両方のパラメーターで日付時刻データを使用することができます。ただし、データの時刻部分に UTC オフセット(タイムゾーンインジケーター)が含まれる場合には、注意が必要で

関数

す。

UTC オフセットがない日付時刻データ

時刻データにUTC オフセットが含まれない場合は、日付時刻値の時刻部分は、AGE() で実行される日付演算には影響しません。

UTC オフセットがある日付時刻データ

片方または両方のパラメーターにある時刻データにUTC オフセットが含まれる場合は、日付時刻値の時刻部分は、AGE() で実行される日付演算には影響することがあります。演算を実行する前に、Analytics は自動的にUTC オフセットを調整します。これにより、深夜 0 時を境に調整が前後する場合、結果が1日変わることがあります。

詳細については、「UTC オフセットが時刻日付式に与える影響」 ページ 899を参照してください。

締切日フィールドの使用

締切日のリテラル日付値を必要とするAGEコマンドとは異なり、AGE() 関数では締切日フィールドを使用できません。

例：

```
AGE(支払日, 期日)
```

このようにAGE() 関数を使用することは、式で2つの日付フィールドを減算して日付間の差を求めることと同等です。

例：

```
期日 - 支払日
```

パラメーターの詳細

日付/日付時刻/文字列または締切日に指定された日付時刻フィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付または日付時刻値の指定

日付/日付時刻/文字列または締切日にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` や `"20141231"` のように、値を逆引用符、一重引用符、または二重引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。文字の時刻値ではコロンを使用できません。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。

形式の例	リテラル値の例
YYYYMMDD	`20141231` "20141231"
YYMMDD	`141231` "141231"
YYYYMMDD hhmmss	`20141231 235959` "20141231 235959"
YYMMDDthhmm	`141231t2359` "141231t2359"
YYYYMMDDThh	`20141231T23` "20141231T23"
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500` "20141231 235959-0500"
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01` "141231 2359+01"
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

ALLTRIM() 関数

入力文字列から先頭と末尾のスペースを除去した文字列を返します。

構文

```
ALLTRIM(文字列)
```

パラメーター

名前	型	説明
文字列	文字	先頭と末尾のスペースを除去するフィールド、式、リテラル値。

出力

文字。

例

基本的な例

"Vancouver" が返されます。

```
ALLTRIM(" Vancouver ")
```

"New York" が返されます。

```
ALLTRIM(" New York ")
```

高度な例

文字フィールドの連結

ALLTRIM の使用に適する場面は、名前フィールドと姓フィールドなどの文字フィールドを連結した後のフィールドで、連結された値同士の間には複数の空白が入らないようにスペースを削除する場
合です。

```
DEFINE FIELD Full_Name COMPUTED ALLTRIM(名前) + " " + ALLTRIM(姓)
```

改行なしスペースの削除

改行なしスペースは ALLTRIM() 関数で削除されません。

先頭または末尾の改行なしスペースを削除する必要がある場合は、次の式を使用して演算
フィールドを作成します。

```
DEFINE FIELD Description_cleaned COMPUTED ALLTRIM(REPLACE(説明, CHR  
(160), CHR(32)))
```

REPLACE() 関数がすべての改行なしスペースを標準のスペースに置換してから、ALLTRIM() が
すべての先頭または末尾の標準スペースを削除します。

備考

機能の仕組み

ALLTRIM() 関数は文字列から先頭と末尾のスペースを除去します。文字列内の、先頭と末尾にないス
ペースは、除去されません。

関連する関数

文字列から先頭のスペースのみを除去したい場合はLTRIM() 関数を、末尾のスペースのみを除去した
い場合はTRIM() 関数を使用してください。

ASCII() 関数

指定された文字のASCIIコードを返します。

構文

```
ASCII(文字)
```

パラメーター

名前	型	説明
文字	文字	ASCIIコードの確認対象となる文字。 引用符で囲まれた文字、あるいは複数文字の文字列、フィールド、または式を指定します。複数文字の列を指定した場合、最初の文字だけが評価されます。

出力

数値。

例

基本的な例

65 が返されます。

```
ASCII("A")
```

49 が返されます。

```
ASCII("1")
```

高度な例

タブ文字で始まるレコードの抽出

"Description" フィールドの先頭にタブ文字が入っているレコードを抽出します。タブ文字の ASCII コードは "9" です。

```
EXTRACT RECORD TO "Tab_Entries.acl" IF ASCII(Description) = 9
```

備考

出力不可能文字がないかどうかのテスト

以下の出力不可能文字がないかどうかをテストするには、ASCII() を使用します。

- NULL - ASCII "0"
- タブ - ASCII "9"
- ラインフィード (LF) - ASCII "10"
- キャリッジリターン (CR) - ASCII "13"

関連する関数

ASCII() は CHR() の逆関数です。

AT() 関数

文字値における部分文字列の特定の出現の開始位置を示す数値を返します。

構文

```
AT(出現番号, 検索文字列, 検索されるテキスト)
```

パラメーター

名前	型	説明
出現番号	数値	位置を返す検索文字列の出現箇所 (インスタンス) たとえば、検索文字列の最初の出現箇所の開始位置を返させる場合は、1を指定します。
検索文字列	文字	検索されるテキスト内から検索する部分文字列。この値では大文字と小文字が区別されます。 検索文字列が二重引用符を含んでいる場合は、検索文字列の値を一重引用符で囲む必要があります。 <pre>AT(1, "test", Description)</pre>
検索されるテキスト	文字	検索される値。 テーブル内の複数のフィールドで検索したい場合は、検索されるテキストパラメーターで2つ以上のフィールドを連結することができます。 <pre>AT(1, "test", Description+Summary)</pre>

出力

数値。検索文字列の指定された出現の開始バイト位置を返します。文字列が検出されなかった場合はゼロ値を返します。

例

基本的な例

出現箇所が見つかった場合

4 が返されます。

```
AT(1, "-", "604-669-4225")
```

8 が返されます。

```
AT(2, "-", "604-669-4225")
```

出現箇所が見つらなかった場合

この値には 3 番目のハイフンがないため、0 が返されます。

```
AT(3, "-", "604-669-4225")
```

この値には 4 番目の小文字の "a" がないので、0 が返されます。

```
AT(4, "a", "Alabama")
```

文字のグループ

5 が返されます。

```
AT(2, "iss", "Mississippi")
```

フィールド内の検索

Invoice_Number(請求書番号) フィールド内の各値における最初のハイフンのバイト位置が返されます。

```
AT(1, "-", Invoice_Number)
```

高度な例

2つ目のハイフンが10番目のバイト位置より後に現れる請求書番号を見つける

AT() 関数を使って以下のようなフィルターを作成することにより、テーブル内の請求書番号の一貫性を分析することができます。請求書番号に2つ以上のハイフンが入っていて2つ目のハイフンが10バイト目の位置より後に現れる、すべてのレコードが、このフィルターにより抽出されます。

```
SET FILTER TO AT(2, "-", Invoice_Number) > 10
```

備考

AT() の使用に適する場面

この関数は、文字値における以下の開始位置を取得する場合に使用することができます。

- 部分文字列の開始位置
- 部分文字列の後続の出現箇所の開始位置

フィールド内に同じ部分文字列が複数回出現することを確認したいだけである場合は、OCCURS() 関数がより良い代替手段です。詳細については、「OCCURS() 関数」ページ 2385を参照してください。

出現番号が出現箇所の数より大きい場合に、値が返されます。

出現番号が検索されるテキスト内における部分文字列の実際の出現数より大きい場合には、関数は部分文字列の出現を見つけられないので0を返します。

連結されたフィールドと戻り値

複数のフィールド内を検索した場合、見つかったインスタンスに対する戻り値は、指定したすべてのフィールドにおける検索文字列の開始位置になります。連結したフィールドは1つのフィールドのように扱われません。ただし、ALLTRIM() 関数を使用して個々のフィールドからスペースを除去しなければ、各フィールドの先頭と末尾のスペースを含むフィールドとなります。

たとえば、それぞれ幅が8バイトである2つのフィールドから文字列の最初の出現箇所を検索した場合に、その文字列が2番目のフィールドの先頭で見つかったときは、戻り値は9になります。

BETWEEN() 関数

指定された値が範囲内にあるかどうかを示す論理値を返します。

構文

```
BETWEEN( 値, 最小値, 最大値)
```

パラメーター

名前	型	説明
値	文字 数値 日付時刻	テストするフィールド、式、またはリテラル値。
最小値	文字 数値 日付時刻	範囲の最小値。 フィールド、式、またはリテラル値にすることができます。
最大値	文字 数値 日付時刻	範囲の最大値。 フィールド、式、またはリテラル値にすることができます。

メモ

T(true) と評価する範囲には、**最小値**と**最大値**が含まれます。

文字の範囲については、"SET EXACT の動作" ページ 2172を参照してください。

出力

論理。値が**最小値**以上かつ**最大値**以下である場合は、T(true) を返します。そうでない場合はF(false) を返します。

例

基本的な例

数値型の入力値

T が返されます。

```
BETWEEN(500,400,700)
```

F が返されます。

```
BETWEEN(100,400,700)
```

文字型の入力値

T が返されます。

```
BETWEEN("B","A","C")
```

文字の比較で大文字と小文字が区別され、小文字 "b" は大文字 "A" と "C" の間には位置しないため、F が返されます。

```
BETWEEN("b","A","C")
```

日付時刻型の入力値

T が返されます。

```
BETWEEN(`141230`,`141229`,`141231`)
```

Login_time フィールドの値のうち、午前 7 時 0 分 0 秒から午前 9 時 0 分 0 秒までのすべての値を指定した場合は 'T' が返され、それ以外の値を指定した場合は 'F' が返されます。

```
BETWEEN(Login_time,`t070000`,`t090000`)
```

SET EXACT の動作

Last_Name フィールドの値のうち、"C" から "K" の文字で始まるすべての値を指定した場合は 'T' が返され、それ以外の値を指定した場合は 'F' が返されます (SET EXACT を OFF にしておく必要があります)。

```
BETWEEN>Last_Name, "C", "K")
```

Last_Name フィールドの値のうち、"C" から "J" の文字で始まるすべての値を指定した場合は 'T' が返され、それ以外の値を指定した場合は 'F' が返されます (SET EXACT を ON にしておく必要があります)。単一の文字 "K" を指定した場合には、T が返されます。

```
BETWEEN>Last_Name, "C", "K")
```

フィールドへの入力値

Invoice_Date フィールドの値のうち、2014 年 9 月 30 日から 2014 年 10 月 30 日までのすべての値を指定した場合は 'T' が返され、それ以外の値を指定した場合は 'F' が返されます。

```
BETWEEN Invoice_Date, `20140930`, `20141030` )
```

請求日 (**Invoice_Date**) が発注日 (**PO_Date**) と支払日 (**Paid_Date**) の間にないすべてのレコードに対しては 'T' が返され、それ以外のレコードに対しては 'F' が返されます。

```
NOT BETWEEN Invoice_Date, PO_Date, Paid_Date )
```

Invoice_Amount フィールドの値のうち、2014 年 9 月 30 日から 2014 年 10 月 30 日までのすべての値を指定した場合は 'T' が返され、それ以外の値を指定した場合は 'F' が返されます。

```
BETWEEN Invoice_Amount, 1000, 5000 )
```

高度な例

給与の範囲を表示するためのフィルターを作成する

次の例は、**Employee_List** サンプルテーブルを開き、給与が40,000.00ドル以上、50,000.00ドル以下の従業員のみが含まれるように、表示されるレコードを制限するフィルターを適用しています。

```
OPEN Employee_List
SET FILTER TO BETWEEN(Salary, 40000.00, 50000.00)
```

変化する範囲内の日付を検索するフィルターを作成する

会社の旅費交通費システムのデータを、会社のクレジットカード データに結合するテーブルを作成しました。会社のクレジットカードに課金されたホテルの宿泊料金が払い戻された従業員を検出したいと思います。

金額フィールドで2つのデータのセットを結合し、ホテルの宿泊日と旅費の発生日を使用して、2つの金額が同じ費用を指していることを確認する計画です。問題は、費用システムの日付が会社のクレジットカード データのホテルの日付と1、2日ずれる可能性があることです。

以下の例は、**Joined_expense_data** テーブルを開き、ホテルの宿泊日の範囲内の費用発生日を検索するフィルターを適用します。実際の日付値ではなくフィールドを使用すると、日付とともに範囲が代わります。

```
OPEN Joined_expense_data
SET FILTER TO BETWEEN(T_E_date, Arrival_date-2, Arrival_date+2) OR
BETWEEN(T_E_date, Departure_date-2, Departure_date+2)
```

備考

サポートされているデータ型

BETWEEN() 関数への入力には数値、文字、日付時刻のデータを指定できます。データ型を混在させることはできません。3つの入力はすべて同じデータ型に属している必要があります。

AND 演算子の代わりに BETWEEN() を使用する

AND 演算子を使用する式の代わりに BETWEEN() 関数を使用することができます。

例：

```
BETWEEN(Invoice_Amount, 1000, 5000)
```

上記の関数は、次の式と同等です。

```
Invoice_Amount >= 1000 AND Invoice_Amount <= 5000
```

最小値と最大値の順序

BETWEEN() 関数に指定する最小値と最大値の順序は重要ではありません。値の大小関係は Analytics が自動的に識別します。

次の例ではいずれも T が返されます。

```
BETWEEN(2500, 1000, 5000)
```

```
BETWEEN(2500, 5000, 1000)
```

数値型入力値の小数点以下の精度

比較する数値型入力値の小数点以下の精度が異なる場合、比較は高い方の精度に合わせて行われます。

1.23 は 1.23 と等しいため、T が返されます。

```
BETWEEN(1.23, 1.23, 1.25)
```

小数点第 3 位まで考慮されるため、1.23 は 1.234 より小さいと判定されるので、F が返されます。

```
BETWEEN(1.23, 1.234, 1.25)
```

文字データ

大文字と小文字の区別

文字データが使用される場合、BETWEEN() 関数は大文字と小文字を区別します。文字を比較する場合、"a" は "A" と同等ではありません。

F が返されます。

関数

```
BETWEEN("B", "a", "C")
```

大文字と小文字のどちらかに統一されていないデータを操作する場合は、BETWEEN() 関数を使用する前に UPPER() 関数を使用して、値をすべて大文字に変換することができます。

T が返されます。

```
BETWEEN(UPPER("B"), UPPER("a"), UPPER("C"))
```

部分一致

文字比較では、部分一致がサポートされます。

値が最小値より大きいと見なされる場合があります。

値 "AB" は最小値 "ABC" より小さいように見えますが、T が返されます。

```
BETWEEN("AB", "ABC", "Z")
```

最大値が値より小さいと見なされる場合があります。

値 "ZZ" は最大値 "Z" より大きいように見えますが、T が返されます。

```
BETWEEN("ZZ", "ABC", "Z")
```

メモ

文字比較において、短い値は、一致を成す長い値の先頭に現れる必要があります。

部分一致と SET EXACT

部分一致は、Analytics のデフォルト設定である SET EXACT = OFF の場合に有効になります。SET EXACT = ON にすると、部分一致は無効になり、比較値が一致を成すには、正確に一致しなければなりません。

SET EXACT が ON の場合、上記の例はどちらも False になります。

SET EXACT([正確な文字比較を行う](#) オプション) の詳細については、"SET コマンド" ページ 2081 を参照してください。

SET EXACT の OFF または ON への設定

BETWEEN() 関数で [正確な文字比較を行う](#) オプションが使用されないようにしたい場合は、[オプション](#) ダイアログボックスの [テーブル](#) タブ([ツール > オプション](#)) でこのオプションが選択されていないことを確認してください。

スクリプトを使用している場合は、BETWEEN() 関数が現れる前に `SET EXACT OFF` コマンドを追加します。必要に応じて、`SET EXACT ON` コマンドによって前の状態に戻すことができます。

日付時刻パラメーター

関数への入力として指定された日付、日付時刻、または時刻フィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式、日付時刻書式、または時刻書式でも使用することができます。

日付型、日付時刻型、時刻型の入力値を混在させる

BETWEEN() 関数の3つの入力値に日付値、日付時刻値、および時刻値を混在させることは禁止されていませんが、これら日付時刻のサブタイプを混在させると、意味のない結果が返される可能性があります。

日付時刻値の日付部分のみに関心があっても、時刻部分はまだ計算の一部を構成しているため、Analytics は対応するシリアル値を使用して、日付時刻の計算を処理しています。

次の例で考えてみましょう。

2014年12月31日は最小値と最大値によって指定された範囲に含まれるため、Tが返されます。

```
BETWEEN(`20141231`, `20141229`, `20141231`)
```

2014年12月31日は最小値と最大値によって指定された範囲に含まれるように見えますが、Fが返されます。

```
BETWEEN(`20141231 120000`, `20141229`, `20141231`)
```

これら2つの式の対応するシリアル値を見ると、どうして2番目の式がfalseと評価されるのかがわかりません。

シリアル値である値がシリアル値である最大値と等しいため、Tが返されます。

```
BETWEEN(42003.000000, 42001.000000, 42003.000000)
```

シリアル値である値がシリアル値である最大値より大きいため、Fが返されます。

```
BETWEEN(42003.500000, 42001.000000, 42003.000000)
```

2つの日付が同一であっても、シリアル値の `42003.500000` は `42003.000000` より大きいため、範囲外になります。`0.500000` は 12:00 PM に相当するシリアル値です。

日付時刻サブタイプ同士を一致させる

混在する日付時刻サブタイプによって発生する可能性のある問題を回避するには、サブタイプ同士を一致させる関数を使用します。

たとえば、次の式では、上記の2番目の式と同じ初期値が使用されていますが、FでなくTが返されません。

```
BETWEEN(CTOD(DATE(`20141231
120000`, "YYYYMMDD")), "YYYYMMDD"), `20141229`, `20141231`)
```

リテラル日付、日付時刻、または時刻値の指定

関数への入力のいずれかにリテラルの日付値、日付時刻値、または時刻値を指定する場合は、次の表内の書式に制限されます。また、`20141231`のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
thhmmss	`t235959`

形式の例	リテラル値の例
Thhmm	`T2359`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

BINTOSTR() 関数

ZONED または EBCDIC 文字データから変換された Unicode 文字データを返します。"Binary to String" の省略形です。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

構文

```
BINTOSTR(文字列, 文字列のタイプ)
```

パラメーター

名前	型	説明
文字列	文字	Unicode 文字エンコードに変換する ZONED 値または EBCDIC 値。
文字列のタイプ	文字	変換元の書式。次のいずれかの値を指定する必要があります。 <ul style="list-style-type: none">"A" - 変換元は ZONED(ASCII) データです。"E" - 変換元は EBCDIC データです。

出力

文字。

例

基本的な例

式 ZONED(-6448,4) は値 -6448 を文字型書式 "644Q" に変換していますが、Analytics の Unicode 版では BINTOSTR() を使って ZONED() の出力を Unicode 文字に変換する必要があります。

Unicode 書式の "644Q" が返されます。

```
BINTOSTR(ZONED(-6448,4), "A")
```

備考

BINTOSTR() の使用に適する場面

この関数は、ZONED() および EBCDIC() 関数からの戻り値を Unicode 値に変換する場合に使用することができます。

メモ

この関数を Analytics の Unicode 版の ZONED() および EBCDIC() の戻り値に適用しない場合は、エンコードが正確に解釈されないので、戻り値の表示が不正確になります。

BIT() 関数

現在のレコードにおける指定されたバイト位置のバイナリ表記を 8 文字の文字列として返します。

構文

```
BIT(バイト位置)
```

パラメーター

名前	型	説明
バイト位置	数値	バイナリ値として返されるバイト位置。

出力

文字。

例

基本的な例

8 番目のバイトに "1" が含まれる場合に、"00110001" が返されます。

```
BIT(8)
```

9 番目のバイトに "A" が含まれる場合に、"01000001" が返されます。

```
BIT(9)
```

17 番目のバイトに "a" が含まれる場合に、"01100001" が返されます。

```
BIT(17)
```

高度な例

BIT () および SUBSTRING () を使用した値の抽出

バイト 17 に 8 つの信用フラグが含まれているとします。

3 番目のビットが 1 に設定されたすべての顧客 ("出荷対象外" の顧客) のレコードを抽出するには、次のように指定します。

```
EXTRACT IF SUBSTRING(BIT(17), 3, 1) = "1"
```

この例では、SUBSTRING() 関数を使って、3 番目のビットの値を抽出しています。

備考

機能の仕組み

BIT() は、指定されたバイト位置にあるバイトを 1 と 0 で構成される 8 文字の列に変換します。

BIT() の使用に適する場面

BIT() は、バイト中の個々のビットを調べる場合に使用できます。

関連する関数

指定されたバイト位置にある文字を読み出したい場合は、BYTE() 関数を使用してください。

BLANKS() 関数

指定された数の空白スペースを含んでいる文字列を返します。

構文

```
BLANKS( 数 )
```

パラメーター

名前	型	説明
カウント	数値	挿入する空白スペースの数。

出力

文字。

例

基本的な例

" " が返されます。

```
BLANKS(5)
```

"ABC Corporation" を返します。

```
"ABC" + BLANKS(1) + "Corporation"
```

備考

BLANKS() の使用に適する場面

BLANKS() 関数を使用できる場面は、フィールドを一致させる場合や、スクリプトで変数を初期化する場合、あるいはフィールドの書式設定や文字列の連結を行うときに空白を挿入する場です。

BYTE() 関数

現在のレコード内で指定されたバイト位置に格納されている文字を返します。

構文

```
BYTE(バイト位置)
```

パラメーター

名前	型	説明
バイト位置	数値	文字値として返されるバイト位置。 バイト位置はフィールド定義とは関係なく、レコード内での位置(1から開始)を参照します。

出力

文字。

例

基本的な例

"1"を値とするIDフィールドで始まるレコードに対して、"1"が返されます。

```
byte(112)
```

高度な例

一貫した書式設定を使用した印刷ファイルまたは PDF 中のレコードを識別する

BYTE() を使用して、データファイル内のレコードのうち、特定の文字が特定のバイト位置に存在するレコードを識別することができます。これは、ドキュメント全体を通してデータが一貫した方法で書式設定されている、印刷イメージ(レポート)ファイルまたは Adobe Acrobat(PDF)ファイルで典型的なケースです。

たとえば、バイト位置 113 にピリオドのあるレコードを見つけて抽出するには、次のコマンドを使用します。

```
EXTRACT RECORD IF BYTE(113) = "."TO "Output.fil"
```

備考

BYTE() の使用に適する場面

BYTE() 関数を使用すると、フィールドを定義しなくても、レコード内の特定の位置にある内容を調べることができます。

EBCDIC データに対する BYTE() の使用

EBCDIC データに対してこの関数を使用した場合は、返される値も EBCDIC になります。この値を文字値と比較することはできないかもしれません。

関連する関数

指定されたバイト位置のバイナリ表記を取得したい場合は、BIT() 関数を使用してください。

CDOW() 関数

指定された日付または日付時刻の曜日を返します。"Character Day of Week" の省略形です。

構文

```
CDOW(日付/日付時刻, 長さ)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	曜日を返すフィールド、式、またはリテラル値。
長さ	数値	出力文字列の長さを指定する 1 から 9 の間の値。曜日名の省略形を表示するには、より小さな値を指定してください。

出力

文字。

例

基本的な例

2014 年 12 月 31 日が水曜日 (Wednesday) に当たり、長さが 9 のため、"Wednesday" が返されます。

```
CDOW(`20141231`, 9)
```

2014 年 12 月 31 日が水曜日 (Wednesday) に当たり、長さが 9 のため、"Wed" が返されます。

```
CDOW(`20141231 235959`, 3)
```

Invoice_date フィールドの各値に対して曜日の正式名称が返されます。

```
CDOW(Invoice_date, 9)
```

Receipt_timestamp フィールドの各値に対して曜日の省略名称が返されます。

```
CDOW(Receipt_timestamp, 3)
```

高度な例

日付の曜日を識別するフィールドを追加する

CDOW() 関数を使用して、日付フィールドの全日付の曜日を識別するための演算フィールドを作成することができます。演算フィールドを作成したら、ビューの日付列の横にそれを追加できます。

```
DEFINE FIELD Name_of_Day COMPUTED CDOW(Trans_Date, 3)
```

フィルターを作成して、週末に発生した取引をテストする

CDOW() 関数を使用して、フィルターを作成し、週末に発生した取引を抽出することができます。

```
SET FILTER TO CDOW(Trans_Date, 3) = "Sat" OR CDOW(Trans_Date, 3) = "Sun"
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

長さパラメーターが曜日名より短い場合、曜日名は指定された長さに切り詰められます。長さパラメーターが曜日名より長い場合、曜日名は空白スペースで埋められます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、`20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。	

関連する関数

曜日を数字(1 ~ 7)として返したい場合は、CDOW() でなく DOW() を使用してください。

CHR() 関数

指定された ASCII コードに対応する文字を返します。

構文

```
CHR(数値)
```

パラメーター

名前	型	説明
数値	数値	1 から 255 の間の有効な数式。

出力

文字。

例

基本的な例

"A" が返されます。

```
CHR(65)
```

"1" が返されます。

```
CHR(49)
```

高度な例

通貨フィールドの値に英国のポンド記号(£)を付加する

Invoice_Amount フィールドの金額の前にポンド記号(ASCII コード 163)を付加する演算フィールドを作成します。数値型の **Invoice_Amount** フィールドがまず文字型のフィールドに変換されて、次いで先頭と末尾の空白額が削除されます。

```
DEFINE FIELD Currency_UK COMPUTED CHR(163)+ALLTRIM(STRING(Invoice_
Amount, 12))
```

備考

CHR() の使用に適する場面

CHR() 関数は、キーボードから直接入力できない文字や画面に表示できない文字など、あらゆる ASCII コードに対応する文字を返す場合に使用できます。CHR() を使用して、フィールドやレコードの中からこれら特定の文字を見つけることができます。

NUL の参照

ASCII の NUL(ヌル)文字は Analytics でテキスト修飾子として使用されるため、これに相当する `CHR(0)` を参照すると、予測できない結果をもたらす可能性があります。できれば参照を避けてください。

関連する関数

CHR() は ASCII() の逆関数です。

CLEAN() 関数

文字列内で、最初の無効な文字とそれ以降のすべての文字を空白に置き換えます。

構文

```
CLEAN(文字列 <,追加指定の無効文字>)
```

パラメーター

名前	型	説明
文字列	文字	デフォルトで設定されている無効文字および追加指定の無効文字を削除する対象となる値
追加指定の無効文字 省略可能	文字	文字列から削除する対象となる、デフォルトで設定されている無効な文字以外の無効な文字。追加指定の無効文字は複数、指定できます。 <pre>" ,;\"</pre> <p>タブ文字、ヌル文字、およびキャリッジリターン/ラインフィード文字はデフォルトの無効な文字であり、自動的に削除されるため、指定する必要はありません。</p> <p>二重引用符を追加指定の無効文字として指定するには、追加指定の無効文字に、一重引用符で囲んだ二重引用符を指定してください。</p> <pre>''''</pre>

出力

文字。

例

基本的な例

"ABC " ("ABC" および 4 つの空白) が返されます。

```
CLEAN("ABC%DEF", "%")
```

"1234.56 " ("1234.56" および 6 つの空白) が返されます。

```
CLEAN("1234.56,111,2", ",")
```

備考

CLEAN() の使用に適する場面

この関数は、無効なデータを含むフィールドが正しく印刷されるようにするために使用します。また、この関数を使用すると、フィールドの一部を分離することができます。たとえば、姓と名前の両方が入っている顧客フィールドから姓を分離できます。

無効な一重引用符または二重引用符の指定

無効文字として一重引用符と二重引用符を両方とも指定するには、CLEAN() 関数を入れ子にする必要があります。

```
CLEAN(CLEAN(string, '''), '''')
```

Automatic CLEAN()

Analytics スクリプトに `SET CLEAN ON` を追加することによって、すべての文字フィールドに対して自動的に CLEAN() 関数を適用できます。このオプションを使用して、追加指定の無効文字を指定することはできません。

CMOY() 関数

指定された日付または日付時刻の月の名前を返します。"Character Month of Year" の省略形です。

構文

```
CMOY(日付/日付時刻, 長さ)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	月の名前を返すフィールド、式、またはリテラル値。
長さ	数値	出力文字列の長さを指定する 1 から 9 の間の値。月の名前の省略形を表示するには、より小さな値を指定してください。

出力

文字。

例

基本的な例

"December" が返されます。

```
CMOY(`20141231`, 9)
```

"Dec" が返されます。

```
CMOY(`20141231 235959`, 3)
```

Receipt_timestamp フィールドの各値に対して月の省略名称が返されます。

```
CMOY(Receipt_timestamp, 3)
```

Invoice_date フィールドの各値に対して月の正式名称が返されます。

```
CMOY(Invoice_date, 9)
```

Invoice_date フィールドの各値の 15 日後に当たる月の正式名称が返されます。

```
CMOY(Invoice_date + 15, 9)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

長さパラメーターが月の名前より短い場合、月の名前は指定された長さに切り詰められます。長さパラメーターが月の名前より長い場合、月の名前は空白スペースで埋められます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、`{20141231}` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2 つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 'L' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC(Coordinated Universal Time:

関数

協定世界時)からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。	

関連する関数

月を数字 (1 ~ 12) として返したい場合は、CMOY() でなく MONTH() を使用してください。

COS() 関数

ラジアン単位で表された角度のコサインを、小数点以下 15 桁の精度で返します。

構文

```
COS(ラジアン)
```

パラメーター

名前	型	説明
ラジアン	数値	ラジアン単位の角度。

出力

数値。

例

基本的な例

指定したラジアン数に対して 0.500000000000000 が返されます。

```
COS(1.047197551196598)
```

高度な例

入力値として度を使用する

関数

60 度の余弦である 0.5000000000000000 が返されます。

```
COS(60 * PI( )/180)
```

小数点第 3 位までの小数に丸める

60 度の余弦を小数点第 3 位に丸めた 0.500 が返されます。

```
DEC(COS(60 * PI( )/180),3)
```

備考

Mantissa Arc Test の実行

Analytics の 3 つの三角関数 SIN()、COS()、および TAN() は、ベンフォードの法則に関連する Mantissa Arc Test の実行をサポートします。

度のラジアンへの変換

入力値が度で表されている場合は、PI() 関数を使用して角度をラジアンに変換できます。(度 * PI()/180) = ラジアンとなります。必要に応じて、DEC() 関数を使用して、戻り値を丸めたり切り捨てたりすることができます。

CTOD() 関数

文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。

構文

```
CTOD(文字/数値 <,書式>)
```

パラメーター

名前	型	説明
文字/数値	文字 数値	日付に変換する、あるいは日付を抽出するフィールド、式、またはリテラル値。
書式 省略可能	文字	文字/数値を日付書式にした値。YYYYMMDD または YYMMDD を除く日付書式を使用した値を指定する場合には、書式が必須です。書式の例: "DD/MM/YYYY" メモ 書式パラメーターを必要とする日付時刻値で CTOD 関数を使用する場合は、書式の日付部分のみを指定してください。時刻部分を指定してはいけません。例: <pre>CTODT("31/12/2014 23:59:59", "DD/MM/YYYY hh:mm:ss")</pre> 時刻部分を指定すると、結果が表示されなくなります。

出力

日付時刻。日付値は、現在 Analytics に設定されている日付の表示書式を使用して出力されます。

例

基本的な例

文字リテラルの入力値

`20141231` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、31 Dec 2014 が返されます。

```
CTOD("20141231")
```

```
CTOD("31/12/2014", "DD/MM/YYYY")
```

```
CTOD("20141231 235959")
```

数値リテラルの入力値

`20141231` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、31 Dec 2014 が返されます。

```
CTOD(20141231)
```

```
CTOD(31122014, "DDMMYYYY")
```

```
CTOD(20141231.235959)
```

フィールドへの文字型入力値

指定した文字フィールドの各値が現在の Analytics 日付表示形式を使用した日付として返されます。

```
CTOD(Invoice_date, "DD/MM/YYYY")
```

```
CTOD(Receipt_timestamp)
```

フィールドへの数値型入力値

指定した数値フィールドの各値が現在の Analytics 日付表示形式を使用した日付として返されます。

```
CTOD(Due_date, "DDMMYYYY")
```

```
CTOD(Payment_timestamp)
```

高度な例

文字フィールドまたは数値フィールドを日付と比較する

日付を表す値が格納されている文字フィールドまたは数値フィールドに対して日付と比較するには、CTOD() 関数を使用します。

下のフィルターでは、次の2つの値が比較されています。

- 書式 DDMMYYYY で数値として日付を格納する数値型の **Due_date** フィールド
- リテラル日付値、2014 年 7 月 1 日

```
SET FILTER TO CTOD(Due_date, "DDMMYYYY") < `20140701`
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

必須の日付書式

日付値または日付時刻値を含んでいる文字フィールドおよび数値フィールドは、次の表内の書式と一致している必要があります。日付時刻値は、そのデータ型に有効な日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があり、かつ、2つの間に区切り文字を入れる必要があります。

関数

日付、または日付時刻値の日付部分は、Analytics でサポートされており、そのデータ型に有効であれば、どのような日付書式でも使用することができます。ただし、YYYYMMDD と YYMMDD 以外の書式は、書式で正しく定義されてさえいれば使用できます。

日付書式	区切り文字形式	時刻書式
文字フィールド		
YYYYMMDD	単一の空白スペース	hhmmss hh:mm:ss
YYMMDD	文字 't'	hhmm hh:mm
書式で定義されている場合は、データ型に有効な、Analytics がサポートする任意の日付書式	文字 'T'	hh
		+/-hhmm +/-hh:mm (UTC オフセット)
		+/-hh (UTC オフセット)
		<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>
数値フィールド		
YYYYMMDD	小数点	hhmmss
YYMMDD		hhmm
書式で定義されている場合は、データ型に有効な、Analytics がサポートする任意の日付書式		hh

他の日付時刻変換関数

文字または数値から日付時刻への変換

関数	説明
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

CTODT() 関数

文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。

構文

```
CTODT(文字/数値 <,書式>)
```

パラメーター

名前	型	説明
文字/数値	文字 数値	日付時刻に変換するフィールド、式、またはリテラル値。
書式 省略可能	文字	文字/数値を日付書式にした値。値の日付部分に対し、YYYYMMDD または YYMMDD を除く日付書式を使用した値を指定する場合には、書式が必須です。書式の例: "DD/MM/YYYY"

出力

日付時刻。日付時刻値は、現在 Analytics に設定されている日付と時刻の表示書式を使用して出力されます。

例

基本的な例

文字リテラルの入力値

`20141231t235959` が返されます。これは、現在の Analytics 日付時刻表示書式である DD MMM YYYY hh:mm:ss を適用した 31 Dec 2014 23:59:59 として表示されます。

```
CTODT("20141231 235959")
```

```
CTODT("31/12/2014 23:59:59", "DD/MM/YYYY hh:mm:ss")
```

数値リテラルの入力値

`20141231t235959` が返されます。これは、現在の Analytics 日付時刻表示書式である DD MMM YYYY hh:mm:ss を適用した 31 Dec 2014 23:59:59 として表示されます。

```
CTODT(20141231.235959)
```

```
CTODT(31122014.235959, "DDMMYYYY.hhmmss")
```

フィールドへの文字型入力値

Receipt_timestamp フィールドの各値が現在の Analytics の日付表示書式を適用した日付時刻として返されます。

```
CTODT(Receipt_timestamp, "DD/MM/YYYY hh:mm:ss")
```

フィールドへの数値型入力値

数値フィールド、**Payment_timestamp** の各値が、現在の Analytics の日付表示書式を適用した日付時刻として返されます。

```
CTODT(Payment_timestamp, "DD/MM/YYYY hh:mm:ss")
```

高度な例

文字フィールドまたは数値フィールドを日付時刻と比較する

日付時刻を表す値が格納されている文字フィールドまたは数値フィールドに対して日付時刻を比較するには、CTODT() 関数を使用します。

下のフィルターでは、次の 2 つの値が比較されています。

- 日付時刻を書式 DD/MM/YYYY hh:mm:ss の文字データとして格納する文字フィールド、**Receipt_timestamp**
- リテラル日付時刻値、2014 年 7 月 1 日 13:30:00

```
SET FILTER TO CTODT(Receipt_timestamp, "DD/MM/YYYY hh:mm:ss") <
'20140701t133000'
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

必須の日付時刻書式

日付時刻値を含んでいる文字フィールドおよび数値フィールドは、次の表内の書式と一致している必要があります。日付時刻値は、そのデータ型に有効な日付、区切り文字、および時刻の書式を任意に組み合わせ使用することができます。日付を時刻の前に置く必要があり、かつ、2 つの間に区切り文字を入れる必要があります。

値の日付部分は、Analytics でサポートされており、そのデータ型に有効であれば、どのような日付書式でも使用することができます。ただし、YYYYMMDD と YYMMDD 以外の書式は、書式で正しく定義されてさえいれば使用できます。書式を使用する場合は、時刻書式も指定する必要があり、次の表に示されている時刻書式のいずれかである必要があります。

Analytics は、日付時刻値の日付部分と時刻部分の間の区切り文字を自動的に認識するため、書式に区切り文字を指定する必要はありません。指定したい場合は、区切り文字を指定してもかまいません。

日付書式	区切り文字形式	時刻書式
文字フィールド		
YYYYMMDD	単一の空白スペース	hhmmss hh:mm:ss
YYMMDD	文字 't'	hhmm hh:mm
書式で定義されている場合は、データ型に有	文字 'T'	hh

日付書式	区切り文字形式	時刻書式
効な、Analytics がサポートする任意の日付書式		
		+/-hhmm +/-hh:mm (UTC オフセット)
		+/-hh (UTC オフセット)
		<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>
数値フィールド		
YYYYMMDD	小数点	hhmmss
YYMMDD		hhmm
書式で定義されている場合は、データ型に有効な、Analytics がサポートする任意の日付書式		hh

他の日付時刻変換関数

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

CTOT() 関数

文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

構文

```
CTOT(文字/数値)
```

パラメーター

名前	型	説明
文字/数値	文字 数値	時刻に変換する、あるいは時刻を抽出するフィールド、式、またはリテラル値。

出力

日付時刻。時刻値は、現在 Analytics に設定されている時刻の表示書式を使用して出力されます。

例

基本的な例

文字リテラルの入力値

`t235959` が返されます。これは、現在の Analytics 時刻表示書式である hh:mm:ss を適用した 23:59:59 として表示されます。

```
CTOT("t235959")
```

```
CTOT("23:59:59")
```

関数

```
CTOT("20141231 235959")
```

数値リテラルの入力値

`t235959` が返されます。これは、現在の Analytics 時刻表示書式である hh:mm:ss を適用した 23:59:59 として表示されます。

```
CTOT(.235959)
```

```
CTOT(0.235959)
```

```
CTOT(20141231.235959)
```

フィールドへの文字型入力値

文字フィールド `Login_time` の各値が、現在の Analytics の時刻表示書式を適用した時刻として返されます。

```
CTOT(Login_time)
```

フィールドへの数値型入力値

数値フィールド、`Payment_timestamp` の各値が、現在の Analytics の時刻表示書式を適用した、日付部分なしの時刻として返されます。

```
CTOT(Payment_datetime)
```

高度な例

文字フィールドまたは数値フィールドを時刻と比較する

時刻を表す値が格納されている文字フィールドまたは数値フィールドに対して時刻と比較するには、`CTOT()` 関数を使用します。

下のフィルターでは、次の2つの値が比較されています。

- 時刻を数値データとして格納した数値型の `Login_time` フィールド
- リテラル時刻値 09:30:00

```
SET FILTER TO CTOT(Login_time) > `t093000`
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

必須の日付時刻書式

時刻値または日付時刻値を含んでいる文字フィールドおよび数値フィールドは、次の表内の書式と一致している必要があります。

時刻値は、区切り文字と時刻の書式を任意に組み合わせて使用することができます。関数が正しく動作するためには、時刻値の前に区切り文字を置くか、あるいは時刻の要素の間にコロンを入れる必要があります。

日付時刻値は、そのデータ型に有効な日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2つの間に区切り文字を入れる必要があります。

文字または数値の日付値を日付に変換する場合、あるいは、文字または数値の日付時刻値から日付を抽出し、それを日付として返す場合は、`CTOD()` 関数を使用してください。

文字または数値の日付時刻値を日付時刻に変換する場合は、`CTODT()` 関数を使用してください。

日付書式	区切り文字形式	時刻書式
文字フィールド		
YYYYMMDD	単一の空白スペース	hhmmss hh:mm:ss
YYMMDD	文字 't'	hhmm hh:mm
	文字 'T'	hh

関数

日付書式	区切り文字形式	時刻書式
		+/-hhmm +/-hh:mm (UTC オフセット)
		+/-hh (UTC オフセット)
		メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。)
数値フィールド		
YYYYMMDD	小数点	hhmmss
YYMMDD		hhmm
		hh

他の日付時刻変換関数

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

CUMIPMT() 関数

期間の範囲中に貸付金に対して支払う累積利息を返します。

構文

CUMIPMT(利率, 期間, 金額, 開始期, 終了期 <, 種類>)

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	支払期間の総数。
金額	数値	貸付金の元金。
開始期	数値	計算対象となる最初の期。 開始期には0は指定できません。
終了期	数値	計算対象となる終了の期。 終了期には、支払期間の総数を超える数を指定することはできません。
種類 省略可能	数値	支払いのタイミング: ○ 0 - 期末払い ○ 1 - 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間を指定する際には、1期あたりの利率を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利5%の2年間の貸付金または投資に対して月払いする場合は、利率に0.05/12、期間に2 * 12を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に0.05、期間に2を指定します。

出力

数値。

例

基本的な例

\$275,000 の貸付金を年利 6.5 パーセントで 25 年間にわたって返済する場合 (支払期日は月末です) の、2 年目に支払う利息の合計額、(\$) 17437.23 が返されます。

```
CUMIPMT(0.065/12, 12*25, 275000, 13, 24, 0)
```

上記の貸付金について、初年度に支払う利息の合計額、(\$) 17741.31 が返されます。

```
CUMIPMT(0.065/12, 12*25, 275000, 1, 12, 0)
```

備考

関連する関数

CUMPRINC() 関数は CUMIPMT() 関数に対して補完的役割を果たします。

CUMIPMT() 関数は単一の期間に支払われた利息を計算します。

CUMPRINC() 関数

期間の範囲中に貸付金に対して支払う累積元金を返します。

構文

CUMPRINC(利率, 期間, 金額, 開始期, 終了期 <, 種類 >)

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	支払期間の総数。
金額	数値	貸付金の元金。
開始期	数値	計算対象となる最初の期。 開始期には0は指定できません。
終了期	数値	計算対象となる終了の期。 終了期には、支払期間の総数を超える数を指定することはできません。
種類 省略可能	数値	支払いのタイミング: ○ 0 - 期末払い ○ 1 - 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間を指定する際には、1期あたりの利率を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利5%の2年間の貸付金または投資に対して月払いする場合は、利率に0.05/12、期間に2 * 12を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に0.05、期間に2を指定します。

出力

数値。

例

基本的な例

\$275,000 の貸付金を年利 6.5 パーセントで 25 年間にわたって返済する場合 (支払期日は月末です) の、2 年目に支払う元金の合計額、\$4844.61 が返されます。

```
CUMPRINC(0.065 / 12, 12*25, 275000, 13, 24)
```

上記の貸付金の最初の月に支払う元金の額、\$367.24 が返されます。

```
CUMPRINC(0.065/12, 12*25, 275000, 1, 1, 0)
```

備考

関連する関数

CUMIPMT() 関数は CUMPRINC() 関数に対して補完的役割を果たします。

CUMPPMT() 関数は単一の期間に支払われた元金を計算します。

DATE() 関数

指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。

構文

```
DATE(<日付/日付時刻> <,書式>)
```

パラメーター

名前	型	説明
日付/日付時刻 省略可能	日付時刻	日付を抽出するフィールド、式、またはリテラル値。これを省略した場合は、現在のオペレーティングシステム日付が返されます。
書式 省略可能	文字	出力文字列に適用する書式。例: "DD/MM/YYYY" このパラメーターを省略した場合は、現在の Analytics の日付表示書式が使用されます。日付/日付時刻を省略した場合に、書式を指定することはできません。

出力

文字。

例

基本的な例

"20141231" に対して現在の Analytics 日付表示書式を適用した値が返されます。

```
DATE(`20141231 235959`)
```

"31-Dec-2014" が返されます。

```
DATE(`20141231 235959`, "DD-MMM-YYYY")
```

現在のオペレーティングシステム日付が、現在の Analytics の日付表示書式を適用した文字列として返されます。

```
DATE()
```

Receipt_timestamp フィールドの各値が、現在の Analytics の日付表示書式を適用した文字列として返されます。

```
DATE(Receipt_timestamp)
```

Receipt_timestamp フィールドの各値に対して、指定した日付表示書式を適用した文字列が返されます。

```
DATE(Receipt_timestamp, "DD/MM/YYYY")
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

出力文字列の長さ

出力文字列の長さは常に 12 文字です。指定した出力書式、または Analytics の日付の表示書式が 12 文字より短い場合は、出力文字列の末尾に空白が埋められます。

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

書式を使用して出力文字列の表示方法を制御する場合は、サポートされている任意の Analytics 日付表示書式を使用することができます。例：

- DD/MM/YYYY
- MM-DD-YY
- DD MMM YYYY

書式は、一重引用符または二重引用符を使用して指定する必要があります。たとえば、`"DD MMM YYYY"` のように指定します。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	<code>`20141231`</code>
YYMMDD	<code>`141231`</code>
YYYYMMDD hhmmss	<code>`20141231 235959`</code>
YYMMDDthhmm	<code>`141231t2359`</code>
YYYYMMDDThh	<code>`20141231T23`</code>
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	<code>`20141231 235959-0500`</code>
YYMMDD hhmm+/-hh (UTC オフセット)	<code>`141231 2359+01`</code>
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

関連する関数

日付時刻値として現在のオペレーティングシステム日付を返すには、DATE() の代わりに TODAY() を使用してください。

他の日付時刻変換関数

日付時刻から文字への変換

関数	説明
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

DATETIME() 関数

日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。

構文

```
DATETIME(<日付時刻> <,書式>)
```

パラメーター

名前	型	説明
日付時刻 省略可能	日付時刻	変換するフィールド、式、またはリテラル値。これを省略した場合は、現在のオペレーティングシステム日付が返されます。
書式 省略可能	文字	出力文字列に適用する書式。例: "DD/MM/YYYY" このパラメーターを省略した場合は、現在の Analytics の日付表示書式が使用されます。日付/日付時刻を省略した場合に、書式を指定することはできません。

出力

文字。

例

基本的な例

リテラル日付時刻の入力値

"20141231 235959" に対して現在の Analytics 日付時刻表示書式を適用した値が返されます。

```
DATETIME(`20141231 235959`)
```

"31-Dec-2014 11:59 P" が返されます。

```
DATETIME(`20141231 235959`, "DD-MMM-YYYY hh:mm A")
```

現在のオペレーティングシステム日付時刻に対して、現在のAnalyticsの日付時刻表示書式を適用した文字列が、返されます。

```
DATETIME()
```

フィールドへの入力値

Receipt_timestamp フィールドの各値が、現在のAnalyticsの日付時刻表示書式を適用した文字列として返されます。

```
DATETIME(Receipt_timestamp)
```

Receipt_timestamp フィールドの各値に対して、指定した日付時刻表示書式を適用した文字列が返されます。

```
DATETIME(Receipt_timestamp, "DD/MM/YYYY hh:mm:ss")
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

出力文字列の長さ

出力文字列の長さは常に27文字です。指定した出力書式、またはAnalyticsの日付と時刻の表示書式が27文字より短い場合は、出力文字列の末尾に空白が埋められます。

パラメーターの詳細

日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付時刻書式でも使用することができます。

書式を使用して出力文字列の表示方法を制御する場合は、次の表内の書式に制限されます。

- 日付、時刻、および AM/PM の書式を任意に組み合わせて使用することができます。
- 日付は時刻の前に置く必要があります。Analytics は出力文字列の区切り文字として単一のスペースを自動的に使用するので、両者の間に区切り文字を置く必要はありません。
- AM/PM 書式は省略可能で、最後に置きます。
- 書式は、一重引用符または二重引用符を使用して指定する必要があります。

例: "DD-MMM-YYYY hh:mm:ss AM"

日付書式	時刻書式	AM/PM 書式	例
サポートされるすべての Analytics 日付表示書式	hh:mm:ss	なし 24 時間制	"DD/MM/YYYY hh:mm:ss"
	hhmmss	AM または PM 12 時間制	"MMDDYY hhmmss PM"
	hh:mm	A または P 12 時間制	"DD-MMM-YYYY hh:mm A"
	hhmm		
	hh		

リテラル日付時刻値の指定

日付時刻にリテラルの日付時刻値を指定する場合は、次の表内の書式に制限されます。また、`20141231 235959` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があり、かつ、2 つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh	`141231 2359+01`

形式の例	リテラル値の例
(UTC オフセット)	
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

他の日付時刻変換関数

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。

関数

関数	説明
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime"の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time"の省略形です。

DAY() 関数

指定された日付または日付時刻から日にちを抽出し、それを数値(1 ~ 31)として返します。

構文

```
DAY(日付/日付時刻)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	日にちを抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

31 が返されます。

```
DAY(`20141231`)
```

```
DAY(`20141231 235959`)
```

Invoice_date フィールドの各値に対して日付が返されます。

```
DAY(Invoice_date)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`

形式の例	リテラル値の例
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。	

関連する関数

以下の各値を返させたい場合は、それぞれ該当する関数を使用します。

- 曜日を数字 (1 ~ 7) として返させる場合 : DAY() でなく DOW() を使用
- 曜日の名前を返させる場合 : DAY() でなく CDOW() を使用

DBYTE() 関数

レコード内の指定されたバイト位置にある Unicode 文字を返します。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

構文

```
DBYTE(バイト位置)
```

パラメーター

名前	型	説明
バイト位置	数値	文字値として返されるバイト位置。 意味のある値を返すために、2 バイト文字の開始場所を指定する必要があります。これは単に、バイト位置パラメーターには奇数を指定してくださいということです。

出力

文字。

例

基本的な例

次の例は、11 文字 (22 バイト) の値、**美丽 10072DOE** を含んでいる Unicode 値に適用されたときのこの関数の動作を示しています。

"丽" が返されます。

```
DBYTE(3)
```

"D" が返されます。

```
DBYTE(17)
```

"E" が返されます。

```
DBYTE(21)
```

備考

DBYTE() の使用に適する場面

DBYTE() 関数は、フィールドを定義しなくてもレコード内の特定の位置にある内容を調べる場合に使用できます。

DEC() 関数

指定された小数点以下の桁数で値または数式の結果を返します。

構文

```
DEC(数値, 小数位)
```

パラメーター

名前	型	説明
数値	数値	小数位の数を調節する値または結果。 <ul style="list-style-type: none">○ 整数 - 小数点桁数が末尾のゼロとして数値の最後に追加されます。○ 小数 - 小数点桁数が減る場合、数値は切り捨てられずに、端数処理されます。小数点桁数が増える場合、末尾のゼロが数値の最後に追加されます。
小数位	数値	戻り値で使用する小数点以下の桁数。 メモ DEC() を使用して、結果の小数点精度を増やすことはできません。 小数点精度を増やす方法については、「数式の端数処理と小数点精度の制御」ページ 874を参照してください。

出力

数値。

例

基本的な例

7.00 が返されます。

```
DEC(7, 2)
```

7.565 が返されます。

```
DEC(7.5647, 3)
```

7.56470 が返されます。

```
DEC(7.5647, 5)
```

高度な例

日歩の計算

`Annual_rate` というフィールドの日歩を小数点第 6 位まで求めるには、次のように指定します。

```
DEC(Annual_rate, 6) / 365
```

備考

DEC() の使用に適する場面

この関数は、フィールドで小数位を調整する場合や、値や計算結果の小数位を指定した桁数に丸める場合に使用できます。

DEC() は固定小数点端数処理を元に戻すことはできません

DEC() 関数を使用して、数式で固定小数点演算で実行される標準の端数処理を元に戻すことはできません。

例

Analytics で次の一連の式を検討します。

```
1.1 * 1.1 = 1.2  
1.1 * 1.10 = 1.21  
DEC(1.1 * 1.1, 2) = 1.20
```

固定小数点の端数処理は、`1.1 * 1.1`の結果は1.2で、1.21ではありません。これは端数処理されていない結果です。DEC()を使用して、小数点2桁の結果を指定しても、小数点2桁の精度は作成されません。代わりに、末尾のゼロが追加され、精度を上げずに、指定された小数点桁数を作成します。

小数点精度を増やす方法については、"数式の端数処理と小数点精度の制御" ページ 874を参照してください。

関連する関数

値を最も近い整数に丸めたい場合は、"ROUND() 関数" ページ 2483 関数を使用してください。

DHEX() 関数

Unicode 文字列を 16 進数の文字列に変換します。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

構文

```
DHEX(フィールド)
```

パラメーター

名前	型	説明
フィールド	文字	16 進数の文字列に変換する Unicode 文字列。

出力

文字。

例

基本的な例

"004100420043003100320033" が返されます。

```
DHEX("ABC123")
```

備考

機能の仕組み

DHEX() 関数は、各 2 バイト文字をビッグエンディアンで、最上位の 2 バイトから順に格納して表示します。

各文字は 4 文字のコードで表されます。出力文字列は、フィールド値の 4 倍の長さで、16 進数の値を構成する 0 から 9 までの数字と A から F までの文字を含んでいます。

関連する関数

DHEX() は、16 進数の文字列を Unicode 文字列に変換する HTOU() 関数の逆関数です。

DICECOEFFICIENT() 関数

2つの指定された文字列のダイス係数を返します。これは、2つの文字列間の類似度を測定したものです。

構文

```
DICECOEFFICIENT(文字列 1, 文字列 2 <,n-gram>)
```

パラメーター

名前	型	説明
文字列1	文字	比較の最初の文字列。
文字列2	文字	比較の2番目の文字列。
<i>n-gram</i> 省略可能	数値	使用する <i>n-gram</i> の長さ。 1以上の整数を指定します。 <i>n-gram</i> の長さを大きくすると、2つ文字列の間の類似度の基準が厳しくなります。 長さを指定しない場合、デフォルトの長さの2が使用されます。 <i>n-gram</i> は、ダイス係数計算の構成要素であり、比較対象となる2つの文字列にとって、構成要素であると同時に重なり合う、部分文字列(文字ブロック)です。 詳細については、「備考」ページ 2241を参照してください。

出力

数値。値は2つの文字列のダイス係数です。これは、同一の2つの文字列の *n-gram* の合計数の割合を表します。範囲は0.0000 ~ 1.0000です。

例

基本的な例

n-gram の長さが結果に及ぼす影響

下の 3 つの例は、同じ 2 つの文字列を比較したものです。返される類似度は、指定した *n*-gram の長さによって変わります。

次の例では 0.9167 が返されます(デフォルトの *n*-gram 長さ(2) を使用すると、2 つの文字列間の *n*-gram の同一度が 92% になる)。

```
DICECOEFFICIENT("125 SW 39TH ST, Suite 100","Suite 100, 125 SW 39TH ST")
```

次の例では 1.0000 が返されます(*n*-gram 長さに 1 を使用すると、2 つの文字列間の *n*-gram の同一度が 100% になる)。

```
DICECOEFFICIENT("125 SW 39TH ST, Suite 100","Suite 100, 125 SW 39TH ST", 1)
```

次の例では、0.8261 が返されます(*n*-gram 長さに 3 を使用すると、2 つの文字列間の *n*-gram の同一度が 83% になる)。

```
DICECOEFFICIENT("125 SW 39TH ST, Suite 100","Suite 100, 125 SW 39TH ST", 3)
```

フィールドへの入力値

デフォルトの *n*-gram 長さ 2 を使用した場合の、文字列「125 SW 39TH ST, Suite 100」と "Address" フィールドの各値との間のダイス係数が返されます。

```
DICECOEFFICIENT(Address,"125 SW 39TH ST, Suite 100")
```

高度な例

要素間の順序を入れ替えた文字列の使用

要素間の順序を入れ替えた文字列を検索する場合に、DICECOEFFICIENT() を最適化するには、 n -gram 長さを小さくし、重要でない文字を削除します。

次の例では 0.7368 が返されます(デフォルトの n -gram 長さ(2) を使用すると、2 つの文字列間の n -gram の同一度が 76% になる)。

```
DICECOEFFICIENT("John Smith","Smith, John")
```

次の例では、1.0000 が返されます(姓と名 の間のカンマを除外し、 n -gram 長さ 1 を使用すると、2 つの文字列間の n -gram の同一度が 100% になる)。

```
DICECOEFFICIENT("John Smith", EXCLUDE("Smith, John", ","), 1)
```

「125 SW 39TH ST, Suite 100」に対する類似度のランキング

次の例は、「125 SW 39TH ST, Suite 100」と "Address" フィールドの各値との間のダイス係数を表示する演算フィールド **Dice_Co** を作成しています。

```
DICECOEFFICIENT(Address,"125 SW 39TH ST, Suite 100")
```

「125 SW 39TH ST, Suite 100」との類似度に基づいてすべての "Address" フィールド値をランキングするには、演算フィールド **Dice_Co** をビューに追加し、そのフィールドの降順でクイックソートします。

「125 SW 39TH ST, Suite 100」に対するあいまい重複の抽出

次の例は、「125 SW 39TH ST, Suite 100」との指定した類似度以上の類似度を持つすべての "Address" フィールド値を抽出するフィルターを作成しています。

```
SET FILTER TO DICECOEFFICIENT(Address,"125 SW 39TH ST, Suite 100") > 0.5
```

式の数値を変更すると、フィルターリングされる値の類似度を調整することができます。

備考

DICECOEFFICIENT() の使用に適する場面

DICECOEFFICIENT() 関数は、ほぼ同一の値(あいまい重複)を検出する場合に使用できます。また、同一またはほぼ同一の内容で、要素間の順序を入れ替えた文字列を検索する場合にも使用できます。例:

- 数字が入れ替わった電話番号、社会保障番号
- 形式が異なる複数の同じ住所

機能の仕組み

DICECOEFFICIENT() は2つの評価された文字列間のダイス係数を返します。ダイス係数は、文字列間の類似度を測定したもので、0.0000 ~ 1.0000 で示されます。返された値が大きいと、2つの文字列間の類似度が高くなります。

- **1.0000** - 各文字列が同じ文字から構成されていることを示します。ただし、文字の順序と大文字小文字の表記は異なる場合があります。
- **0.7500** - 2つの文字列の n -gram の同一度が75% になることを意味します。
- **0.0000** - 2つの文字列に共有 n -grams(以下で説明)がないか、計算で使用される n -gram の指定された長さが比較対象の2つの文字列のうちの短い方の文字列よりも長いことを意味します。

使用上のヒント

- **フィルターリングまたは並べ替え** - ダイス係数に基づいてフィールドの値をフィルターリングするか並べ替えると、比較文字列に最も類似している値が特定されます。
- **大文字と小文字の区別** - この関数では大文字と小文字が区別されないため、"SMITH" は "smith" と同じであると判断されます。
- **先頭と末尾にあるスペース** - フィールド内の先頭と末尾にあるスペースを自動的に除去するため、パラメーターとしてフィールドを指定するときに TRIM() および ALLTRIM() 関数を使う必要はありません。

ダイス係数の計算方法

ダイス係数は、同一の2つの文字列の n -gram の合計数の割合を表します。

まず、ダイス係数は比較対象の2つの文字列を n -gram 文字ずつに分割して計算されます。 n -grams (または q -grams) は長さが n の重なり合うサブ文字列または重なり合う文字ブロックです。 n gram パラメーターを使用して n の長さを指定するか、デフォルトの長さの2を使用できます。

2つの名前は n -grams に分割されます。

ここに名前、"John Smith"と"Smith, John D."があり、長さ2文字および3文字の n -gram に分割します。アンダースコアはスペースを示します。内部スペースと句読点は文字と見なされます。

n -gram の長さ	"John Smith" n -gram	"Smith, John D." n -gram
2	Jo oh hn n_ _S Sm mi it th	Sm mi it th h, _ _J Jo oh hn n_ _D .D.
3	Joh ohn hn n_ _S _Sm Smi mit ith	Smi mit ith th, h, _ _J _Jo Joh ohn hn n_ _D _D.

ダイス係数の計算式

2つの比較対象の文字列に対して n -grams が確立されたら、次の式を使用して計算が完了します。

- $2 \times$ 共有 n -grams の数 / 両方の文字列の n -grams の数

"共有 n -grams" は両方の文字列に表示される n -grams です。たとえば、 n -gram 長さ2 (AB | BC と BC | CD) の場合、「ABC」と「BCD」は n -gram 「BC」を共有します。

ダイス係数の計算例

次の表では、異なる n -gram の長さを使用して、2つの文字列「John Smith」および「John D. Smith」のダイス係数を計算する例を示します。

同じ文字列のペアの n -gram の長さが大きくなると、ダイス係数値が小さくなり、類似度が低いことを示します。文字列は同じですが、類似度の基準がより厳しくなります。文字列をより長い n -grams に分割する場合、共有と見なされるには、より長い連続文字が n -gram と一致する必要があります。

この点については、 n -gram の長さを大きくすると、文字の相対位置の重みが大きくなると考えることもできます。対照的に、長さ1の n -gram を使用しているときには、文字の相対位置は考慮されません。相対位置は、文字列内での絶対的位置ではなく、文字列の、他の文字列からの相対的位置のことです。

ヒント

特に入れ替えを検索している場合は、 n -gram 長さ1を使用します。

n -gram の長さ	"John Smith" n -gram	"Smith, John D." n -gram	共有 n -grams	ダイス係数
1	J o h n n_ _S m i t h (10 n -grams)	S m i t h , _ _J o h n _ _D .D. (14 n -grams)	10	$2 \times 10 / (10 + 14) = 0.8333$
2 (デフォルト)	Jo oh hn n_ _S Sm mi it th (9 n -grams)	Sm mi it th h, _ _J Jo oh hn n_ _D .D. (13 n -grams)	8	$2 \times 8 / (9 + 13) = 0.7273$

<i>n</i> -gram の長さ	"John Smith" <i>n</i> -gram	"Smith, John D." <i>n</i> -gram	共有 <i>n</i> -grams	ダイス係数
3	Joh ohn hn_ n_S _Sm Smi mit ith (8 <i>n</i> -grams)	Smi mit ith th, h,_ ,_J _Jo Joh ohn hn_ n_D _D. (12 <i>n</i> -grams)	6	$2 \times 6 / (8+12) =$ 0.6000
4	John ohn_ hn_S n_Sm _Smi Smit mith (7 <i>n</i> -grams)	Smit mith ith, th,_ h,_J ,_Jo _Joh John ohn_ hn_D n_D. (11 <i>n</i> -grams)	4	$2 \times 4 / (7+11) =$ 0.4444

DICECOEFFICIENT() と ISFUZZYDUP() および LEVDIST() の比較

DICECOEFFICIENT() 関数とレーベンシュタイン距離を使用する ISFUZZYDUP() および LEVDIST() の主な違いの 1 つは、DICECOEFFICIENT() は比較対象の 2 つの文字列の文字または文字ブロックの相対位置を重視しないか、完全に無視するという点です。これに対して、レーベンシュタイン距離に基づく関数では相対位置が重要になります。

比較対象値(要素入れ替えの考慮あり)

要素全体の配置が入れ替わる可能性がある住所などの文字列を比較する場合は、DICECOEFFICIENT() を使用した方がよいことがあります。たとえば、同じ住所で「Suite」要素が入れ替わっている場合は、DICECOEFFICIENT() を使用すると高い類似度が示されますが、LEVDIST() では高い相違性が示されます。

住所ペア	ダイス係数 (デフォルトの <i>n</i> -gram 2)	レーベンシュタイン距離
<ul style="list-style-type: none"> ◦ 125 SW 39TH ST, Suite 100 ◦ Suite 100, 125 SW 39TH ST 	0.9167	22 (レーベンシュタイン距離が大きくなれば、2 つの文字列間の相違も大きくなります)

比較対象値(要素入れ替えの考慮なし)

入れ替えが問題ではない場合、LEVDIST() はより有用な結果を提供する可能性があります。たとえば、同じ会社名に異なる句読点が含まれている場合、DICECOEFFICIENT() を使用すると高い相違度が示されますが、LEVDIST() では高い類似度が示されます。

会社名の組み合わせ	ダイス係数 (デフォルトの <i>n</i> -gram 2)	レーベンシュタイン距離
<ul style="list-style-type: none"> ◦ AVS, Inc 	0.3750	3

会社名の組み合わせ	ダイス係数 (デフォルトの n -gram 2)	レーベンシュタイン距離
◦ A.V.S. Inc		

DIGIT() 関数

指定された位置にある Packed データ型のバイトから、上位または下位の数字を返します。

構文

```
DIGIT(バイト位置, 位置)
```

パラメーター

名前	型	説明
バイト位置	数値	レコード内のバイト位置。
位置	数値	返させる数字： <ul style="list-style-type: none">バイトの上位 4 ビットを返させる場合は 1 を指定します。バイトの下位 4 ビットを返させる場合は 2 を指定します。

出力

数値。

例

基本的な例

小数点以下 2 桁で、10 バイト目から始まる、値 123.45(00 12 34 5C) のパックフィールドは、データレコードでは次の形式で表されます。

	バイト 10	バイト 11	バイト 12	バイト 13
上位 4 ビット (1)	0	1	3	5
下位 4 ビット (2)	0	2	4	C

次の例では 3 が返されます(12 バイト目の上位 4 ビットに格納されている数字を取得します)。

```
DIGIT(12, 1)
```

次の例では 4 が返されます(12 バイト目の上位 4 ビットに格納されている数字を取得します)。

```
DIGIT(12, 2)
```

備考

機能の仕組み

DIGIT() は、1 バイトの半分をそれぞれ区分し、位置パラメーターで指定されたバイトの値を 0 から 15 までの数字で返します。

DIGIT() の使用に適する場面

DIGIT() は個々のハーフバイト(上位バイトまたは下位バイト)にアクセスする場合に使用できます。Unisys アプリケーションなどの、ハーフバイト(4 ビット)単位で配列されたパックフィールドを使用するアプリケーションで作業を行う場合は、これが必要です。

DOW() 関数

指定された日付または日付時刻の曜日を表す数値(1 ~ 7)を返します。"Day of Week" の省略形です。

構文

```
DOW(日付/日付時刻)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	曜日の数値を返す対象となるフィールド、式、またはリテラルの値。

出力

数値。

例

基本的な例

2014年12月31日は4番目の曜日である水曜日に当たるため、4が返されます。

```
DOW(`20141231`)
```

```
DOW(`20141231 235959`)
```

Invoice_date フィールドの各値の曜日を表す数値が返されます。

```
DOW(Invoice_date)
```

高度な例

週末に発生する取引の特定

DOW() 関数を使用して、週末に発生する取引を特定することができます。次のフィルターは、Trans_Date フィールドの日付のうち、土曜日または日曜日となる日付を抽出します。

```
SET FILTER TO DOW(Trans_Date) = 7 OR DOW(Trans_Date) = 1
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、`20141231` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります、かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 'y' または 'T' です。
- **時刻値** - 24時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

関連する関数

以下の各値を返させたい場合は、それぞれ該当する関数を使用します。

- 曜日の名前を返させる場合 : DOW() でなく CDOW() を使用
- 日付を数字 (1 ~ 7) として返させる場合 : DOW() でなく DAY() を使用

DTOU() 関数

Analytics 日付値を指定された言語およびロケール書式の Unicode 文字列に変換します。"Date to Unicode" の省略形です。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

構文

```
DTOU(<日付> <,ロケールコード> <,スタイル>)
```

パラメーター

名前	型	説明
日付 省略可能	日付時刻	<p>Unicode 文字列に変換するフィールド、式、またはリテラル値。これを省略した場合は、現在のオペレーティングシステム日付が使用されます。</p> <p>日付には日付時刻値を指定できますが、値の時刻部分は無視されません。単独の時刻値はサポートされていません。</p> <p>フィールドまたはリテラルの日付値を指定できます。</p> <ul style="list-style-type: none"> フィールド-にはどのような日付書式でも使用することができます(ただし、フィールド定義で正しく書式を定義している場合)。 リテラル-には YYYYMMDD 書式または YYMMDD 書式のいずれかを使用する必要があります。例: `20141231` <p>サポートされている最も古い日付値は 1969 年 12 月 31 日です。</p>
ロケール 省略可能	文字	<p>出力文字列の言語(と、オプションで、特定の国や地域に関連付けられている言語バージョン)を指定するロケールコード。</p> <p>たとえば、`zh` では中国語が指定され、`pt_BR` ではブラジルのポルトガル語が指定されます。</p> <p>これを省略した場合は、コンピューターに設定されているデフォルトのロケールが使用されます。言語コードを指定したが、国コードを指定しなかった場合は、言語コードにデフォルトで設定されている国コードが適用されます。</p> <p>日付を指定しなかった場合に、ロケールを指定することはできません。</p> <p>ロケールコードに関する情報については、www.unicode.org を参照してください。</p>

関数

名前	型	説明
スタイル 省略可能	数値	Unicode 文字列に使用する日付書式スタイル。この書式スタイルは、指定したロケールの標準書式スタイルと一致します。 <ul style="list-style-type: none">0 - "Sunday, September 18, 2016" などの完全仕様の書式1 - "September 18, 2016" などの長い書式2 - "Sep 18, 2016" などの中ぐらいの長さの書式3 - "9/18/16" などの短い数値型書式 これを省略した場合は、デフォルト値の 2 が使用されます。日付とロケールを指定しなかった場合に、スタイルを指定することはできません。

出力

文字。

例

基本的な例

リテラルの入力値

"31 de dezembro de 2014" が返されます。

```
DTOU(`20141231`, "pt_BR", 1)
```

"31 grudnia 2014" が返されます。

```
DTOU(`20141231`, "pl", 1)
```

フィールドの入力値

Invoice_date フィールドの数値の各日付に対する Unicode 文字列が返されます。

```
DTOU(Invoice_date, "zh", 1)
```

出力では正式の日付スタイルが使用されます。

"星期三, 2014 十二月 31" が返されます(地域識別子の指定なし)。

```
DTOU(`20141231`, "zh", 0)
```

"2014年12月31日 星期三" が返されます(地域識別子の指定あり)。

```
DTOU(`20141231`, "ja_JP", 0)
```

出力では長い形式の日付スタイルが使用されます。

"2014 十二月 31" が返されます(地域識別子の指定なし)。

```
DTOU(`20141231`, "zh", 1)
```

"2014年12月31日" が返されます(地域識別子の指定あり)。

```
DTOU(`20141231`, "ja_JP", 1)
```

備考

関連する関数

DTOU() は、Unicode 文字列を日付に変換する UTOD() 関数の逆関数です。

EBCDIC() 関数

EBCDIC 文字エンコードに変換された文字列を返します。

構文

```
EBCDIC(文字列)
```

パラメーター

名前	型	説明
文字列	文字	EBCDIC に変換する値。

出力

文字。

例

基本的な例

Returns "ñòó@Æ '...@â£K":

```
EBCDIC("123 Fake St.")
```

高度な例

```
エクスポート対象となる、EBCDIC でエンコードされる  
フィールドの作成
```

EBCDIC エンコードを必要とするアプリケーションへエクスポートするために、**ADDRESS** フィールドの値の EBCDIC でエンコードされた値を格納するフィールドを作成するには、次のように指定します。

```
DEFINE FIELD Name_Exp COMPUTED EBCDIC(Name)
```

備考

EBCDIC() の使用に適する場面

この関数は、データを EBCDIC(Extended Binary Coded Decimal Interchange Code) 文字エンコードに変換する場合に使用できます。EBCDIC 文字エンコードは、主として z/OS のような IBM メインフレームオペレーティングシステム上で使用されます。

EFFECTIVE() 関数

貸付金に適用される実効年利率を返します。

構文

```
EFFECTIVE(名目利率, 期間)
```

パラメーター

名前	型	説明
名目利率	数値	名目年利率。
期間	数値	年間の複利計算回数。 メモ 整数を指定してください。小数部分を指定しても切り捨てられるためです。

出力

数値。利率は小数点以下 8 桁で計算されます。

例

基本的な例

次の例では、クレジットカードの年利が 18% で、未払い分に対して毎月複利計算される場合の実効年利率、0.19561817(19.56%) が返されます。

```
EFFECTIVE(0.18, 12)
```

備考

実効年利率とは

貸付金の実効年利率とは、未払い額に対する利息に対して毎月または毎日複利計算が行われた場合の、実際に支払われた年利率です。

関連する関数

NOMINAL()関数はEFFECTIVE()関数の逆関数です。

EOMONTH() 関数

指定された日付から起算して、指定された月数だけ前または後の月の末日の日付を返します。

構文

```
EOMONTH(<日付/日付時刻> <,月数>)
```

パラメーター

名前	型	説明
日付/日付時刻 省略可能	日付時刻	月末の日付を計算するための起算日となるフィールド、式、またはリテラル値。これを省略した場合は、現在のオペレーティングシステム日付から月末の日付が求められます。 メモ 日付/日付時刻には日付時刻値を指定できますが、値の時刻部分は無視されます。
月数 省略可能	数値	日付/日付時刻の前または後にある、月を示す数字。これを省略した場合は、デフォルトの0(ゼロ)が使用されます。 日付/日付時刻を省略した場合に、月数を指定することはできません。

出力

日付時刻。日付値は、現在 Analytics に設定されている日付の表示書式を使用して出力されます。

例

基本的な例

入力値なし

現在のオペレーティングシステム日付の月末の日付が返されます。

```
EOMONTH( )
```

リテラルの入力値

`20140131` に対して現在の ACL 日付表示書式である DD MMM YYYY を適用した、31 Jan 2014 が返されます。

```
EOMONTH(`20140115`)
```

`20140430` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、30 Apr 2014 が返されます。

```
EOMONTH(`20140115`, 3)
```

`20131031` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、31 Oct 2013 が返されます。

```
EOMONTH(`20140115`, -3)
```

フィールドの入力値

Invoice_date フィールドの各日付の 3 か月後の月の末日が返されます。

```
EOMONTH(Invoice_date, 3)
```

Invoice_date フィールドの各日付に 15 日の猶予期間を足した日付から 3 か月後の月の末日が返されます。

```
EOMONTH(Invoice_date + 15, 3)
```

Invoice_date フィールドの各日付の月の初日が返されます。

```
EOMONTH(Invoice_date, -1) + 1
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

日付時刻書式

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付値には、次のいずれかの書式を使用する必要があります。

- YYYYMMDD
- YYMMDD

リテラル日付値はバッククォートで囲む必要があります。例: ``20141231``

月数の値の動作

- **値が正の場合** - 出力日付は指定された日付/日付時刻より新しい日付になります。
- **値が負の場合** - 出力日付は指定した日付/日付時刻より前の日付になります。
- **値を省略した場合、または '0' (ゼロ) を使用した場合** - 出力日付は日付/日付時刻が存在する月の末日になります。

月の初日の日付を返させる

月の初日の日付を返させるには、EOMONTH() 関数の結果に 1 を加算します。

次の例では `20140201` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、01 Feb 2014 が返されます。

```
EOMONTH(`20140115`) + 1
```

関連する関数

指定された日付から起算して、指定された月数だけ前または後の月の末日の日付ではなく正確な日付を返したい場合は、GOMONTH() 関数を使用してください。

EXCLUDE() 関数

指定した文字を除外する文字列を返します。

構文

```
EXCLUDE(文字列, 取り除く文字)
```

パラメーター

名前	型	説明
文字列	文字	文字を除外するフィールド、式、またはリテラル値。
取り除く文字	文字	取り除く文字のリスト。 取り除く文字の中に二重引用符を指定する場合は、文字のリスト全体を一重引用符で囲む必要があります。 例: <code>'"/'</code>

出力

文字。

例

基本的な例

入力文字列からすべての数字を取り除いた "Alberni Street" が返されます。

```
EXCLUDE("1550 Alberni Street", "0123456789")
```

Product_Number フィールドにあるすべての値からスラッシュとシャープ記号を取り除いたものが返されます。

```
EXCLUDE(Product_Number, "/"#")
```

備考

機能の仕組み

EXCLUDE() 関数は、文字列内の各文字を、取り除く文字に列挙された文字と比較します。一致した文字は出力文字列から取り除かれます。

たとえば、`EXCLUDE("123-45-4536", "-")` の出力は "123454536" になります。

一致する文字がない場合

文字列と取り除く文字の間に一致する文字がない場合は、関数の出力は文字列と同じになります。

たとえば、`EXCLUDE("ABC", "D")` の出力は "ABC" です。

大文字と小文字の区別

EXCLUDE() 関数では大文字と小文字が区別されます。取り除く文字に "ID" を指定した場合、"id#94022" からこれらの文字は取り除かれませんが、大文字と小文字の両方が混在している可能性がある場合は、UPPER() 関数を使用して文字列を大文字に変換します。

例：

```
EXCLUDE(UPPER("id#94022"), "ID#")
```

使用上のヒント

含める文字のセットが大きく、取り除く文字のセットが小さい場合は、EXCLUDE() を使用します。

一重引用符と二重引用符を取り除く

引用符は文字列の前後の区切り文字として使用されるため、一重引用符と二重引用符を取り除くには、引用符のタイプごとに EXCLUDE() が使用されるように、EXCLUDE() を入れ子にする必要があります。

```
EXCLUDE(EXCLUDE(field_to_process, "'"), '"')
```

関連する関数

EXCLUDE() 関数は INCLUDE() 関数の逆関数です。

EXP() 関数

数式の指数値(底 10)を指定された小数点以下の桁数で返します。

構文

```
EXP(数値, 小数位)
```

パラメーター

名前	型	説明
数値	数値	指数値を返す対象となる数値フィールド、式、値。
小数位	数値	戻り値に含める小数点以下桁数。

出力

数値。

例

基本的な例

1000.00 が返されます。

```
EXP(3, 2)
```

72443.596007 が返されます。

```
EXP(4.86, 6)
```

高度な例

立方根を求める

フィールド X の立方根を小数点以下 2 桁で表すフィールドを作成するには、次のようにします。

```
DEFINE FIELD cube_root COMPUTED EXP(LOG(X, 6) / 3, 2)
```

ヒント

ある値の指数 n の累乗根は、値の対数を n で割ってから、その結果の指数を取ることによって求められます。

備考

機能の仕組み

この関数は数式の指数値 (底は 10) を返します。指数値は 10 の n 乗と定義されています。たとえば、3 の指数値は 10^3 、つまり 1000 です。

EXP() の使用に適する場面

EXP() 関数は、複雑な数学的計算を必要とする財務アプリケーションで使用します。EXP() 関数は指数演算子 (^) と同様の演算を行いますが、LOG() 関数も実行するアプリケーションで使用すると有用です。

関連する関数

指数の逆は対数です。したがって、EXP() は LOG() 関数の逆関数です。

FILESIZE() 関数

指定されたファイルのサイズをバイト数で返します。ファイルが存在しない場合は -1 を返します。

構文

```
FILESIZE(ファイル名)
```

パラメーター

名前	型	説明
ファイル名	文字	<p>ファイルの名前。</p> <p>ファイルが Analytics プロジェクトと同じフォルダーに存在する場合は、ファイルのパスを指定する必要はありません。</p> <p>他のフォルダーに存在するファイルの場合は、相対パスまたは絶対パスを指定します。例：</p> <ul style="list-style-type: none">"results\test_output.fil""c:\results\test_output.fil" <p>メモ</p> <p>テーブル名ではなく、Analytics テーブルの物理データファイル名 (.fil) を指定する必要があります。</p>

出力

数値。

例

基本的な例

14744 が返されます。

```
FILESIZE("Inventory.fil")
```

サイズを調べるファイルが Analytics プロジェクトと同じフォルダーにない場合は、ファイルの相対パスか絶対パスを指定します。

6018 が返されます。

```
FILESIZE("C:\ACL Data\Sample Data Files\Backup\Ap_Trans.fil")
```

高度な例

ファイルが存在しない場合にスクリプトを実行する

Metaphor_Inventory_2002.fil ファイルが存在しない場合にのみ **import_data** というスクリプトを実行するには、次のコマンドを入力します。

```
DO SCRIPT import_data IF FILESIZE("Metaphor_Inventory_2002.fil") = -1
```

Analytics コマンド ログにファイルのサイズを記録する

Analytics コマンド ログに **Metaphor_Inventory_2002.fil** のサイズを記録するには、**CALCULATE** コマンドを使用します。

```
CALCULATE FILESIZE("Metaphor_Inventory_2002.fil")
```

FIND() 関数

指定された文字列が、特定のフィールド中またはレコード全体のどこかに存在するかどうかを示す論理値を返します。

メモ

FIND() 関数と "FIND コマンド" ページ 1809 は Analytics の 2 つの異なる機能であり、大きな違いがあります。

構文

```
FIND(文字列 <, 検索フィールド>)
```

パラメーター

名前	型	説明
文字列	文字	検索する文字列。この検索では大文字と小文字が区別されません。
検索フィールド 省略可能	文字	検索場所のフィールドまたは変数。省略した場合は、未定義部分もすべて含めたレコードの全体が検索されます。

出力

論理。指定された文字列の値が見つかった場合は T(true)、そうでない場合は F(false) を返します。

例

基本的な例

レコード全体を検索する

次の例では、単一のフィールド、複数のフィールドのいずれか、または未定義部分の値が文字列 "New York" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND("New York")
```

単一のフィールド内の検索

次の例では、**City** フィールドの値が文字列 "New York" であるすべてのレコードに対し、'T' が返されません。それ以外のレコードに対しては 'F' が返されます。

```
FIND("New York", City)
```

City フィールドの値が文字列 "Ne" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND("Ne", City)
```

次の例では、**City** フィールドの値が1つ以上のスペースと文字列 "New York" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND(" New York", City)
```

次の例では、`v_search_term` 変数の値と完全一致または部分一致する値を **Description** フィールドに持つすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND(v_search_term, Description)
```

複数のフィールド内の検索

次の例では、**City** フィールドか **City_2** フィールドのいずれかの値が文字列 "New York" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND("New York", City+City_2)
```

次の例では、**City** フィールドか **City_2** フィールドのいずれかの値が文字列 "New York" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND("New York", City) OR FIND("New York", City_2)
```

他の関数との組み合わせ

次の例では、**Last_Name_2** フィールドの値が **Last_Name** フィールドの値と完全一致または部分一致するすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FIND(ALLTRIM>Last_Name), Last_Name_2)
```

備考

FIND() の使用に適する場面

FIND() 関数は、指定された文字列が特定のフィールド、2 つ以上のフィールドのいずれか、またはレコード内のいずれかのフィールドに存在するかどうかを調べる場合に使用できます。

文字列が一致する仕組み

文字列の値は、正確に一致することもあれば、長い文字列の中に含まれることもあります。フィールド内の先頭スペースは、文字列値の先頭に 1 つ以上のスペースを入れない限り、検索には影響しません。

レコード全体を検索する

任意指定の検索フィールドが指定されていない場合は、未定義部分もすべて含めたレコードの全体が検索されます。レコード全体が検索される場合には、フィールドの境界は無視され、フィールドの末尾のスペースは文字として扱われます。

メモ

レコード全体を検索する場合は、物理レコードが検索されます。演算フィールドおよび関連フィールドは検索されません。

フィールドのサブセットを検索する

テーブル内のフィールドのサブセット内を検索したい場合は、2 つ以上のフィールドを連結して検索フィールドに指定することができます。たとえば、**City** フィールドと **City_2** フィールドの両方で文字列 "New York" を検索するには、次の関数を使用します。

```
FIND("New York", City+City_2)
```

連結したフィールドは1つのフィールドのように扱われます。ただし、ALLTRIM() 関数を使用して個々のフィールドからスペースを除去しなければ、各フィールドの先頭と末尾のスペースを含むフィールドとなります。

また、各フィールド内を別々に検索する式を作成することもできます。

```
FIND("New York", City) OR FIND("New York", City_2)
```

文字列の先頭にスペースが含まれている場合には、この2つの手法の検索結果は異なってくる可能性があります。

大文字と小文字の区別と正確な文字比較

FIND() 関数は大文字と小文字を区別せず、ASCII 文字とEBCDIC 文字の両方を検索します。この関数は、**正確な文字比較を行う**]オプション(SET EXACT ON/OFF) の影響を受けません。

演算フィールド内を検索する

演算フィールド内を検索するには、**検索フィールド**にその演算フィールドの名前を指定する必要があります。たとえば、Vendor_City が住所から市区町村を抽出する演算フィールドの場合は、次のように指定します。

```
FIND("New York", Vendor_City)
```

関連フィールド内を検索する

関連付けられたフィールド内を検索するには、**検索フィールド**値にフィールドの完全修飾名(つまり、<テーブル>.<フィールド名>)を指定する必要があります。

```
FIND("New York", Vendor.Vendor_City)
```

日付時刻データまたは数値データを検索する

レコードレベルでの日付時刻データまたは数値データの検索には、FIND() 関数を使用できます。日付時刻または数値の検索では、**検索フィールド**の指定はサポートされていません。

数値または日付時刻文字列は引用符で囲む必要があるとともに、ビューにおける書式ではなくソースデータの書式と完全に一致する必要があります。

FIND() 関数を使用して演算フィールドや関連フィールド内で日付時刻データや数値データを検索することはできません。

メモ

FIND() 関数は、日付時刻データや数値データをあまりうまく検索できないため、お勧めしません。

FINDMULTI() 関数

指定された1つまたは複数の文字列が、特定のフィールド中、またはレコード全体のどこかに存在するかどうかを示す論理値を返します。

構文

```
FINDMULTI({検索対象 | RECORD}, 文字列1 <, ...n>)
```

パラメーター

名前	型	説明
検索対象 RECORD	文字	検索場所となるフィールドまたは変数。 未定義部分もすべて含めたレコードの全体を検索するには、キーワード、RECORDを指定します。 また、複数のフィールド名を連結したフィールドリストを指定することもできます。 <pre>Field_1+Field_2+Field_3</pre>
文字列1 <, ...n>	文字	検索する1つまたは複数の文字列。複数の検索文字列を指定する場合は、カンマで区切ります。 <pre>FINDMULTI(RECORD, "Joa", "Jim", "Joh")</pre> 検索は大文字と小文字を区別しません。

出力

論理。指定された文字列の値のいずれかが見つかった場合はT(True)、そうでない場合はF(False)を返します。

例

基本的な例

レコード全体を検索する

次の例では、特定のフィールド、複数のフィールド、または未定義部分に "New York" または "Chicago" を含んでいるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されません。

```
FINDMULTI(RECORD, "New York", "Chicago")
```

単一のフィールド内の検索

次の例では、**City** フィールドの値が "New York" または "Chicago" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI(City, "New York", "Chicago")
```

次の例では、**City** フィールドの値が文字列 "Ne" または "Chi" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI(City, "Ne", "Chi")
```

次の例では、**City** フィールドの値が、"New York" または "Chicago" の前に1つ以上のスペースが付加された文字列であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI(City, " New York", " Chicago")
```

次の例では、*v_search_term* 変数のいずれかの値と完全一致または部分一致する値を **Description** フィールドに持つすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されません。

```
FINDMULTI(Description, v_search_term_1, v_search_term_2, v_search_term_3)
```

複数のフィールド内の検索

次の例では、**City** フィールドか **City_2** フィールドのどちらかの値が文字列 "New York" または "Chicago" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI(City+City_2, "New York", "Chicago")
```

次の例では、**City** フィールドか **City_2** フィールドのどちらかの値が文字列 "New York" または "Chicago" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI(City, "New York", "Chicago") または FINDMULTI(City_2, "New York", "Chicago")
```

他の関数との組み合わせ

次の例では、**Last_Name_1** フィールドの値が **Last_Name_2** または **Last_Name_3** フィールドの値と完全一致または部分一致するすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
FINDMULTI>Last_Name_1, ALLTRIM>Last_Name_2), ALLTRIM>Last_Name_3))
```

備考

FINDMULTI() の使用に適する場面

FINDMULTI() 関数は、指定された文字列のいずれかが特定のフィールド、2 つ以上のフィールドのいずれか、またはレコード内のいずれかのフィールドに存在するかどうかを調べる場合に使用できます。

文字列が一致する仕組み

文字列の値は、正確に一致することもあれば、長い文字列の中に含まれることもあります。フィールド内の先頭スペースは、文字列値の先頭に 1 つ以上のスペースを入れない限り、検索には影響しません。

レコード全体を検索する

検索対象フィールド パラメーターの代わりに RECORD を指定した場合には、レコードは、未定義部分もすべて含め、全体が検索されます。レコード全体が検索される場合には、フィールドの境界は無視され、フィールドの末尾のスペースは文字として扱われます。

メモ

レコード全体を検索する場合は、物理レコードが検索されます。演算フィールドおよび関連フィールドは検索されません。

フィールドのサブセットを検索する

テーブル内のフィールドのサブセット内を検索したい場合は、検索対象パラメーターの中で2つ以上のフィールドを連結することができます。たとえば、City フィールドと City_2 フィールドの両方で文字列 "New York" または "Chicago" を検索するには、次の関数を使用します。

```
FINDMULTI(City+City_2, "New York", "Chicago")
```

連結したフィールドは1つのフィールドのように扱われます。ただし、ALLTRIM() 関数を使用して個々のフィールドからスペースを除去しなければ、各フィールドの先頭と末尾のスペースを含むフィールドとなります。

また、各フィールド内を別々に検索する式を作成することもできます。

```
FINDMULTI(City, "New York", "Chicago") または FINDMULTI(City_2, "New York", "Chicago")
```

文字列の値に先頭スペースが含まれている場合、この2つの手法の検索結果は異なる場合があります。

大文字と小文字の区別と正確な文字比較

FINDMULTI() 関数は大文字と小文字を区別せず、ASCII 文字とEBCDIC 文字の両方を検索します。この関数は、**正確な文字比較を行う**]オプション(SET EXACT ON/OFF)の影響を受けません。

演算フィールド内を検索する

演算フィールド内を検索するには、検索対象にその演算フィールドの名前を指定する必要があります。たとえば、Vendor_City が住所から市区町村を抽出する演算フィールドの場合は、次のように指定します。

```
FINDMULTI(Vendor_City, "New York", "Chicago")
```

関連フィールド内を検索する

関連付けられたフィールド内を検索するには、検索対象の値としてフィールドの完全修飾名(つまり、<テーブル>.<フィールド名>)を指定する必要があります。

```
FINDMULTI(Vendor.Vendor_City, "New York", "Chicago")
```

日付時刻データまたは数値データを検索する

レコードレベルで日付時刻データまたは数値データを検索するには、FINDMULTI() 関数を RECORD とともに使用します。日付時刻または数値の検索では、検索対象フィールドの指定はサポートされていません。

数値または日付時刻の文字列値は引用符で囲む必要があり、ビューにおける書式ではなくソースデータの書式と完全に一致する必要があります。

FINDMULTI() 関数を使用して演算フィールドや関連フィールド内で日付時刻データや数値データを検索することはできません。

メモ

FINDMULTI() 関数は、日付時刻データや数値データをあまりうまく検索できないため、お勧めしません。

FREQUENCY() 関数

連続する先頭の正の数値桁について、ベンフォードの法則で期待される頻度を小数点以下 8 桁で返します。

構文

```
FREQUENCY(数字列)
```

パラメーター

名前	型	説明
数字列	文字	頻度を識別する数字 (0 ~ 9) を含んでいる文字列。数字列は正数である必要があります。先頭のゼロは無視されます。

出力

数値。

例

基本的な例

0.00998422 が返されます。

```
FREQUENCY("43")
```

0.00000000 が返されます。

```
FREQUENCY("87654321")
```

メモ

結果は0.00000000495になりますが、Analytics は計算結果を小数点以下 8 桁の精度にするため、ゼロ値が返されます。

備考

機能の仕組み

連続する先頭の正の数値桁について、ベンフォードの法則で予測される頻度を小数点以下 8 桁で返します。この結果を使用して、特定の状況に限定されたベンフォード検査を実施できます。

特定の数値の組み合わせにこの関数を使用する

特定の数字の組み合わせに着目したい場合は、BENFORD コマンドの代わりにこの関数を使用できます。たとえば、承認する請求金額が制限されている保険請求を監査する場合に、限度額をほんの少し下回るだけの請求金額について調べるためなどに FREQUENCY() 関数を使用することができます。

承認される限度額の \$5,000 に近い請求金額を調べるために、\$4,900 から \$4,999 の範囲を選択することができます。まず、レコードの総数を求めます。次に、フィルターを使用して LEADING() 関数で 49 が返されるレコードの数を求めます。最後に、2 つの件数の比率と、FREQUENCY("49") を実行して得た値とを比較します。

この方法は、百万件のレコードを持つテーブルで完全な分析を実行するよりも短時間で実行でき、テーブルやコマンド ログのサイズも抑えられます。

6 桁を超える文字列を指定する

6 桁を超える文字列を指定すると結果がゼロになる場合があります。6 桁を超える文字列の計算では、Analytics の制限である小数点以下 8 桁よりも高い精度が必要になる場合があります。

FTYPE() 関数

フィールドまたは変数のデータカテゴリや、Analytics プロジェクト項目の種類を識別する文字を返します。

構文

```
FTYPE(フィールド名文字列)
```

パラメーター

名前	型	説明
フィールド名文字列	文字	フィールド名、変数名、または Analytics プロジェクト項目の名前。 フィールド名文字列は引用符で囲みます。 <pre>FTYPE("Amount")</pre>

出力

文字。この関数はフィールド、変数、または Analytics プロジェクト項目の種類を示す以下の文字の1つを返します。

- "C" - 文字フィールド
- "N" - 数値フィールド
- "D" - 日付時刻フィールド
- "L" - 論理フィールド
- "c" - 文字変数
- "n" - 数値変数
- "d" - 日付時刻変数
- "l" - 論理変数
- "b" - Analytics スクリプト
- "y" - Analytics テーブルレイアウト
- "w" - Analytics ワークスペース
- "i" - Analytics インデックス
- "r" - Analytics レポート
- "a" - Analytics ログファイル
- "U" - 未定義

例

基本的な例

次の例では、*num* 変数に値 4 を代入した後、その種類を検査しています。

"n" が返されます。

```
ASSIGN num = 4  
FTYPE("num")
```

高度な例

フィールドのデータ型のテスト

数値型の **Amount** フィールドを必要とするスクリプト、つまりアナリティクスがあるとします。そのフィールドが適切な型であるかを調べてから、スクリプトを実行する必要があります。

次のコマンドは、**Amount** が数値フィールドの場合のみ Script_1 を実行します。

```
OPEN Invoices  
DO Script_1 IF FTYPE("Amount") = "N"
```

テーブルまたは Analytics プロジェクト項目が存在するかどうかをテストする

次のコマンドは、プロジェクトに Invoices というテーブルがある場合に限り、Script_1 を実行します。

```
DO Script_1 IF FTYPE("Invoices") <> "U"
```

FVANNUITY() 関数

一定の利率を使って計算した一連の預金額の将来価値を返します。将来価値は、毎月の預金額と複利利息の合計です。

構文

```
FVANNUITY(利率, 期間, 支払金額 <, 種類>)
```

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	支払期間の総数。
支払金額	数値	1期間あたりの支払金額。 年金期間中、支払金額は毎回同じ金額でなければなりません。
種類 省略可能	数値	支払いのタイミング: ○ 0 - 期末払い ○ 1 - 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間、支払金額を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利5%の2年間の貸付金または投資に対して月払いする場合は、利率に0.05/12、期間に2 * 12を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に0.05、期間に2を指定します。

出力

数値。結果は小数点以下2桁まで計算されます。

例

基本的な例

月次支払額

2年間、毎月月初に支払う\$1,000の将来価値(月利1%の複利)として、(\$)27243.20が返されます。

```
FVANNUITY(0.01, 12, 1000, 1)
```

同じ年金の1年後の将来価値として、(\$)12809.33が返されます。

```
FVANNUITY(0.01, 2*12, 1000, 1)
```

年次支払額

2年間、毎年年末に支払う\$12,000の将来価値(年利12%の複利)として、(\$)25440.00が返されます。

```
FVANNUITY(0.12, 2, 12000, 0)
```

高度な例

年金の計算

年金の計算では、次の4つの変数が使用されます。

- **現在価値または将来価値** - 下の例では\$21,243.39と\$26,973.46
- **1期間あたりの支払金額** - 下の例では\$1,000.00
- **1期間あたりの利率** - 下の例では月あたり1%
- **期間の数** - 下の例では24か月

これらの変数のうち3つの値がわかっている場合は、Analytics関数を使って残りの変数の値を計算できます。

求めたい値:	使用する Analytics 関数
現在価値	PVANNUIITY() 21243.39 を返す: <input type="text" value="FVANNUIITY(0.01, 12, 1000)"/>
将来価値	FVANNUIITY() 26973.46 を返す: <input type="text" value="FVANNUIITY(0.01, 24, 1000)"/>
1 期間あたりの支払金額	PMT() 1000 を返す: <input type="text" value="PMT(0.01, 24, 21243.39)"/>
1 期間あたりの利率	RATE() 0.00999999(1%) を返す: <input type="text" value="RATE(24, 1000, 21243.39)"/>
期間の数	NPER() 24.00 を返す: <input type="text" value="NPER(0.01, 1000, 21243.39)"/>

年金の式

期末年金(期末払い)の現在価値を計算する式:

$$PV_A = Pmt \left[\frac{1 - \frac{1}{(1+i)^N}}{i} \right]$$

$$21243.39 = 1000 \left[\frac{1 - \frac{1}{(1+0.01)^{24}}}{0.01} \right]$$

期末年金(期末払い)の将来価値を計算する式:

$$FV_A = Pmt \left[\frac{(1+i)^N - 1}{i} \right]$$
$$26973.46 = 1000 \left[\frac{(1+0.01)^{24} - 1}{0.01} \right]$$

備考

関連する関数

PVANNUIITY() 関数は FVANNUIITY() 関数の逆関数です。

FVLUMPSUM() 関数

一定の利率を使って計算した現在の総額の将来価値を返します。

構文

```
FVLUMPSUM( 利率, 期間, 金額)
```

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	期間の総数。
金額	数値	最初の期間の冒頭に支払う投資(出資)額。

メモ

利率、期間を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。結果は小数点以下 2 桁まで計算されます。

例

基本的な例

月ごとに複利が付く場合

総額 \$1,000を月利 1% の複利で 2 年間運用した場合の将来価値として、(\$) 1269.73 が返されます。

```
FVLUMPSUM(0.01, 2*12, 1000)
```

同じ投資の 1 年後の将来価値として、(\$) 1126.83 が返されます。

```
FVLUMPSUM(0.01, 12, 1000)
```

月利 1% の複利で 2 年間運用した場合の \$21,455.82 の将来価値として、(\$) 27243.20 が返されます。

```
FVLUMPSUM(0.01, 2*12, 1000)
```

半年ごとに複利が付く場合

総額 \$1,000を年利 12% の年複利で 2 年間運用した場合の将来価値として、(\$) 1262.48 が返されます。

```
FVLUMPSUM(0.12/2, 2*2, 1000)
```

1 年ごとに複利が付く場合

総額 \$1,000を年利 12% の年複利で 2 年間運用した場合の将来価値として、(\$) 1254.408 が返されます。

```
FVLUMPSUM(0.12, 2, 1000)
```

備考

将来価値とは

運用する総額の将来価値は、初期投資元本と複利の合計です。

関連する関数

PVLUMPSUM() 関数は FVLUMPSUM() 関数の逆関数です。

FVSCHEDULE() 関数

一連の利率を使って計算した現在の総額の将来価値を返します。

構文

```
FVSCHEDULE(元金, 利率1 <, 利率2...>)
```

パラメーター

名前	型	説明
元金	数値	初期投資額。
利率1, 利率2...	数値	同じ長さを持つ複数の期間における一連の利率。 メモ タイプが同じ複数の期間により、複数の月、年、またはその他の期間を表すことができます。 1 期間あたりの利率 を指定する必要があります。したがって、利率の1つが年 5% で、期間が月の場合は、0.05/12 を指定します。

出力

数値。結果は小数点以下 2 桁まで計算されます。

例

基本的な例

\$1000 の総額を 1 年目に 10%、2 年目に 9%、3 年目に 7% の年複利で運用した場合の将来価値として、(\$) 1282.93 が返されます。

```
FVSCHEDULE(1000, 0.1, 0.09, 0.07)
```

関数

備考

運用する総額の将来価値は、初期投資元本と複利の合計です。

GETOPTIONS() 関数

指定された Analytics オプションの現在の設定 (オプション] ダイアログ ボックスの設定) を返します。

構文

```
GETOPTIONS(オプション)
```

パラメーター

名前	型	説明
オプション	文字	<p>設定を返す対象となる Analytics オプション。</p> <p>オプションの名前は、以下の一覧に示されるとおり正確に指定する必要があります。また引用符で囲む必要があります。</p> <ul style="list-style-type: none">◦ separators - Analytics の 3 つの区切り文字の現在の設定を次の順序で返します。<ul style="list-style-type: none">• 小数点の記号• 桁区切り記号• リストの区切り文字 <p>メモ</p> <p>現時点では、"separators" が、GETOPTIONS() 関数で指定できる唯一のオプションです。</p>

出力

文字。

例

基本的な例

3 つの Analytics 区切り文字の現在の設定が返されます。例: ".,,"

```
GETOPTIONS("separators")
```

高度な例

スクリプト内での GETOPTIONS() の使用

スクリプトで Analytics の区切り文字のうち 1 つ以上を変更する必要がある場合、GETOPTIONS() 関数により現在の設定を検出することができます。現在の設定を変数に格納しておき、スクリプトの最後で元に戻すことができます。

```
ASSIGN v_SeparatorsSetting = GETOPTIONS("separators")
SET SEPARATORS ",.;"
<スクリプト内容>
SET SEPARATORS "%v_SeparatorsSetting%"
```

備考

3 つの Analytics 区切り文字が、**オプション** ダイアログボックスにある以下のオプションとして指定されています。

- 小数点の記号
- 桁区切り記号
- リストの区切り文字

GOMONTH() 関数

指定された日付から、指定された月数前または月数後の日付を返します。

構文

```
GOMONTH(日付/日付時刻, 月数)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	出力日付を求めるフィールド、式、またはリテラル値。
月数	数値	日付/日付時刻の前または後にある、月を示す数字。 メモ 日付/日付時刻には日付時刻値を指定できますが、値の時刻部分は無視されます。

出力

日付時刻。日付値は、現在 Analytics に設定されている日付の表示書式を使用して出力されます。

例

基本的な例

リテラルの入力値

現在の Analytics 日付表示書式 DD MMM YYYY の形式を `20140415` に適用した値、15 Apr 2014 が返されます。

```
GOMONTH(`20140415`, 3)
```

関数

`20131015` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、15 Oct 2013 が返されます。

```
GOMONTH(`20140115`, -3)
```

`20140430` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、30 Apr 2014 が返されます(無効な日付である 2014 年 4 月 31 日が返されないように日付が丸められます)。

```
GOMONTH(`20140330`, 1)
```

```
GOMONTH(`20140331`, 1)
```

`20140501` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、01 May 2014 が返されます。

```
GOMONTH(`20140331`, 1)
```

フィールドの入力値

Invoice_date フィールドの各日付の 3 か月後の日付が返されます。

```
GOMONTH(Invoice_date, 3)
```

Invoice_date フィールドの各日付に猶予期間 15 日を加えた日付から 3 か月後の日付が返されます。

```
GOMONTH(Invoice_date + 15, 3)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

日付時刻書式

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付値には、次のいずれかの書式を使用する必要があります。

- YYYYMMDD
- YYMMDD

リテラル日付値はバッククォートで囲む必要があります。例: ``20141231``

月数の値の動作

- **値が正の場合** - 出力日付は指定された日付/日付時刻より新しい日付になります。
- **値が負の場合** - 出力日付は指定した日付/日付時刻より前の日付になります。
- **値を省略した場合、または'0'(ゼロ)を使用した場合** - 出力日付は日付/日付時刻と同じになります。

実在しない日付にしないうための日付の丸め処理

日付/日付時刻と月数の組み合わせが実在しない日付を作り出す場合は、GOMONTH() 関数は「日付の丸め」を使用して、同月内の最も近い有効な日付を返します。

次の例の場合、31 Apr 2014 は無効な日付であるため、`20140430` (30 Apr 2014) が返されます。

```
GOMONTH(`20140331`,1)
```

関連する関数

指定された日付から、指定された月数だけ前または後に当たる月の、正確な日付ではなく末日の日付を返したい場合は、EOMONTH() 関数を使用してください。

HASH() 関数

入力値に基づいてソルト付き暗号化ハッシュ値を返します。

構文

```
HASH(フィールド <,ソルト値>)
```

パラメーター

名前	型	説明
フィールド	文字 数値 日付時刻 論理	ハッシュする値。
ソルト値 省略可能	文字 数値	使用するソルト値。1から10までの <code>PASSWORD</code> 識別番号か、または文字列を指定できます。 これを省略した場合は、Analyticsのデフォルトのソルト値が使用されます。 ソルト値は128文字に制限されています。それより長いソルト値を指定した場合は、自動的に128文字に切り詰められます。 詳細については、「ソルト値」ページ 2299を参照してください。

出力

文字。

例

基本的な例

Analytics のデフォルトのソルト値を使用する場合

"819A974BB91215D58E7753FD5A42226150100A0763087CA7DECD93F3C3090405" が返されます。

```
HASH("555-44-3322")
```

`Credit_card_num` フィールド内の各番号のハッシュ値が返されます。

```
HASH(Credit_card_num)
```

ユーザー指定のソルト値を使用する場合

"AD1E7D9B97B6F6B5345AB13471A74C31EBE6630CA2622BB7E8C280E9FBEE1F17" が返されます。

```
HASH("555443322", "マイ ソルト値 123")
```

高度な例

同一のハッシュ値が確実に生成されるようにする

クリアテキスト値を統一して同一のハッシュ値を生成するために、`HASH()` とともにその他の関数を使用します。

次の一連の例について考えてください。最初の2つの例では、クリアテキスト値の大文字と小文字の違いによって出力のハッシュ値がまったく変更されるかどうか注目してください。

"DF6789E1EC65055CD9CA17DD5B0BEA5892504DFE7661D258737AF7CB9DC46462" が返されます。

```
HASH("John Smith")
```

"3E12EABB5940B7A2AD90A6B0710237B935FAB68E629907927A65B3AA7BE6781D" が返されます。

```
HASH("John Smith")
```

次の例では、UPPER() 関数を使って大文字と小文字の違いを統一することで、同一のハッシュ値が生成されます。

"3E12EABB5940B7A2AD90A6B0710237B935FAB68E629907927A65B3AA7BE6781D" が返されます。

```
HASH(UPPER("John Smith"))
```

HASH() を使用した大きなテキストのブロック同士の比較

HASH() 関数を使用して、2つのコメントフィールド内のテキストのブロックが同一であるかどうかをテストすることができます。

このテストを実行するには、以下と同様の2つの演算フィールドを作成してから、同一でないテキストブロックを見つけるためのフィルターを作成します。

```
DEFINE FIELD Hash_1 COMPUTED HASH(Comment_Field_1)
DEFINE FIELD Hash_2 COMPUTED HASH(Comment_Field_2)
SET FILTER TO Hash_1 <> Hash_2
```

コメントフィールドが別々のテーブルにある場合は、各テーブルにHASH() 演算フィールドを作成し、その演算フィールドを共通のキーフィールドとして使用してそれら2つのテーブルを不一致結合します。結合された出力テーブル内のレコードは、同一でないテキストブロックを表します。

備考

HASH() の使用に適する場面

HASH() 関数は、クレジットカード番号、給与情報、および社会保障番号などの機密データを保護する場合に使用できます。

機能の仕組み

HASH() では一方向のエンコードが行われます。クリアテキストのデータを使用してハッシュ値を生成することはできますが、その後、そのハッシュ値のエンコードを解除したり、ハッシュ値を復号したりすることはできません。

特定のクリアテキスト値は常に同じハッシュ値を生成するため、ハッシュされたクレジットカード番号のフィールドで重複を検索したり、ハッシュされたクレジットカード番号の2つのフィールドを結合したりすることができます。この操作を同等のクリアテキストフィールドで実行した場合、結果は同じになります。

機密データの保護

機密データをサーバー上に保存することを避けるために、ローカルでHASH() 関数を使用して演算フィールドを作成することができます。ハッシュされたフィールドとその他必要なフィールドを抽出すると同時に、クリアテキストフィールドを除外して、新しいテーブルを作成します。分析でサーバー上の新しいテーブルを使用して、一度その結果を得たら、ハッシュされたデータのクリアテキストバージョンを確認する必要がある場合には、元のテーブルを参照することができます。

初期使用以外で、機密データをローカルに保存することが禁止されている場合は、ハッシュ値を使って新しいテーブルを作成した後、元のテーブルを削除してもかまいません。削除しても、クリアテキスト値について元のソースシステムを参照することができます。

クリアテキスト値は同一である必要あり

同一のハッシュ値を生成するためには、2つのクリアテキスト値はまったく同じでなければなりません。たとえば、同じクレジットカード番号でもハイフンがあるかどうかによって、また、同じ名前でも先頭文字が大文字かすべて大文字かどうかによって、異なるハッシュ値になります。

クリアテキスト値を統一するために、INCLUDE()、EXCLUDE()、またはUPPER()などの関数をHASH() 関数に組み込む必要があるかもしれません。

先頭と末尾のスペースはHASH() 関数によって自動的に除去されるため、TRIM() 関数やALLTRIM() 関数を使用する必要はありません。

先頭や末尾にあるスペースに意味がある場合の処理

値の先頭や末尾にあるスペースで、値間の違いを表しているデータがある場合は、スペースを別の文字に置き換えてから、値のハッシュ値を求める必要があります。

ハッシュ処理を行う前に、フィールド値内のスペースをアンダースコア文字(`_`)に置き換えます。

```
HASH(REPLACE(フィールド名, " ", "_"))
```

HASH() で使用される暗号化アルゴリズム

HASH() は、SHA-2 暗号化ハッシュアルゴリズムを使用します。このアルゴリズムでは、入力値の長さに関係なく、64 バイトの固定長のハッシュされた出力が生成されます。クリアテキストの入力値は 64 バイトより長くてもかまいません。

ソルト値

機能の仕組み

HASH() 関数によって提供される保護は、ハッシュする前にソルト値を自動的に追加することによって強化されます。ソルト値は、ソースデータ値と連結される英数字の文字列です。連結された文字列全体を使用して、ソルト付きのハッシュされた値が生成されます。この手法は、ハッシュされた値の復号技術に対する耐性を強めます。

任意で独自のソルト値を指定する

ソルト値を指定しない限り、固定のデフォルトのソルト値が使用されます。次のいずれかの方法を使用して、ソルト値を指定できます。

- **クリアテキスト文字列としてのソルト値**

英数字の文字列を指定します。例：

```
HASH(Credit_card_num, "my salt value")
```

- **パスワードとしてのソルト値**

HASH() 関数と併せて PASSWORD コマンドを使用し、1 から 10 までの PASSWORD 識別番号を指定します。例：

```
PASSWORD 3 "ソルト値を入力"  
EXTRACT FIELDS HASH(Credit_card_num, 3) TO "Protected_table"
```

メモ

HASH() 関数のフィールドを抽出するには、その前に PASSWORD のソルト値を入力しておく必要があります。

HASH() とともに PASSWORD 識別番号を使用することの利点は、クリアテキストのソルト値をあらわにしないで済むことです。

詳細については、「PASSWORD コマンド」ページ 2015を参照してください。

パスワード方式のガイドライン

このパスワード方式は、スクリプトの先頭、あるいはスクリプト内で HASH() 関数が現れる前に、パスワードの入力を求めるスクリプトで使用することを目的としています。

PASSWORD の割り当ては Analytics を閉じるときに削除されるため、パスワード方式は演算フィールドでの使用には適していません。

さらに、パスワードベースのソルト値を使用している演算フィールドは、Analytics を再度開くときにビューから自動的に削除されます。この削除は、デフォルトのソルト値を使用したハッシュ値の再計算を避けるために必要です。再計算される値は、ユーザー指定したソルト値を使って計算された元のハッシュ値とは異なることとなります。

HEX() 関数

ASCII 文字列を 16 進数の文字列に変換します。

構文

```
HEX(フィールド)
```

パラメーター

名前	型	説明
フィールド	文字	16 進数の文字列に変換する ASCII 文字列。

出力

文字。

例

基本的な例

"3132333435" が返されます。

```
HEX("12345")
```

Count フィールドの値に対する 16 進数の文字列が返されます。

```
HEX(Count)
```

備考

機能の仕組み

この関数は、指定されたフィールド値または式と等価な 16 進数の文字列を返します。CR(キャリッジリターン)、LF(ラインフィード)、NUL(ヌル) など画面に表示できない文字を含んでいるフィールドの正確な内容を確認する必要がある場合に、この関数を使用できます。

戻り値の長さ

戻り値は、フィールド値の 2 倍の長さを持つ文字列になります。0 から 9 までの数字と A から F までの文字(10 から 15 までの数字に相当) によって 16 進数の値を表します。

式でなくフィールドを入力値として使用する

一般に、この関数は式ではなくフィールドに適用します。式に適用すると、式の内部記憶形式が表示され、ほとんどの場合意味がないためです。

HOUR() 関数

指定された時刻または日付時刻から時間を抽出し、それを 24 時間制の数値として返します。

構文

```
HOUR(時刻/日付時刻)
```

パラメーター

名前	型	説明
時刻/日付時刻	日付時刻	時間を抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

23 が返されます。

```
HOUR(`t235959`)
```

```
HOUR(`20141231 235959`)
```

Call_start_time フィールドの各値の時間数が返されます。

```
HOUR(Call_start_time)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

時刻/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような時刻書式または日付時刻書式でも使用することができます。

リテラル時刻または日付時刻値の指定

日付時刻にリテラルの時刻値または日付時刻値を指定する場合は、次の表内の書式に制限されません。また、``20141231 235959`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **時刻値** - 以下の表に示す任意の時刻の書式を使用することができます。関数が正しく動作するためには、単独の時刻値の前に区切り文字を使用する必要があります。有効な区切り文字は文字 't' または 'T' です。24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。
- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2 つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 't' または 'T' です。

形式の例	リテラル値の例
thhmmss	`t235959`
Thhmm	`T2359`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`

関数

形式の例	リテラル値の例
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

HTOU() 関数

16 進数の文字列を Unicode 文字列に変換します。"Hexadecimal to Unicode" の省略形です。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

構文

```
HTOU(16進数文字列)
```

パラメーター

名前	型	説明
16 進数文字列	文字	Unicode 文字列に変換する 16 進数の文字列。この文字列には "20AC" などの 16 進値のみを指定できます。

出力

文字。

例

基本的な例

"ABC123" が返されます。

```
HTOU("004100420043003100320033")
```

高度な例

金額に通貨記号を付加する

金額単位フィールドを新しいテーブルに抽出する必要があるとします。このフィールドには、ユーロ通貨記号(€)と、数値型の **Amount** フィールドの値が表示されます。

```
EXTRACT HTOU("20AC") + STRING(Amount, 10) AS "Currency_Amount" TO  
Display_Table
```

この EXTRACT コマンドを実行すると、HTOU() により、ユーロ記号 "€" と、STRING() によって文字に変換された **Amount** の値が併せて表示されます。**Amount** の元の値が 2000 の場合は、**Currency_Amount** の値は "€2000" となります。

備考

関連する関数

HTOU() は、Unicode 文字列を 16 進数文字列に変換する DHEX() 関数の逆関数です。

INCLUDE() 関数

指定した文字のみを含む文字列を返します。

構文

```
INCLUDE(文字列, 含める文字)
```

パラメーター

名前	型	説明
文字列	文字	含める文字に制限するフィールド、式、リテラル値。
含める文字	文字	含める文字のリスト。 含める文字の中に二重引用符を指定する場合は、文字のリスト全体を一重引用符で囲む必要があります。 例: <code>'"/'</code> メモ 含める文字のリストに指定した文字が文字列にない場合には、その文字は戻り値に含まれません。

出力

文字。

例

基本的な例

入力文字列の中から数字のみを選択した "123" が返されます。

```
INCLUDE("123 Main St.", "0123456789")
```

関数

入力文字列の中から数字のみを選択した "1231234" が返されます。

```
INCLUDE("123-123-4", "1243")
```

入力文字列には "D" が含まれていないため、"" が返されます(つまり、何も返されません)。

```
INCLUDE("ABC", "D")
```

備考

機能の仕組み

INCLUDE() 関数は、文字列内の各文字を、含める文字に列挙された文字と比較します。一致した文字は出力文字列に追加されます。

一致する文字がない場合

文字列と含める文字の間に一致する文字がない場合、関数の出力は空になります。

大文字と小文字の区別

INCLUDE() 関数では大文字と小文字が区別されます。このため、含める文字に "ID" を指定しても、これらの文字は "id#94022" には含まれていないことになります。大文字と小文字の両方が混在している可能性がある場合は、UPPER() 関数を使用して文字列を大文字に変換します。

例：

```
INCLUDE(UPPER("id#94022"), "ID0123456789")
```

使用上のヒント

含める文字のセットが小さく、取り除く文字のセットが大きい場合は、INCLUDE() を使用します。

関連する関数

INCLUDE() 関数は EXCLUDE() 関数の逆関数です。

INSERT() 関数

元の文字列の指定のバイト位置に、指定した文字列が挿入された内容が返されます。

構文

```
INSERT(文字列, 挿入するテキスト, 位置)
```

パラメーター

名前	型	説明
文字列	文字	テキストを挿入するフィールド、式、またはリテラル値。
挿入するテキスト	文字	挿入するテキスト。
位置	数値	挿入テキストを文字列に挿入する文字の位置。

出力

文字。

例

基本的な例

"aXXXbcde" が返されます。

```
INSERT("abcde", "XXX", 2)
```

"XXXabcde" が返されます。

```
INSERT("abcde", "XXX", 0)
```

"abcde" の長さが5バイトのため、バイト位置 8 でなく6に"XXX"を挿入した"abcdeXXX"が返されません。

```
INSERT("abcde", "XXX", 8)
```

備考

機能の仕組み

INSERT() 関数は、指定された文字またはスペースを文字列の指定された桁に挿入します。

INSERT() の使用に適する場面

INSERT() は、データを正規化する場合に使用できます。データの正規化は、書式を設定する場合や、重複するデータを突き合わせる場合、また、フィールド同士が一致している必要のある JOIN および DEFINE RELATION コマンドを使用する場合などに行います。

たとえば、パーツ番号フィールドが、あるファイルでは"12345"という形式になっていて、別のファイルでは"12-345"という形式になっているとします。最初のファイルでは、INSERT() 関数を使って、3 桁目にハイフン(-)を挿入できます。

位置についてのガイドライン

- 位置の値が文字列の長さより大きい場合、挿入するテキストの値は文字列の最後に挿入されません。
- 位置が0または1の場合は、挿入するテキストは文字列の先頭に挿入されます。

二重引用符を挿入する

含める文字の中に二重引用符を指定する場合は、二重引用符を一重引用符で囲む必要があります。

例: `'''`

INT() 関数

数式またはフィールド値の整数値を返します。

構文

```
INT(数値)
```

パラメーター

名前	型	説明
数値	数値	整数に変換するフィールドまたは数式。指定された値が小数を含んでいる場合、小数点以下は四捨五入されないで切り捨てられます。

出力

数値。

例

基本的な例

7 が返されます。

```
INT(7.9)
```

-7 が返されます。

```
INT(-7.9)
```

IPMT() 関数

単一の期間で貸付金に対して支払われた利息を返します。

構文

IPMT(利率, 指定期間, 期間, 金額 <, 種類>)

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
指定期間	数値	支払利息を確認する期間。
期間	数値	支払期間の総数。
金額	数値	貸付金の元金。
種類 省略可能	数値	支払いのタイミング: ○ 0- 期末払い ○ 1- 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のあ
る期間を使用する必要があります。

例:

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。

例

基本的な例

\$275,000 の貸付金を年利 6.5 パーセントで 25 年間にわたって返済する場合 (支払期日は月末です) の、1 年目に支払う利息、(\$) 1489.58 が返されます。

```
IPMT(0.065/12, 1, 12*25, 275000, 0)
```

貸付金の最終月において支払う利息、(\$) 10.00 が返されます。

```
IPMT(0.065/12, 300, 12*25, 275000, 0)
```

備考

関連する関数

PPMT() 関数は IPMT() 関数に対して補完的役割を果たします。

CUMIPMT() 関数は一連の期間にわたって支払われた利息を計算します。

ISBLANK() 関数

入力値が空白かどうかを示す論理値を返します。

構文

```
ISBLANK(文字列)
```

パラメーター

名前	型	説明
文字列	文字	空白データを調べる値。

出力

論理。文字列パラメーターの値が空白の場合はT(true)、そうでない場合はF(false)を返します。

例

基本的な例

Fが返されます。

```
ISBLANK(" A")
```

Tが返されます。

```
ISBLANK(" ")
```

```
ISBLANK("")
```

Address フィールド内のすべての値が空白の場合は 'T'、そうでない場合は 'F' が返されます。

```
ISBLANK(Address)
```

備考

ISBLANK() の使用に適する場面

ISBLANK() は、データが欠落したフィールドを識別するために、分析プロジェクトのデータの整合性フェーズで使用することができます。このようなフィールドは、ソース データの問題を示している可能性があります。

空白の入力値とは

この関数が true と評価されるには、入力値が次のいずれかである必要があります。

- 完全に空白 (つまり、スペースのみが含まれる)
- 長さがゼロの文字列

ただし、この関数ではソース データ内にある真の空白が識別されるだけで、ビュー内で空白として表示される無効な文字は識別されません。

NULL 文字

ISBLANK() では、NULL 文字を値として持つ文字フィールドを使用した場合、有効な結果が返されない可能性があります。Analytics では、文字列の終端を示すために NULL 文字が使用されています。このため、ISBLANK() 関数は NULL 文字があった場合、それ以降の文字を空白も含めてすべて読み取りません。

ISDEFINED() 関数

指定されたフィールドが定義されている場合はT(true)、そうでない場合はF(false)を返します。

構文

```
ISDEFINED(文字列)
```

パラメーター

名前	型	説明
文字列	文字	存在を確認するフィールドまたは変数の名前。値は、引用符で囲んだ文字列として入力する必要があります。 <pre>ISDEFINED("v_numeric_limit")</pre>

出力

論理。

例

基本的な例

`v_numeric_limit` が変数またはフィールドとして定義されている場合はT、それ以外の場合はFが返されます。

```
ISDEFINED("v_numeric_limit")
```

高度な例

ISDEFINED() を使ってフィールドをテストする

次の例では、**Limit** フィールドの値に基づいてレコードを抽出する前に、このフィールドがテーブルに定義されているかどうかを ISDEFINED() 関数を使ってテストしています。

```
OPEN Metaphor_Employees_US  
IF ISDEFINED("Limit") EXTRACT RECORD IF Limit > 50000 TO  
"HighLimit.fil"
```

ISFUZZYDUP() 関数

文字列が、比較文字列のあいまい重複であるかどうかを示す論理値を返します。

構文

```
ISFUZZYDUP(文字列1, 文字列2, レーベンシュタイン距離 <, 相違のパーセント>)
```

パラメーター

名前	型	説明
文字列1	文字	比較の最初の文字列。
文字列2	文字	比較の2番目の文字列。
レーベンシュタイン距離	数値	2つの文字列があいまい重複と認定されるために、それらの文字列間で許容される最大のレーベンシュタイン距離。 レーベンシュタイン距離の値は1未満にすることや、10を超えることはできません。 レーベンシュタイン距離の値を大きくすると、あいまい度が高い値(相互の関連性が低い値)が含まれるため、結果の件数が多くなります。
相違のパーセント 省略可能	数値	'相違のパーセント'(相違度)の上限。 相違のパーセントについては、"機能の仕組み" ページ 2322を参照してください。 相違のパーセントの値は1未満にすることや、99を超えることはできません。 相違のパーセントの値を大きくすると、長さに比べて相違の割合が高い値が含まれるようになるため、結果の件数が多くなります。 このパラメーターを省略した場合は、ISFUZZYDUP() 関数の処理時に相違のパーセントが考慮されません。

出力

論理。文字列の値があいまい重複である場合はT(true)、そうでない場合はF(false)が返されます。

例

基本的な例

"Smith" を "Smythe" に変えるには 2 回の編集が必要ですが、レーベンシュタイン距離値が 1 にすぎないため、F が返されます。

```
ISFUZZYDUP("Smith","Smythe", 1, 99)
```

"Smith" を "Smythe" に変えるには 2 回の編集が必要ですが、レーベンシュタイン距離値が 2 のため、T が返されます。

```
ISFUZZYDUP("Smith","Smythe", 2, 99)
```

"SMITH" を "smith" に変えるには編集が不要 (ISFUZZYDUP() では大文字と小文字が区別されない) であり、レーベンシュタイン距離値が 1 のため、T が返されます。

```
ISFUZZYDUP("SMITH","smith", 1, 99)
```

Last_name フィールド内の個々の値が、文字列 "Smith" のあいまい重複であるかどうかを示す論理値 (T または F) が返されます。

```
ISFUZZYDUP>Last_Name,"Smith", 3, 99)
```

高度な例

相違のパーセントの使用方法

相違のパーセントは、ISFUZZYDUP() で返される誤検出の数を減らすための手段になります。

相違のパーセントを指定しなかった場合

"abc" を "Smith" に変えるには 5 回の編集が必要ですが、レーベンシュタイン距離値が 5 のため、T が返されます。

```
ISFUZZYDUP("abc", "Smith", 5)
```

相違のパーセントを指定した場合

"abc" が "Smith" から指定したレーベンシュタイン距離内にあっても、長さ3の文字列に対して5回の編集を行うと相違のパーセントが167%になり、指定した相違のパーセントである99%を超えるため、Fが返されます。

```
ISFUZZYDUP("abc", "Smith", 5, 99)
```

相違のパーセントの詳細については、「機能の仕組み」見開きページを参照してください。

"Smith" に対するあいまい重複の抽出

Last_name フィールド内から "Smith" に対するあいまい重複であるすべての値を抽出するフィルターを作成するには、次のように指定します。

```
SET FILTER TO ISFUZZYDUP>Last_Name, "Smith", 3, 99)
```

レーベンシュタイン距離または相違のパーセントを変更すると、フィルターされる値の相違の量を調整することができます。

業者名に対するあいまい重複の抽出

Vendor_Name フィールド内から "Pacific Lighting and Electrical Supply, Inc." に対するあいまい重複であるすべての値を抽出するフィルターを作成するには、次のように指定します。

```
SET FILTER TO ISFUZZYDUP(Vendor_Name, "Pacific Lighting and Electrical  
Supply, Inc.", 2, 99)
```

ISFUZZYDUP() 関数と追加の関数を使用して、フィルターの効果を改善します。

ISFUZZYDUP() と OMIT() を使用すると、次の結果が返されます。

- Pacific Lighting and Electrical Supply, Inc.
- Pacific Lighting and Electrical Supply
- Pacific Lighting & Electrical Supply, Inc.

```
SET FILTER TO ISFUZZYDUP(OMIT(Vendor_Name, ".,&,and,Inc,Ltd,"),
"Pacific Lighting Electrical Supply", 2, 99)
```

ISFUZZYDUP() と OMIT() と UPPER() を使用すると、次の結果が返されます。

- Pacific Lighting and Electrical Supply, Inc.
- Pacific Electrical and Lighting Supply, Inc.

```
SET FILTER TO ISFUZZYDUP(SORTWORDS(UPPER(Vendor_Name)), SORTWORDS
(UPPER("Pacific Lighting and Electrical Supply, Inc.")), 2, 99)
```

ISFUZZYDUP() と SORTWORDS()、OMIT()、UPPER() を使用すると、次の結果が返されま
す。

- Pacific Lighting and Electrical Supply, Inc.
- Pacific Lighting and Electrical Supply
- Pacific Lighting & Electrical Supply, Inc.
- Pacific Electrical and Lighting Supply, Inc.

```
SET FILTER TO ISFUZZYDUP(SORTWORDS(UPPER(OMIT(Vendor_Name,
".,&,and,Inc,Ltd,"))), SORTWORDS(UPPER("Pacific Lighting Electrical
Supply")), 2, 99)
```

備考

ISFUZZYDUP() の使用に適する場面

ISFUZZYDUP() 関数は、ほぼ同一の値(あいまい重複)の検出や、手作業で入力されたデータで一貫性のないつづりを見つける場合に使用できます。

機能の仕組み

ISFUZZYDUP() 関数は、2つの文字列間のレーベンシュタイン距離を計算し、相違のパーセントを求めます。

ISFUZZYDUP() は次の場合に T(true) と評価されます。

- 実際のレーベンシュタイン距離が、指定したレーベンシュタイン距離の値以下の場合
- 実際の相違のパーセントが、相違のパーセントの値(指定されている場合)以下の場合

レーベンシュタイン距離

レーベンシュタイン距離は、ある文字列を別の文字列にするために必要な、1文字の編集の最小回数を示す値です。

詳細については、"LEVDIST() 関数" ページ 2335を参照してください。

相違のパーセント

相違のパーセントは、異なる2つの評価対象文字列のうちの短い方の文字列に対する長い方の文字列のパーセントです。

相違のパーセントは、2つの文字列間のレーベンシュタイン距離を使用する、次のような Analytics での内部計算の結果です。

$\text{レーベンシュタイン距離} / \text{短い文字列内の文字数} \times 100 = \text{相違のパーセント}$

任意である相違のパーセントを使用することで、ISFUZZYDUP() で返される誤検出の数を減らすことができます。

- 相違のパーセントの上限は99%です。99%では、文字列の置き換えがすべて禁止されるため、文字列が同一になります。
- 長さに比べて多数回の編集が必要な文字列は除外されます。

使用上のヒント

- **大文字と小文字の区別** - この関数では大文字と小文字が区別されないため、"SMITH" は "smith" と同じであると判断されます。
- **末尾にあるスペース** - フィールド内の末尾にあるスペースはこの関数により自動的に除去されるため、パラメーターとしてフィールドを指定するときにTRIM() 関数を使う必要はありません。
- **要素の並べ替え** - SORTWORDS() 関数は、フィールド値の個別の要素を連続する順序に並べ替えることで、ISFUZZYDUP() 関数の効果を高めることができます。

アドレスの構成要素などの要素を並べ替えると、同じ情報を含む、形式が異なり、相互により密接に類似した2つの文字列を作成できます。より密接な類似性により、文字列の組み合わせが相互のあいまい重複として選択される確率が上がります。

- **一般的要素の除去** - OMIT() と EXCLUDE() 関数は、フィールド値から "Corporation" や "Inc."、カンマ、ピリオド、アンパサンド (&) 文字などの一般的要素を除去することによって、ISFUZZYDUP() 関数の有効性を高めることができます。

一般的要素および句読点の除去により、ISFUZZYDUP() の文字列比較は、意味のある違いが発生する可能性のある文字列の部分だけに集中されます。

FUZZYDUP コマンドと ISFUZZYDUP() 関数の相違点

FUZZYDUP コマンドでは、フィールド内のすべてのあいまい重複を特定し、それらを完全ではないグループにまとめて、全体的に網羅的である結果を出力します。

これに対し、ISFUZZYDUP() 関数は、特定の文字値に対するあいまい重複と完全重複を含む、単一かつ網羅的な一覧を生成します。

FUZZYDUP コマンドと ISFUZZYDUP() 関数はともに完全重複を特定します。しかし、ISFUZZYDUP() 関数を使用する場合には、FUZZYDUP コマンドとは違って完全重複を除外することができません。

「網羅的」とは

「網羅的」とは、検査値から指定された相違の度合いの範囲内であれば、検査値に関連する検査フィールド内の位置に関係なく、すべての値が返されることを意味します。

ISFUZZYDUP() 関数は、FUZZYDUP コマンドによって生成された非網羅的なグループでは分析目的に適していない場合や、特定の文字値のあらゆるあいまい重複について直接綿密に調べる必要がある場合に役に立ちます。

関連する関数

- **LEVDIST()** - は、レーベンシュタイン距離に基づいて文字列を比較するためのもう1つの手段です。
ISFUZZYDUP() とは異なり、LEVDIST() では既定で大文字と小文字が区別されます。
- **DICECOEFFICIENT()** - は、文字列を比較するときに、文字または文字ブロックの相対位置を重視しないか、または完全に無視します。
- **SOUNDSLIKE()** および **SOUNDEX()** - は、正字法の比較(綴り)ではなく、発音記号の比較(発音)に基づいて文字列を比較します。

LAST() 関数

文字列の末尾から指定された数の文字を返します。

構文

```
LAST(文字列, 長さ)
```

パラメーター

名前	型	説明
文字列	文字	文字を返すフィールド、式、またはリテラル値。
長さ	数値	返させる文字数。

出力

文字。

例

基本的な例

"Savings" が返されます。

```
LAST("Account Type: Savings", 7)
```

"efghi" が返されます。

```
LAST("abcdefghi", 5)
```

"fghi" が返されます。

```
LAST("abcdefghi ", 5)
```

文字列が指定された長さである6より短いので、短い分のスペースが出力の先頭に追加されるため、" abc" が返されます。

```
LAST("abc", 6)
```

備考

末尾のスペースによって結果が空白になる場合

文字列の末尾にスペースがあるために、LAST() 関数によって生成される結果が空白になる場合があります。

たとえば、LAST("6483-30384 ", 3) の出力は" "になります。

ALLTRIM() 関数とLAST() を組み合わせて使用して文字列内の末尾のスペースをすべて取り除くことができます。

たとえば、LAST(ALLTRIM("6483-30384 "), 3) では"384" が返されます。

文字列の先頭にある文字を返す

文字列の先頭にある指定数の文字を返すには、SUBSTR() 関数を使用します。詳細については、"SUBSTR() 関数" ページ 2531を参照してください。

LEADING() 関数

指定された数の先頭桁を含んでいる文字列を返します。

構文

```
LEADING(数値, 先頭桁数)
```

パラメーター

名前	型	説明
数値	数値	先頭桁を返す値。
先頭桁数	数値	返される先頭桁数。

出力

文字。

例

基本的な例

リテラル数値の入力値

623 が返されます。

```
LEADING(6234.56, 3)
```

62345 が返されます。

```
LEADING(6234.56, 3)
```

末尾にゼロを追加する

000 が返されます。

```
LEADING(0.00, 3)
```

00000 が返されます。

```
LEADING(0.00, 5)
```

35500 が返されます。

```
LEADING(3.55, 5)
```

備考

LEADING() を使用すると、数値フィールドから数字を文字列として抽出し、小数点やドル記号などの非数値の要素を取り除くことができます。

LEADINGZEROS() 関数

文字列または数値の先頭にゼロを追加します。

構文

```
LEADINGZEROS(文字列/数値, 長さ)
```

パラメーター

名前	型	説明
文字/数値	文字 数値	先頭のゼロを追加するフィールド、式、またはリテラル値。 先頭と末尾のスペースは、先頭のゼロが追加される前に、文字値、および文字式の結果から、自動的に切り捨てられます。
長さ	数値	出力文字列の長さ。 メモ 文字/数値の値のうち、長さを超える部分は左から切り捨てられます。

出力

文字。

例

基本的な例

文字型の入力値

次の例では、長さが文字列/数値の文字数より大きいので、先頭に3つのゼロが追加されて "00235" が返されます。

```
LEADINGZEROS("235", 6)
```

次の例では、長さが文字/数値の文字数より小さいので、結果が左から切り詰められて"35"が返されません。

```
LEADINGZEROS("235", 2)
```

整数の入力値

"235"が返されます。

```
LEADINGZEROS(235, 3)
```

次の例では、長さが文字列/数値の桁数より大きいので、先頭に2つのゼロが追加されて"00235"が返されます。

```
LEADINGZEROS(235, 5)
```

小数の入力値

LEADINGZEROS() は小数点を削除するため、"023585"を返します。

```
LEADINGZEROS("235", 6)
```

負の入力値

LEADINGZEROS() は負の記号を削除するため、"0644894"を返します。

```
LEADINGZEROS(-6448.94, 7)
```

高度な例

数値を含む文字フィールドに先頭ゼロを追加する

Employee_Number フィールドには値 "254879" があります。値を先頭ゼロ埋めの 10 桁の文字列に変換する必要があります。

```
COMMENT "0000254879" を返します
ASSIGN v_str_length = 10
LEADINGZEROS(Employee_Number, v_str_length)
```

テーブルを関連付ける際にキーフィールドを一致させる

2つのテーブル、Ar と Customer がある場合に、これらをキーフィールドである CustNum で関連付けてさらに分析するとします。ただし、2つのキーフィールドに異なるデータ形式があり、値が一致しなくなります。

テーブル	キーフィールド	データ型	フィールド長	例
<ar>	CustNum	数値	4	4455
Customer	CustNo	文字	6 (数値の先頭をゼロ埋め)	"004455"

フィールド同士を関連付けの際に調整するには、Ar テーブル内に、LEADINGZEROS() 関数を使用する演算フィールドを作成します。次いで、この演算フィールドを使って関連付けを行います。

```
OPEN Customer
INDEX ON CustNo TO Customer_on_CustNo
OPEN Ar
COMMENT CustNum フィールドを文字データ型に変換して先頭にゼロを追加する演算フィールド CustNum_Zeros を作成します。
DEFINE FIELD CustNum_Zeros COMPUTED LEADINGZEROS(CustNum,6)
COMMENT 新しく作成した演算フィールド CustNum_Zeros を使って Ar テーブルを関連付けます。
DEFINE RELATION CustNum_Zeros WITH Customer INDEX Customer_on_CustNo
```

備考

機能の仕組み

この関数は、指定する出力値が入力値の長さより大きい場合に、先頭のゼロを出力に追加します。この関数は、さまざまな種類の文字および数値の入力を許可し、文字列を出力します。この関数は、Unicode および非 Unicode の Analytics で同様に動作します。

この関数は、小切手番号や、発注番号、請求書番号フィールドなどの、先頭にゼロを必要とするフィールドを正規化させるためによく利用されます。

入力長さと出力値

入力長さ	出力値
長さ未満の文字列/数値	先頭ゼロを追加
長さと等しい文字列/数値	ゼロが追加されない
長さより大きい文字列/数値	値は左から切り捨てられる

負の記号、千の位の桁区切り、小数点

入力値	結果
数値	LEADINGZEROS() では、数値型の入力値から、負の記号、千の位の桁区切り、小数点が削除されます。 削除された記号は、入力値の長さに含まれません。
文字	LEADINGZEROS() では、文字型の入力値からは、負の記号、千の位の桁区切り、小数点は削除されません。 削除された記号は、入力値の長さに含まれます。

LENGTH() 関数

文字列に含まれている文字数を返します。

構文

```
LENGTH(文字列)
```

パラメーター

名前	型	説明
文字列	文字	長さを検出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

15 が返されます。

```
LENGTH("ABC Corporation")
```

テーブルレイアウト内の **Description** フィールドの、文字数で表した長さを返します。

```
LENGTH(Description)
```

高度な例

住所関連フィールドの各住所の長さを表示する

Vendor_Street フィールド内の各住所の、文字数で表した長さを表示する演算フィールドを作成します。住所の値から最初に削除される先頭と末尾のスペースは、長さにカウントされません。

```
DEFINE FIELD Address_Length COMPUTED LENGTH(ALLTRIM(Vendor_Street))
```

備考

機能の仕組み

LENGTH() 関数はスペースも含めて文字列中の文字数を計算して、その数を返します。

末尾のスペース

末尾のスペースは文字としてカウントされます。末尾のスペースをカウントしたくない場合は、TRIM() または ALLTRIM() 関数を使用して末尾のスペースを削除します。例：

```
LENGTH(TRIM(Vendor_Street))
```

フィールドの値の長さを表示する演算フィールドを作成する場合に、末尾のスペースを削除しないときは、値ごとにフィールドの最大長が表示されます。

LEVDIST() 関数

指定された2つの文字列間のレーベンシュタイン距離を返します。これは、2つの文字列がどの程度異なっているかを測る数値です。

構文

```
LEVDIST(文字列1, 文字列2 <,大文字と小文字の区別>)
```

パラメーター

名前	型	説明
文字列1	文字	比較の最初の文字列。
文字列2	文字	比較の2番目の文字列。
大文字と小文字の区別 省略可能	論理	文字列同士を大文字と小文字を区別して比較する場合は <code>T</code> 、大文字と小文字を区別しないで比較する場合は <code>F</code> を指定します。 これを省略した場合は、デフォルト値の <code>T</code> が使用されます。

出力

数値。この値は、2つの文字列間のレーベンシュタイン距離です。

例

基本的な例

"smith" を "Smythe" に変えるには、2回の置換と1回の挿入が必要なため、3が返されます。

```
LEVDIST("smith", "Smythe")
```

大文字と小文字が無視されるので、"smith's" を "Smythe" に変えるために必要なのは2回の置換だけのため、2が返されます。

```
LEVDIST("smith's","Smythes",F)
```

Last_name フィールドの各値と文字列 "Smith" の間のレーベンシュタイン距離が返されます。

```
LEVDIST(TRIM>Last_Name),"Smith")
```

高度な例

"Smith" に対する値をランク付けする

"Smith" と **Last_Name** フィールドの各値とのレーベンシュタイン距離を表示する演算フィールド、**Lev_Dist** を作成します。

```
DEFINE FIELD Lev_Dist COMPUTED LEVDIST(TRIM>Last_Name),"Smith", F)
```

ビューに演算フィールド **Lev_Dist** を追加して、この演算フィールドの昇順にクイックソートを行うことで、**Last_name** フィールドのすべての値を "Smith" との相違の量によってランク付けすることができます。

"Smith" に対するあいまい重複の抽出

Last_name フィールドから、"Smith" とのレーベンシュタイン距離が指定された範囲内であるすべての値を抽出するフィルターを作成するには、次のように指定します。

```
SET FILTER TO LEVDIST(TRIM>Last_Name),"Smith", F) < 3
```

式の数字を変更すると、フィルターされる値のレーベンシュタイン距離の量を調整することができます。

備考

LEVDIST() の使用に適する場面

LEVDIST() 関数は、ほぼ同一の値(あいまい重複)の検出や、手作業で入力されたデータで一貫性のないつづりを見つける場合に使用できます。また、LEVDIST() は完全な重複も識別します。

機能の仕組み

LEVDIST() 関数は、2つの評価される文字列間のレーベンシュタイン距離を返します。レーベンシュタイン距離とは、ある文字列を別の文字列にするために必要な、1文字の編集の最小回数を示す値です。

必要な編集を行うたびに、レーベンシュタイン距離の値に1が加算されます。レーベンシュタイン距離が大きいほど、2つの文字列間の相違が大きいこととなります。距離がゼロ(0)ということは、文字列がまったく同じということです。

編集の種類

編集には次の3つの種類があります。

- 挿入
- 削除
- 置換

転置(隣接する2つの文字の位置を入れ替える)は、レーベンシュタインアルゴリズムでは認識されないため、2回の編集、具体的には2回の置換と見なされます。

英数字以外の文字

句読点、特殊文字、および空白は、普通の文字や数字と同様、単一の文字として扱われます。

大文字と小文字

大文字と小文字の区別を使用して大小文字の区別をオフにしない限り、文字に使用されている大文字または小文字の変更が1回の置換として数えられます。

文字の位置

レーベンシュタイン距離は文字の位置を考慮に入れます。同じ文字の組み合わせでも文字の順序が異なると、レーベンシュタイン距離が変化する可能性があります。

2が返されます。

```
LEVDIST("abc", "dec")
```

3 が返されます。

```
LEVDIST("abc", "dec")
```

LEVDIST() と TRIM() の併用

LEVDIST() を使って "Smith" などのリテラル文字列を文字フィールドと比較する場合、正確な結果を確保するためには、TRIM() 関数を使用してフィールドから末尾の空白を除去しておく必要があります。

2つのフィールドを比較する場合は、フィールドごとに TRIM() 関数を使用してください。

レーベンシュタインアルゴリズムは空白を文字として数えるため、末尾に空白があると、それらは2つの文字列を同一にするために必要な編集回数の計算にすべて含まれます。

関連する関数

- **ISFUZZYDUP()** - は、レーベンシュタイン距離に基づいて文字列を比較するためのもう1つの手段です。
LEVDIST() のデフォルトの動作とは異なり、ISFUZZYDUP() では大文字と小文字が区別されません。
- **DICECOEFFICIENT()** - は、文字列を比較するときに、文字または文字ブロックの相対位置を重視しないか、または完全に無視します。
- **SOUNDSLIKE() および SOUNDEX()** - は、正字法の比較(綴り)ではなく、発音記号の比較(発音)に基づいて文字列を比較します。

LOG() 関数

数式またはフィールド値の対数(底 10)を、指定された小数点以下の桁数で返します。

構文

```
LOG(数値, 小数位)
```

パラメーター

名前	型	説明
数値	数値	対数を確認する値。
小数位	数値	戻り値の小数点以下の桁数。

出力

数値。

例

基本的な例

3.0000 が返されます。

```
LOG(1000, 4)
```

4.86 が返されます。

```
LOG(72443, 2)
```

高度な例

立方根を求める

フィールド X の立方根を小数点以下 2 桁で表すフィールドを作成するには、次のようにします。

```
DEFINE FIELD Cube_root COMPUTED EXP(LOG(X, 6) / 3, 2)
```

メモ

ある値の指数 n の累乗根は、値の対数を n で割ってから、その結果の指数を取ることによって求められます。

備考

機能の仕組み

数値の対数とは、その数値を生成するために必要な、10 を底とする指数(すなわち累乗)です。たとえば、1000 の対数は 3 です。

関連する関数

LOG() は EXP() 関数の逆関数です。

LOWER() 関数

アルファベット文字を小文字に変換した文字列を返します。

構文

```
LOWER(文字列)
```

パラメーター

名前	型	説明
文字列	文字	小文字に変換するフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

"abc" が返されます。

```
LOWER("ABC")
```

"abc 123 def" が返されます。

```
LOWER("abc 123 DEF")
```

"abcd 12" が返されます。

```
LOWER("AbCd 12")
```

Last_Name フィールドのすべての値を小文字に変換した値が返されます。

```
LOWER>Last_Name)
```

備考

機能の仕組み

LOWER() 関数は文字列内のすべてのアルファベットを小文字に変換します。アルファベット以外の文字は変換されません。

LOWER() の使用に適する場面

LOWER() 関数は、大文字と小文字が混在するデータや、どちらの表記が使用されているか不明なデータを検索する場合、または小文字形式のデータが必要な場合に使用できます。

LTRIM() 関数

入力文字列から先頭のスペースを除去した文字列を返します。

構文

```
LTRIM(文字列)
```

パラメーター

名前	型	説明
文字列	文字	先頭のスペースを削除するフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

どちらの例も、LTRIM() 関数では末尾のスペースが除去されないことに注目してください。

"Vancouver " が返されます。

```
LTRIM(" Vancouver ")
```

"New York " が返されます。

```
LTRIM(" New York ")
```

高度な例

改行なしスペースの削除

改行なしスペースはLTRIM()関数で削除されません。

先頭の改行なしスペースを削除する必要がある場合は、次の式を使用して演算フィールドを作成します。

```
DEFINE FIELD Description_cleaned COMPUTED LTRIM(REPLACE(Description,  
CHR(160), CHR(32)))
```

REPLACE()関数がすべての改行なしスペースを標準のスペースに置換してから、LTRIM()がすべての先頭の標準スペースを削除します。

備考

機能の仕組み

LTRIM()関数は先頭スペースだけを除去します。文字列の内部スペースおよび後続スペースは除去されません。

関連する関数

LTRIM()の関連関数として、文字列の末尾のスペースを除去するTRIM()関数と、先頭と末尾の両方のスペースを除去するALLTRIM()関数があります。

MAP() 関数

文字列が、ワイルドカード文字やリテラル文字を含んでいる指定された書式文字列と一致するかどうかを示す論理値を返します。

構文

```
MAP(文字列, 書式)
```

パラメーター

名前	型	説明
文字列	文字	一致をテストするフィールド、式、またはリテラル値。
書式	文字	<p>文字列と比較するデータパターンまたは文字列。</p> <p>書式には、ワイルドカード文字、リテラル文字、これら2つの組み合わせを指定できます。</p> <pre>"\9\9\9-999-9999"</pre> <p>次のワイルドカード文字がサポートされています。</p> <ul style="list-style-type: none">"X"- すべてのアルファベットに一致します(a ~ z、A ~ Z、欧州文字)。このワイルドカード文字には大文字と小文字の区別はありません。つまり、"X"と"x"のいずれをも使用できます。"9"- すべての数字に一致します(0 ~ 9)。"1"- 空白以外のすべての文字に一致します。"?"- 空白を含むすべての文字に一致します。"\"- 直後の文字をリテラルとして指定するエスケープ文字。ワイルドカード文字(X、x、9、!、?)のどれかと文字どおり一致させたい場合は、エスケープ文字を使用します。"\"- リテラルの円記号(¥)を指定しています。

出力

論理。一致するものが見つかった場合はT(true)、そうでない場合はF(false)を返します。

例

基本的な例

シンプルな検索パターン

T が返されます。

```
MAP("ABC Plumbing", "xxx")
```

文字列には 4 桁の数字が要求されているのに、3 桁の数字しかないため、F が返されます。

```
MAP("045", "9999")
```

ワイルドカードをエスケープする

リテラル文字 "X" で始まってその後に任意の文字が続く値に対してのみ 'T' を返すことが目的である場合は、書式パラメーター "XX" を使用することで、最初の "X" をワイルドカードとしてでなくリテラルとして解釈されるようにします。

T が返されます。

```
MAP("XA-123", "XX")
```

```
MAP("GC-123", "XX")
```

```
MAP("XA-123", "\\XX")
```

F が返されます。

```
MAP("GC-123", "\\XX")
```

フィールドとパターン

次の例では、2 つの文字の後に 5 つの数字が続く構成であるか、またはこの構成で始まっている請求書番号が含まれるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されず。

関数

```
MAP(Invoice_Number, "XX99999")
```

次の例では、請求書番号が"AB12345"であるか、または"AB12345"で始まるすべてのレコードに対して、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
MAP(Invoice_Number, "AB12345")
```

次の例では、"AB"の後に5つの数字が続く構成であるか、またはこの構成で始まっている請求書番号が含まれるすべてのレコードに対し、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
MAP(Invoice_Number, "AB99999")
```

次の例では、SSN フィールドの値が社会保障番号の標準形式と一致しないすべてのレコードに対し、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
NOT MAP(SSN, "999-99-9999")
```

高度な例

「859-」で始まる 10 文字の製品コードを持つレコードを抽出する

「859-」で始まる 10 文字以上の製品コードを持つレコードのみを抽出するには、次の IF ステートメントと MAP() 関数を使用します。

```
EXTRACT RECORD IF MAP(Product_Code, "85\9-999999") TO "Long_Codes_859"
```

備考

MAP() の使用に適する場面

MAP() 関数は、英数字データからパターンや特定の書式を検索する場合に使用できます。このパターンや書式にはワイルドカード文字、リテラル文字、またはその双方の組み合わせを指定できます。

大文字と小文字の区別

MAP() 関数で2つのリテラル値を比較する場合、大文字と小文字は区別されません。たとえば、"a" は "A" と同等ではありません。

文字列のデータが大文字と小文字のどちらかに統一されていない場合は、MAP() 関数を使用する前に UPPER() 関数を使用して、大文字と小文字のどちらかのみになるように値を変換することができます。

例：

```
MAP(UPPER(Invoice_Number), "AB99999")
```

部分一致

MAP() では、部分一致がサポートされる場合とサポートされない場合があります。

MAP() における部分一致は、**正確な文字比較を行う**オプション(SET EXACT ON/OFF)の影響を受けるわけではありません。

部分一致がサポートされる場合とは

部分一致がサポートされるのは、書式の値が文字列の値より短い場合です。

書式が7文字で、文字列が9文字のため、Tが返されます。

```
MAP("AB1234567", "AB99999")
```

メモ

T(True) が返されるには、書式の値が文字列の値の先頭に出現している必要があります。

部分一致がサポートされない場合とは

部分一致がサポートされないのは、書式の値が文字列の値より長い場合です。

書式が7文字で、文字列が6文字のため、Fが返されます。

```
MAP("AB1234567", "AB99999")
```

書式が文字列より長い場合には、結果は常にFalseになります。

空白の処理

空白は文字として扱われ、次のいずれかの方法で処理できます。

関数

- 文字どおり空白に一致させる。そのように処理するには、書式の値の該当する位置に空白を挿入します。
- ワイルドカード "?" を使用する。これは、空白を含むすべての文字に一致します。

必要に応じて、文字列パラメーターに対し、TRIM()、LTRIM() または ALLTRIM() 関数を使用して先頭または末尾の空白を除去することにより、テキストの文字とテキスト中の空白のみが比較されるようにすることができます。

フィールドの連結

テーブル内の複数のフィールド内の値を検索したい場合は、2 つ以上のフィールドを連結して文字列に指定することができます。連結したフィールドは 1 つのフィールドのように扱われます。ただし、ALLTRIM() 関数を使用して個々のフィールドから空白を除去しなければ、各フィールドの先頭と末尾の空白を含むフィールドとなります。

MASK() 関数

2つの文字列の最初の各バイトに対してビットごとのAND演算を実行します。

構文

```
MASK(文字の値, 文字のマスク)
```

パラメーター

名前	型	説明
文字の値	文字	バイトを調べる文字列。
文字のマスク	文字	バイトを調べる相手となる文字列(マスク値)。

出力

文字。出力は、ビットごとのAND演算のバイナリ結果に対する文字表記です。

例

基本的な例

3(00110011)と6(00110110)のビットごとのAND演算の結果である"2"(00110010)が返されます。

```
MASK("3", "6")
```

備考

MASK() の使用に適する場面

MASK() 関数は、1 バイト データにおける特定のビット パターン(特定のビットが1に設定されているかどうかは問わない)を識別する場合に使用できます。

機能の仕組み

MASK() 関数は、文字の値と文字のマスクの最初の文字のバイナリ表記に対してビットごとの AND 論理演算を実行します。比較対象となる2つのバイトが一度に1ビットずつ比較されて、第3のバイナリ値への出力が行われます。

対応する各ビット同士の比較結果は1か0になります。

文字の値のビット	文字のマスクのビット	結果
0	0	0
0	1	0
1	0	0
1	1	1

比較対象となる文字列が1バイトより長い

比較対象となる文字列のどちらかが1バイトより長い場合、後続の文字は無視されます。

MATCH() 関数

指定された値が比較対象の値のうちのどれかと一致するかどうかを示す論理値を返します。

構文

```
MATCH(比較値, テスト <,...n>)
```

パラメーター

名前	型	説明
比較値	文字 数値 日付時刻	一致をテストするフィールド、式、またはリテラル値。
テスト <,...n>	文字 数値 日付時刻	比較値と比較するフィールド、式、またはリテラル値。 必要な数のテスト値を指定できますが、指定したすべての値は同じデータ型である必要があります。 <pre>MATCH(比較値, `20140930`, `20141030`)</pre>

メモ

MATCH() 関数への入力値には文字、数値、日付時刻のデータを指定できます。データ型を混在させることはできません。入力値はすべて同じデータ型に属している必要があります。

出力

論理。1つでも一致するものが見つかった場合は T(true)、そうでない場合は F(false) を返します。

例

基本的な例

メモ

特に指定のない限り、SET EXACT は OFF(デフォルト設定)であることを前提として、文字比較の値を返します。

リテラル値をテストする

T が返されます。

```
MATCH("ABC", "BCD", "CDE", "AB")
```

F が返されます。

```
MATCH(98, 99, 100, 101)
```

フィールドをテストする

次の例では、**Vendor_City** フィールドの値が "Phoenix"、"Austin"、または "Los Angeles" であるすべてのレコードに対し、T が返されます。それ以外のレコードに対しては 'F' が返されます。

```
MATCH(Vendor_City, "Phoenix", "Austin", "Los Angeles")
```

次の例では、**Vendor_City** フィールドの値が "Phoenix"、"Austin"、または "Los Angeles" でないすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
NOT MATCH(Vendor_City, "Phoenix", "Austin", "Los Angeles")
```

次の例では、**Vendor_City** フィールドの値が "PHOENIX"、"AUSTIN"、または "LOS ANGELES"(大文字か小文字かを問わない)であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

Vendor_City フィールドの値が、大文字に変換されてから、大文字の都市名と比較されます。

```
MATCH(UPPER(Vendor_City), "PHOENIX", "AUSTIN", "LOS ANGELES")
```

複数のフィールドをテストする

次の例では、**Vendor_City**、**City**、または **City_2** フィールドの値が "Phoenix" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
MATCH("Phoenix", Vendor_City, City, City_2)
```

SET EXACT の動作

次の例では、**Product_Code** フィールド内の製品コードが "A"、"D"、または "F" であるか、もしくは "A"、"D"、"F" で始まるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
MATCH(Product_Code, "A", "D", "F")
```

次の例では、**Product_Code** フィールド内の製品コードが1文字の "A"、"D"、または "F" であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます (SET EXACT を ON にしておく必要があります)。

```
MATCH(Product_Code, "A", "D", "F")
```

2つのフィールドを比較する

次の例では、業者と従業員の住所が同一の値であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

業者と従業員の住所の形式を統一する必要がある場合は、その他の関数を使用します。

```
MATCH(Vendor_Address, Employee_Address)
```

日付同士を比較する

次の例では、請求日が2014年9月30日か2014年10月30日であるすべてのレコードに対し、'T' が返されます。それ以外のレコードに対しては 'F' が返されます。

```
MATCH(Invoice_Date, `20140930`, `20141030`)
```

高度な例

異常な在庫レコードを抽出する

Inventory_Value_at_Cost フィールドと演算フィールド **Cost_x_Quantity** の間で金額が異なるすべてのレコードに対し、'T' が返されます。

```
EXTRACT RECORD IF NOT MATCH(Inventory_Value_at_Cost, Cost_x_Quantity)
TO "Non_matching_amounts"
```

部門 101、103、および 107 のレコードを抽出する

以下のように IF ステートメントと MATCH() 関数を使用することで、部門 101、103、または 107 に関連するレコードのみを抽出できます。

```
EXTRACT RECORD IF MATCH(Dept, "101", "103", "107") TO "Three_
Departments"
```

備考

OR 演算子の代わりに MATCH() を使用する

AND 演算子を使用する式の代わりに MATCH() 関数を使用することができます。

例：

```
MATCH(City, "Phoenix", "Austin", "Los Angeles")
```

上記の関数は、次の式と同等です。

```
City="Phoenix" OR City="Austin" OR City="Los Angeles"
```

数値型入力値の小数点以下の精度

比較する数値型入力値の小数点以下の精度が異なる場合、比較は高い方の精度に合わせて行われます。

1.23 は 1.23 と等しいため、T が返されます。

```
MATCH(1.23, 1.23, 1.25)
```

小数点第 3 位まで考慮されるため、1.23 は 1.234 と等しくないと判定されるので、F が返されます。

```
MATCH(1.23, 1.23, 1.25)
```

文字パラメーター

大文字と小文字の区別

文字データが使用される場合、MATCH() 関数は大文字と小文字を区別します。文字を比較する場合、"a" は "A" と同等ではありません。

F が返されます。

```
MATCH("a", "A", "B", "C")
```

大文字と小文字のどちらかに統一されていないデータを操作する場合は、MATCH() 関数を使用する前に UPPER() 関数を使用して、値をすべて大文字か小文字に変換することができます。

T が返されます。

```
MATCH(UPPER("a"), UPPER("A"), UPPER("B"), UPPER("C"))
```

部分一致

文字比較では、部分一致がサポートされます。比較するどちらか一方の値がもう一方の値の一部である場合に、これらの値が一致と見なされます。

次の例ではいずれも T が返されます。

```
MATCH("AB", "ABC")
```

```
MATCH("ABC", "AB")
```

メモ

短い値は、一致を成す長い値の先頭に出現している必要があります。

部分一致とSET EXACT

部分一致は、Analytics のデフォルト設定である SET EXACT = OFF の場合に有効になります。SET EXACT = ON にすると、部分一致は無効になり、比較値が一致を成すには、正確に一致しなければなりません。

SET EXACT が ON の場合、上記の例はどちらも False になります。

SET EXACT(**正確な文字比較を行う** オプション)の詳細については、"SET コマンド" ページ 2081を参照してください。

SET EXACT の OFF または ON への設定

MATCH() 関数で **正確な文字比較を行う** オプションが使用されないようにしたい場合は、**オプション** ダイアログボックスの **テーブル** タブ(**ツール > オプション**) でこのオプションが選択されていないことを確認してください。

スクリプトを使用している場合は、MATCH() 関数が現れる前に `SET EXACT OFF` コマンドを追加します。必要に応じて、`SET EXACT ON` コマンドによって前の状態に戻すことができます。

日付時刻パラメーター

関数への入力として指定された日付、日付時刻、または時刻フィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式、日付時刻書式、または時刻書式でも使用することができます。

日付型、日付時刻型、時刻型の入力値を混在させる

MATCH() 関数の入力値に日付値、日付時刻値、および時刻値を混在させることは禁止されていませんが、これら日付時刻のサブタイプを混在させると、意味のない結果が返される可能性があります。

日付時刻値の日付部分のみに関心があっても、時刻部分はまだ計算の一部を構成しているため、Analytics は対応するシリアル値を使用して、日付時刻の計算を処理しています。

次の例で考えてみましょう。

2014 年 12 月 31 日は 2 番目のテスト値と一致しているため、T が返されます。

```
MATCH(`20141231`, `20141229`, `20141231`)
```

比較値と2番目のテスト値は同一の日付 2014 年 12 月 31 日を持っているにもかかわらず、F が返されます。

```
MATCH(`20141231 120000`, `20141229`, `20141231`)
```

これら2つの式の対応するシリアル値を見ると、どうして2番目の式がfalseと評価されるのかがわかりません。

比較値のシリアル番号が2番目のテスト値のシリアル番号と等しいため、T が返されます。

```
MATCH(42003.000000, 42001.000000, 42003.000000)
```

比較値のシリアル番号がどのテスト値とも等しくないため、F が返されます。

```
MATCH(42003.500000, 42001.000000, 42003.000000)
```

シリアル番号 `42003.500000` と `42003.000000` の間では、日付部分は一致していますが、時刻部分は一致していません。`0.500000` は 12:00 PM に相当するシリアル値です。

日付時刻サブタイプ同士を一致させる

混在する日付時刻サブタイプによって発生する可能性のある問題を回避するには、サブタイプ同士を一致させる関数を使用します。

たとえば、次の式では、上記の2番目の式と同じ初期値が使用されていますが、F でなく T が返されます。

```
MATCH(CTOD(DATE(`20141231 120000`, "YYYYMMDD")), "YYYYMMDD"), `20141229`, `20141231`)
```

リテラル日付、日付時刻、または時刻値の指定

関数への入力のいずれかにリテラルの日付値、日付時刻値、または時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 ' ' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time:

協定世界時)からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
thhmmss	`t235959`
Thhmm	`T2359`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

MAXIMUM() 関数

一連の数値中の最大値、または一連の日付時刻値中の最も新しい値を返します。

構文

```
MAXIMUM( 値1, 値2 <, ...n> )
```

パラメーター

名前	型	説明
値1、値2 <, ...n>	数値 日付時刻	カンマによって区切られた、比較し合う値。 すべての値は同じデータ型である必要があります。 また、日付時刻値は同じサブタイプである必要があります。この関数の1回の実行において、日付値、日付時刻値、時刻値を混在させることはできません。

出力

数値または日付時刻。

例

基本的な例

リテラル数値の入力値

7 が返されます。

```
MAXIMUM(4, 7)
```

8 が返されます。

関数

```
MAXIMUM(4, 7, 3, 8)
```

8.00 が返されます。

```
MAXIMUM(4, 7.25, 3, 8)
```

リテラル日付時刻の入力値

`20161231` が返されます。

```
MAXIMUM(`20161231`, `20161229`, `20161230`)
```

`20161231 23:59:59` が返されます。

```
MAXIMUM(`20161231 235959`, `20161229 235959`)
```

`23:59:59` が返されます。

```
MAXIMUM(`.235957`, `.235959`, `.235958`)
```

フィールドへの入力値

各レコードについて、次の3つのフィールドのうち、最も新しい日付が返されます。

```
MAXIMUM(PO_Date, Invoice_Date, Payment_Date)
```

高度な例

演算フィールドを作成してデフォルトの最小額を設定する

期限経過勘定のテーブルに支払利息という演算フィールドを作成してデフォルトの最小額 \$1.00 を設定するには、次の式を使用します。

```
DEFINE FIELD Interest_Due COMPUTED MAXIMUM(BALANCE * ANNUAL_RATE, 1)
```

残高に利率を掛けた値が\$1.00より小さい場合、MAXIMUM()関数は1を返します。そうでない場合、MAXIMUM()は計算した利息を返します。

四半期の終了日より後の日付を検出する

複数のフィールドにまたがる日付が四半期の終了日を過ぎているかどうかを判断するには、下記のような式を使用した演算フィールドを作成します。

```
DEFINE FIELD Past_Qtr COMPUTED MAXIMUM(PO_Date, Invoice_Date, Payment_Date, `20160331`)
```

- 2016年3月31日以前の日付が含まれるすべてのレコードに対し、`20160331`が返されます。
- 2016年3月31日より後の日付が含まれるレコードに対しては、上記3つのフィールドのうち、最新の日付が返されます。

備考

数値の比較における小数点以下の桁数の動作

比較し合う数値の小数点以下の桁数が異なる場合、比較結果は最も大きい小数点以下の桁数に合わせて調整されます。

次の例では20.400が返されます。

```
MAXIMUM(3.682, 10.88, 20.4)
```

DECIMALS()関数を使用して、値パラメーターの小数点以下の桁数を調整することができます。

次の例では20.40が返されます。

```
MAXIMUM(DECIMALS(3.682, 2), 10.88, 20.4)
```

MINIMUM() 関数

一連の数値中の最小値、または一連の日付時刻値中の最も古い値を返します。

構文

```
MINIMUM(値1, 値2 <, ...n>)
```

パラメーター

名前	型	説明
値1, 値2<, ...n>	数値 日付時刻	カンマによって区切られた、比較し合う値。 すべての値は同じデータ型である必要があります。 また、日付時刻値は同じサブタイプである必要があります。この関数の1回の実行において、日付値、日付時刻値、時刻値を混在させることはできません。

出力

数値または日付時刻。

例

基本的な例

リテラル数値の入力値

4 が返されます。

```
MINIMUM(4, 7)
```

3 が返されます。

```
MINIMUM(4, 7, 3, 8)
```

3.00 が返されます。

```
MINIMUM(4, 7.25, 3, 8)
```

リテラル日付時刻の入力値

`20161229` が返されます。

```
MINIMUM(`20161231`, `20161229`, `20161230`)
```

`20161229 23:59:59` が返されます。

```
MINIMUM(`20161231 235959`, `20161229 235959`)
```

`23:59:57` が返されます。

```
MINIMUM(`.235957`, `.235959`, `.235958`)
```

フィールドへの入力値

各レコードについて、次の3つのフィールドのうち、最も古い日付が返されます。

```
MINIMUM(PO_Date, Invoice_Date, Payment_Date)
```

高度な例

複数のフィールドの値のうち、最も低い値を識別する

"Cost"(費用)、"Sale_Price"(販売価格)、“Discount_Price”(割引価格)フィールドの値のうち、最も低い値を識別するには、次の式を使用した演算フィールドを作成します。

```
DEFINE FIELD Low_Value COMPUTED MINIMUM(Cost, Sale_Price, Discount_Price)
```

四半期の開始日より前の日付を検出する

複数のフィールドにまたがる日付が四半期の開始日より前であるかどうかを判断するには、下記のような式を使用した演算フィールドを作成します。

```
DEFINE FIELD Pre_Qtr COMPUTED MINIMUM(PO_Date, Invoice_Date, Payment_Date, `20160101`)
```

- 2016年1月1日以後の日付が含まれるレコードに対し、`20160101` が返されます。
- 2016年1月1日より前の日付が含まれるレコードに対しては、上記3つのフィールドのうち、最も古い日付が返されます。

備考

数値の比較における小数点以下の桁数の動作

比較し合う数値の小数点以下の桁数が異なる場合、比較結果は最も大きい小数点以下の桁数に合わせて調整されます。

次の例では3.600が返されます。

```
MINIMUM(3.6,10.88, 20.482)
```

DECIMALS() 関数を使用して、値パラメーターの小数点以下の桁数を調整することができます。

次の例では3.60が返されます。

```
MINIMUM(3.6,10.88, DECIMALS(20.482, 2))
```

省略形 MIN()

ACLScriptでは、省略形のMINは関数を一意に識別しないにもかかわらず、MINIMUM()関数の省略形として使用することができます。これは、関数名を省略するときの規定の要件です。

MIN() は MINUTE() の省略形である可能性も考えられますが、Analytics では、省略形 MIN() を MINIMUM() 関数のために取っています。

MINUTE() 関数

指定された時刻または日付時刻から分数を抽出し、それを数値として返します。

構文

```
MINUTE(時刻/日付時刻)
```

パラメーター

名前	型	説明
時刻/日付時刻	日付時刻	分数を抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

59 が返されます。

```
MINUTE(`t235930`)
```

```
MINUTE(`20141231 235930`)
```

Call_start_time フィールドの各値の分数が返されます。

```
MINUTE(Call_start_time)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

スクリプトにおける MINUTE() の省略形

ACLScript で MINUTE() 関数を短縮する場合は、少なくとも最初の4文字 (MINU) を使用する必要があります。Analytics では、MINIMUM() 関数の省略形 MIN が予約されています。

パラメーターの詳細

時刻/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような時刻書式または日付時刻書式でも使用することができます。

リテラル時刻または日付時刻値の指定

日付時刻にリテラルの時刻値または日付時刻値を指定する場合は、次の表内の書式に制限されません。また、`20141231 235959` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **時刻値** - 以下の表に示す任意の時刻の書式を使用することができます。関数が正しく動作するためには、単独の時刻値の前に区切り文字を使用する必要があります。有効な区切り文字は文字 't' または 'T' です。24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。
- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があり、かつ、2 つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 't' または 'T' です。

形式の例	リテラル値の例
thhmmss	`t235959`
Thhmm	`T2359`
YYYYMMDD hhmmss	`20141231 235959`

形式の例	リテラル値の例
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。	

MOD() 関数

2つの数値を除算して余りを返します。

構文

```
MOD(数値, 除数)
```

パラメーター

名前	型	説明
数値	数値	除算する数値。
除数	数値	数値を除算する数値。 数値または除数、あるいはこれら両方が小数を含んでいる場合、出力の小数点以下の精度は、入力値で小数点以下の桁数が多い方と同じになります。たとえば、MOD(45.35, 5.3)の出力は2.95です。

出力

数値。

例

基本的な例

3が返されます。

```
MOD(93, 10)
```

2.0が返されます。

関数

```
MOD(66, 16.00)
```

3.45 が返されます。

```
MOD(53.45, 10)
```

高度な例

創業記念日を計算する

前回の創業記念日からの経過月数を表示するフィールドを定義するには、次のように指定します。

```
DEFINE FIELD Months_since_last_anniversary COMPUTED MOD(Months_of_
service, 12)
```

備考

MOD() の使用に適する場面

MOD() 関数は、2つの数値が等分できるかどうかをテストしたり、除算の余りを分離させたりする場合に使用します。この関数は、一方の数値を別の数値で割り、その余りを返します。

MONTH() 関数

指定された日付または日付時刻から月を抽出し、それを数値(1 ~ 12)として返します。

構文

```
MONTH(日付/日付時刻)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	月を抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

12 が返されます。

```
MONTH(`20141231`)
```

```
MONTH(`20141231 235959`)
```

Invoice_date フィールドの各値に対して月が返されます。

```
MONTH(Invoice_date)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`

形式の例	リテラル値の例
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
メモ UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。	

関連する関数

月の名前を返したい場合は、MONTH() でなく CMOY() を使用してください。

NOMINAL() 関数

貸付金の名目年利率を返します。

構文

```
NOMINAL(実効利率, 期間)
```

パラメーター

名前	型	説明
実効利率	数値	実効年利率。
期間	数値	年間の複利計算回数。 メモ 整数を指定してください。小数部分を指定しても切り捨てられるためです。

出力

数値。利率は小数点以下 8 桁で計算されます。

例

基本的な例

19.56% の実効年利を請求されるクレジットカードの未払い分に対し、名目年利率 0.17998457 (18%) が返されます。

```
NOMINAL(0.1956, 12)
```

備考

名目年利率とは

貸付金の名目年利率とは、未払い額に対する利息を考慮せずに毎月または毎日複利計算が行われると仮定した場合の名目利率または公示利率です。

関連する関数

EFFECTIVE() 関数は NOMINAL() 関数の逆関数です。

NORMDIST() 関数

正規分布データセットのランダム変数が指定した値以下になるか、指定した値とまったく同じになる確率を返します。

構文

```
NORMDIST(x, 平均, 標準偏差, 累計)
```

パラメーター

名前	型	説明
x	数値	確率を計算する値。
平均	数値	データセットの平均値。
標準偏差	数値	データセットの標準偏差。標準偏差値は0より大きくなければなりません。
累計	論理	Tを指定すると、ランダム変数がx以下である確率(累計確率)を計算します。Fを指定すると、ランダム変数がxに等しい確率(単純な確率)を計算します。

出力

数値。

例

基本的な例

0.908788780274132 が返されます。

```
NORMDIST(42, 40, 1.5, T)
```

0.109340049783996 が返されます。

```
NORMDIST(42, 40, 1.5, F)
```

NORMSINV() 関数

標準正規分布で指定された確率に関連付けられたzスコアを返します。zスコアは、標準正規分布の平均を基準にした、値の標準偏差数です。

構文

```
NORMSINV( 確率 )
```

パラメーター

名前	型	説明
確率	数値	zスコアを計算する確率。

出力

数値。

例

基本的な例

1.333401745213610 が返されます。

```
NORMSINV(0.9088)
```

NOW() 関数

現在のオペレーティングシステム時刻を日付時刻データ型で返します。

構文

```
NOW()
```

パラメーター

この関数にはパラメーターはありません。

出力

日付時刻。

例

基本的な例

現在のオペレーティングシステム時刻が、現在の Analytics の時刻表示書式を使用して表示される日付時刻値 ('t235959' など) として返されます。

```
NOW()
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896

関数

- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

関連する関数

現在のオペレーティングシステム時刻を文字列として返させたい場合には、NOW() でなく TIME() を使用してください。

NPER() 関数

貸付金を完済するのに必要な期間の数値を返します。

構文

NPER(利率, 支払金額, 金額 <, 種類 >)

パラメーター

名前	型	説明
利率	数値	1 期あたりの利率。
支払金額	数値	1 期間あたりの支払金額。
金額	数値	貸付金の元金。
種類 省略可能	数値	支払いのタイミング: ◦ 0 - 期末払い ◦ 1 - 期首払い 支払いのタイミングが省略された場合は、デフォルト値の 0 が使用されます。

出力

数値。

例

基本的な例

毎月末までに \$1,856.82 を支払うことによって年利 6.5% の貸付金、\$275,000 を完済するのに必要な月数、300.00 が返されます。

関数

```
NPER(0.065/12, 1856.82, 275000, 0)
```

毎月末までに\$2,000を支払うことによって同じ貸付金を完済するのに必要な月数、252.81が返されます。

```
NPER(0.065/12, 1856.82, 275000, 0)
```

毎月月初までに\$2,000を支払うことによって同じ貸付金を完済するのに必要な月数、249.92が返されます。

```
NPER(0.065/12, 1856.82, 275000, 0)
```

高度な例

年金の計算

年金の計算では、次の4つの変数が使用されます。

- 現在価値または将来価値 - 下の例では\$21,243.39と\$26,973.46
- 1期間あたりの支払金額 - 下の例では\$1,000.00
- 1期間あたりの利率 - 下の例では月あたり1%
- 期間の数 - 下の例では24か月

これらの変数のうち3つの値がわかっている場合は、Analytics関数を使って残りの変数の値を計算できます。

求めたい値:	使用する Analytics 関数
現在価値	PVANNUIITY() 21243.39を返す: <pre>FVANNUIITY(0.01, 12, 1000)</pre>
将来価値	FVANNUIITY() 26973.46を返す: <pre>FVANNUIITY(0.01, 24, 1000)</pre>
1期間あたりの支払金額	PMT()

求めたい値:	使用する Analytics 関数
	1000 を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">PMT(0.01, 24, 21243.39)</div>
1 期間あたりの利率	RATE() 0.00999999(1%) を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">RATE(24, 1000, 21243.39)</div>
期間の数	NPER() 24.00 を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">NPER(0.01, 1000, 21243.39)</div>

年金の式

期末年金(期末払い)の現在価値を計算する式:

$$PV_A = Pmt \left[\frac{1 - \frac{1}{(1+i)^N}}{i} \right]$$

$$21243.39 = 1000 \left[\frac{1 - \frac{1}{(1+0.01)^{24}}}{0.01} \right]$$

期末年金(期末払い)の将来価値を計算する式:

$$FV_A = Pmt \left[\frac{(1+i)^N - 1}{i} \right]$$

$$26973.46 = 1000 \left[\frac{(1+0.01)^{24} - 1}{0.01} \right]$$

OCCURS() 関数

部分文字列が指定された文字値内に現れる回数を数えて返します。

構文

```
OCCURS(文字列, 検索語)
```

パラメーター

名前	型	説明
文字列	文字	検索される値。 テーブル内の複数のフィールド内の値を検索したい場合は、2つ以上のフィールドを連結することができます。 <pre>OCCURS(First_Name+Last_Name,"John")</pre>
検索語	文字	検索対象となる値。検索は大文字と小文字を区別します。

出力

数値。

例

基本的な例

2 が返されます。

```
OCCURS("abc/abc/a","ab")
```

3 が返されます。

```
OCCURS("abc/abc/a","a")
```

Invoice_Number フィールド内の各値におけるハイフンの出現回数が返されます。

```
OCCURS(Invoice_Number, "-")
```

高度な例

ハイフンが2つ以上入っている請求書番号を見つける

テーブル内の Invoice_Number にハイフンが1つしか入ってはいけない場合は、OCCURS() 関数を使って、2つ以上のハイフンが含まれる Invoice_Number を抽出するフィルターを作成します。

```
SET FILTER TO OCCURS(Invoice_Number, "-") > 1
```

あるフィールドの値の他フィールドへの出現箇所を見つける

あるフィールドの値の他フィールドへの出現箇所を見つけるには、OCCURS() を使用します。たとえば、Last_Name の値が Full_Name フィールドに出現しているレコードを抽出するフィルターを作成できます。

```
OCCURS(Full_Name, ALLTRIM>Last_Name)) = 1
```

式に ALLTRIM() 関数を追加すると、Last_Name フィールドの先頭と末尾のスペースがすべて取り除かれ、テキスト値のみが比較されるようになります。

大文字と小文字を区別する検索を実行する

FIND() 関数とは異なり、OCCURS() 関数は大文字と小文字を区別するため、大小文字を区別した検索を実行できます。

関数

次の式は、**Vendor_Name** フィールドに大文字の "UNITED EQUIPMENT" という名前が入っているすべてのレコードを抽出します ("United Equipment" の出現箇所は無視)。

```
SET FILTER TO OCCURS(UPPER(Vendor_Name), "UNITED EQUIPMENT") > 0
```

大文字と小文字の違いを無視して "United Equipment" のすべての出現箇所を見つけたい場合は、UPPER() 関数を使って検索フィールドの値を大文字に変換します。

```
OCCURS(Vendor_Name, "UNITED EQUIPMENT") > 0
```

OFFSET() 関数

指定されたバイト数によって補正された開始位置からフィールドの値を返します。

構文

```
OFFSET(フィールド, バイト数)
```

パラメーター

名前	型	説明
フィールド	文字 数値 日付時刻	フィールド名。
バイト数	数値	任意の正の数式。

出力

戻り値は、入力のフィールドパラメーターと同じデータ型になります。

例

基本的な例

"Number" というフィールドに値 "1234567890" が入っており、"Offset_Number" という開始位置 1、長さ 3、小数位なしの重複フィールドを定義するとします。OFFSET() 関数を使用して、フィールド内の数字の位置を変えることができます。

123 が返されます。

```
OFFSET(Offset_Number,0)
```

234 が返されます。

関数

```
OFFSET(Offset_Number,1)
```

789 が返されます。

```
OFFSET(Offset_Number,6)
```

備考

フィールドの開始位置の一時的な補正としてこの関数を使用できます。フィールドの開始位置が可変であるデータを処理している場合、これは有用です。

条件演算フィールドで OFFSET() 関数を使用する場合、IF テストで参照されるフィールドも補正される点に注意してください。

OMIT() 関数

指定した1つ以上の部分文字列が削除された文字列を返します。

構文

```
OMIT(文字列1, 文字列2 <,大文字と小文字の区別>)
```

パラメーター

名前	型	説明
文字列1	文字	1つ以上のサブ文字列を削除するフィールド、式、またはリテラル値。
文字列2	文字	削除対象となる1つ以上の部分文字列。 <ul style="list-style-type: none">複数の部分文字列を区切るにはカンマを使用します。その部分文字列が削除したい部分文字列の一部である場合のみ、カンマの後にスペースを使用します。部分文字列のいずれかに二重引用符文字が出現している場合は、文字列2パラメーターの前後を二重引用符でなく一重引用符で囲む必要があります('')。カンマを削除するには、部分文字列のリストの最後に単一のカンマを置き、その直後に閉じる引用符を続けます(下記の最後の例を参照してください)。
大文字と小文字の区別 省略可能	論理	部分文字列で大文字と小文字が区別されるようにするには T を指定し、大文字と小文字の違いが無視されるようにするには F を指定します。 大文字と小文字の区別を省略した場合は、デフォルト値の T が使用されます。

出力

文字。

例

基本的な例

リテラル文字の入力値

"Intercity Couriers" が返されます。

```
OMIT("Intercity Couriers Corporation", " Corporation, Corp.")
```

"Inter-city Couriers" が返されます。

```
OMIT("Inter-city Couriers Corp.", " Corporation, Corp.")
```

メモ

この戻り値と上記の例の戻り値との間のレーベンシュタイン距離は1です。一般的要素が削除されていない場合、2つの例の間の距離は8となり、値がお互いのあいまい重複としての検出を逃れる可能性があります。

フィールドへの入力値

"Corporation" および "Inc." などの一般的な要素を含む業者名フィールドのすべての値を返します。削除済み:

```
OMIT(Vendor_Name, " Corporation, Corp., Corp, Inc., Inc, Ltd., Ltd")
```

"Corporation" および "Inc." などの一般的な要素を含む業者名フィールドのすべての値を返します。削除済み:

```
OMIT(Vendor_Name, " ,.,Corporation,Corp,Inc,Ltd")
```

メモ

上記の2つの例は同じ結果を返しますが、2番目の例の構文の方がより効率的です。

Vendor_Name フィールドのすべての値から "Corporation" や "Corp" などの要素とすべてのカンマを削除した値が返されます。

```
OMIT(Vendor_Name, " Corporation, Corp,")
```

備考

OMIT()により、部分文字列ごとに削除を行うことができます。

OMIT()関数は、文字列から1つ以上の部分文字列を削除します。これは、CLEAN()、EXCLUDE()、INCLUDE()、およびREMOVE()などの関数とは異なり、文字単位ではなく、部分文字列単位で文字列と一致し、削除を行います。サブ文字列の削除により、文字列から特定の単語、略語、あるいは繰り返し現れる文字の並びを、文字列の残り部分に影響を与えることなく削除することができます。

あいまい比較用のヘルパー関数

OMIT()は、テストフィールド値から"Corporation"や"Inc."、カンマ、ピリオド、アンパサンド(&)文字などの一般的要素を除去することによって、ISFUZZYDUP()関数、FUZZYDUPコマンド、FUZZYJOINコマンドの効果を高めることができます。

一般的要素と句読点の除去により、曖昧比較は、意味のある違いが発生する可能性のある文字列の部分だけに集中されます。

部分文字列間の順序が結果に及ぼす影響

削除する部分文字列を複数指定する場合、文字列2にそれらを列挙した順序が出力結果に影響を与えることがあります。

OMIT()関数の処理時、最初の部分文字列がそれを含んでいるすべての値から削除され、次に2番目の部分文字列がそれを含んでいるすべての値から削除され、というように続きます。もし、ある部分文字列が別の文字列の一部を構成している場合、たとえば"Corp"と"Corporation"という場合には、先に短い部分文字列を削除すると、長い部分文字列を含んでいる値も変えてしまい("Corporation"が"oration"になります)、長い部分文字列が検出されなくなります。

このような状況を回避するために、長い部分文字列を、それに含まれるあらゆる短い部分文字列よりも先に指定してください。例：

```
OMIT(Vendor_Name," Corporation, Corp., Corp")
```

まず特殊文字を削除するようにしてください。

句読点、特殊文字、およびスペースなどの単一文字の部分文字列を指定して、文字列内の一般的な構成要素をさらに減らすことができます。最初にピリオドや空白などの単一文字を削除する方が効率的かもしれません。これにより、その後指定する必要のある部分文字列の変化形の数が減ります。上記の3番目と4番目の例を比較してください。どちらも同じ結果を返しますが、4番目の例の方がより効率的です。

空白やスペースの扱い

部分文字列内の空白やスペースは他の文字と同様に扱われます。サブ文字列の一部として削除する各スペースを明示的に指定する必要があります。たとえば、空白をまったく入れないでアンパサンドを指定した場合("&")、"Ricoh Sales & Service" は "Ricoh Sales Service" になります。空白を入れた場合("& & ")、"Ricoh Sales & Service" は "Ricoh SalesService" になります。

サブ文字列の一部ではない空白を指定する場合、サブ文字列は検索されません。たとえば、アンパサンドと空白("& & ")を指定する場合、"Ricoh Sales&Service" は変更されません。

カンマを使って複数の部分文字列を区切る場合、カンマの後にスペースを入れるのは、そうすることで、削除したい実際の部分文字列と一致するときのみです。

空白を扱う手法の1つは、最初にフィールドからすべての空白を削除することです。これは、ほかのどの部分文字列を指定するより先に、空白を単一文字の部分文字列として指定することで行えます。

OMIT() を使用した結果を見直す

OMIT() を使って演算フィールドを作成した後、そのフィールドの内容を見直して、文字列の意味のある部分を不注意に削除していないかを確認してください。たとえば、"Co" を削除すると、"Company" に対してよく用いられる略語 (Co) が取り除かれますが、それはまた、"Coca-Cola" の2か所から文字 "Co" を除くことにもなります。

PACKED() 関数

パックデータ型に変換された数値データを返します。

構文

```
PACKED(数値, 結果の長さ)
```

パラメーター

名前	型	説明
数値	数値	変換する数値またはフィールド。
結果の長さ	数値	出力文字列で使用するバイトの数。

出力

数値。

例

基本的な例

整数と小数の入力値

00075C が返されます。

```
PACKED(75, 3)
```

```
PACKED(7.5, 3)
```

桁を切り捨てて出力する

00000012456D が返されます。

```
PACKED(-12.456, 6)
```

456D が返されます。

```
PACKED(-12.456, 2)
```

高度な例

メインフレームを更新する 8 バイトのフィールドを作成する

メインフレームへのアップロード対象として、各従業員の給与をパック型の数値にした 8 バイトのフィールドを作成する場合は、次のように指定します。

```
EXTRACT PACKED(SALARY, 8) AS "Salary_Export" TO "export"
```

備考

パック データとは

パック データ型は、メインフレームオペレーティングシステムで使用されるデータ型で、最小の記憶域を使用する形式で数値を格納します。パック データ型は 1 バイトにつき 2 桁の数字を格納し、最終バイトは、値が正負いずれであるかを示します。

PACKED() の使用に適する場面

PACKED() 関数は、数値データをメインフレームシステムへエクスポートするために、数値データをパックされた形式に変換する場合に使用できます。

戻り値が切り捨てられる場合とは

結果の長さの値が数値の長さより短い場合は、余分な桁が切り詰められます。

PI() 関数

円周率 π の値を、小数点以下 15 桁で返します。

構文

```
PI( )
```

パラメーター

この関数にはパラメーターはありません。

出力

数値。

例

基本的な例

3.141592653589793(小数点以下 15 桁での円周率 π の値) が返されます。

```
PI( )
```

60 度のラジアンに相当する 1.047197551196598 が返されます。

```
60 * PI( )/180
```

高度な例

入力値として度を使用する

60度の正弦に相当する0.866025403784439が返されます。

```
SIN(60 * PI( )/180)
```

備考

PI() の使用に適する場面

PI() 関数は、角度をラジアンに変換する場合に使用できます。 $(度 * PI()/180) = \text{ラジアン}$ となります。ラジアンは、Analytics の3つの数学関数 SIN()、COS()、TAN() で入力が必要不可欠です。

PMT() 関数

貸付金を完済するのに必要な定期払込金額(元金 + 利息)を返します。

構文

PMT(利率, 期間, 金額 <, 種類>)

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	支払期間の総数。
金額	数値	貸付金の元金。
種類 省略可能	数値	支払いのタイミング: ◦ 0 - 期末払い ◦ 1 - 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。

例

基本的な例

\$275,000 の貸付金を年利 6.5 パーセントで 25 年間で完済する場合 (支払期日は月末です) の月次支払額 (元金 + 利息)、(\$) 1856.82 が返されます。

```
PMT(0.065/12, 12*25, 275000, 0)
```

同じ貸付金を完済する場合 (支払期日は月初です) の月次支払額 (元金 + 利息)、(\$) 1846.82 が返されます。

```
PMT(0.065/12, 12*25, 275000, 1)
```

高度な例

年金の計算

年金の計算では、次の 4 つの変数が使用されます。

- **現在価値または将来価値** - 下の例では \$21,243.39 と \$ 26,973.46
- **1 期間あたりの支払金額** - 下の例では \$1,000.00
- **1 期間あたりの利率** - 下の例では月あたり 1%
- **期間の数** - 下の例では 24 か月

これらの変数のうち 3 つの値がわかっている場合は、Analytics 関数を使って残りの変数の値を計算できます。

求めたい値:	使用する Analytics 関数
現在価値	PVANNUITY() 21243.39 を返す: <pre>FVANNUITY(0.01, 12, 1000)</pre>
将来価値	FVANNUITY() 26973.46 を返す:

求めたい値:	使用する Analytics 関数
	FVANNUITY(0.01, 24, 1000)
1 期間あたりの支払金額	PMT() 1000 を返す: PMT(0.01, 24, 21243.39)
1 期間あたりの利率	RATE() 0.00999999(1%) を返す: RATE(24, 1000, 21243.39)
期間の数	NPER() 24.00 を返す: NPER(0.01, 1000, 21243.39)

年金の式

期末年金(期末払い)の現在価値を計算する式:

$$PV_A = Pmt \left[\frac{1 - \frac{1}{(1+i)^N}}{i} \right]$$

$$21243.39 = 1000 \left[\frac{1 - \frac{1}{(1+0.01)^{24}}}{0.01} \right]$$

期末年金(期末払い)の将来価値を計算する式:

$$FV_A = Pmt \left[\frac{(1+i)^N - 1}{i} \right]$$

$$26973.46 = 1000 \left[\frac{(1+0.01)^{24} - 1}{0.01} \right]$$

PPMT() 関数

単一の期間で貸付金に対して支払われた元金を返します。

構文

PPMT(利率, 指定期間, 期間, 金額 <, 種類>)

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
指定期間	数値	元金の支払いを確認する期間。
期間	数値	支払期間の総数。
金額	数値	貸付金の元金。
種類 省略可能	数値	支払いのタイミング: ○ 0- 期末払い ○ 1- 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のあ
る期間を使用する必要があります。

例:

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。

例

基本的な例

\$275,000 の貸付金を年利 6.5 パーセントで 25 年間にわたって返済する場合 (支払期日は月末です) の、1 年目に支払う元金、(\$) 367.24 が返されます。

```
PPMT(0.065/12, 1, 12*25, 275000, 0)
```

上記の貸付金の最終月において支払う元金、(\$) 1846.82 が返されます。

```
PPMT(0.065/12, 300, 12*25, 275000, 0)
```

備考

関連する関数

IPMT() 関数は PPMT() 関数に対して補完的役割を果たします。

CUMPRINC() 関数は一連の期間にわたって支払われた元金を計算します。

PROPER() 関数

各単語の最初の文字を大文字に、残りの文字を小文字に設定した文字列を返します。

構文

```
PROPER(文字列)
```

パラメーター

名前	型	説明
文字列	文字	適切な大文字または小文字に変換するフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

"John Doe" が返されます。

```
PROPER("JOHN DOE")
```

"John Doe" が返されます。

```
PROPER("john doe")
```

"1550 Albern St." が返されます。

関数

```
PROPER("1550 ALBERNI st.")
```

"Bill O'Hara" が返されます。

```
PROPER("BILL O'HARA")
```

Company_Name フィールドのすべての値を適切な大文字と小文字に変換した値が返されます。

```
PROPER(Company_Name)
```

備考

機能の仕組み

PROPER() 関数は、文字列の最初の文字と、空白の次に来る任意の文字を大文字に変換します。ハイフン、アポストロフィ、アンパサンド(&)、および他のいくつかの句読点と特殊文字に続く文字も大文字に変換されます。ほかのすべてのアルファベット文字は小文字に変換されます。

PROPER() の使用に適する場面

PROPER() の最も一般的な使い方は、すべて大文字かすべて小文字でデータソースに保存されている名前を適切な大文字と小文字書式に変換して、定型書簡やレポートに名前が正しく表示されるようにすることです。

PROPERTIES() 関数

指定された Analytics プロジェクト 項目のプロパティ情報を返します。

構文

```
PROPERTIES(名前, オブジェクトの種類, 情報の種類)
```

パラメーター

名前	型	説明
名前	文字	<p>情報を必要とする Analytics プロジェクト 項目の名前。名前は大文字と小文字を区別しません。</p> <p>プロジェクト 項目が Analytics テーブルである場合は、データファイル名ではなくテーブルレイアウト名を指定します。たとえば、「<code>january_invoices.fil</code>」ではなく「<code>Invoices</code>」と指定します。</p> <p>PROPERTIES() 関数を使用してアクティブなテーブルの名前を返させる場合は、「名前 <code>activetable</code>」を指定します。</p>
オブジェクトの種類	文字	<p>名前で参照する Analytics プロジェクト 項目の種類。</p> <p>メモ 現在のところ、サポートされているプロジェクト 項目の種類は <code>"table"</code> のみです。</p>
情報の種類	文字	<p>Analytics プロジェクト 項目について必要な情報の種類。</p> <p>詳細については、「プロパティ情報の種類」ページ 2408を参照してください。</p>

出力

文字。出力文字列の最大長は260文字です。プロパティ情報が見つからない場合は、空の文字列が返されます。

例

基本的な例

Analytics データファイル(.fil)に関する情報

"Ap_Trans.fil" が返されます。

```
PROPERTIES("Ap_Trans", "table", "filename")
```

"C:\ACL DATA\Sample Data Files" が返されます。

```
PROPERTIES("Ap_Trans", "table", "filepath")
```

開かれている Analytics テーブルに関する情報

"Ap_Trans" が返されます。

```
PROPERTIES("activetable", "table", "open")
```

外部データソースに関する情報

"Trans_May.xls" が返されます。

```
PROPERTIES("Trans_May", "table", "sourcename")
```

"C:\Project Data\Monthly Invoices_Excel" が返されます。

```
PROPERTIES("Trans_May", "table", "sourcepath")
```

"EXCEL" が返されます。

```
PROPERTIES("Trans_May", "table", "sourcetype")
```

備考

ファイル情報

"file" で始まる種類の情報は、Analytics テーブルと関連付けられている Analytics データ ファイル(.fil) に関する情報を提供します。

ソース情報

"source" で始まる種類の情報は、Analytics テーブルと関連付けることができる外部データソースに関する情報を提供します。PROPERTIES() 関数を使用して報告できる外部データソースは、Analytics テーブルの更新をサポートしている次の外部データソースだけです。

- Microsoft Excel
- Microsoft Access
- 区切り文字付きテキスト
- Adobe Acrobat(PDF)
- 印刷イメージ(レポート)
- SAP プライベート ファイル形式/DART
- XML
- XBRL
- ODBC データソース

プロパティ情報の種類

次の表は、PROPERTIES() 関数が返すことのできるプロパティ情報の種類の一覧です。Analytics テーブルが、PROPERTIES() 関数で現在使用できる唯一の Analytics プロジェクト項目です。

オブジェクトの種類	情報の種類	戻り値
"table"	"filename"	Analytics テーブルに関連付けられているデータ ファイルの名前。
	"filepath"	Analytics テーブルに関連付けられているデータ ファイルのパス。
	"filesize"	Analytics テーブルに関連付けられているデータ ファイルのサイズ(KB)。
	"filemodifiedat"	Analytics テーブルに関連付けられているデータ ファイルが最後に変更された日付と時刻。
	"sourcename"	Analytics テーブルに関連付けられているデータ ソースの名前。 データ ソースは、Excel、Access、PDF、XML、または区切り文字付きテキストファイルなどの外部ファイルか、ODBC データ ソースである可能性があります。
	"sourcepath"	Analytics テーブルに関連付けられているデータ ソースのパス。 ODBC データ ソースではサポートされません。
	"sourcetype"	Analytics テーブルに関連付けられているデータ ソースの種類。
	"sourcesize"	Analytics テーブルに関連付けられているデータ ソースのサイズ(KB)。 ODBC データ ソースではサポートされません。
	"sourcemodifiedat"	Analytics テーブルに関連付けられているデータ ソースが最後に変更された日付と時刻。 ODBC データ ソースではサポートされません。
	"open"	現在アクティブな Analytics テーブルの名前。 メモ 複数の Analytics テーブルを同時に開くことは可能ですが、ユーザー インターフェイスでは一度に1つのテーブルしかアクティブにすることができません。

PVANNUIITY() 関数

一定の利率を使って計算した一連の将来価値の現在価値を返します。現在価値は現在の総額です。

構文

```
PVANNUIITY(利率, 期間, 支払金額 <,種類>)
```

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	支払期間の総数。
支払金額	数値	1期間あたりの支払金額。 年金期間中、支払金額は毎回同じ金額でなければなりません。
種類 省略可能	数値	支払いのタイミング: ◦ 0- 期末払い ◦ 1- 期首払い 支払いのタイミングが省略された場合は、デフォルト値の0が使用されます。

メモ

利率、期間、支払金額を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のある期間を使用する必要があります。

例:

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。結果は小数点以下 2 桁まで計算されます。

例

基本的な例

月次支払額

2年間、月利 1% の複利で毎月月初に \$1,000 を支払う場合の現在価値として、(\$) 21455.82 が返されます。

```
PVANNUIITY(0.01, 2*12, 1000, 1)
```

年次支払額

2年間、年利 12% の複利で毎年年末に \$12,000 を支払う場合の現在価値として、(\$) 20280.61 が返されます。

```
PVANNUIITY(0.12, 2, 12000, 0)
```

高度な例

年金の計算

年金の計算では、次の 4 つの変数が使用されます。

- **現在価値または将来価値** - 下の例では \$21,243.39 と \$ 26,973.46
- **1 期間あたりの支払金額** - 下の例では \$1,000.00
- **1 期間あたりの利率** - 下の例では月あたり 1%
- **期間の数** - 下の例では 24 か月

これらの変数のうち 3 つの値がわかっている場合は、Analytics 関数を使って残りの変数の値を計算できます。

求めたい値:	使用する Analytics 関数
現在価値	PVANNUIITY() 21243.39 を返す:

求めたい値:	使用する Analytics 関数
	<code>FVANNUIITY(0.01, 12, 1000)</code>
将来価値	FVANNUIITY() 26973.46 を返す: <code>FVANNUIITY(0.01, 24, 1000)</code>
1 期間あたりの支払金額	PMT() 1000 を返す: <code>PMT(0.01, 24, 21243.39)</code>
1 期間あたりの利率	RATE() 0.00999999(1%) を返す: <code>RATE(24, 1000, 21243.39)</code>
期間の数	NPER() 24.00 を返す: <code>NPER(0.01, 1000, 21243.39)</code>

年金の式

期末年金(期末払い)の現在価値を計算する式:

$$PV_A = Pmt \left[\frac{1 - \frac{1}{(1+i)^N}}{i} \right]$$

$$21243.39 = 1000 \left[\frac{1 - \frac{1}{(1+0.01)^{24}}}{0.01} \right]$$

期末年金(期末払い)の将来価値を計算する式:

$$FV_A = Pmt \left[\frac{(1+i)^N - 1}{i} \right]$$

$$26973.46 = 1000 \left[\frac{(1 + 0.01)^{24} - 1}{0.01} \right]$$

備考

関連する関数

FVANNUIITY() 関数は PVANNUIITY() 関数の逆関数です。

PVLUMPSUM() 関数

一定の利率を使って計算する特定の将来総額を生成するのに必要な現在価値を返します。現在価値は現在の総額です。

構文

```
PVLUMPSUM( 利率, 期間, 金額)
```

パラメーター

名前	型	説明
利率	数値	1期あたりの利率。
期間	数値	期間の総数。
金額	数値	将来総額(最終期間の期末時の価値)

メモ

利率、期間を指定する際には、**1期あたりの利率**を確実に指定するため、整合性のある期間を使用する必要があります。

例：

- 年利 5% の 2 年間の貸付金または投資に対して月払いする場合は、利率に 0.05/12、期間に 2 * 12 を指定します。
- 同じ貸付金または投資に対して年払いする場合は、利率に 0.05、期間に 2 を指定します。

出力

数値。結果は小数点以下 2 桁まで計算されます。

例

基本的な例

月ごとに複利が付く場合

月利 1% の複利で 2 年間運用した場合に将来総額 \$1,269.73 を生み出せる初期投資元本として、(\$) 1000.00 が返されます。

```
PVLUMPSUM(0.01, 2*12, 1000)
```

月利 1% の複利で 2 年間運用した場合に将来総額 \$1,000 を生み出せる初期投資元本として、(\$) 787.57 が返されます。

```
FVLUMPSUM(0.01, 2*12, 1000)
```

月利 1% の複利で 2 年間運用した場合に将来総額 \$27,243.20 を生み出せる初期投資元本として、(\$) 21455.82 が返されます。

```
FVLUMPSUM(0.01, 2*12, 1000)
```

半年ごとに複利が付く場合

年利 12% の半年複利で 2 年間運用した場合に将来総額 \$1,000 を生み出せる初期投資元本として、(\$) 792.09 が返されます。

```
FVLUMPSUM(0.12/2, 2*2, 1000)
```

1 年ごとに複利が付く場合

年利 12% の複利で 2 年間運用した場合に将来総額 \$1,000 を生み出せる初期投資元本として、(\$) 797.19 が返されます。

```
FVLUMPSUM(0.12/2, 2*2, 1000)
```

備考

現在価値とは

運用する総額の現在価値とは、特定の時間枠内に特定の将来総額を生み出せる初期元本のことで、将来価値は、元本と複利の合計です。

関連する関数

FVLUMPSUM() 関数は PVLUMPSUM() 関数の逆関数です。

PYDATE() 関数

外部のPython スクリプトの関数によって計算された日付値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYDATE("Py ファイル,Py 関数" <, フィールド|値 <,...n>>)
```

パラメーター

名前	型	説明
Py ファイル、Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する Py関数は、Python の <code>datetime.date</code> オブジェクトを返す必要があります。</p>
フィールド 値 <,...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

日付時刻。

例

基本的な例

`20160630` が返されます。

```
PYDATE('hello,due_date', `20160531`, 30)
```

次の例は、日付と、日数としての猶予期間を入力として取って、請求書の支払期日を計算する外部 Python スクリプトです。請求日が**2016-05-31**で、猶予期間が30日の場合、支払期日は"2016-06-30"となります。

```
#!/python
日付時刻インポート時間差分から

def due_date(inv_date, period):
    return(inv_date + timedelta(period))
```

高度な例

演算フィールドを定義する

Python スクリプトを使って支払期日を計算する演算フィールドを `Ap_Trans` テーブルに定義します。

```
OPEN AP_Trans
DEFINE FIELD due_date COMPUTED
WIDTH 27
    PYDATE( "hello,due_date" ,Invoice_Date, Pay_Period)
```

PYDATETIME() 関数

外部の Python スクリプトの関数によって計算された日付時刻値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYDATETIME("Pyファイル,Py関数" <, フィールド|値 <,...n>>)
```

パラメーター

名前	型	説明
Py ファイル、Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する <i>Py</i>関数は、Python の datetime オブジェクトを返す必要があります。</p>
フィールド 値 <,...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

日付時刻。

例

基本的な例

`20170101t0500` が返されます。

```
PYDATETIME("hello, combine_date_time", `20170101`, `t0500`)
```

次の例は、日付の引数と時間の引数を入力として取り、これらを合成した datetime オブジェクトを返す外部 Python スクリプトです。

```
# hello.py content
日付時刻インポート日付時刻から

def combine_date_time(d,t):
    return datetime.combine(d,t)
```

高度な例

日付時刻に時刻を付加する

`20160101t2230` が返されます。

```
PYDATETIME("hello,add_time", `20160101 150000`, `t073000`)
```

次の例は、日付時刻と時間を入力として取り、日付時刻に時刻を付加する外部 Python スクリプトです。データの例: 2016-01-01 15:00:00 + 7 時間 30 分 00 秒 = 2016-01-01 22:30:00

```
# hello.py content
日付時刻インポート時間差分から
日付時刻インポート日付時刻から
```

関数

```
日付時刻インポート時刻から  
def add_time(start, time_to_add):  
    return start + timedelta(hours=time_to_add.hour, minutes=time_to_  
add.minute, seconds=time_to_add.second)
```

PYLOGICAL() 関数

外部のPython スクリプトの関数によって計算された論理値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYDATE("Py ファイル,Py 関数" <, フィールド|値 <,...n>>)
```

パラメーター

名前	型	説明
Py ファイル、Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する Py関数は、Python の真理値を返す必要があります。</p>
フィールド 値 <,...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

論理。

例

基本的な例

F が返されます。

```
PYLOGICAL( "hello,str_compare", "basketball", "baseball", "b" )
```

次の例は、*char*として渡される文字カウントを使って、*str1*と*str2*を比較する外部 Python スクリプトです。

```
# hello.py content
def str_compare(str1, str2, char):
    return str1.count(char) > str2.count(char)
```

高度な例

フィールドを使用する

Vendor_NameとVendor_Cityを比較したときの真理値が返されます。

```
PYLOGICAL( "hello,str_compare" Vendor_Name, Vendor_City, 'b' )
```

次の例は、*char*として渡される文字カウントを使って、*str1*と*str2*を比較する外部 Python スクリプトです。

```
# hello.py content
def str_compare(str1, str2, char):
    return str1.count(char) > str2.count(char)
```

PYNUMERIC() 関数

外部の Python スクリプトの関数によって計算された数値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYNUMERIC(Py ファイル, Py 関数, 小数点以下の桁数 <, フィールド|値 <, ...n>>)
```

パラメーター

名前	型	説明
Py ファイル、Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する Py関数は、Python の数値型を返す必要があります。</p>
小数点以下の桁数	数値	<p>戻り値に含める小数点以下の桁数。正の整数である必要があります。</p>
フィールド 値 <, ...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

数値。

例

基本的な例

35.00 が返されます。

```
PYNUMERIC("hello,get_nth_percent", 2, 80, 120, 30, 45, 30, 100, 35, 45)
```

次の例は、複数の値を含む、動的にサイズ設定したリストにおける、要求したパーセンタイルの値を返す外部 Python スクリプトです。

```
# hello.py content
演算インポート上限から
def get_nth_percent(percentage, *values):
    input_length = len(values)
    position = ceil((percentage/100.00) * input_length)
    return values[position-1]
```

PYSTRING() 関数

外部の Python スクリプトの関数によって計算された文字値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYSTRING("Py ファイル, Py 関数", 長さ <, フィールド|値 <, ...n>>)
```

名前	型	説明
Py ファイル, Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する Py関数は、Python の string オブジェクトを返す必要があります。</p>
長さ	数値	返される文字列に割り当てられる長さ。
フィールド 値 <, ...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

文字。

例

基本的な例

"my test" が返されます。

```
PYSTRING('hello,main', 20, "my")
```

次の例は、文字列を入力として取り、その文字列に"test"を連結する外部 Python スクリプトです。

```
#!/python
# hello.py content
def main(str):
    str2 = str + ' test'
    return(str2)
```

高度な例

部分文字列が返される

次の例では、Vendor_Name フィールドから最後の2文字が削除されて、削除後の部分文字列が返されます。

```
PYSTRING( "hello,sub_set", LENGTH(Vendor_Name), ALLTRIM(Vendor_Name),
LENGTH(ALLTRIM(Vendor_Name)), 0, LENGTH(ALLTRIM(Vendor_Name)) - 2
```

次の例は、文字列 (str)、文字列長 (length)、2つの桁位置 (p1 と p2) を入力として取る外部 Python スクリプトです。この関数により、p1(位置 1) と p2(位置 2) の間にある部分文字列が返されます。

```
#hello.py content
def sub_set(str, length, p1, p2):
    if p1 >= 0 and p2 < length and p1 < p2:
        str2 = str[p1:p2]
    else:
        str2 = str
    return str2
```

PYTIME() 関数

外部の Python スクリプトの関数によって計算された時刻値を返します。Python によるデータ処理は Analytics の外部で行われます。

構文

```
PYTIME("Py ファイル,Py 関数" <, フィールド|値 <,...n>>)
```

パラメーター

名前	型	説明
Py ファイル、Py 関数	文字	<p>実行する Python スクリプトの名前、カンマ、値を返す関数の名前。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>たとえば、「myScript,myFunction」と指定します。</p> </div> <p>Python スクリプトを指定する場合、ファイル拡張子は省略してください。呼び出す関数から、同じ Python スクリプトやその他のスクリプトにある他の関数を呼び出すことができますが、実行するスクリプトはすべて、実行前に <code>PYTHONPATH</code> システム環境変数のフォルダーに入れておく必要があります。</p> <p>詳細については、「Python の Analytics 連携用設定」ページ 2697を参照してください。</p> <p>メモ 使用する Py関数は、Python の datetime.time オブジェクトを返す必要があります。</p>
フィールド 値 <,...n> 省略可能	文字 数値 日付時刻 論理	<p>Python 関数の引数として使用するフィールド、式、リテラル値から成るこのリスト。値は呼び出す関数に指定順に渡されます。</p> <p>Python スクリプトの関数定義を満たすのに必要な数の引数を指定できます。</p> <p>メモ 文字入力から先頭と末尾の空白を除去するには、<code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「ALLTRIM() 関数」ページ 2163を参照してください。</p>

出力

日付時刻。

例

基本的な例

`t2122` が返されます。

```
ASSIGN v_time_part = PYTIME("hello,get_time", `20160101 212223`)
```

外部 Python スクリプト:

```
# hello.py content
日付時刻 インポート時刻から
日付時刻 インポート日付から

def get_time(timestamp):
    return timestamp.time();
```

RAND() 関数

指定した境界内にあるランダム数を返します。

構文

```
RAND( 数値 )
```

パラメーター

名前	型	説明
数値	数値	<p>乱数の数値境界。</p> <p>小数点以下の桁数を持つ数値を指定した場合には、生成される乱数は、数値と同じ小数点以下の桁数を持つこととなります。</p> <ul style="list-style-type: none">○ 正の数値 -を入力した場合には、返される乱数はゼロ以上、指定した数値未満となります。 <p>0 ~ 99 の数値が返されます。</p> <pre>RAND(100)</pre> <ul style="list-style-type: none">○ 負の数値 -を入力した場合には、返されるランダム数は指定した数値以上、ゼロ未満となります。 <p>-1 ~ 100 の数値が返されます。</p> <pre>RAND(-100)</pre>

出力

数値。

例

基本的な例

278.61 が返されます。

```
RAND(1000.00)
```

3781 が返されます。

```
RAND(10000)
```

メモ

戻り値は関数の実行ごとに異なります。

備考

RAND() では結果を複製できない

RAND() 関数を同じ数値に対して2度続けて使用した場合、異なる結果が返されます。RAND() 関数には、RANDOM コマンドと違ってシード値がありません。

重複する乱数は生成される可能性あり

RAND() を使用して、乱数をテーブルのすべてのレコードに割り当てる演算フィールドを作成する場合は、重複する乱数が生成されることがあります。乱数が一意である保証はありません。

テーブルのレコード数に対する数値の比率が大きいほど、生成される数が一意である確率が高くなります。

乱数は動的に更新される

簡易並べ替えを実行したり、フィルターを適用したり、列の配置を変えたり、ビューをスクロールしたりするような操作を実行するたびに、RAND() 関数の演算フィールドにより新しい乱数セットが生成されます。

乱数セットを固定するには、**抽出**]ダイアログボックスで **ビュー**]または **フィールド**]オプションを使用して、新しいテーブルにそのデータを抽出します。

RATE() 関数

期間ごとの利率を返します。

構文

```
RATE(期間, 支払金額, 金額)
```

パラメーター

名前	型	説明
期間	数値	支払期間の総数。
支払金額	数値	1 期間あたりの支払金額。
金額	数値	貸付金の元金。

メモ

RATE() 関数では支払は期末に行われることが前提となっています。

出力

数値。利率は小数点以下 8 桁で計算されます。

例

基本的な例

\$275,000 の 25 年ローンに対して毎月 \$1856.82 を支払う場合の月利率として、0.00541667(0.54%) が返されます。

```
RATE(12*25, 1856.82, 275000)
```

同じローンの年利率として 0.06500004(6.5%) が返されます。

```
RATE(12*25, 1856.82, 275000)*12
```

高度な例

名目利率から実効利率を求める

RATE() 関数では名目利率が計算されます。RATE() の結果から実効利率を求めるには、EFFECTIVE() 関数を使用します。

前記例のローンの実効利率として 0.06715155 (6.7%) が返されます。

```
EFFECTIVE((RATE(12*25, 1856.82, 275000)*12), 12*25)
```

年金の計算

年金の計算では、次の4つの変数が使用されます。

- 現在価値または将来価値 - 下の例では \$21,243.39 と \$ 26,973.46
- 1 期間あたりの支払金額 - 下の例では \$1,000.00
- 1 期間あたりの利率 - 下の例では月あたり 1%
- 期間の数 - 下の例では 24 か月

これらの変数のうち3つの値がわかっている場合は、Analytics 関数を使って残りの変数の値を計算できます。

求めたい値:	使用する Analytics 関数
現在価値	PVANNUITY() 21243.39 を返す: <pre>FVANNUITY(0.01, 12, 1000)</pre>
将来価値	FVANNUITY() 26973.46 を返す: <pre>FVANNUITY(0.01, 24, 1000)</pre>
1 期間あたりの支払金額	PMT()

求めたい値:	使用する Analytics 関数
	1000 を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">PMT(0.01, 24, 21243.39)</div>
1 期間あたりの利率	RATE() 0.00999999(1%) を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">RATE(24, 1000, 21243.39)</div>
期間の数	NPER() 24.00 を返す: <div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;">NPER(0.01, 1000, 21243.39)</div>

年金の式

期末年金(期末払い)の**現在価値**を計算する式:

$$PV_A = Pmt \left[\frac{1 - \frac{1}{(1+i)^N}}{i} \right]$$

$$21243.39 = 1000 \left[\frac{1 - \frac{1}{(1+0.01)^{24}}}{0.01} \right]$$

期末年金(期末払い)の**将来価値**を計算する式:

$$FV_A = Pmt \left[\frac{(1+i)^N - 1}{i} \right]$$

$$26973.46 = 1000 \left[\frac{(1+0.01)^{24} - 1}{0.01} \right]$$

RDATE() 関数

R の関数またはスクリプトによって計算された日付値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RDATE(rScript/rCode <,フィールド/値 <,...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- \"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
フィールド/値 <,...n> 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で「 <code>value1, value2 ... valueN</code> 」を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「 <code>ALLTRIM()</code> 関数」ページ 2163 を参照してください。

出力

日付時刻。

例

基本的な例

`20160530` が返されます。

```
RDATE("as.Date(value1, '%m-%d-%Y')", "05-30-16")
```

高度な例

外部 R スクリプトを使用する

文字列から変換された日付が返されます。

```
RDATE("a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]",  
dateText)
```

外部 R スクリプト (sample.r) :

```
dateForm <- function(dateText) {  
  return(as.Date(dateText,format='%y%m%d'))  
}  
dateForm(value1)
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダーの `ac1r1ang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

RDATETIME() 関数

R の関数またはスクリプトによって計算された日付時刻値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RDATETIME(rScript/rCode <,フィールド/値 <,...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- \"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
フィールド/値 <,...n> 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で「 <code>value1, value2 ... valueN</code> 」を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「 <code>ALLTRIM()</code> 関数」ページ 2163 を参照してください。

出力

日付時刻。

例

基本的な例

現在の日付と時刻に45分が加算されます。

```
RDATETIME("Sys.time() + value1",2700)
```

高度な例

外部 R スクリプトを使用する

次の例では、外部 R 関数にフィールドとリテラル値を渡すことで、45分の日付時刻フィールドに加算されます。

```
RDATETIME("a<-'c:\\scripts\\sample.r');a[[1]]", start_date, 2700)
```

外部 R スクリプト (sample.r) :

```
add_time <- function(start, sec) {  
  return(start + sec)  
}  
add_time(value1, value2)
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダーの `ac1r1ang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

システム タイムゾーン

グリニッジ標準時 (GMT) は、Analytics で使用される R 環境のデフォルトの現在のタイムゾーンです。

RECLLEN() 関数

現在のレコードの長さを返します。

構文

```
RECLLEN( )
```

パラメーター

この関数にはパラメーターはありません。

出力

数値。

例

基本的な例

次の例では、長さが 110 であるすべてのレコードが抽出されます。

```
EXTRACT RECORD IF RECLLEN( ) = 110 TO "Extract.fil"
```

備考

特定の長さのレコードを識別したり、予測より短いレコードを検査したりするには、RECLLEN() 関数を使用します。この関数はレコード長を簡単に調べられる方法を提供するので、印刷イメージ(レポート)ファイルで作業している場合に有用です。

- 固定長のレコードでは、戻り値は一定(レコード長)です。
- 可変長のレコードでは、レコードによって戻り値は異なります。

RECNO() 関数

現在のレコード番号を返します。

構文

```
RECNO( )
```

パラメーター

この関数にはパラメーターはありません。

出力

数値。

例

基本的な例

次の例は、番号 10 から 20 のレコードを新しい Analytics テーブルに抽出します。

```
EXTRACT RECORD IF BETWEEN(RECNO( ),10,20) TO "Subset.fil"
```

備考

RECNO() 関数を使用すると、レコード番号をテーブルに出力したり、特定のレコードのテーブル内の相対的な位置を確認できます。

インデックス付きのテーブルとインデックス付きでないテーブル

この関数は現在の論理レコード番号を返します。

- テーブルにインデックスが付いていない場合、RECNO() 関数は値 1 から始まり、テーブル内のレコードごとに 1 ずつ増えていきます。この場合、論理レコード番号と物理レコード番号は同じです。
- テーブルにインデックスが付いている場合も RECNO() 関数による操作は変わりませんが、物理的ではなく論理的な順番でレコードをカウントします。

SEEK または FIND コマンドを実行する

SEEK または FIND コマンドを実行すると、レコード番号は 1 にリセットされます。

レコードを並べ替える

テーブル内のレコードを並べ替えても、RECNO() によって生成されたレコード番号の順序は変わりません。レコードに元々関連付けられていたレコード番号を保持するには、レコードを並べ替える前に、**[フィールド]** オプションを使用してこのデータを新しいテーブルに抽出してください。

RECOFFSET() 関数

現在のレコードを基準にして、指定された n 番目にあるレコードのフィールド値を返します。

構文

```
RECOFFSET(フィールド, レコードの数)
```

パラメーター

名前	型	説明
フィールド	文字 数値 日付時刻	値を取得するフィールドの名前。
レコードの数	数値	現在のレコードから数えたレコード数。現在のレコードより後のレコードを指定する場合は正の数、現在のレコードより前のレコードを指定する場合は負の数にします。

出力

文字、数値、または日付時刻。戻り値は、入力のフィールドパラメーターと同じデータカテゴリに属します。

例

基本的な例

次のレコードの *Amount* 値が返されます。

```
RECOFFSET(Amount,1)
```

前のレコードの *Amount* 値が返されます。

```
RECOFFSET(Amount, -1)
```

高度な例

演算フィールド内で RECOFFSET を使用する

演算フィールド *Next_Amount* には、次のレコードが同じ顧客番号である場合に限り、次のレコードの *Amount* フィールドの値が表示されます。

この演算フィールドをスクリプトで定義するには、次の構文を使用します。

```
DEFINE FIELD Next_Amount COMPUTED  
RECOFFSET(Amount,1) IF RECOFFSET(Customer,1) = Customer  
0
```

次のレコードの顧客番号が現在のレコードの顧客番号と同じ場合に限り、*Next_Amount* は次のレコードの *Amount* フィールドの値になります。そうでない場合、*Next_Amount* には値ゼロが割り当てられます。

備考

RECOFFSET() 関数は現在のレコードを基準にして、指定された前後 n 番目にあるレコードのフィールド値を返します。

RECOFFSET() の使用に適する場面

この関数は高度な比較テストを行う場合によく使用されます。

この関数は、現在のレコードのフィールドと別のレコードのフィールドで値を比較するために使用できます。たとえば、現在のレコードの金額と前のレコードの金額の差額を計算する演算フィールドを追加する場合などに使用します。

テーブルの開始または終了位置

この関数では、フィールドにテーブルの開始または終了位置を指定した場合、数値フィールドに対してはゼロ、文字フィールドに対しては空白の文字列、日付フィールドに対しては 1900/01/01 が返されます。現在のレコードと比較するレコードがそれ以上存在しないため、このような場合には、関数は空白の出力を返します。

REGEXFIND() 関数

正規表現で指定されたパターンが文字列内に現れるかどうかを示す論理値を返します。

構文

```
REGEXFIND(文字列, パターン)
```

パラメーター

名前	型	説明
文字列	文字	パターンとの一致をテストするフィールド、式、またはリテラル値。
パターン	文字	検索するパターン文字列(正規表現)。 パターンには、リテラル文字、メタ文字、またはこれら2つの組み合わせを指定できます。リテラル文字には、すべての英数字、一部の句読点、および空白が含まれます。 検索は大文字と小文字を区別します。つまり、英字の大文字と小文字は明確に指定しなければならないということです。

出力

論理。指定されたパターンの値が見つかった場合はT(true)、そうでない場合はF(false)を返します。

例

基本的な例

英字のパターン

次の例では、Vendor_City フィールドの値が"Phoenix"、"Austin"、または"Los Angeles"であるすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Vendor_City, "Phoenix|Austin|Los Angeles")
```

次の例では、ラストネームが"John"または"Jon"で始まるすべてのレコードに対し、Tが返されます。たとえば、Jon、Johnson、Johnston、Jonson、Jonston、Jonesなどが該当します。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Last_Name, "^Joh?n")
```

次の例では、ラストネームが"John"または"Jon"であるすべてのレコードに対してのみ、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Last_Name, "^Joh?n\b")
```

数値文字のパターン

次の例では、請求書番号に"98"を含んでいるすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "98")
```

次の例では、請求書番号が"98"で始まるすべてのレコードに対し、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "\b98")
```

次の例では、請求書番号が"98"で終わるすべてのレコードに対し、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "98\b")
```

次の例では、請求書番号の左端から5桁目と6桁目に"98"を含んでいるすべてのレコードに対し、'T'が返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "\b\d\d\d\d98")
```

```
REGEXFIND(Invoice_Number, "\b\d{4}98")
```

文字が混在するパターン

次の例では、製品コードが3つの数字、1つのハイフン、6つの文字の順になっているすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Product_Code, "\b\d{3}-[a-zA-Z]{6}\b")
```

次の例では、製品コードが3つ以上の数字、1つのハイフン、6つ以上の文字の順になっているすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Product_Code, "\b\d{3,}-[a-zA-Z]{6}")
```

次の例では、英数字の請求書番号の左端から5桁目と6桁目に"98"を含んでいるすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "\b\w{4}98")
```

次の例では、請求書番号に次の文字をすべて含んでいるすべてのレコードに対し、Tが返されます。それ以外のレコードに対してはFが返されます。

- 左端から4桁目までに任意の文字
- 左端から5桁目と6桁目に"98"

```
REGEXFIND(Invoice_Number, "\b.{4}98")
```

次の例では、請求書番号に"98"を含んでおり、その前に1～4文字があるすべてのレコードに対し、Tが返されます。それ以外のレコードに対しては'F'が返されます。

```
REGEXFIND(Invoice_Number, "\b.{1,4}98")
```

次の例では、請求書番号に次の文字をすべて含んでいるすべてのレコードに対し、Tが返されます。それ以外のレコードに対してはFが返されます。

- 左端から3桁目までに任意の文字
- 左端から4桁目に"5"または"6"
- 左端から5桁目と6桁目に"98"

```
REGEXFIND(Invoice_Number, "\b.{3}[56]98")
```

次の例では、請求書番号の次の文字をすべて含んでいるすべてのレコードに対し、Tが返されます。それ以外のレコードに対してはFが返されます。

- 左端から2桁目までに任意の文字
- 左端から3桁目と4桁目に"55"または"56"
- 左端から5桁目と6桁目に"98"

```
REGEXFIND(Invoice_Number, "\b.{2}(55|56)98")
```

備考

機能の仕組み

REGEXFIND() 関数は正規表現を使って Analytics でデータを検索します。

正規表現は、さまざまな検索操作を実行する特殊文字であるメタ文字と、リテラル文字を組み合わせた、強力で柔軟性のある検索文字列です。

例:

```
REGEXFIND(Last_Name, "Sm(i|y)the{0,1}")
```

これはグループ `()`、交替 `|`、量指定子 `{ }` のメタ文字を使用して、**Last_Name** フィールド内から "Smith"、"Smyth"、"Smithe"、または "Smythe" を見つけるための正規表現を作成しています。

照合は順次実行される

文字列の値とパターンの値との照合は、順次実行されます。上記の例:

- "S" が **Last_Name** フィールドの左端から1桁目と照合される
- "m" が2桁目と照合される
- "i" と "y" が3桁目と照合される
- "t" が4桁目と照合される
- "h" が5桁目と照合される
- "e" が6桁目と照合される(照合対象に6桁目が存在する場合)

REGEXFIND() の使用に適する場面

REGEXFIND() は、シンプルまたは複雑なパターン一致を使ってデータを検索する場合に使用できます。

正規表現の構文を見るのは初めてである場合は特に、正規表現の作成は難しいかもしれません。より単純な FIND()、MATCH()、または MAP() などの Analytics 検索関数を使用して、検索の目的を達成できる可能性もあります。

検索要件がこれらの単純な関数の能力を超えている場合は、正規表現を使用することで、検索文字列の作成にほぼ無制限の柔軟性を与えられます。

REGEXFIND() によるスペースの扱い

スペース(空白)は、文字列とパターンのどちらに指定した場合でも文字として扱われるので、スペースを扱う際には注意が必要です。

パターンに空白を空白そのものとして指定するには、スペースを入力するか、またはメタ文字の `\s` を使用します。メタ文字を使用すると、特に複雑なパターンを指定する場合に、パターン内のスペースが読みやすくなるため、見落とされにくくなります。

フィールドの連結

同時に複数のフィールドにわたって検索したい場合は、2つ以上のフィールドを連結して文字列に指定することができます。

例:

```
REGEXFIND(Vendor_Name+Vendor_Street,"Hardware.*Main")
```

Vendor_Name と **Vendor_Street** の両方のフィールドで、"Hardware" と "Main" という単語間に0個以上の文字が入っているパターンを検索します。

名前に "Hardware" という単語を含み、"Main" という通り沿いに立地する企業は、この正規表現に一致します。"Hardware on Main" という企業(名)は一致します。

連結したフィールドは1つのフィールドのように扱われます。ただし、`ALLTRIM()` 関数を使用して個々のフィールドからスペースを除去しなければ、各フィールドの先頭と末尾のスペースを含むフィールドとなります。

連結するフィールド間の順序は重要

`REGEXFIND()` は、パターン内の文字を指定された順序で検索するため、フィールドを連結する順序は検索に影響を与えます。上の式で **Vendor_Name** と **Vendor_Street** を逆にした場合、得られる結果が少なくなる可能性があります。

正規表現のメタ文字

次の表は、`REGEXFIND()` および `REGEXREPLACE()` で使用できるメタ文字の一覧と、それぞれのメタ文字で実行される操作を示しています。

このほかの正規表現構文もあって、Analytics でサポートされていますが、もっと複雑です。それらの構文の完全な説明はこのガイドには記載されていません。正規表現について説明している多数のリソースがインターネットで入手できます。

Analytics は、**ECMAScript** の正規表現の実装を使用しています。ほとんどの正規表現の実装は、共通のコア構文を使用しています。

メモ

Analytics における現在の正規表現の実装は、英語以外の言語の検索を完全にサポートしていません。

メタ文字	説明
.	任意の1文字に一致します(改行文字を除く)。
?	直前のリテラル、メタ文字、または要素と0回または1回一致します。
*	直前のリテラル、メタ文字、または要素と0回以上一致します。
+	直前のリテラル、メタ文字、または要素と1回以上一致します。
{}	直前のリテラル、メタ文字、または要素と指定した回数一致します。正確な数、範囲、または終わりが決められていない範囲を指定できます。 例: <ul style="list-style-type: none"> ○ a{3} は "aaa" と一致します。 ○ X{0,2}L は "L"、"XL"、および "XXL" と一致します。 ○ AB\d{2,-}YZ は、英数字の識別子で、接頭辞に "AB-"、接尾辞に "-YZ"、および本体部分に2つ以上の数字を持つすべての識別子と一致します。
[]	角かっこで囲まれた文字の中のいずれかに一致します。 例: <ul style="list-style-type: none"> ○ [aeiou] は a、e、i、o、または u と一致します。 ○ [^aeiou] は a、e、i、o、または u 以外の文字と一致します。 ○ [A-G] は大文字の英字 A から G の範囲にある任意の文字と一致します。 ○ [A-Ga-g] は大文字の英字 A から G、または小文字の英字 a から g の範囲にある任意の文字と一致します。 ○ [5-9] は 5 から 9 の範囲にある任意の数字と一致します。
()	文字の並びまたはブロックを定義するグループを作成します。その後、グループは単一ユニットとして扱うことができます。 例: <ul style="list-style-type: none"> ○ S(ch)?mid?th?これは、"Smith" または "Schmidt" と一致します。 ○ (56A.*){2} は、"56A" の並びが少なくとも2回出現するすべての英数字の識別子と一致します。 ○ (56A)*.*\1 は、"56A" の並びが少なくとも2回出現し、2つの出現の間にハイフンがあるすべての英数字の識別子と一致します。
\	エスケープ文字は、直後の文字がリテラルであることを示します。メタ文字と文字どおり一致させたい場合は、エスケープ文字を使用します。たとえば、\ (は左かっこを見つけ、\ \\ はバックスラッシュを見つけます。 次の文字のいずれかと完全に一致させたい場合は、エスケープ文字を使用します。 ^\$. * + ? = ! : \ () [] { } アンパサンド (&) やアットマーク (@) などの他の句読点文字は、エスケープ文字を必要としません。

メタ文字	説明
\int	<p>前にかっこ () を使って定義したグループを繰り返し使用することを指定します。 <i>int</i> は、前に定義したグループの、その他のグループからの位置 (先頭から数える) を特定する整数です。このメタ文字は、REGEXFIND() と REGEXREPLACE() いずれのパターンパラメーターでも使用できます。</p> <p>例:</p> <ul style="list-style-type: none"> ◦ (123).*\1 は、数字 "123" のグループが少なくとも 2 回出現するすべての識別子と一致します。 ◦ ^(\d{3})*\1 は、先頭の 3 桁の数字が繰り返し現れるすべての識別子と一致します。 ◦ ^(\d{3})*\1.*\1 は、先頭の 3 桁の数字が繰り返し少なくとも 2 回現れるすべての識別子と一致します。 ◦ ^(\D)(\d)*\2\1 は、先頭の英字と数字が反転して再び現れるすべての識別子と一致します。
\$int	<p>ターゲット文字列で見つかったグループが置換文字列として使用されることを指定します。 <i>int</i> は、対象文字列における、そのグループの、その他のグループからの位置 (先頭から数える) を特定する整数です。このメタ文字は、REGEXREPLACE() の新しい文字列パラメーターで使用できます。</p> <p>例:</p> <ul style="list-style-type: none"> ◦ さまざまな電話番号の表示形式と一致させるために (\d{3})[-]?(\d{3})[-]?(\d{4}) というパターンを使用している場合、新しい文字列として (\$1)-\$2-\$3 を使用すると、それ自体を数字と置き換えて、表示形式を標準化することができます。999 123-4567 と 9991234567 はどちらも (999)-123-4567 になります。
	<p>パイプ () の前後にある文字、文字のブロック、または式と一致します。</p> <p>例:</p> <ul style="list-style-type: none"> ◦ a b は a または b と一致します。 ◦ abc def は "abc" または "def" と一致します。 ◦ Sm(il)yth は "Smith" または "Smyth" と一致します。 ◦ [a-c][Q-S][x-z] は a、b、c、Q、R、S、x、y、z のいずれかの文字と一致します。 ◦ \s - は空白文字またはハイフンと一致します。
\w	<p>単語に使用される任意の文字と一致します (a から z、A から Z、0 から 9、およびアンダースコア _)。</p>
\W	<p>単語に使用される文字以外の任意の文字と一致します (a から z、A から Z、0 から 9、またはアンダースコア _ 以外)。</p>
\d	<p>任意の数字 (10 進数字) と一致します。</p>
\D	<p>10 進数字以外の任意の 1 文字と一致します。</p>
\s	<p>スペース (空白文字) と一致します。</p>
\S	<p>空白文字以外の任意の文字と一致します。</p>
\b	<p>単語の境界 (\w 文字と \W 文字の間) と一致します。</p> <p>単語の境界自体に領域は取りません。例:</p> <ul style="list-style-type: none"> ◦ "United Equipment" には単語の境界が 4 つあります。スペースの左右に 1 つずつ、文字列のはじめと終わりに 1 つずつです。"United Equipment" は、正規表現 \b\w*\b\W\b\w*\b と一致します。

メタ文字	説明
	<p>ヒント</p> <p>単語の境界は、単語を区切るスペース以外の、カンマやピリオドなどの文字によっても区切ることができます。</p> <p>たとえば、次の式は True と評価されます。</p> <pre>REGEXFIND("jsmith@example.net", "\bexample\b")</pre>
^	<p>文字列の先頭と一致します。</p> <p>角かっこ [] 内では、^ は指定内容の否定になります。</p>
\$	<p>文字列の末尾と一致します。</p>

関連する関数

一致パターンを検索、置換したい場合は、"REGEXREPLACE() 関数" 次のページを使用します。

REGEXREPLACE() 関数

正規表現と一致する文字列のすべてのインスタンスを新しい文字列で置き換えます。

構文

```
REGEXREPLACE(文字列, パターン, 新しい文字列)
```

パラメーター

名前	型	説明
文字列	文字	パターンとの一致をテストするフィールド、式、またはリテラル値。
パターン	文字	検索するパターン文字列(正規表現)。 パターンには、リテラル文字、メタ文字、またはこれら2つの組み合わせを指定できます。リテラル文字には、すべての英数字、一部の句読点、および空白が含まれます。 検索は大文字と小文字を区別します。つまり、英字の大文字と小文字は明確に指定しなければならないということです。
新しい文字列	文字	パターンと一致するすべての値と置き換える文字列。 置換文字列には、リテラル文字、元の文字列に基づく文字のグループ(<code>\$int</code> 要素を使用)、またはこれら2つの組み合わせを含めることができます。

出力

文字。

例

基本的な例

スペースを操作する

次の例では、文字列間の複数のスペースを単一スペースに置換することで、"AB CD EF" が返されます。

```
REGEXREPLACE("AB CD EF", "\s+", " ")
```

文字フィールドのデータに対し、単語間の間隔を単一スペースに統一したデータが返されます。

```
REGEXREPLACE(文字フィールド, "\s+", " ")
```

文字フィールドのデータに対し、単語間の間隔を単一スペースに統一したデータが返されます。新しい文字列としてリテラルスペースの代わりにBLANKS()関数を使用すると、スペースが視認しやすくなり、見落とされる可能性が低くなります。

```
REGEXREPLACE(character_field, "\s+", BLANKS(1))
```

電話番号を標準的な書式にする

"(123) 456-7890" が返されます。つまり、電話番号 '1234567890' が標準的な書式に置き換えられました。

```
REGEXREPLACE(SUBSTR("1234567890",1,14), "(\d{3})[\s-]*(\d{3})[\s-]*(\d{4})", "($1) $2-$3")
```

Telephone_Number フィールド内の数字を標準的な書式にした値が返されます。

```
REGEXREPLACE(Telephone_Number, ".*(\d{3})[\s-\.\.]*(\d{3})[\s-\.\.]*(\d{4})", "($1) $2-$3")
```

テキストから "123-456-7890" が抽出されます。

関数

```
REGEXREPLACE("Tel num: 123-456-7890 (office)", "(.*)(\d{3}[\s-]\.)*\d{3}[\s-\.]*\d{4})(.*)", "$2")
```

Comment フィールドのテキスト内から電話番号を抽出し、標準的な書式に置き換えます。

```
REGEXREPLACE(Comment, "(.*)(\d{3}[\s-]\.)*(\d{3}[\s-\.]*\d{4})(.*)", "( $2) $3-$4")
```

一般的な書式を抽出する

文字列("1ABC-123aa")によって指定された値の一般的な書式を示す"9XXX-999xx"が返されます。

```
REGEXREPLACE(REGEXREPLACE(REGEXREPLACE("1ABC-123aa", "\d", "9"), "[a-z]", "x"), "[A-Z]", "X")
```

Invoice_Number フィールド内のすべての識別子の一般的な書式が返されます。

```
REGEXREPLACE(REGEXREPLACE(REGEXREPLACE(Invoice_Number, "\d", "9"), "[a-z]", "x"), "[A-Z]", "X")
```

氏名を標準的な書式に置き換える

"John David Smith" が返されます。

```
REGEXREPLACE("Smith, John David", "^(\\w+), (\\s\\w+)(\\s\\w+)?(\\s\\w+)?", "$2$3$4 $1")
```

Full_Name フィールド内の氏名の構成要素が正式な順序である「ファースト (ミドル) (ミドル) ラスト」として返されます。

```
REGEXREPLACE(Full_Name, "^(\\w+), (\\s\\w+)(\\s\\w+)?(\\s\\w+)?", "$2$3$4 $1")
```

メモ

氏名のデータには、アポストロフィが入った名前など、さまざまな要素の複雑な結合が存在していることがあります。氏名のデータにおける変化が多ければ、必要となる正規表現はこの例で提供されているものより通常、複雑になります。

HTML マークアップの削除

"https://www.flgov.com/wp-content/uploads/orders/2020/EO_20-166.pdf" を返します。

```
REGEXREPLACE("<a href='https://www.flgov.com/wp-content/uploads/orders/2020/EO_20-166.pdf' target='blank'>https://www.flgov.com/wp-content/uploads/orders/2020/EO_20-166.pdf</a>", "<[^>]*>", ' ')
```

HTML マークアップが削除された **Source_URL_Link** フィールドのハイパーリンクを返します。

```
REGEXREPLACE(Source_URL_Link, "<[^>]*>", ' ')
```

備考

機能の仕組み

REGEXREPLACE() 関数では、正規表現を使って、データ内で一致するパターンを見つけ、その一致する値を新しい文字列で置き換えることができます。

例：

```
REGEXREPLACE(文字フィールド, "\s+", " ")
```

これは、テキスト文字の間にある 1 つ以上のスペースを単一のスペースに置き換えることにより、文字データにおける間隔を標準化しています。

REGEXREPLACE() 関数の検索部分は、REGEXFIND() 関数の検索部分と同じです。両方の関数に共通する検索機能の詳細については、"REGEXFIND() 関数" ページ 2447 を参照してください。

REGEXREPLACE() の使用に適する場面

REGEXREPLACE() は、Analytics でシンプルまたは複雑なパターン一致を使ってデータを検索、置換したい場合に使用できます。

文字列をそれ自体に置換する

`$int` 要素を使用すると、文字列をそれ自体に置換できます。これにより、意味のあるデータ部分を維持しながら、周囲のデータや混合されたデータを標準化または省略することが可能になります。

上には、電話番号と名前を使用するいくつかの例が示されています。

`$int` 要素を使用するには、まずパターン値をカッコ () で囲むことでグループを作成する必要があります。詳細については、"REGEXFIND() 関数" ページ 2447を参照してください。

文字の順次の照合を避ける

REGEXREPLACE() 関数を入れ子にすることで、文字の順次の照合を避け、他の部分文字列との位置関係にかかわらず、部分文字列を置き換えることができます。

次の2つの例における課題は、数字と文字がランダムな順序で出現している英数字のソースデータから一般的な書式を抽出することです。このように数字と文字の順序が不明な場合、どのようにしてパターン文字列を指定できるでしょうか？

解決策は、以下に例を示すように、まず内側のREGEXREPLACE() 関数を使って数字を検索、置換し、次に外側のREGEXREPLACE() 関数を使って文字を検索、置換することです。

次の例の場合には、"999XXX" が返されます。

```
REGEXREPLACE(REGEXREPLACE("123ABC", "\d", "9"), "[A-Z]", "X")
```

次の例の場合には、"9X9X9X" が返されます。

```
REGEXREPLACE(REGEXREPLACE("1A2B3C", "\d", "9"), "[A-Z]", "X")
```

置換文字列の長さとの切り詰め

REGEXREPLACE() を使用して演算フィールドを作成した場合、演算フィールドの長さは元のフィールドの長さと同じになります。

置換文字列の長さがターゲット文字列の長さを超えており、文字列全体の長さが増えた場合、その増えた文字列の長さを演算フィールドの長さに収めることができないときには、文字列が切り詰められます。

まず、ターゲット文字列の末尾の文字から切り詰められ、次に置換文字列の末尾から切り詰められます。以下に切り詰めの例を示します。

文字列	パターン	新しい文字列	フィールド長	結果	切り詰め後の文字
x123x	123	A	5	xAx	なし
x123x	123	ABC	5	xABCx	なし
x123x	123	ABCD	5	xABCD	x
x123x	123	ABCDE	5	xABCD	x、E
x123x	123	ABCDE	6	xABCDE	x
x123x	123	ABCDE	7	xABCDEx	なし

切り詰めを避ける方法

切り詰められないようにするには SUBSTR() 関数を使用してフィールド長を大きくしておきます。次の2番目の例を参照してください。

置換文字列 "E" と既存の文字 "x" を切り捨てた "xABCD" が返されます。

```
REGEXREPLACE("x123x", "123", "ABCDE")
```

すべての置換文字列と、置換されなかった既存の文字列で構成される、"xABCDEx" が返されます。

```
REGEXREPLACE(SUBSTR("x123x", 1, 10), "123", "ABCDE")
```

正規表現のメタ文字

次の表は、REGEXFIND() および REGEXREPLACE() で使用できるメタ文字の一覧と、それぞれのメタ文字で実行される操作を示しています。

このほかの正規表現構文もあって、Analytics でサポートされていますが、もっと複雑です。それらの構文の完全な説明はこのガイドには記載されていません。正規表現について説明している多数のリソースがインターネットで入手できます。

Analytics は、ECMAScript の正規表現の実装を使用しています。ほとんどの正規表現の実装は、共通のコア構文を使用しています。

メモ

Analytics における現在の正規表現の実装は、英語以外の言語の検索を完全にサポートしていません。

メタ文字	説明
.	任意の1文字に一致します(改行文字を除く)。
?	直前のリテラル、メタ文字、または要素と0回または1回一致します。
*	直前のリテラル、メタ文字、または要素と0回以上一致します。
+	直前のリテラル、メタ文字、または要素と1回以上一致します。
{}	直前のリテラル、メタ文字、または要素と指定した回数一致します。正確な数、範囲、または終わりが決められていない範囲を指定できます。 例: <ul style="list-style-type: none"> <code>a{3}</code> は "aaa" と一致します。

メタ文字	説明
	<ul style="list-style-type: none"> ○ X{0,2}L は "L"、"XL"、および "XXL" と一致します。 ○ AB-\d{2,}-YZ は、英数字の識別子で、接頭辞に "AB-"、接尾辞に "-YZ"、および本体部分に2つ以上の数字を持つすべての識別子と一致します。
[]	<p>角かっこで囲まれた文字の中のいずれかに一致します。</p> <p>例：</p> <ul style="list-style-type: none"> ○ [aeiou] は a、e、i、o、または u と一致します。 ○ [^aeiou] は a、e、i、o、または u 以外の文字と一致します。 ○ [A-G] は大文字の英字 A から G の範囲にある任意の文字と一致します。 ○ [A-Ga-g] は大文字の英字 A から G、または小文字の英字 a から g の範囲にある任意の文字と一致します。 ○ [5-9] は 5 から 9 の範囲にある任意の数字と一致します。
()	<p>文字の並びまたはブロックを定義するグループを作成します。その後、グループは単一ユニットとして扱うことができます。</p> <p>例：</p> <ul style="list-style-type: none"> ○ S(ch)?mid?th?これは、"Smith" または "Schmidt" と一致します。 ○ (56A.*){2} は、"56A" の並びが少なくとも 2 回出現するすべての英数字の識別子と一致します。 ○ (56A).*.*\1 は、"56A" の並びが少なくとも 2 回出現し、2 つの出現の間にハイフンがあるすべての英数字の識別子と一致します。
\	<p>エスケープ文字は、直後の文字がリテラルであることを示します。メタ文字と文字どおり一致させたい場合は、エスケープ文字を使用します。たとえば、\(は左かっこを見つけ、\\ はバックスラッシュを見つけます。</p> <p>次の文字のいずれかと完全に一致させたい場合は、エスケープ文字を使用します。</p> <p>^\$. *+ ? = ! : \ () [] { }</p> <p>アンパサンド (&) やアットマーク (@) などの他の句読点文字は、エスケープ文字を必要としません。</p>
\int	<p>前にかっこ () を使って定義したグループを繰り返し使用することを指定します。<i>int</i> は、前に定義したグループの、その他のグループからの位置 (先頭から数える) を特定する整数です。このメタ文字は、REGEXFIND() と REGEXREPLACE() いずれのパターンパラメーターでも使用できます。</p> <p>例：</p> <ul style="list-style-type: none"> ○ (123).*\1 は、数字 "123" のグループが少なくとも 2 回出現するすべての識別子と一致します。 ○ ^(\d{3}).*\1 は、先頭の 3 桁の数字が繰り返し現れるすべての識別子と一致します。 ○ ^(\d{3}).*\1.*\1 は、先頭の 3 桁の数字が繰り返し少なくとも 2 回現れるすべての識別子と一致します。 ○ ^(\D\d)-.*\2\1 は、先頭の英字と数字が反転して再び現れるすべての識別子と一致します。
\$int	<p>ターゲット文字列で見つかったグループが置換文字列として使用されることを指定します。<i>int</i> は、対象文字列における、そのグループの、その他のグループからの位置 (先頭から数える) を特定する整数です。このメタ文字は、REGEXREPLACE() の新しい文字列パラメーターで使用できます。</p> <p>例：</p> <ul style="list-style-type: none"> ○ さまざまな電話番号の表示形式と一致させるために (\d{3})[-]?(\d{3})[-]?(\d{4}) というパターンを使用している場合、新しい文字列として (\$1)-\$2-\$3 を使用すると、それ自体を数字と置き換えて、表示形式を標準化

メタ文字	説明
	することができます。999 123-4567 と 9991234567 はどちらも (999)-123-4567 になります。
	パイプ()の前後にある文字、文字のブロック、または式と一致します。 例： <ul style="list-style-type: none"> ◦ a b は a または b と一致します。 ◦ abc def は "abc" または "def" と一致します。 ◦ Sm(i y)th は "Smith" または "Smyth" と一致します。 ◦ [a-c][Q-S][x-z] は a、b、c、Q、R、S、x、y、z のいずれかの文字と一致します。 ◦ \s は空白文字またはハイフンと一致します。
\w	単語に使用される任意の文字と一致します(a から z、A から Z、0 から 9、およびアンダースコア_)。
\W	単語に使用される文字以外の任意の文字と一致します(a から z、A から Z、0 から 9、またはアンダースコア_以外)。
\d	任意の数字(10進数字)と一致します。
\D	10進数字以外の任意の1文字と一致します。
\s	スペース(空白文字)と一致します。
\S	空白文字以外の任意の文字と一致します。
\b	単語の境界(\w文字と\W文字の間)と一致します。 単語の境界自体に領域は取りません。例： <ul style="list-style-type: none"> ◦ "United Equipment" には単語の境界が4つあります。スペースの左右に1つずつと、文字列のはじめと終わりに1つずつです。"United Equipment" は、正規表現 <code>\b\w*\b\W\b\w*\b</code> と一致します。 <p>ヒント 単語の境界は、単語を区切るスペース以外の、カンマやピリオドなどの文字によっても区切ることができます。 たとえば、次の式は True と評価されます。</p> <pre>REGEXFIND("jsmith@example.net", "\bexample\b")</pre>
^	文字列の先頭と一致します。 角かっこ[]内では、^は指定内容の否定になります。
\$	文字列の末尾と一致します。

関連する関数

一致パターンを置換せずに検索したいだけの場合は、"REGEXFIND() 関数" ページ 2447 を使用します。

REMOVE() 関数

指定した文字のみを含む文字列を返します。

構文

```
REMOVE(文字列, 有効な文字)
```

パラメーター

名前	型	説明
文字列	文字	文字を削除するフィールド、式、またはリテラル値。
有効な文字	文字	文字列のうち、保持する文字。 有効な文字の中に二重引用符を指定する場合は、文字のリスト全体を一重引用符で囲む必要があります。 例: <code>'"/'</code> メモ 指定する文字が文字列にない場合、戻り値に含まれません。

出力

文字。

例

基本的な例

"ABC123 " が返されます。

```
REMOVE("ABC 123 XX4", "ABC123")
```

関数

"ABC123XX " が返されます。

```
REMOVE("zABC 123 XX4", "ABCX123")
```

"1234 " が返されます。

```
REMOVE("ABC 123 XX4", "1234567890")
```

Product_Number フィールドのすべての値から数字以外のすべての文字を削除した値が返されます。

```
REMOVE(Product_Number, "0123456789")
```

備考

メモ

現在では、REMOVE 関数の代わりに INCLUDE() 関数と EXCLUDE() 関数が使われるようになりました。

しかし、旧バージョンとの互換性を保つために、現行バージョンの Analytics でも REMOVE() を使用できます。

機能の仕組み

REMOVE() 関数は文字データから不要な文字を取り除き、固定長文字列を返します。

REMOVE() の使用に適する場面

REMOVE() 関数は、住所フィールドなど、書式が統一されていないデータフィールドの書式を統一する場合に使用できます。また、編集が不十分なフィールドから句読点やその他の無効な文字を削除する場合にも使用できます。

さらに、SORT または JOIN コマンドの使用、重複の照合、あるいはレポートの出力を行う前にフィールド内のデータをクリーンにする場合にも、使用できます。

大文字と小文字の区別

REMOVE() 関数では大文字と小文字が区別されます。このため、有効な文字に "ID" を指定しても、これらの文字は "id#94022" には含まれていないことになります。大文字と小文字の両方が混在している可能性がある場合は、UPPER() 関数を使用して文字列を大文字に変換します。

例：

```
REMOVE(UPPER("id#94022"), "ID0123456789")
```

関連する関数

REMOVE() 関数は INCLUDE() 関数と似ていますが、次の点が異なります。

- REMOVE() では、削除された文字の代わりとして、空白が出力の末尾に追加されます。文字列の元の長さが保持されます。
- INCLUDE() では空白が追加されません。

REPEAT() 関数

指定された回数だけ部分文字列を繰り返す文字列を返します。

構文

```
REPEAT(文字列, 回数)
```

パラメーター

名前	型	説明
文字列	文字	繰り返すフィールド、式、またはリテラル値。
カウント	数値	文字列の値を繰り返す回数。

出力

文字。

例

基本的な例

"ABCABCABC" が返されます。

```
REPEAT("ABC", 3)
```

"000000000" が返されます。

```
REPEAT("0", 9)
```

備考

REPEAT() の使用に適する場面

REPEAT()関数は、変数を定数値や空白で初期化したり、演算フィールドにデフォルト値を設定したりする場合に使用することができます。

REPLACE() 関数

指定された文字列のすべてのインスタンスを新しい文字列で置き換えます。

構文

```
REPLACE(文字列, 元のテキスト, 新しいテキスト)
```

パラメーター

名前	型	説明
文字列	文字	文字が置き換えられる値。
元のテキスト	文字	置き換える文字。検索は大文字と小文字を区別します。
新しいテキスト	文字	元のテキストの値を置き換えるテキスト。

出力

文字。

例

基本的な例

"a12345efg" が返されます。

```
REPLACE("abcdefg", "bcd", "12345")
```

"Rd." が返されます。

```
REPLACE("Road", "Road", "Rd.")
```

"ac" が返されます。

```
REPLACE("abc", "b", "")
```

高度な例

指定された文字を削除する

REPLACE() 関数では、指定した文字列を空文字列()で置き換えることによって、指定文字列をソース文字列から削除することもできます。

"1234 Scott" が返されます。

```
REPLACE("1234 Scott rd.", "rd.", "")
```

フィールド長の調整

新しいテキスト("ABC")が元のテキスト("X")より長い場合は、置換後の文字列のフィールド長が置換結果を収めることができるように自動的に拡張されます。

フィールド長が3文字から5文字に拡張されることで、"9ABC9" が返されます。

```
REPLACE("9X9", "X", "ABC")
```

フィールド長は、2回目以降の置換では自動拡張されないため、すべての新しい文字を収めることができるほど十分に長くない場合には切り詰められます。

"9ABC9A" が返されます。

```
REPLACE("9X9", "X", "ABC")
```

切り詰められないようにするには、BLANKS() 関数またはリテラルスペースを使用して文字列の長さを大きくしておきます。

"9ABC9ABC" が返されます。

```
REPLACE("9X9X" + BLANKS(2), "X", "ABC")
```

```
REPLACE("9X9X" + " ", "X", "ABC")
```

結果の文字列が文字列より短くなる場合には、フィールド長を同じにするため、結果の文字列にスペースが追加されます。

"9X9 " が返されます。

```
REPLACE("9X9", "X", "ABC")
```

備考

機能の仕組み

REPLACE() 関数は既存の文字列のすべてのインスタンスを新しい文字列で置き換えます。

"1234 Scott Road" が返されます。

```
REPLACE("1234 Scott rd.", "rd.", "")
```

REPLACE() の使用に適する場面

REPLACE() 関数は、住所フィールドなどの書式が統一されていないデータフィールドを正規化する場合や、編集が不十分なフィールドの無効な文字を置換する場合に使用します。重複の検査や、テーブルの結合または関連付けなどの操作を正確に実行するためには、正規化または標準化された書式のデータを必要とします。

大文字と小文字の区別

REPLACE() 関数では大文字と小文字が区別されます。`"RD."` を元のテキストに指定する際、文字列内の値が小文字の場合は、一致が検出されないため、値は新しいテキストの値に置き換えられません。

文字列に大文字と小文字の両方が混在している可能性がある場合は、最初に UPPER() 関数を使用してすべての文字を大文字に変換します。

"1234 SCOTT ROAD" が返されます。

```
REPLACE(UPPER("1234 Scott rd."), "RD.", "ROAD")
```

不用意な置換が行われないようにする

REPLACE() 式を作成する際は、不用意な置換が行われないように、文字列内に存在し得る元のテキストのすべてのインスタンスを把握しておいてください。

"Richard" の最後の 2 文字が "rd" のため、"645 RichaRoad Road" が返されます。

```
REPLACE("645 Richard rd ", "rd", "Road")
```

元のテキストの値の前後にスペースを追加すると(" rd ")、名前に含まれる "rd" のインスタンスのように、前にスペースがないインスタンスは置換されないようになります。

"645 Richard Road" が返されます。

```
REPLACE("645 Richard rd ", "rd", "Road")
```

REVERSE() 関数

文字の順番を逆にした文字列を返します。

構文

```
REVERSE(文字列)
```

パラメーター

名前	型	説明
文字列	文字	順序を逆にするフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

"E DCBA" が返されます。

```
REVERSE("ABCD E")
```

RJUSTIFY() 関数

文字列の末尾の空白をすべて先頭に移動させ、指定された文字列と同じ長さの右寄せした文字列を返します。

構文

```
RJUSTIFY(文字列)
```

パラメーター

名前	型	説明
文字列	文字	右寄せにするフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

" ABC" が返されます。

```
RJUSTIFY("ABC ")
```

備考

RJUSTIFY() の使用に適する場面

RJUSTIFY() 関数は文字フィールドを右寄せする場合に使用します。

RLOGICAL() 関数

R の関数またはスクリプトによって計算された論理値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RLOGICAL(rScript/rCode <,フィールド/値 <,...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- \"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
フィールド/値 <,...n> 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で「 <code>value1, value2 ... valueN</code> 」を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「 <code>ALLTRIM()</code> 関数」ページ 2163 を参照してください。

出力

論理。

例

基本的な例

T が返されます。

```
RLOGICAL("(value1>0.6) & (value2>0.7) & (value3>0.5)", 0.8, 0.9, 0.55)
```

高度な例

外部 R スクリプトを使用する

金額と上下限值を入力として取ります。一連の論理比較に基づいて、真理値が返されます。

```
RLOGICAL("a<-'c:\\scripts\\sample.r';a[[1]]", expense_amt, threshold_low, threshold_hi)
```

外部 R スクリプト (sample.r) :

```
test_truth <- function(amt, low, hi) {  
  return(((amt > low) & (amt < hi)) | ((amt==low) | (amt==hi)))  
}  
test_truth(value1, value2, value3)
```

変数に格納された R コードの使用

次の例は、AND ロジックを使って 3 つのフィールドの論理テストを行っています。

```
v_rcode = "(value1>0.6) & (value2>0.7) & (value3>0.5)"  
RLOGICAL(v_rcode, PACKED, MICRO_LONG, ACCPAC)
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログ ファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダの `acrlang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

RNUMERIC() 関数

R の関数またはスクリプトによって計算された数値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RNUMERIC(rScript/rCode, 小数点以下の桁数 <,フィールド/値 <,...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- \"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
小数位	数値	戻り値に含める小数点以下の桁数。正の整数である必要があります。
フィールド/値 <,...n> 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で <code>[value1, value2 ... valueN]</code> を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、"ALLTRIM() 関数" ページ 2163 を参照してください。

出力

数値。

例

基本的な例

100 に対し、小数点以下の桁数を 10 にした 100.0000000000 が返されます。

```
RNUMERIC("print(value1)", 10, 100)
```

高度な例

変数への R コードの格納

100 に対し、小数点以下の桁数を 10 にした 100.0000000000 が返されます。

```
ASSIGN v_rcode = "print(value1)"  
RNUMERIC(v_rcode, 10, 100)
```

外部ファイルに書き込む

簡単な加算を行い、R 内で `sink` 関数を使って、`RNUMERIC()` 関数に付加したコメントをファイルに書き込みます。

```
RNUMERIC("foo<-function(x,y){x+y};attr(foo, 'comment') <- 'foo は簡単な  
加算を行います';sink('c:/temp/result.txt');attributes(foo);sink  
(NULL);foo(value1,value2)",0, amt, gross)
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログ ファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダーの `acrlang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

ROOT() 関数

数式の平方根を返します。

構文

```
ROOT(数値, 小数位)
```

パラメーター

名前	型	説明
数値	数値	平方根を計算する数式。 数値が負の数の場合は、ゼロが返されます。
小数位	数値	出力で使用する小数点以下桁数。

出力

数値。

例

基本的な例

10.00 が返されます。

```
ROOT(100, 2)
```

31.6228 が返されます。

```
ROOT(1000, 4)
```

備考

機能の仕組み

ROOT()関数は、数式やフィールド値の平方根を、指定された小数点以下の桁数で返します。この場合、返す値は適切に丸められます。

ROOT()の使用に適する場面

3乗根などその他のルート関数を実行する場合は、LOG()を使用してください。

ROUND() 関数

数値を丸めた整数を返します。

構文

```
ROUND(数値)
```

パラメーター

名前	型	説明
数値	数値	最も近い整数に丸める値。

出力

数値。

例

基本的な例

7 が返されます。

```
ROUND(7.2)
```

8 が返されます。

```
ROUND(7.5)
```

-8 が返されます。

```
ROUND(-7.5)
```

高度な例

通貨額を丸める

残高を最も近いドル金額に丸めたフィールドを作成するには、次のように指定します。

```
DEFINE FIELD Nearest_dollar_value COMPUTED ROUND(Balance)
```

備考

機能の仕組み

ROUND() は数値を最も近い整数に丸めた値を返します。

	正の値	負の値
次の整数に切り上げます	≥ 0.5	< 0.5
次の整数に切り下げます	< 0.5	≥ 0.5

小数点以下を特定の桁数で丸める

小数点以下を特定の桁数で丸める必要がある場合は、「DEC() 関数」ページ 2233 関数を使用してください。ROUND() 関数は、0 の小数位を指定した DEC() 関数と同じです。

```
ROUND(数値)
```

上記のコマンドは、次のコマンドと同等です。

```
DEC(数値, 0)
```

RSTRING() 関数

R の関数またはスクリプトによって計算された文字列の値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RSTRING(rScript/rCode, 長さ <, フィールド|値 <, ...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- "\"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
長さ	数値	返される文字列に割り当てられる長さ。
フィールド 値 <,... <i>n</i> > 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で <code>[value1, value2 ... valueN]</code> を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、"ALLTRIM() 関数" ページ 2163 を参照してください。

出力

文字。

例

基本的な例

"abc123" が返されます。

```
RSTRING("print(paste(value1,value2,sep=''))",6,"abc","123")
```

高度な例

外部 R スクリプトを使用する

xとyを連結し、スペース文字で区切って単一の文字列に格納します。

```
RSTRING("a<-source('./sample.r');a[[1]]",50, FirstName, LastName)
```

外部 R スクリプト (sample.r) :

```
conc <- function(x, y) {  
  paste(x, y, sep=" ")  
}  
print(conc(value1, value2))
```

変数に格納された R コードの使用

xとyを連結し、スペース文字で区切って単一の文字列に格納します。

```
ASSIGN v_script = "conc <- function(x, y){paste(x, y, sep=' ')};conc  
(value1, value2)"  
RSTRING(v_script, 50, FirstName, LastName)
```

テーブルの UUID を生成する R スクリプトを使用する

リザルトにアップロードする、例外から成るテーブルを作成しているとします。レコードごとに、一意であることが保証されている識別子が必要です。このようなフィールドを生成するには、R の **uuid** パッケージを使って、レコードごとに一意の主キー値を作成します。

```
EXTRACT RSTRING("uuid::UUIDgenerate()", 36) AS "id", first_name, last_name, birthdate TO export_table
```

ヒント

uuid パッケージをインストールするには、R.exe を開いて次のコマンドを実行します。

```
install.packages("uuid")
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダの `acrlang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

RTIME() 関数

R の関数またはスクリプトによって計算された時刻値を返します。R によるデータ処理は Analytics の外部で行われます。

構文

```
RTIME(rScript|rCode <,フィールド|値 <,...n>>)
```

パラメーター

名前	型	説明
<i>rScript</i> / <i>rCode</i>	文字	実行する R コードのスニペットまたは R スクリプトの絶対または相対パス。 外部ファイルを使用せずに直接 R コードを入力する場合、コード内では、前後の引用符文字はエスケープしても使用することはできません。 <ul style="list-style-type: none">有効 - <code>'var <- \"test\"'</code>無効 - <code>'var <- \"'test\"'</code>
<i>フィールド 値 <,...n></i> 省略可能	文字 数値 日付時刻 論理	R スクリプトまたは R コード スニペットの引数として使用するフィールド、式、リテラル値から成るリスト。 値は呼び出す関数に指定順に渡されます。また、値を参照するには、R コード内で「 <code>value1, value2 ... valueN</code> 」を使用します。 R コードの関数定義を満たすのに必要な数の引数を指定できます。 メモ 文字入力から先頭と末尾の空白を除去するには、 <code>ALLTRIM(str)</code> のように <code>ALLTRIM()</code> 関数を使用します。詳細については、「 <code>ALLTRIM()</code> 関数」ページ 2163 を参照してください。

出力

日付時刻。

例

基本的な例

`t0545` が返されます。

```
RTIME("value1+2700", `t0500`)
```

高度な例

外部 R スクリプトを使用する

次の例では、外部 R 関数にフィールドとリテラル値を渡すことで、45 分が時刻フィールドに加算されます。

```
RTIME("a<-source('c:\\scripts\\sample.r');a[[1]]", end_time, 2700)
```

外部 R スクリプト(sample.r) :

```
add_time <- function(start, sec) {  
  return(start + sec)  
}  
add_time(value1, value2)
```

備考

R からデータを返す

R スクリプトを呼び出す場合、`source` 関数を使用して、戻りオブジェクトを変数に割り当てます。次に、R 関数から返されて戻りオブジェクトに格納された値に、次のようにアクセスできます。

```
# 'a' はレスポンス オブジェクトを格納し、a[[1]] はデータ値にアクセスしています  
"a<-source('c:\\scripts\\r_scripts\\sample.r');a[[1]]"
```

R のログファイル

R 言語のメッセージは、Analytics によりプロジェクト フォルダーの `ac1r1lang.log` ファイルに記録されます。このファイルを使って、R のエラーをデバッグします。

システム タイムゾーン

グリニッジ標準時 (GMT) は、Analytics で使用される R 環境のデフォルトの現在のタイムゾーンです。

SECOND() 関数

指定された時刻または日付時刻から秒数を抽出し、それを数値として返します。

構文

```
SECOND(時刻/日付時刻)
```

パラメーター

名前	型	説明
時刻/日付時刻	日付時刻	秒数を抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

30 が返されます。

```
SECOND(`t235930`)
```

```
SECOND(`20141231 235930`)
```

Call_start_time フィールドの各値の秒数が返されます。

```
SECOND(Call_start_time)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

時刻/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような時刻書式または日付時刻書式でも使用することができます。

リテラル時刻または日付時刻値の指定

日付時刻にリテラルの時刻値または日付時刻値を指定する場合は、次の表内の書式に制限されません。また、``20141231 235959`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **時刻値** - 以下の表に示す任意の時刻の書式を使用することができます。関数が正しく動作するためには、単独の時刻値の前に区切り文字を使用する必要があります。有効な区切り文字は文字 't' または 'T' です。24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。
- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があり、かつ、2 つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 't' または 'T' です。

形式の例	リテラル値の例
thhmmss	`t235959`
Thhmm	`T2359`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`

形式の例	リテラル値の例
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

SHIFT() 関数

入力値の最初の文字のビットを左または右に移動させた1文字の文字列を返します。

構文

```
SHIFT(文字, 移動させるビット数)
```

パラメーター

名前	型	説明
文字	文字	ビットを移動する値。
移動させるビット数	数値	文字値を移動させるビット数を指定します。 <ul style="list-style-type: none">この値が正の場合 - 文字は左側に移動します。この値が負の場合 - 文字は右側に移動します。 ビット数が15より大きいか-15より小さい場合、結果はバイナリのゼロ、CHR(0)になります。

出力

文字。

例

基本的な例

文字 "X" または CHR(88) が返されます(00010110 が 01011000 になります)。

```
SHIFT(CHR(22), 2)
```

BS(後退) または CHR(8) が返されます(00010000 が 00001000 になります)。

関数

```
SHIFT(CHR(16), -1)
```

アクセント文字または CHR(96) が返されます(10011011 が 01100000 になります)。

```
SHIFT(CHR(155), 5)
```

備考

SHIFT() の使用に適する場面

SHIFT() 関数を BYTE() や CHR()、MASK() 関数と組み合わせて使用すると、レコード内の個々のビットを分離させたり移動させることができます。

SIN() 関数

ラジアン単位で表された角度のサインを、小数点以下 15 桁の精度で返します。

構文

```
SIN(ラジアン)
```

パラメーター

名前	型	説明
ラジアン	数値	ラジアン単位の角度。

出力

数値。

例

基本的な例

指定したラジアン数(30 度に相当)に対し、その正弦である 0.500000000000000 が返されます。

```
SIN(0.523598775598299)
```

30 度に対し、その正弦値である 0.500000000000000 が返されます。

```
SIN(30 * PI( )/180)
```

高度な例

入力値として度を使用する

30度の正弦を小数点第3位に丸めた0.500が返されます。

```
DEC(SIN(30 * PI( )/180),3)
```

備考

Mantissa Arc Test の実行

Analytics の3つの三角関数 SIN()、COS()、および TAN() は、ベンフォードの法則に関連する Mantissa Arc Test の実行をサポートします。

度のラジアンへの変換

入力値が度で表されている場合は、PI() 関数を使用して角度をラジアンに変換できます。(度 * PI()/180) = ラジアンとなります。必要に応じて、DEC() 関数を使用して、戻り値を丸めたり切り捨てたりすることができます。

SORTWORDS() 関数

連続する順序で並べ替えられた個別の単語を含む文字列を返します。

構文

```
SORTWORDS(文字列)
```

パラメーター

名前	型	説明
文字列	文字	<p>並べ替える単語を含む文字列または式。</p> <p>「単語」とは、任意の文字、またはスペース区切りの文字のグループです。</p> <ul style="list-style-type: none">○ 実際の単語または名前○ 文字データ型を使用する数値○ 略語や特定するプレフィックスなどの文字、数字、特殊文字のグループ <p>単語間の複数のスペースは自動的に1つのスペースに変換されます。先頭または末尾のスペースは自動的に切り取られます。</p>

出力

文字。

例

基本的な例

リテラル文字の入力値

"1 2 A Z a z" を返します(非 Unicode Analytics)

```
SORTWORDS("Z a 2 z A 1")
```

関数

"1 2 a A z Z" を返します(Unicode Analytics)

```
SORTWORDS("Z a 2 z A 1")
```

"1 2 A A Z Z" を返します

```
SORTWORDS(UPPER("Z a 2 z A 1"))
```

"CA, FL NY, TX," を返します

```
SORTWORDS("CA, TX, NY, FL")
```

"CA FL NY TX" を返します

```
SORTWORDS(OMIT("CA, TX, NY, FL", ","))
```

フィールドへの入力値

住所要素が連続する順序に並べ替えられた **Vendor_Address** フィールドのすべての値を返します。

```
SORTWORDS(Vendor_Address)
```

高度な例

住所要素を並べ替え、あいまい重複一致を改善する

Analytics で各種あいまい一致を実行するときには、SORTWORDS() をヘルパー関数として使用できます。

レーベンシュタイン距離に対する SORTWORDS() の効果

まず、形式が異なる2件の同じ住所でレーベンシュタイン距離を確認します。

SORTWORDS() 関数を使用しないと、返されるレーベンシュタイン距離は22です。そのような大きいレーベンシュタイン距離は、2つの文字列が同じ住所ではないことを示唆しています。

```
LEVDIST("125 SW 39TH ST, Suite 100", "Suite 100, 125 SW 39TH ST")
```

では、SORTWORDS() 関数を追加します。返されるレーベンシュタイン距離は2となり、大幅に小さくなります。これは2つの文字列が同じ住所であることを示しています。

```
LEVDIST(SORTWORDS("125 SW 39TH ST, Suite 100"), SORTWORDS("Suite 100, 125 SW 39TH ST"))
```

「125 SW 39TH ST, Suite 100」に対するあいまい重複の抽出

次の例は、「125 SW 39TH ST, Suite 100」との指定したレーベンシュタイン距離内にあるすべての "Vendor_Address" フィールド値を抽出するフィルターを作成します。

```
SET FILTER TO LEVDIST(SORTWORDS(Vendor_Address), SORTWORDS("125 SW 39TH ST, Suite 100"), F) < 3
```

```
SET FILTER TO ISFUZZYDUP(SORTWORDS(Vendor_Address), SORTWORDS("125 SW 39TH ST, Suite 100"), 3, 99)
```

式のレーベンシュタイン距離(③)を増減させると、フィルターされた値の距離の度合いを調整することができます。

レーベンシュタイン距離の詳細については、「LEVDIST() 関数」ページ 2335を参照してください。

備考

概要のビデオ

この関数の概要を説明する動画については、「[Fuzzy Matching Using SORTWORDS\(\)](#)」(英語のみ)をご覧ください。

SORTWORDS() で使用される並べ替え順序

SORTWORDS() 関数は、**並べ替え順序** オプション(**ツール > オプション > テーブル**)で指定された並べ順を使用します。デフォルトの並べ替え順序は次のとおりです。

詳細については、「**並べ替え順序** オプションとソート順序」ページ 1184を参照してください。

Analytics のエディション	並べ替え順]のデフォルト	関連付けられているソート順序
非 Unicode	システム デフォルト (ASCII)	数字、大文字、小文字、小文字: <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">0, 1, 2... A, B, C... a, b, c...</div> たとえば、"Z" は "a" の前にソートされます。
Unicode	混合言語 (UCA) (Unicode 照合アルゴリズム)	数字、小文字と大文字の混合: <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">0, 1, 2... a, A, b, B, c, C...</div> たとえば、"a" は "Z" の前にソートされます。

大文字と小文字の区別

SORTWORDS() は大文字と小文字を区別します。ご使用の Analytics のエディション(非 Unicode または Unicode)によって、文字列の大文字と小文字が並べ替えに影響する場合があります。

大文字と小文字が並べ替えに影響しないようにしたい場合は、SORTWORDS() と一緒に UPPER() 関数を使用できます。

```
SORTWORDS(UPPER("文字列"))
```

SORTWORDS() は曖昧一致を改善できます

SORTWORDS() は、曖昧一致コマンドのほか、曖昧一致関数を使用するフィルターや条件の効果を高めることができます。

- "FUZZYJOIN コマンド" ページ 1817
- "FUZZYDUP コマンド" ページ 1811
- "ISFUZZYDUP() 関数" ページ 2319
- "LEVDIST() 関数" ページ 2335
- "DICECOEFFICIENT() 関数" ページ 2238

注意

曖昧一致コマンドまたは関数と SORTWORDS() を使用する場合は、SORTWORDS() を比較対象の両方の文字列または両方のフィールドに適用する必要があります。2 つの文字列またはフィールドのいずれかにのみ関数を適用すると、曖昧一致の結果が大幅に低下することがあります。

レーベンシュタイン距離アルゴリズム

レーベンシュタイン距離アルゴリズムに基づくコマンドと関数を使用するときには、SORTWORDS() が特に有効です。この関数は、2つの文字列の同一または類似した要素が別の位置にあると、大きい相違値を返します。最初に要素を並べ替えると、レーベンシュタインアルゴリズムによって返される相違値を大幅に減らすことができます。結果セットの誤検出が少なくなり、真の検出が多くなります。

ダイス係数アルゴリズム

ダイス係数アルゴリズムに基づくコマンドや関数を使用するときには、SORTWORDS() が有効な場合があります。ただし、効果の改善がいつでも得られるわけではありません。設計上、ダイス係数アルゴリズムにより、要素の位置の重要度が最小化されるため、レーベンシュタイン距離アルゴリズムよりも要素の並べ替えの影響が少なくなります。

データの性質によっては、SORTWORDS() によって、ダイス係数アルゴリズムが低いスコアを返し、実際には効果が低下する場合があります。本番設定でダイス係数アルゴリズムと SORTWORDS() を使用するかどうかを決定する前に、サンプルデータのセットをテストしてください。

もう一つの考慮事項は、ダイス係数の n -gram 長さが短くなると、SORTWORDS() を使用する利点がそれほど大きくなるということです。 n -gram 長さを減らすと、ダイス係数アルゴリズムは要素の位置の重要度を大幅に最小化します。

SOUNDEX() 関数

ほかの文字列との発音の比較に使用できる、指定された文字列の soundex コードを返します。

構文

```
SOUNDEX(名前)
```

パラメーター

名前	型	説明
名前	文字	評価する文字式。

出力

文字。4 文字の soundex コードを返します。

例

基本的な例

発音は同じだが、綴りが異なる単語

次の2つの例では、綴りが異なっても発音が同じのため、同じ soundex コードが返されます。

F634 が返されます。

```
SOUNDEX("Fairdale")
```

F634 が返されます。

```
SOUNDEX("Faredale")
```

発音が似ている単語

次の2つの例では異なる soundex コードが返されますが、これら2つの単語は発音が似ているため、近い値が返されます。

J525 が返されます。

```
SOUNDEX("Jonson")
```

J523 が返されます。

```
SOUNDEX("Jonston")
```

発音が異なる単語

次の2つの例では、2つの単語がまったく似ていないため、互いから非常に遠い soundex コードが返されます。

S530 が返されます。

```
SOUNDEX("Smith")
```

M235 が返されます。

```
SOUNDEX("MacDonald")
```

フィールドへの入力値

Last_name フィールドの各値の soundex コードが返されます。

```
SOUNDEX>Last_Name)
```

高度な例

一致する soundex コードを特定する

Last_Name フィールドの各値の soundex コードを表示する演算フィールド、**Soundex_Code** を作成します。

```
DEFINE FIELD Soundex_Code COMPUTED SOUNDEX>Last_Name)
```

ビューに演算フィールド **Soundex_Code** を追加し、その演算フィールドで重複検査を実行することによって、一致するあらゆる soundex コードを特定することができます。

```
DUPLICATES ON Soundex_Code OTHER Last_Name PRESORT OPEN TO "Possible_Dupes.fil"
```

一致する soundex コードは、**Last_name** フィールドの関連する文字値が重複している可能性があることを示しています。

備考

SOUNDEX() の使用に適する場面

SOUNDEX() 関数は発音の似た値を検索する場合に使用できます。発音の類似は、重複している可能性がある値や、手作業で入力されたデータで不統一な綴りを見つける 1 つの方法です。

機能の仕組み

SOUNDEX() は、評価される文字列の American Soundex コードを返します。コードはすべて、1 文字の後に 3 つの数字が続きます。たとえば、"F634" などです。

Soundex コードの意味

- コードの最初の文字は、評価される文字列中の最初の文字を表します。
- コード内の各番号は、6 つの American Soundex グループのうちの 1 つを表します。これらのグループは、発音が似た子音で構成されています。

これらのグループに基づいて、soundex 処理は、評価される文字列のうち、先頭文字の後に来る最初の 3 つの子音をエンコードします。

Soundex 処理で無視される内容

Soundex 処理では以下が無視されます。

- 大文字の使用
- 母音

- 子音 "H"、"W"、"Y"
- エンコードされた3つの子音の後に来るすべての子音

返されるコード内の末尾にある1つ以上のゼロ(0)は、評価される文字列には、最初の文字の後に子音が3つ未満しかないことを示しています。

Soundex 処理にある制限

SOUNDSLIKE() 関数と SOUNDEX() 関数のどちらにも、ある制限があります。

- soundex アルゴリズムは、英語で発音された言葉を処理するように設計されており、他の言語で使用した場合には、有効性の度合いが違ってきます。
- soundex 処理は発音の照合を実行しますが、一致する単語は同じ文字で始まっていなければなりません。これはつまり、発音が同じであるいくつかの単語は一致しないということです。

たとえば、"F" で始まる単語と "Ph" で始まる単語は発音は同じですが、これらは決して一致しません。

関連する関数

- **SOUNDSLIKE()** - 文字列の発音を比較するもう1つの方法。
- **ISFUZZYDUP()** および **LEVDIST** - は、発音記号の比較(発音)ではなく、正字法の比較(綴り)に基づいて文字列を比較します。
- **DICECOEFFICIENT()** - は、文字列を比較するときに、文字または文字ブロックの相対位置を重視しないか、または完全に無視します。

SOUNDSLIKE() 関数

文字列が比較文字列と発音学的に一致しているかどうかを示す論理値を返します。

構文

```
SOUNDSLIKE(名前, 発音の似ている名前)
```

パラメーター

名前	型	説明
名前	文字	比較の最初の文字列。
発音の似ている名前	文字	比較の2番目の文字列。

出力

論理。比較する値同士の発音が一致している場合はT(true)、そうでない場合はF(false)が返されません。

例

基本的な例

"Fairdale" と "Faredale" はどちらも soundex コードが F634 のため、T が返されます。

```
SOUNDSLIKE("Fairdale", "Faredale")
```

"Jonson" の soundex コードは J525、"Jonston" の soundex コードは J523 のため、F が返されます。

```
SOUNDSLIKE("Jonson", "Jonston")
```

Last_name フィールドの各値の soundex コードが文字列 "Smith" の soundex コードと一致するかどうかを示す論理値 (T または F) が返されます。

```
SOUNDSLIKE>Last_name, "Smith")
```

高度な例

"Smith" と発音が似ている値を抽出する

Last_name フィールドの、"Smith" と発音が似ているすべての値を抽出するフィルターを作成するには、次のように指定します。

```
SET FILTER TO SOUNDSLIKE>Last_name, "Smith")
```

備考

SOUNDSLIKE() の使用に適する場面

SOUNDSLIKE() 関数は発音の似た値を検索する場合に使用できます。発音の類似は、重複している可能性がある値や、手作業で入力されたデータで不統一な綴りを見つける 1 つの方法です。

機能の仕組み

SOUNDSLIKE() 関数はまず、比較対象となる文字列を 4 文字の American Soundex コードに変換します。このコードは、各文字列の最初の文字と、その後続く最初の 3 つの子音に基づきます。

次に、各文字列のコードを比較し、それらが一致するかどうかを示す論理値を返します。

soundex コードの詳細については、"SOUNDEX() 関数" ページ 2503 を参照してください。

大文字と小文字の区別

この関数では大文字と小文字が区別されないため、"SMITH" は "smith" と同じであると判断されます。

Soundex 処理にある制限

SOUNDSLIKE() 関数と SOUNDEX() 関数のどちらにも、ある制限があります。

関数

- soundex アルゴリズムは、英語で発音された言葉処理するように設計されており、他の言語で使用した場合には、有効性の度合いが違ってきます。
- soundex 処理は発音の照合を実行しますが、一致する単語は同じ文字で始まっていなければなりません。これはつまり、発音と同じであるいくつかの単語は一致しないということです。

たとえば、"F" で始まる単語と "Ph" で始まる単語は発音が同じですが、これらは決して一致しません。

関連する関数

- **SOUNDEX()** - 文字列の発音を比較するもう1つの方法。
- **ISFUZZYDUP()** および **LEVDIST** - は、発音記号の比較(発音)ではなく、正字法の比較(綴り)に基づいて文字列を比較します。
- **DICECOEFFICIENT()** - は、文字列を比較するときに、文字または文字ブロックの相対位置を重視しないか、または完全に無視します。

SPLIT() 関数

文字列のうちの指定された部分を返します。

構文

```
SPLIT(文字列, 区切り文字, セグメント <, テキスト修飾子>)
```

パラメーター

名前	型	説明
文字列	文字	セグメントを抽出するフィールド、式、またはリテラル値。
区切り文字	文字	セグメントを区切る文字または文字列。 詳細については、「区切り文字の動作」ページ 2512を参照してください。
セグメント	数値	抽出するセグメント。 抽出するセグメントを指定するには、番号を使用します。たとえば、3番目のセグメントを抽出するには、「3」を指定します。
テキスト修飾子 省略可能	文字	テキストのセグメントの開始位置と終了位置を示す文字または文字列。 テキスト修飾子のペアの内部に出現する区切り文字は、区切り文字でなくテキストとして読み取られます。 テキスト修飾子は引用符で囲む必要があります。一重引用符のテキスト修飾子は二重引用符で、二重引用符テキスト修飾子は一重引用符で、それぞれ囲む必要があります。 ヒント この任意のパラメーターは、区切り文字とテキスト修飾子が含まれるソースデータをインポートして操作するときに役立つ場合があります。

出力

文字。

例

基本的な例

カンマで区切られたセグメント

"seg1" が返されます。

```
SPLIT("seg1,seg2,seg3", ",", 1)
```

"seg3" が返されます。

```
SPLIT("seg1,seg2,seg3", ",", 3)
```

"" が返されます(3番目のセグメントは空です)。

```
SPLIT("seg1,seg2,,seg4", ",", 3)
```

文字列およびスペースの区切り文字

"seg3" が返されます。

```
SPLIT("seg1/*seg2/*seg3", "/*", 3)
```

"Doe" が返されます。

```
SPLIT("Jane Doe", " ", 2)
```

区切り文字をテキスト修飾子でエスケープする

区切り文字でなくテキストとして読み取られるカンマが含まれている "Doe, Jane" が返されます。

```
SPLIT('"Doe, Jane","Smith, John"', ",", 1, '"')
```

高度な例

クレジット カード番号から数字を抽出する

SPLIT() 関数を使用して、クレジット カード番号からダッシュを取り除くことができます。

変数を使用してクレジット カード番号の各セグメントを取得した後、それらのセグメントを連結して別の変数に格納します。

```
ASSIGN seg1 = SPLIT("4150-2222-3333-4444", "-", 1)
ASSIGN seg2 = SPLIT("4150-2222-3333-4444", "-", 2)
ASSIGN seg3 = SPLIT("4150-2222-3333-4444", "-", 3)
ASSIGN seg4 = SPLIT("4150-2222-3333-4444", "-", 4)
ASSIGN ccNum = seg1 + seg2 + seg3 + seg4
```

ccNum の値は "4150222233334444" です。

この例は SPLIT() 関数について説明していますが、EXCLUDE() 関数を使用すると、より効率的にダッシュを取り除くことができることに留意してください。

備考

機能の仕組み

SPLIT() 関数は、文字データをスペースやカンマなどの区切り文字に基づいてセグメントに区切り、指定されたセグメントを返します。

SPLIT() の使用に適する場面

SPLIT() 関数は、レコードまたはフィールドからデータの特定のセグメントを抽出する場合に使用できます。このセグメントは、個々のレコードまたはフィールドの同じ位置に存在する必要があります。

区切り文字の動作

区切り文字は、ソース文字列においてデータをセグメントとして区切ったり指定したりするものです。

多数のセグメントがある文字列では、セグメントの大部分は2つの区切り文字に挟まれています。これに対し、最初のセグメントの前と、最後のセグメントの後には区切り文字がありません。

ソース文字列が区切り文字で始まっている場合、最初の区切り文字の前のセグメントはセグメント 1 として処理されます。

関数

"seg1" が返されます。

```
SPLIT("seg1,seg2,seg3", ",", 1)
```

ソース文字列の先頭が区切り文字の場合、セグメント 1 はヌルと見なされます。その区切り文字の後のセグメントはセグメント 2 として処理されます。

"seg1" が返されます。

```
SPLIT("seg1,seg2,seg3", ",", 1)
```

大文字と小文字の区別

区切り文字またはテキスト修飾子として、大文字や小文字がある文字を指定する場合、その文字と、データ内の区切り文字やテキスト修飾子の間で、大文字と小文字の違いも一致している必要があります。

関連する関数

SPLIT() と SUBSTR() はともに、長いソース文字列からデータのセグメントを返します。

- SPLIT() は区切り文字を使ってセグメントを識別します。
- SUBSTR() は数値文字の位置を使ってセグメントを識別します。

STOD() 関数

シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。

構文

```
STOD(シリアル日付 <,開始日>)
```

パラメーター

名前	型	説明
シリアル日付	数値	変換するフィールド、式、またはリテラル値。 シリアル日付には、シリアル日付またはシリアル日付時刻を指定できません。シリアル日付時刻の日付部分のみが考慮されます。時刻部分は無視されます。
開始日 省略可能	日付時刻	シリアル日付の計算基準となる開始日。これを省略した場合は、デフォルトの開始日である 1900 年 1 月 1 日を使用されます。

出力

日付時刻。日付値は、現在 Analytics に設定されている日付の表示書式を使用して出力されます。

例

基本的な例

`20141231` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、31 Dec 2014 が返されます。

```
STOD(42003)
```

`20181231` に対して現在の Analytics 日付表示書式である DD MMM YYYY を適用した、31 Dec 2018 が返されます。

関数

```
STOD(42003, `19040101`)
```

Invoice_Date フィールドの各シリアル日付値に対応する日付が返されます。

```
STOD(Invoice_Date)
```

高度な例

開始日を 1900-01-01 より前の日付に変更する

Analytics の最早の開始日である 1900 年 1 月 1 日より前の値に開始日を変更するには、日付の算術を行います。

1. デフォルトの開始日を使ってシリアル日付を変換します。
2. 実際の開始日から 1900-01-01 までの日数を差し引きます。

次の例は、開始日として 1899-01-01 を使用した場合に `20131231` と評価されます。

```
STOD(42003) - 365
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

機能の仕組み

STOD() 関数を使用すると、シリアル日付を標準の日付に変換することができます。Analytics のシリアル日付は、1900 年 1 月 1 日からの経過日数を表します。

シリアル日付	相当する標準の日付
1	1900 年 1 月 2 日

シリアル日付	相当する標準の日付
365	1900年12月31日
42003	2014年12月31日
0	有効でない

シリアル日付の詳細については、「シリアル日付時刻」ページ 896を参照してください。

Analytics のシリアル日付と Excel シリアルの日付との比較

Analytics のシリアル日付は Microsoft Excel のシリアル日付に似ています。注意すべき類似のキーポイントが1つと、相違のキーポイントが1つあります。これら2つのポイントには関連性はありません。

類似のポイント

Analytics と Excel はどちらも、1900 年を 366 日ある閏年として扱います。1900 年は実際には閏年ではありませんでしたが、Excel は Lotus 1-2-3 との互換性を保つために、この年を閏年として処理しました。

差異の点

Analytics のシリアル日付は、Excel のシリアル日付から 1 日ずれています。Excel では、1900 年 1 月 1 日はシリアル日付 '1' になります。Analytics では、1900 年 1 月 1 日はカウントされないため、1900 年 1 月 2 日がシリアル日付 '1' になります。

開始日

一部のソースデータファイルは 1900 年 1 月 1 日以外の開始日を使用する場合があります。開始日は、ソースデータファイルの開始日と一致させることができます。シリアル日付の計算基準となる開始日。

ソースデータファイルの開始日	次を指定します。	詳細
1900年1月1日	<code>STOD(日付フィールド)</code>	1900年1月1日はデフォルト開始日であるため、開始日を指定する必要はありません。
1901年1月1日	<code>STOD(日付フィールド, `19010101`)</code>	開始日 `19010101` を指定し、ソースデータファイルで使用される1901年1月1日の開始日と合わせます。
1899年1月1日	<code>STOD(日付フィールド) - 365</code>	1900年1月1日より前の開始日を指定することはできません。ソースデータファイルが1900年1月1日より前の開始日を使用する場合は、STOD()関数の出力結果から適当な日数を減算する日付時刻式を作成することができます。

他の日付時刻変換関数

シリアルから日付時刻への変換

関数	説明
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime"の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time"の省略形です。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date"の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime"の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time"の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

STODT() 関数

シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime"の省略形です。

構文

```
STODT(シリアル日付時刻 <,開始日>)
```

パラメーター

名前	型	説明
シリアル日付時刻	数値	変換するフィールド、式、またはリテラル値。 日付と時刻の部分が小数点で区切られたシリアル日付時刻値が必要です。たとえば、42003.75000のように指定します。
開始日 省略可能	日付時刻	シリアル日付の計算基準となる開始日。これを省略した場合は、デフォルトの開始日である1900年1月1日を使用されます。

出力

日付時刻。日付時刻値は、現在 Analytics に設定されている日付と時刻の表示書式を使用して出力されます。

例

基本的な例

開始日を調整しない場合

`20141231t060000` に対して現在の Analytics 日付時刻表示書式である DD MMM YYYY hh:mm:ss PM を適用した、31 Dec 2014 06:00:00 AM が表示されます。

関数

```
STODT(42003.25000)
```

`20141231t191530` が返されますが、表示上は、現在の Analytics 日付時刻表示書式である DD MMM YYYY hh:mm:ss PM を適用した 31 Dec 2014 07:15:30 PM となります。

```
STODT(42003.802431)
```

開始日を調整した場合

`20181231t120000` が返されますが、表示上は、現在の Analytics 日付時刻表示書式である DD MMM YYYY hh:mm:ss PM を適用した 31 Dec 2018 12:00:00 PM として表示されます。

```
STODT(42003.50000, `19040101`)
```

入力用フィールド

Receipt_timestamp フィールドの各シリアル日付時刻値に対応する日付時刻が返されます。

```
STODT(Receipt_datetime)
```

高度な例

開始日を 1900-01-01 より前の日付に変更する

Analytics の最早の開始日である 1900 年 1 月 1 日より前の値に開始日を変更するには、日付の算術を行います。

1. デフォルトの開始日を使ってシリアル日付時刻を変換します。
2. 実際の開始日から 1900-01-01 までの日数を差し引きます。

次の例は、開始日として 1899-01-01 を使用した場合に `20131231t180000` と評価されます。

```
STODT(42003.75000) - 365
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

機能の仕組み

STODT() 関数を使用すると、シリアル日付時刻を標準の日付時刻に変換することができます。Analytics のシリアル日付時刻は、1900 年 1 月 1 日からの経過日数を表し、小数点以下は、24 時間を 1 とする、24 時間の小数を表します。

シリアル日付時刻	相当する標準の日付時刻
1.25	1900 年 1 月 2 日午前 6 時 0 分 0 秒
365.75000	1900 年 12 月 31 日午後 6 時 0 分 0 秒
42003.79167	2014 年 12 月 31 日午後 7 時 0 分 0 秒
42003.802431	2014 年 12 月 31 日午後 7 時 15 分 30 秒
42003.00000	2014 年 12 月 31 日午前 12 時 0 分 0 秒
42003.50000	2014 年 12 月 31 日午後 12 時 0 分 0 秒
0.0	有効でない

シリアル日付時刻の詳細については、"シリアル日付時刻" ページ 896を参照してください。

Analytics のシリアル日付と Excel シリアルの日付との比較

Analytics のシリアル日付は Microsoft Excel のシリアル日付に似ています。注意すべき類似のキーポイントが 1 つと、相違のキーポイントが 1 つあります。これら 2 つのポイントには関連性がありません。

類似のポイント

Analytics と Excel はどちらも、1900 年を 366 日ある閏年として扱います。1900 年は実際には閏年ではありませんでしたが、Excel は Lotus 1-2-3 との互換性を保つために、この年を閏年として処理しました。

差異の点

Analytics のシリアル日付は、Excel のシリアル日付から 1 日ずれています。Excel では、1900 年 1 月 1 日はシリアル日付 '1' になります。Analytics では、1900 年 1 月 1 日はカウントされないため、1900 年 1 月 2 日がシリアル日付 '1' になります。

開始日

一部のソースデータファイルは 1900 年 1 月 1 日以外の開始日を使用する場合があります。開始日は、ソースデータファイルの開始日と一致させることができます。シリアル日付時刻の計算基準となる開始日。

ソースデータファイルの開始日	次を指定します。	詳細
1900 年 1 月 1 日	STODT(日付時刻フィールド)	1900 年 1 月 1 日はデフォルト開始日であるため、開始日を指定する必要はありません。
1901 年 1 月 1 日	STODT(日付時刻フィールド, `19010101`)	開始日 `19010101` を指定し、ソースデータファイルで使用される 1901 年 1 月 1 日の開始日と合わせます。
1899 年 1 月 1 日	STODT(日付時刻フィールド) - 365	1900 年 1 月 1 日より前の開始日を指定することはできません。ソースデータファイルが 1900 年 1 月 1 日より前の開始日を使用する場合は、STODT() 関数の出力結果から適当日数を減算する日付時刻式を作成することができます。

他の日付時刻変換関数

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STOT()	シリアル時刻、つまり、24 時間を 1 として、24 時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

STOT() 関数

シリアル時刻、つまり、24 時間を 1 として、24 時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

構文

```
STOT(シリアル時刻)
```

パラメーター

名前	型	説明
シリアル時刻	数値	変換するフィールド、式、またはリテラル値。 シリアル時刻には、シリアル時刻またはシリアル日付時刻を指定できません。シリアル日付時刻の時刻部分のみが考慮されます。日付部分は無視されます。

出力

日付時刻。時刻値は、現在 Analytics に設定されている時刻の表示書式を使用して出力されます。

例

基本的な例

`t060000` が返されますが、表示上は、現在の Analytics 時刻表示書式である hh:mm:ss PM を適用した 06:00:00 AM となります。

```
STOT(0.25000)
```

`t191530` が返されますが、表示上は、現在の Analytics 時刻表示書式である hh:mm:ss PM を適用した 07:15:30 PM となります。

```
STOT(0.802431)
```

Login_time フィールドの各シリアル時刻値に相当する標準の時刻が返されます。

```
STOT(Login_time)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

STOT() の使用に適する場面

STOT() 関数は、シリアル時刻を標準の時刻に変換する場合に使用できます。

シリアル時刻とは

Analytics のシリアル時刻は、24 時間を 1 として、24 時間制の時刻を小数として表したものです。

例：

- 1 時間に相当するシリアル時刻は 1/24、つまり 0.04167 です。
- 1 分に相当するシリアル時刻は 1/1440、つまり 0.0006945 です。

シリアル時刻の先頭には、'0'(ゼロ)と小数点、または小数点のみを使用できます。

1.000000 は有効なシリアル時刻ではありません。

シリアル時刻を計算する目的のために、24 時間は 1 に等しいとしていますが、1.000000 は有効なシリアル時刻ではありません。有効なシリアル時刻は 1 未満のすべての小数です。例：0.75000(午後 6 時)

Analytics はシリアル値 1.000000 を、1900 年 1 月 2 日午前 12 時に対応するシリアル日付時刻として扱います。STOT() は日付時刻の日付部分を無視するため、`STOT(1.000000)` は `STOT(0.000000)` と同等となり、どちらも標準時刻の午前 12 時に相当します。

シリアル時刻とそれに相当する標準の時刻

シリアル時刻	相当する標準の時刻
0.00	12:00:00 AM
0.0006945	12:01:00 AM
0.04167	01:00:00 AM
0.0423645	01:01:00 AM
0.042998	01:01:55 AM
0.25	06:00:00 AM
0.50	12:00:00 PM
0.75	06:00:00 PM
0.79167	07:00:00 PM
0.802431	07:15:30 PM
1.00	12:00:00 AM

他の日付時刻変換関数

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を

関数	説明
	抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。
TIME()	指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

STRING() 関数

数値を文字列に変換します。

構文

```
STRING(数値, 長さ <, 書式 >)
```

パラメーター

名前	型	説明
数値	数値	文字に変換する数値。
長さ	数値	出力文字列の長さの文字数。 <ul style="list-style-type: none">長さが数値より長い場合は、出力文字列の先頭にスペースが追加されます。長さが数値より短い場合、出力文字列は左側から切り詰められます。 指定した長さのスペースが十分にあり、任意の書式パラメーターを指定した場合の数値以外の書式文字など、フィールドで最も長い数値が格納されることを確認してください。
書式 省略可能	文字	出力文字列に適用する書式。 書式は二重引用符で囲む必要があります。例: "(9,999.99)" 任意の書式パラメーターを使用して、ソースデータに存在しない出力文字列に書式を追加します。ドル記号、パーセント記号、1つ以上の小数点プレースホルダー、千の位の桁区切り文字、括弧などを追加できます。 メモ 指定する数値以外の書式文字では、数値の長さが増えます。

出力

文字。

例

基本的な例(出力が書式設定されていない)

数値 `125.2`

"125.2" が返されます。

```
STRING(125.2, 6)
```

長さの値が`6`で、数値の桁数と書式文字よりも1文字長いため、出力文字列の先頭には1つのスペースが埋め込まれます。

数値 `-125.2`

"25.2" が返されます。

```
STRING(-125.2, 4)
```

長さの値が`4`で、数値の桁数と書式文字よりも2文字短いため、出力文字列は切り詰められます。
"-125.2" が返されます。

```
STRING(-125.2, 7)
```

長さの値が`7`で、数値の桁数と書式文字よりも1文字長いため、出力文字列の先頭には1つのスペースが埋め込まれます。

基本的な例(出力が書式設定されている)

数値 `125.2`

"25.20" が返されます。

```
STRING(125.2, 6, "(9,999.99)")
```

長さの値が`6`で、指定した書式が適用された後の数値よりも1文字短いため、出力文字列は切り詰められます。

"125.20" が返されます。

```
STRING(125.2, 7, "(9,999.99)")
```

メモ

特定の数値のインスタンスで書式文字が必要ではない場合であっても、書式に指定した文字は、右側から数値の長さの計算に含まれます。上記の例では、数値で正値を表すために括弧が必要でない場合でも、右端の括弧が文字としてカウントされます。

数値 -125.2

" (125.20)" が返されます。

```
STRING(-125.2, 10, "(9,999.99)")
```

長さの値が10で、指定した書式が適用された後の数値よりも2文字長いため、出力文字列の先頭に2つのスペースが埋め込まれます。

基本的な例(フィールド入力)

文字列型の長さ10文字の数値を **Employee_number** フィールドが返されます。必要に応じて、出力文字列は空白が埋め込まれたり、切り捨てられたりします。

```
STRING(Employee_number, 10)
```

備考

出力文字列の書式設定

出力文字列の書式を設定し、ソースデータには含まれていない場合でも、書式を表示することができます。

書式のプレースホルダー桁数

指定する書式で、数字9は桁数のプレースホルダーとして機能します。フィールドの最も長い数値を格納できるように、必ず十分なプレースホルダー桁数を指定してください。たとえば、フィールドの金額が最大\$5,000,000で、小数点2桁までである場合は、9桁のプレースホルダー"\$9,999,999.99"を指定する必要があります。

出力文字列で必要な最小の長さに書式が及ぼす影響

長さで指定する値は、少なくとも、フィールドの最も長い値のすべての数字と、指定するすべての書式文字を格納できる長さが必要です。

最大 \$5,000,000 の金額を含むフィールドの値に、ドル記号と千の位の桁区切り文字を追加する場合は、長さに **13** (9 桁 + 4 文字の数値以外の書式文字) 以上を指定する必要があります。

金額フィールドの数値が、指定した書式が付いた文字列として返されます。

```
STRING(Amount, 13, "$9,999,999.99")
```

\$4,789,123.50 が文字列として返されます。

```
STRING(4789123.50, 13, "$9,999,999.99")
```

関連する関数

STRING() 関数は VALUE() 関数の逆関数です。後者は、文字データを数値データに変換します。

SUBSTR() 関数

文字列のうちの指定された部分文字列を返します。

構文

```
SUBSTR(文字列, 開始位置, 長さ)
```

パラメーター

名前	型	説明
文字列	文字	サブ文字列を抽出するフィールド、式、またはリテラル値。
開始	数値	部分文字列の開始文字位置。 文字列の文字の数値位置は 1 から開始します。文字列 <code>ABCDEF</code> から <code>C</code> で始まるサブ文字列を抽出するには、開始値 3 を指定します。
長さ	数値	部分文字列の文字数。 長さが 0 の場合は、出力は空です。

出力

文字。

例

基本的な例

リテラル文字の入力値

"BCD" が返されます。

```
SUBSTR("ABCDEF", 2, 3)
```

"EF" が返されます。

```
SUBSTR("ABCDEF", 5, 10)
```

構造的な文字データを解析する

"189543" が返されます。

```
SUBSTR("***189543***", 4, 6)
```

"MM/DD/YYYY" として書式設定された日付が含まれる文字フィールドのうち、4桁の年

```
SUBSTR(DATE, 7, 4)
```

高度な例

フィールド長を大きくする

文字フィールドの長さを大きくするのに SUBSTR() を使用できます。フィールド長を大きくすることは、2つのフィールドを結合したり追加したりする前に必要に応じて行うことができる一般的な調整作業です。

次の例は、Product_Description フィールドにスペースを追加して、長さが 50 文字の演算フィールド、Product_Description_Long を作成しています。

```
DEFINE FIELD Product_Description_Long COMPUTED SUBSTR(Product_
Description, 1, 50)
```

備考

機能の仕組み

SUBSTR() 関数は、文字列値内の、開始位置で指定された桁位置から始まる文字列を返します。長さには、返される文字数を指定します。

SUBSTR() がスペースを処理する方法

文字列値の先頭、末尾、または内部のスペースは文字として処理されます。開始位置および長さで取り込まれたスペースは出力文字列に含まれます。

スペースの追加の動作

長さの値が開始位置から文字列の終わりまでの文字数(スペースを含む)を超える場合は、出力の右側にスペースが追加される場合とされない場合があります。

スペースが追加される場合とは

フィールドを作成するコマンド内で SUBSTR() を使用する場合は、出力にスペースが追加されます。

演算フィールドを作成する場合のスペースの追加

次の例は、長さが24文字の物理フィールド、**Product_Description**に基づいて、長さが50文字の演算フィールド、**Product_Description_Long**を作成しています。

```
DEFINE FIELD Product_Description_Long COMPUTED SUBSTR(Product_
Description, 1, 50)
```

物理フィールドを抽出する場合のスペースの追加

次の例は、長さが24文字の物理フィールド、**Product_Description**に基づいて、長さが50文字の演算フィールド、**Product_Description_Long**を新しいテーブルに抽出しています。

```
EXTRACT FIELDS SUBSTR(Product_Description, 1, 50) AS "Product_
Description_Long" TO New_Table
```

スペースが追加されない場合とは

変数定義または式で SUBSTR() を使用する場合は、出力にはスペースが追加されません。

変数を定義する場合にはスペースが追加されない

次の例は、**Product_Description** のフィールド長に基づいて、長さが 24 文字の変数、*v_prod_desc* を作成しています。

```
ASSIGN v_prod_desc = SUBSTR(Product_Description, 1, 50)
```

メモ

SUBSTR() の長さに 50 文字を指定していても、出力は **Product_Description** のフィールド長に制限されます。

関連する関数

SUBSTR() と SPLIT() はともに、長いソース文字列からデータのセグメントを返します。

- SUBSTR() は数値文字の位置を使ってセグメントを識別します。
- SPLIT() は区切り文字を使ってセグメントを識別します。

TAN() 関数

ラジアン単位で表された角度のタンジェントを、小数点以下 15 桁の精度で返します。

構文

```
TAN(ラジアン)
```

パラメーター

名前	型	説明
ラジアン	数値	ラジアン単位の角度。

出力

数値。

例

基本的な例

指定したラジアン数(45 度に相当)に対し、その正接である 0.999999999999999 が返されます。

```
TAN(0.785398163397448)
```

45 度に対し、その正接である 0.999999999999999 が返されます。

```
TAN(45 * PI( )/180)
```

高度な例

入力値として度を使用する

45 度に対し、その正接を小数点以下 3 桁に丸めた 1.000 が返されます。

```
DEC(TAN(45 * PI( )/180),3)
```

備考

Mantissa Arc Test の実行

Analytics の 3 つの三角関数 SIN()、COS()、および TAN() は、ベンフォードの法則に関連する Mantissa Arc Test の実行をサポートします。

度のラジアンへの変換

入力値が度で表されている場合は、PI() 関数を使用して角度をラジアンに変換できます。(度 * PI()/180) = ラジアンとなります。必要に応じて、DEC() 関数を使用して、戻り値を丸めたり切り捨てたりすることができます。

TEST() 関数

指定された文字列がレコード内の特定のバイト位置に現れるかどうかを示す論理値を返します。

構文

```
TEST(バイト位置, 文字列)
```

パラメーター

名前	型	説明
バイト位置	数値	文字列の先頭文字の位置を確認する、テーブルレイアウトでの左からの連番です。 先頭文字の位置を文字列の開始位置とすることができない場合は、その文字列がレコードの別の位置に存在しているとしても、この関数はF (false) と評価されます。
文字列	文字	検索する文字列。 検索は大文字と小文字を区別します。大文字と小文字の両方が混在している可能性がある場合は、UPPER() 関数を使用してすべての文字を大文字に変換します。

出力

論理。指定された文字列がレコード内の指定されたバイト位置から始まっている場合はT(true)、そうでない場合はF(false)を返します。

例

基本的な例

次のようなレコードを例に説明します。

```
Department: Marketing  
.....|.....|.....|.....|.....|
```

Tが返されます。

```
TEST(5, "Department")
```

レコードにおける "Department" の開始バイト位置が6でなく5のため、Fが返されます。

```
TEST(6, "Department")
```

この関数は大文字と小文字が区別されるため、Fが返されます。

```
TEST(5, "DEPARTMENT")
```

高度な例

ページ見出しになっているレコードを抽出する

TEST()を使用することで、"Page:" で始まるすべてのレコードを抽出するフィルターを作成できます。

```
SET FILTER TO TEST(1, "Page:")
```

TIME() 関数

指定された時刻または日付時刻から時刻を抽出し、それを文字データとして返します。また、現在のオペレーティングシステムの時刻を返すこともできます。

構文

```
TIME(<時刻/日付時刻> <,書式>)
```

パラメーター

名前	型	説明
時刻/日付時刻 省略可能	日付時刻	時刻を抽出するフィールド、式、またはリテラル値。これを省略した場合は、現在のオペレーティングシステムの時刻が hh:mm:ss 書式で返されます。
書式 省略可能	文字	出力文字列に適用する書式。例: "hh:mm:ss" このパラメーターを省略した場合は、現在の Analytics の時刻表示書式が使用されます。時刻/日付時刻を省略した場合に、書式を指定することはできません。

出力

文字。

例

基本的な例

リテラルの入力値

現在の Analytics 時刻表示書式である hh:mm:ss を適用した時刻、"23:59:59" が返されます。

```
TIME(`20141231 235959`)
```

"11:59 P" が返されます。

```
TIME(`20141231 235959`, "hh:mm A")
```

現在のオペレーティングシステム時刻が hh:mm:ss 書式 (24 時間制) で文字列として返されます。

```
TIME()
```

入力値用フィールド

Receipt_timestamp フィールドの各値に対して現在の Analytics の時刻表示書式を適用した文字列が返されます。

```
TIME(Receipt_timestamp)
```

Receipt_timestamp フィールドの各値に対して、指定した時刻表示書式を適用した文字列が返されま

```
TIME(Receipt_timestamp, "hh:mm:ss")
```

高度な例

実行するコマンドまたはスクリプトの経過時間を計算する

特定の Analytics コマンドまたはスクリプト全体の実行にかかる時間を計算するために TIME() 関数を使用することができます。

時間を計測するコマンドの直前、またはスクリプトの開始時に、現在のオペレーティングシステムの時間を格納する変数を作成するために次の行を指定します。

```
ASSIGN Time_started = TIME()
```

コマンドの実行直後、またはスクリプトの終了時に次の 2 行を指定します。

最初の行は、コマンドまたはスクリプトが完了した後にオペレーティングシステムの時間を格納する変数を作成します。2行目は終了と開始時間の差を計算して、結果を見やすい書式に変換します。

ヒント

コマンドやスクリプトの経過時間は、CALCULATE ログ エントリをダブルクリックしても見ることができます。

```
ASSIGN Time_finished = TIME()
CALCULATE STOT(CTOT(Time_finished) - CTOT(Time_started))
```

コマンドまたはスクリプトが日付をまたいで実行される場合は、代わりにこの2行を使用します。

```
CALCULATE `T000000` - (CTOT(Time_started) - CTOT(Time_finished))
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

出力文字列の長さ

出力文字列の長さは常に14文字です。指定した出力書式、または Analytics の時刻表示書式が12文字より短い場合は、出力文字列の末尾に空白が埋められます。

パラメーターの詳細

時刻/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような時刻書式または日付時刻書式でも使用することができます。

書式を使用して出力文字列の表示方法を制御する場合は、次の表内の書式に制限されます。時刻とAM/PMの書式を任意に組み合わせて使用することができます。AM/PM書式は省略可能で、最後に置きます。

書式は、一重引用符または二重引用符を使用して指定します。例: "hh:mm:ss AM"

時刻書式	AM/PM 書式	例
hh:mm:ss	なし 24 時間制	"hh:mm:ss"
hhmmss	AM または PM 12 時間制	"hhmmss PM"
hh:mm	A または P 12 時間制	"hh:mm A"
hhmm		
hh		

リテラル時刻または日付時刻値の指定

日付時刻にリテラルの時刻値または日付時刻値を指定する場合は、次の表内の書式に制限されま
す。また、`20141231 235959`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用し
ないでください。

- **時刻値** - 以下の表に示す任意の時刻の書式を使用することができます。関数が正しく動作する
ためには、単独の時刻値の前に区切り文字を使用する必要があります。有効な区切り文字は文
字 't' または 'T' です。24 時間形式で時刻を指定する必要があります。UTC (Coordinated
Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始め
る必要があります。
- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて
使用することができます。日付を時刻の前に置く必要があり、かつ、2 つの間に区切り文字を使用
する必要があります。有効な区切り文字は空白スペース 1 つ、あるいは文字 't' または 'T' です。

形式の例	リテラル値の例
thhmmss	`t235959`
Thhmm	`T2359`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDTthh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`
YYMMDD hhmm+/-hh	`141231 2359+01`

形式の例	リテラル値の例
(UTC オフセット)	
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

関連する関数

日付時刻値として現在のオペレーティングシステム時刻付を返すには、TIME() の代わりに NOW() を使用してください。

他の日付時刻変換関数

日付時刻から文字への変換

関数	説明
DATE()	指定された日付または日付時刻から日付を抽出し、それを文字データとして返します。また、現在のオペレーティングシステム日付を返すこともできます。
DATETIME()	日付時刻を文字列に変換します。また、現在のオペレーティングシステムの日付時刻を返すこともできます。

文字または数値から日付時刻への変換

関数	説明
CTOD()	文字または数値の日付値を日付に変換します。また、文字または数値の日付時刻値から日付を抽出し、それを日付として返すこともできます。"Character to Date" の省略形です。
CTODT()	文字または数値の日付時刻値を日付時刻に変換します。"Character to Datetime" の省略形です。
CTOT()	文字または数値の時刻値を時刻に変換します。また、文字または数値の日付時刻値から時刻を抽出し、それを時刻として返すこともできます。"Character to Time" の省略形です。

シリアルから日付時刻への変換

関数	説明
STOD()	シリアル日付、つまり、整数で表される日付を日付値に変換します。"Serial to Date" の省略形です。
STODT()	シリアル日付時刻、つまり、整数部分と24時間の小数部分で表される日付時刻を日付時刻値に変換します。"Serial to Datetime" の省略形です。
STOT()	シリアル時刻、つまり、24時間を1として、24時間が小数部分で表される時刻を時刻値に変換します。"Serial to Time" の省略形です。

TODAY() 関数

現在のオペレーティングシステム日付を日付時刻データ型で返します。

構文

```
TODAY()
```

パラメーター

この関数にはパラメーターはありません。

出力

日付時刻。

例

基本的な例

現在のオペレーティングシステム日付が、現在の Analytics 日付書式を使用して表示された日付時刻値 (`20141231` など) として返されます。

```
TODAY()
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

関連する関数

現在のオペレーティングシステム日付を文字列として返させたい場合には、`TODAY()` でなく `DATE()` を使用してください。

TRANSFORM() 関数

指定された文字列内の双方向テキストの表示方向を逆にします。

構文

```
TRANSFORM(元の文字列)
```

パラメーター

名前	型	説明
元の文字列	文字	双方向テキストを含むフィールド、式、リテラル値。

出力

文字。

例

基本的な例

入力文字列において、文字列 "XZQB" は入力文字列におけるヘブライ語/双方向文字を表します。入力文字列には、それ以外の文字としては、通常の文字が含まれています。

出力文字列では、"XZQB" の方向は逆になり、"BQZX" が返されます。"XZQB" 以外の文字列は変更されません。

"ABC BQZX 123" が返されます。

```
TRANSFORM("ABC ZYX 123")
```

備考

機能の仕組み

TRANSFORMS() 関数は双方向データを識別して、そのテキストを右から左へ正しく表示します。
この関数で処理されるその他すべての文字列は、変更されず、引き続き左から右へ表示されます。

TRANSFORMS() の使用に適する場面

TRANSFORMS() 関数は、アラビア語やヘブライ語の文字が正しく表示されるように、文字の表示方向を変更する場合に使用できます。

TRIM() 関数

入力文字列から末尾のスペースを除去した文字列を返します。

構文

```
TRIM(文字列)
```

パラメーター

名前	型	説明
文字列	文字	末尾のスペースを除去するフィールド、式、リテラル値。

出力

文字。

例

基本的な例

両方の例において、先頭のスペースと単語間のスペースが TRIM() 関数によって除去されないことに注意してください。

" Vancouver" が返されます。

```
TRIM(" Vancouver ")
```

" New York" が返されます。

```
TRIM(" New York")
```

高度な例

改行なしスペースの削除

改行なしスペースは TRIM() 関数で削除されません。

末尾の改行なしスペースを削除する必要がある場合は、次の式を使用して演算フィールドを作成します。

```
DEFINE FIELD Description_cleaned COMPUTED TRIM(REPLACE(Description,  
CHR(160), CHR(32)))
```

REPLACE() 関数がすべての改行なしスペースを標準のスペースに置換してから、TRIM() がすべての末尾の標準スペースを削除します。

備考

機能の仕組み

TRIM() 関数は末尾のスペースだけを除去します。文字列の内部スペースおよび先頭のスペースは除去されません。

関連する関数

TRIM() 関数の関連関数として、文字列の先頭のスペースを除去する LTRIM() 関数と、先頭と末尾の両方のスペースを除去する ALLTRIM() 関数があります。

UNSIGNED() 関数

符号なしデータ型に変換された数値データを返します。

構文

```
UNSIGNED(数値, 結果の長さ)
```

パラメーター

名前	型	説明
数値	数値	変換する値。
結果の長さ	数値	出力文字列で使用するバイトの数。

出力

数値。

例

基本的な例

000075 が返されます。

```
UNSIGNED(75, 3)
```

```
UNSIGNED(-75, 3)
```

```
UNSIGNED(7.5, 3)
```

2456 が返されます(結果の長さが2のため、4桁しか格納できないので、1が切り詰められます)。

```
UNSIGNED(12456, 2)
```

000000012456 が返されます。

```
UNSIGNED(-12.456, 6)
```

備考

符号なしデータとは

符号なしデータ型は、メインフレームオペレーティングシステムで 사용되는データ型で、1 バイトにつき 2 桁の数字を格納する、最小の記憶域を使用する形式で数値を格納します。符号なしデータ型はパックデータ型と同じですが、このデータ型は、値が正負いずれであるかを示すために最終バイトを使用しません。

UNSIGNED() の使用に適する場面

UNSIGNED() 関数は、メインフレームシステムへのエクスポートのために、数値データを符号なし形式に変換する場合に使用できます。

戻り値が切り捨てられる場合とは

結果の長さの値が数値の長さより短い場合は、余分な桁が切り詰められます。

UPPER() 関数

アルファベット文字を大文字に変換した文字列を返します。

構文

```
UPPER(文字列)
```

パラメーター

名前	型	説明
文字列	文字	大文字に変換するフィールド、式、またはリテラル値。

出力

文字。

例

基本的な例

"ABC" が返されます。

```
UPPER("abc")
```

"ABC 123 DEF" が返されます。

```
UPPER("abc 123 DEF")
```

"ABCD 12" が返されます。

```
UPPER("AbCd 12")
```

Last_Name フィールドのすべての値を大文字に変換した値が返されます。

```
UPPER>Last_Name)
```

備考

機能の仕組み

UPPER() 関数は文字列のすべてのアルファベットを大文字に変換します。アルファベット以外の文字は変換されません。

UPPER() の使用に適する場面

UPPER() は、フィールド、変数、または式のすべての文字を大文字に統一する必要がある場合に使用できます。大文字への統一は、値を比較する場合には特に重要です。

また、UPPER() はレポート内の値を大文字テキストとして書式設定する場合にも使用できます。

UTOD() 関数

書式設定された日付が含まれる Unicode 文字列を Analytics 日付値に変換します。"Unicode to Date" の省略形です。

メモ

この関数は Analytics の Unicode 版に固有です。非 Unicode 版ではサポートされない関数です。

この関数は、デフォルトのインストールとは異なる言語および書式の日付を操作する場合に使用します。変換元の文字列をデフォルトの言語で表す場合は、代わりに CTOD() を使用します。

構文

```
UTOD(文字列 <,ロケール> <,スタイル>)
```

パラメーター

名前	型	説明
文字列	文字	日付に変換する Unicode 文字列。 Unicode 文字列に日付時刻値を入れることはできますが、値の時刻部分は無視されます。単独の時刻値はサポートされていません。 文字列は、日付のロケールのスタイルに必要な入力書式と一致している必要があります。
ロケール 省略可能	文字	出力文字列の言語およびロケール(と、オプションで、特定の国や地域に関連付けられている言語バージョン)を指定するコード。 たとえば、"zh" では中国語が指定され、"pt_BR" ではブラジルのポルトガル語が指定されます。 これを省略した場合は、コンピューターに設定されているデフォルトのロケールが使用されます。言語コードを指定したが、国コードを指定しなかった場合は、言語コードにデフォルトで設定されている国コードが適用されます。 日付を指定しなかった場合に、ロケールを指定することはできません。 ロケールコードに関する情報については、 www.unicode.org を参照してください。
スタイル 省略可能	数値	Unicode 文字列に使用する日付書式スタイル。この書式スタイルは、指定したロケールの標準書式スタイルと一致します。

名前	型	説明
		<ul style="list-style-type: none"> 0 - "Sunday, September 18, 2016" などの完全仕様の書式 1 - "September 18, 2016" などの長い書式 2 - "Sep 18, 2016" などの中ぐらいの長さの書式 3 - "9/18/16" などの短い数値型書式 <p>これを省略した場合は、デフォルト値の2が使用されます。ロケールを指定しなかった場合に、スタイルを指定することはできません。</p> <p>ヒント</p> <p>入力文字列に必要な書式を判定するには、以下のいずれかを行います。</p> <ul style="list-style-type: none"> スタイルとロケールを指定して DTOU() 関数を実行して、値の例を生成します。 この値を出力するには、コマンドラインで DISPLAY コマンドを実行します。 <pre>DISPLAY DTOU(`20160909`, "es_MX", 3)</pre> <ul style="list-style-type: none"> 特定のロケールでのスタイルの標準的な日付書式については、信頼できる情報源を参照してください。

出力

日付時刻。日付値は、現在 Analytics に設定されている日付の表示書式を使用して出力されます。

例

基本的な例

メモ

どの例でも、現在の Analytics 日付表示書式は DD MMM YYYY であるとしてます。

下の例で、中国語 ("zh") および簡体字中国語 ("zh_CN") のロケールコードはさまざまな入力文字列に一致しますが、互いに交換することはできません。

また、正しいスタイルを指定する必要があります。スタイル2を指定した場合、長い形式の Unicode 日付文字列(つまり、スタイルは1)は Analytics 日付を返しません。

リテラルの入力値

31 Dec 2014 として表示される `20141231` を返します。

関数

```
UTOD("31 de dezembro de 2014", "pt_BR", 1)
```

31 Dec 2014 として表示される `20141231` を返します。

```
UTOD("31 grudnia 2014", "pl", 1)
```

フィールドの入力値

`Invoice_date` フィールドの各 Unicode 文字列に対応する数値の日付が返されます。

```
UTOD(Invoice_date, "zh", 1)
```

入力では完全な日付スタイルが使用されます。

31 Dec 2014 として表示される `20141231` を返します(地域 ID 未指定)。

```
UTOD("星期三, 2014 十二月 31", "zh", 0)
```

31 Dec 2014 として表示される `20141231` を返します(地域 ID 指定)。

```
UTOD("2014年12月31日星期三", "zh_CN", 0)
```

入力では長い形式の日付スタイルが使用されます。

31 Dec 2014 として表示される `20141231` を返します(地域 ID 未指定)。

```
UTOD("2014 十二月 31", "zh", 1)
```

31 Dec 2014 として表示される `20141231` を返します(地域 ID 指定)。

```
UTOD("2014年12月31日", "ja_JP", 1)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考

慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

Unicode 文字列を正しく変換する

日付を含んでいる Unicode 文字列を Analytics 日付に正しく変換するには、ロケールパラメーターとスタイルパラメーターを、Unicode 文字列内の日付の言語、国/地域 (該当する場合)、およびスタイルと一致するように指定する必要があります。

関連する関数

UTOD() は、日付を Unicode 文字列に変換する DTOU() 関数の逆関数です。UTOD() に指定する国/地域およびスタイルが不確かな場合は、DTOU() を使って、何を指定したら UTOD() で変換したい入力 Unicode 文字列の形式に一致する出力 Unicode 文字列を生成できるか、さまざまなパラメーターを試してみることができます。

VALUE() 関数

文字列を数値に変換します。

構文

```
VALUE(文字列, 小数位)
```

パラメーター

名前	型	説明
文字列	文字	変換するフィールド、リテラル、または式の値。
小数位	数値	出力に含める小数点以下桁数。

出力

数値。

例

基本的な例

-123.400 が返されます。

```
VALUE("123.4-", 3)
```

123456.00 が返されます。

```
VALUE("$123,456", 2)
```

-77.45 が返されます。

```
VALUE("77.45CR", 2)
```

-123457 が返されます。

```
VALUE(" (123,456.78)", 0)
```

フィールドへの入力値

小数位のない数値としての文字型の値が **Salary**(給与) フィールドに返されます。

```
VALUE(Salary, 0)
```

備考

機能の仕組み

この関数は文字データを数値データに変換します。VALUE() 関数は、Analytics コマンドで使用するために、文字式やフィールド値を数値に変換する必要がある場合に使用できます。

入力数値の書式設定

VALUE() はどのような書式の数値でも受け付けます。入力として、句読点、先頭または末尾の記号、かっこなど、Print データ型で受け付けられる数値書式はすべて使用できます。

負の値

VALUE() 関数は、かっこやマイナス記号などさまざまな負数のインジケータを解釈することができます。さらに、CR(貸方)およびDR(借方)を解釈することができます。例：

-1000.00 が返されます。

```
VALUE("(1000)", 2)
```

```
VALUE("1000CR", 2)
```

小数値と整数値

文字列値に小数点以下の桁を含めない場合には、文字列の数値は整数として処理されます。例：
123.00 が返されます。

```
VALUE("123", 2)
```

小数位に指定された小数点以下の桁数がフィールドや式の小数点以下桁数より少ない場合、結果は丸められます。例：

"10.6" が返されます。

```
VALUE("10.56", 1)
```

関連する関数

VALUE() 関数は STRING() 関数の逆関数です。後者は、数値データを文字データに変換します。

VERIFY() 関数

物理データフィールドのデータが有効かどうかを示す論理値を返します。

構文

```
VERIFY(フィールド)
```

パラメーター

名前	型	説明
フィールド	文字 数値 日付時刻	物理データフィールドである必要があります。

出力

論理。フィールドのデータが有効な場合は T(true)、そうでない場合は F(false) を返します。

例

基本的な例

VERIFY() 関数が False と評価したレコードを新しい Analytics テーブルに抽出します。

```
EXTRACT RECORD IF NOT VERIFY(Address) TO "InvalidEntries.fil"
```

備考

VERIFY() 関数は、フィールドのデータがそのフィールドに指定されたデータ型と一致しているかどうかを判断します。

VERIFY() の使用に適する場面

この関数は、VERIFY コマンドや [オプション] ダイアログ ボックス([ツール> オプション]) の [数値] タブにある [データを検証する] オプションを用いるよりも、検証するフィールドをより厳密に制御できるようにします。この関数を使用して、個々のフィールドのエラーを検出して状況に応じた処理を行えます。

この関数が true と評価される場合とは

関数が true に評価するための条件は次のとおりです。

- 文字フィールドには、文字、数字、および記号などの、有効な印刷可能な文字のみが入っている
- 数値フィールドには、数字、小数点、および通貨記号などの、有効な数値文字のみが入っている
- 日付時刻フィールドには、有効な日付、日付時刻、あるいは時刻のみが入っている

演算フィールドと式

演算フィールドと式は常に T(true) と評価されます。そのため、まず演算フィールドや式を物理フィールドに変換しない限り、それらに対してこの関数を使用することはできません。[抽出] ダイアログ ボックスの [フィールド] オプションを使用して演算フィールドおよび式を抽出することにより、それらを物理フィールドに変換します。

WORKDAY() 関数

2つの日付の間の就業(稼働)日数を返します。

構文

```
WORKDAY(開始日, 終了日 <, 休日>)
```

パラメーター

名前	型	説明
開始日	日付時刻	就業日の計算対象期間の開始日。開始日は期間内に含まれます。
終了日	日付時刻	就業日の計算対象期間の終了日。終了日は期間内に含まれます。
休日 省略可能	文字	<p>週末または休日に当たるため、計算から除外する曜日。休日を省略する場合は、土曜日と日曜日がデフォルトの休日として使用されます。</p> <p>休日を入力する場合は、次の略語をカンマまたはスペースで区切ります。</p> <ul style="list-style-type: none">◦ Mon◦ Tue◦ Wed◦ Thu◦ Fri◦ Sat◦ Sun <p>休日では大文字と小文字が区別されません。英語版以外の Analytics を使用している場合でも、略語は英語で入力する必要があります。</p> <pre>"Fri, Sat, Sun"</pre>

メモ

開始日や終了日には日付時刻値を指定することもできますが、値の時刻部分は無視されます。

開始日が終了日より後の場合は、負の数値が返されます。

出力

数値。就業日の計算対象期間内の就業日数。

例

基本的な例

リテラルの入力値

2015年3月2日月曜日から2015年3月8日日曜日までの就業日数、5が返されます。

```
WORKDAY(`20150302`, `20150308`)
```

日曜日のみが休日の場合の、2015年3月2日月曜日から2015年3月8日日曜日までの就業日数、6が返されます。

```
WORKDAY(`20150302`, `20150308`, "Sun")
```

金曜日と土曜日が休日の場合の、2015年3月1日日曜日から2015年3月7日土曜日までの就業日数、5が返されます。

```
WORKDAY(`20150301`, `20150307`, "Fri Sat")
```

また、関数を使用して、日付範囲の週末の日数を計算することもできます。

2015年3月2日月曜日から2015年3月8日日曜日までの週末の日数、2が返されます。

```
WORKDAY(`20150302`, `20150308`, "Mon, Tue, Wed, Thu, Fri")
```

フィールドの入力値

開始日フィールドの各日付から2015年12月31日までの就業日数が返されます。

```
WORKDAY(Start_date, `20151231`)
```

Start_date フィールドの各日付から **End_date** フィールドの対応する日付までの就業日数が返されます。

- 法定祝日は就業日合計に含まれるので、別個の計算を使用して除外しなければならない場合があります。
- 戻り値が負の場合は、開始日が終了日よりも後であることを示します。

```
WORKDAY(Start_date, End_date)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

日付書式

開始日または終了日に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式でも使用することができます。

開始日または終了日にリテラルの日付値を指定する場合は、`YYYYMMDD` と `YYMMDD` 書式に制限されません。また、`20141231` のように、値を逆引用符で囲む必要があります。

土曜日と日曜日以外の休日

土曜日と日曜日以外の休日を指定すると、月曜日から金曜日までの就業日または週に5日の就業日に基づかないデータで `WORKDAY()` 関数を使用できます。

たとえば、`Sun` を休日に指定すると、月曜日から土曜日までの週6日を就業日とすることができます。

法定祝日の考慮

`WORKDAY()` 関数は法定祝日を考慮しません。つまり、期間内に1日以上法定祝日が含まれている場合、戻り値に期間内の実際の就業日数が反映されないことがあります。

ScriptHub の「就業日数の計算」スクリプト。

法定祝日を考慮する必要がある場合は、ScriptHub で、ユーザー定義の休日のリストを受け入れる「[就業日の計算](#)」スクリプトを使用します。

より期間が長く、多数の祝日が含まれるデータの場合、一般的に、このスクリプトを使用する方が簡単です。詳細については、「ScriptHub からのインポート」ページ 1606を参照してください。

四半期などの3、4日しか祝日を含まない短い期間の場合は、以下で説明する条件付き演算フィールドを簡単に作成することができます。

法定祝日を減算するための条件付き演算フィールド

必要に応じて、条件付き演算フィールドを作成し、WORKDAY() 関数で返された値から法定祝日を差し引くことができます。

たとえば、2015年の第1四半期の場合、特定の期間に含まれる以下の各祝日について、WORKDAY() の戻り値を1ずつ減算できます。

- 2015/01/01
- 2015/01/19
- 2015/2/16

次の例では、この四半期内に開始日と終了日がある複数の期間を示します。

まず、Workdays などの演算フィールドを作成し、四半期中の特定の期間の就業日数を計算します。

```
DEFINE FIELD Workdays COMPUTED WORKDAY(Start_date, End_date)
```

次に、先の演算フィールド(Workdays)に返された値を調整する、Workdays_no_holidays などの条件付き演算フィールドを作成します。

```
DEFINE FIELD Workdays_no_holidays COMPUTED

Workdays-1 IF Start_date = `20150101` AND End_date < `20150119`
Workdays-2 IF Start_date = `20150101` AND End_date < `20150216`
Workdays-3 IF Start_date = `20150101` AND End_date <= `20150331`
Workdays   IF Start_date < `20150119` AND End_date < `20150119`
Workdays-1 IF Start_date < `20150119` AND End_date < `20150216`
Workdays-2 IF Start_date < `20150119` AND End_date <= `20150331`
Workdays-1 IF Start_date = `20150119` AND End_date < `20150216`
Workdays-2 IF Start_date = `20150119` AND End_date <= `20150331`
Workdays   IF Start_date < `20150216` AND End_date < `20150216`
Workdays-1 IF Start_date < `20150216` AND End_date <= `20150331`
Workdays-1 IF Start_date = `20150216` AND End_date <= `20150331`
Workdays   IF Start_date < `20150331` AND End_date <= `20150331`
Workdays
```

メモ

この条件付き演算フィールド内での条件の順序は重要です。

Analytics では、複数の条件を上から順に評価します。あるレコードに対して真と評価される最初の条件が、そのレコードの条件付き演算フィールドの値を割り当てます。真と評価される後続の条件は、割り当てられた値を変更しません。

YEAR() 関数

指定された日付または日付時刻から年を抽出し、それを YYYY 書式の数値として返します。

構文

```
YEAR(日付/日付時刻)
```

パラメーター

名前	型	説明
日付/日付時刻	日付時刻	年を抽出するフィールド、式、またはリテラル値。

出力

数値。

例

基本的な例

2014 が返されます。

```
YEAR(`20141231`)
```

```
YEAR(`141231 235959`)
```

Invoice_date フィールドの各値に対して年が返されます。

```
YEAR(Invoice_date)
```

備考

日付および時刻関数は正しく使用することが難しい場合があります。ヘルプの関数トピックでは、各関数の動作について、具体的かつ詳細に説明します。日付および時刻関数を使用するときの一般的な考慮事項については、次のトピックを参照してください。

- "式での日付時刻の使用" ページ 884
- "シリアル日付時刻" ページ 896
- "UTC オフセットが時刻日付式に与える影響" ページ 899
- "日付と時刻オプション" ページ 135

パラメーターの詳細

日付/日付時刻に指定されたフィールドは、フィールド定義で正しく書式を定義してさえいれば、どのような日付書式または日付時刻書式でも使用することができます。

リテラル日付または日付時刻値の指定

日付/日付時刻にリテラルの日付値または日付時刻値を指定する場合は、次の表内の書式に制限されます。また、``20141231`` のように、値を逆引用符で囲む必要があります。

日付または時刻の個々の要素の間に、スラッシュ(/) やコロン(:) のような区切り文字をいっさい使用しないでください。

- **日付時刻値** - 以下の表に示す日付、区切り文字、および時刻の書式を任意に組み合わせて使用することができます。日付を時刻の前に置く必要があります。かつ、2つの間に区切り文字を使用する必要があります。有効な区切り文字は空白スペース1つ、あるいは文字 't' または 'T' です。
- **時刻値** - 24 時間形式で時刻を指定する必要があります。UTC (Coordinated Universal Time: 協定世界時) からのオフセットは、プラス記号 (+) またはマイナス記号 (-) で始める必要があります。

形式の例	リテラル値の例
YYYYMMDD	`20141231`
YYMMDD	`141231`
YYYYMMDD hhmmss	`20141231 235959`
YYMMDDthhmm	`141231t2359`
YYYYMMDDThh	`20141231T23`
YYYYMMDD hhmmss+/-hhmm (UTC オフセット)	`20141231 235959-0500`

形式の例	リテラル値の例
YYMMDD hhmm+/-hh (UTC オフセット)	`141231 2359+01`
<p>メモ</p> <p>UTC オフセットが設定されているデータのメインの時刻書式で hh を単独で使用しないでください。たとえば、「hh+hhmm」という使い方は避けてください。信頼できない結果になる可能性があります。</p>	

ZONED() 関数

数値データを文字データに変換し、出力の先頭にゼロを追加します。

メモ

Analytics の Unicode 版を使用している場合、ZONED() の戻り値を正しく表示するには、BINTOSTR() 関数の内部で ZONED() 関数をネストする必要があります。

また、ZONED() の戻り値を別の関数のパラメーターとして使用する場合は、ZONED() を BINTOSTR() の内部でネストする必要もあります。

フィールドの値に先頭のゼロを追加することが目的である場合、ZONED() の代わりに LEADINGZEROS() 関数を使用できます。多くの場合、ZONED() よりも LEADINGZEROS() の方が使用が簡単です。LEADINGZEROS() は、Analytics の Unicode 版と非 Unicode 版の両方でサポートされています。

詳細については、「BINTOSTR() 関数」ページ 2179 と「LEADINGZEROS() 関数」ページ 2329 を参照してください。

構文

```
ZONED(数値, 長さ)
```

パラメーター

名前	型	説明
数値	数値	文字に変換する数値。 メモ 数字の文字列を含む文字値に先頭ゼロを追加する場合は、値を ZONED() の入力として使用する前に、VALUE() 関数を使用して文字を数値データ型に変換する必要があります。詳細については、「VALUE() 関数」ページ 2559 を参照してください。
長さ	数値	出力文字列の長さ。

出力

文字。

例

基本的な例

整数の入力値

"235" が返されます。

```
ZONED(235, 3)
```

次の例では、長さが数値の桁数より大きいので、先頭に2つのゼロが追加されて"00235"が返されます。

```
ZONED(235, 5)
```

次の例では、長さが数値の桁数より小さいので、左端の桁が切り詰められて"35"が返されます。

```
ZONED(235, 2)
```

小数の入力値

次の例では、ZONED データ型では小数点がサポートされないため、"23585"が返されます。

```
ZONED(235.85, 5)
```

負の入力値

次の例では、数値が負であるのと、右端の桁である4が"M"で表されるのとで、"64489M"が返されます。

```
ZONED(-6448.94, 6)
```

次の例では、長さが数値の桁数より小さいので左端の2桁が切り詰められるほか、数値が負であるのと、右端の桁である1が"J"で表されるのとで、"489J"が返されます。

```
ZONED(-6448.91, 4)
```

高度な例

数値を含む文字フィールドに先頭ゼロを追加する

`Employee_Number` フィールドには値 "254879" があります。値を先頭ゼロ埋めの 10 桁の文字列に変換する必要があります。

ヒント

VALUE() 関数を使用して文字を数値データに変換してから、ZONED() への入力値として数値を使用します。

```
COMMENT "0000254879" を返します
ASSIGN v_str_length = 10
ASSIGN v_num_decimals = 0
ZONED(VALUE(Employee_Number, v_num_decimals), v_str_length)
```

テーブルを結合する際にキーフィールドを一致させる

2つのテーブル、`Ar` と `Customer` がある場合に、これらを `CustNo` フィールドで結合してさらに分析する必要があります。これら2つのテーブルにはどちらも `CustNo` フィールドがありますが、このフィールドのデータ書式が異なります。

- **Ar テーブルの CustNo フィールド** - 数値フィールド(値の例 : 235)
- **Customer テーブルの CustNo フィールド** - 5 文字の文字フィールド。数値文字フィールドですが、1 ~ 9 でない先頭の桁にはゼロが追加されます("00235" など)。

結合を行う場合、データ型と長さが等しくなるようにフィールド同士を一致させるには、ZONED() 関数を使って `Ar` テーブルの `CustNo` キーフィールドを長さ5の文字フィールドに変換することで、その書式を `Customer` テーブルの `CustNo` キーフィールドの書式に一致させます。

```
OPEN Ar PRIMARY
OPEN Customer SECONDARY
JOIN PKEY ZONED(CustNo,5) FIELDS CustNo Due Amount SKEY CustNo
UNMATCHED TO Ar_Cust OPEN PRESORT SECSORT
```

備考

機能の仕組み

この関数は数値データを文字データに変換し、先頭にゼロを追加して出力します。この関数は、小切手番号や、発注番号、請求書番号フィールドなどの、先頭にゼロを必要とするフィールドを一致させるためによく利用されます。

ZONED() の使用に適する場面

この関数は、正の数値を先頭にゼロを付けた文字値に変換する場合に使用できます。この関数は、キーフィールドとして使用されるフィールドのデータを正規化する場合に役立ちます。

たとえば、あるテーブルでは請求書番号は数値フィールドに 100 という形式で格納されており、別のテーブルでは文字フィールドに 00100 という形式で格納されているとします。ZONED 関数を使用して、数値 100 を文字値 00100 に変換できます。これで請求書番号を比較することができます。

文字列の長さとの戻り値

長さが数値の桁数より大きい場合は、出力値の先頭にゼロが追加されます。長さが数値の桁数より小さい場合は、出力の左側から切り詰められます。数値と長さが同じ長さの場合、ゼロは追加されません。

10 進数

ZONED データ形式は明示的な小数点を含んでいません。

負の数値

入力値である数値が負の場合には、結果では右端の桁が文字として表示されます。

- 右端の桁が 0 の場合は、文字 "0" として表示
- 右端の桁が 1 ~ 9 の場合は、"J" ~ "R" の間の文字として表示

ZSTAT() 関数

標準 Z 統計量を返します。

構文

```
ZSTAT(実際値, 期待値, 母集団)
```

パラメーター

名前	型	説明
実際値	数値	<ul style="list-style-type: none">パラメーターを数値として指定する場合は、先頭の桁や先頭の桁の組合せなど、実際の数値を指定します。パラメーターを比率として指定する場合は、検査する値の予期される比率は 0 ~ 1 の範囲(0 以上 1 以下)で指定します。
期待値	数値	<ul style="list-style-type: none">パラメーターを数値として指定する場合は、先頭の桁や先頭の桁の組合せなど、予期される数値を指定します。パラメーターを比率として指定する場合は、検査する値の予期される比率は 0 ~ 1 の範囲(0 より大きく 1 未満)で指定します。
母集団	数値	検査する項目の総数。この値は 0 より大きい正の整数でなければなりません。

出力

数値。

例

高度な例

```
値が数値のパラメーター
```

過去 10 年のデータを調べた結果、通常は月ごとの高度障害保険請求件数の分布がほぼ一定であることが判明しました。今年の 4 月、5 月、6 月には請求件数が例年より 10 パーセント多く、例年は 1 か月平均 200 件ですが今年は 220 件ありました。7 月と 8 月の件数はやや少なく、それぞれ 193 件と 197 件でした。今年の請求件数の合計は 2,450 件でした。これらの高い数値結果と低い数値結果が有意なものであったかどうかを調べるには、Z 統計を使用します。

4 月から 6 月までの実際の請求件数は 660 件で予想より高いです。この期間の請求件数の期待値は年間の請求件数である 2,450 件の 25 パーセント、つまり 612.5 件です。これらの統計量に対する Z 統計量は 2.193 と計算されます。

```
ZSTAT(660, 612.5, 2450)
```

Z 統計量 1.96 の有意性は 0.05 で、Z 統計量 2.57 の有意性は 0.01 です。したがって、請求件数が偶然高くなる確率は 1/20 と 1/100 の間です。

7 月から 8 月までの実際の請求件数は 390 件で予想より低いです。この期間の請求件数の期待値は年間の請求件数である 2,450 件の 1/6、つまり 408.33 件です。これらの比率に対する Z 統計量は 0.967 と計算されます。

```
ZSTAT(390, 408.33, 2450)
```

これはあまり有意な結果ではありません。1.000 未満の Z 統計量は非常に一般的で、通常は無視できる範囲の値です。

値が比率のパラメーター

過去 10 年のデータを調べた結果、通常は月ごとの高度障害保険請求件数の分布がほぼ一定であることが判明しました。今年の 4 月、5 月、6 月には請求件数が例年より 10 パーセント多く、例年は 1 か月平均 200 件ですが今年は 220 件ありました。7 月と 8 月の件数はやや少なく、それぞれ 193 件と 197 件でした。今年の請求件数の合計は 2,450 件でした。これらの高い数値結果と低い数値結果が有意なものであったかどうかを調べるには、Z 統計を使用します。

4 月から 6 月までの請求件数の実際値は予想より高く、比率 660/2450 で表されます。この期間の請求件数の期待値は年間の請求件数である 2,450 件の 25 パーセントです。これらの比率に対する Z 統計量は 2.193 です。

```
ZSTAT((1.00000000 * 660 / 2450), 0.25, 2450)
```

Z 統計量 1.96 の有意性は 0.05 で、Z 統計量 2.57 の有意性は 0.01 です。したがって、請求件数が偶然高くなる確率は 1/20 と 1/100 の間です。

7月から8月までの実際の請求件数は390件と少ないです。この期間の請求件数の期待値は年間の請求件数である2,450件の1/6、つまり16.6667パーセントです。これらの比率に対するZ統計量は0.967です。

```
ZSTAT((1.00000000 * 390 / 2450), 0.16667, 2450)
```

これはあまり有意な結果ではありません。1.000未満のZ統計量は非常に一般的で、通常は無視できる範囲の値です。

備考

機能の仕組み

ZSTAT()関数は、デジタル解析をはじめとするさまざまな問題解決のタスクに使用する標準Z統計量を計算します。小数点以下3桁の精度で結果を出力します。

ZSTAT()を使用する

ZSTAT()関数を使用すると、指定した期間やカテゴリで得られる結果についての発生頻度を評価することができます。結果のZ統計量が高いほど発生の可能性は低くなります。

たとえば、Z統計量1.96の有意性は0.05(20回に1回発生すると予測される)であるのに対し、Z統計量2.57の有意性は0.01(100回に1回発生すると予測される)です。Z統計量については、統計関係の書籍を参照してください。

ZSTAT()の入力値を指定する

ZSTAT()のパラメーターには数値か比率のいずれかを指定できます。

- 両方の入力値に数値を指定した場合、この関数は浮動小数点演算を使用してZ統計量を計算します。
- 両方の入力値に比率を指定した場合、この関数は固定小数点演算を使用してZ統計量を計算します。この場合、小数の乗数を使用して、丸めを制御する必要があります。
- 実際値または期待値を求める式の中で式を使用する場合、小数の乗算によって結果の精度を指定する必要があります。Analyticsでは小数点以下8桁が使用されるので、1.00000000の乗算によって、得られる範囲で最高の精度の値が返されます。

アナリティクス スクリプト の概要

スクリプトは Analytics でのみ実行されるように制限されていません。標準のスクリプトは、**アナリティクス スクリプト**に変換すると、HighBond プラットフォーム上のロボット アプリでスケジュールおよび実行できるようになります。

メモ

バージョン 16 以降の Analytics は Analytics Exchange(AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[ロボットにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

アナリティクス スクリプトとは

アナリティクス スクリプトまたは「アナリティクス」は、**アナリティクス ヘッダー**を持つ標準のスクリプトです。アナリティクス ヘッダーは、ロボットでスクリプトの実行を可能にする一連の宣言タグです。アナリティクス ヘッダーには、即時、または予定時刻のいずれかに、無人でのアナリティクス スクリプトの実行を可能にする、ユーザーが事前に設定する入力パラメーターが含まれます。

標準のスクリプトをアナリティクス スクリプトに変換する

アナリティクス スクリプトは標準のスクリプトとして作成を開始します。標準のスクリプトをロボットで実行するには、それをアナリティクス スクリプトに変換する必要があります。

1. Analytics でスクリプトを作成してテストします。
2. 該当するアナリティクス ヘッダー タブを追加し、スクリプトをアナリティクス スクリプトにします。
3. ロボットの開発モードで 1 つ以上のアナリティクス スクリプトおよび補助スクリプトを**スクリプト バージョン**としてコミットし、本番モードの有効なスクリプト バージョンをスケジュールおよび実行します。

詳細については、「アナリティクス スクリプトの開発」ページ 2581を参照してください。

アナリティクス ヘッダーの追加

アナリティクス ヘッダーは、スクリプトの先頭行から始まるコメント ブロック内で定義されます。スクリプトがアナリティクス スクリプトであることをアナリティクス ヘッダーに宣言することが、最低限必要です。

```
COMMENT
```

```
//ANALYTIC 欠落している小切手の識別
```

このアナリティクス スクリプトは、欠落している小切手番号を検出します。
END

詳細については、"アナリティクス ヘッダーを使った作業" ページ 2588を参照してください。

補助スクリプトの概要

補助スクリプトは、アナリティクス スクリプトと連動するように設計されている、**アナリティクス ヘッダーがない**標準のスクリプトです。一般的な設計では、アナリティクス スクリプトは DO SCRIPT コマンドを使用して、1つ以上の補助スクリプトを呼び出します。補助スクリプトが完了すると、処理は実行し続けるアナリティクス スクリプトに戻ります。

補助スクリプトはサブスクリプト、セカンダリスクリプト、ユーティリティスクリプト、ヘルパー スクリプトとも呼ばれます。補助スクリプトの使用は必須ではありません。補助スクリプトでは、条件付き、再利用可能、または親アナリティクス スクリプトを含めるために拡張できるスクリプト ロジックのブロックを区分的にすることができます。

補助スクリプトの制限

補助スクリプトはさまざまな方法で使用できますが、アナリティクス ヘッダーは存在しないため、次の2つの制限事項が適用されます。

- **入出力タグがない** - 入出力アナリティクス タグを指定できません。このため、補助スクリプト自体では、入出力パラメーターを作成できません。必要なパラメーターはすべて、親アナリティクス スクリプトのアナリティクス ヘッダーで作成する必要があります。
- **直接実行できない** - ユーザーは補助スクリプトを直接スケジュールまたは実行できません。直接呼び出すか、別の補助スクリプトから間接的に呼び出すかには関係なく、補助スクリプトはアナリティクス スクリプトからのみ呼び出すことができます。

アナリティクス スクリプト の開発

アナリティクス スクリプト の開発において推奨される方法は、まず Analytics で標準の スクリプト を作成、テストすることです。スクリプト が正しく動作している場合は、アナリティクス ヘッダー を追加し、ロボット で実行可能なアナリティクス スクリプト にスクリプト を変換します。

標準の スクリプト の作成については、「スクリプト の基本」 ページ 1431 を参照してください。

メモ

バージョン 16 以降の Analytics は Analytics Exchange(AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポート が終了します。[詳細を確認](#)するか、[ロボット にアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

スクリプト の入力 と出力 の特定

アナリティクス ヘッダー では、**アナリティクス タグ** を使用して、スクリプト の入力 と、エンド ユーザー を利用可能にしたり、後続の スクリプト の入力 として使用したりする スクリプト の出力 を宣言します。さまざまな種類の入力 と出力 が、括弧内に関連するアナリティクス タグ を伴って以下で説明されています。

ヒント

始める前に、必要な入力 と出力 を特定すると、開発 が一層スムーズに進みます。

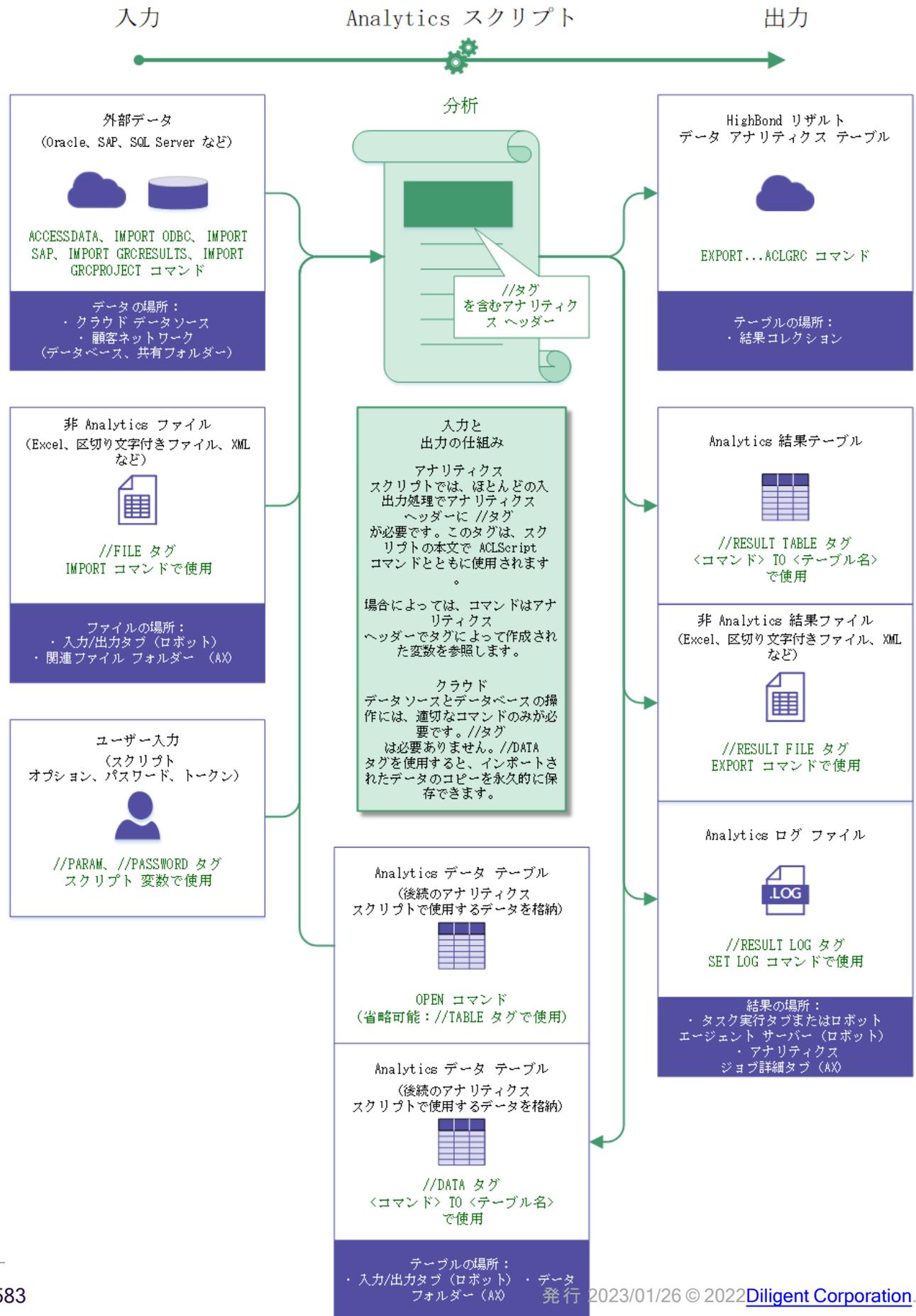
入力	出力
<ul style="list-style-type: none"> Excel ファイルや区切り文字付きファイルなどの非 Analytics ファイル ("FILE タグ" ページ 2613) 切捨額、日付、ID コードなどの入力パラメーター ("PARAM タグ" ページ 2616) パスワード ("PASSWORD タグ" ページ 2626) Analytics のテーブルとフィールド ("TABLE タグ" ページ 2629、"FIELD タグ" ページ 2631) 	<ul style="list-style-type: none"> Analytics および非 Analytics の結果テーブル ("RESULT タグ" ページ 2634) 成功したスクリプトのログファイル ("RESULT タグ" ページ 2634) 後続のスクリプトの入力として使用される Analytics 出力テーブル ("DATA タグ" ページ 2641)

スクリプトの入出力 - 拡大図

次の図は、アナリティクス スクリプト間で受け渡し可能なすべての入力と出力を示します。各入出力は次の項目を示します。

- データ型またはユーザー入力
- 関連付けられたアナリティクス タグ(該当する場合)
- 関連付けられた ACLScript コマンド
- 入出力データ、テーブル、またはファイルの場所

設計と目的によっては、単一のアナリティクス スクリプトに複数の入出力が存在する場合があります。



ソース データへのアクセス

アナリティクス スクリプトによって要求されるソース データにアクセスするための基本的なアプローチは2つあります。

- 接続の自動化
- 手動のアップロード

必要に応じて、同じアナリティクス スクリプトにおいて自由に双方のアプローチが使用できます。

接続の自動化

このアプローチの利点は、スケジュールして実行させるなど、ロボットへのデータのインポートは完全な自動化が可能という点です。

アナリティクス スクリプト本体では、外部のデータソースへの接続、データのインポート、およびデータをコピーした Analytics テーブルの作成において、ACLScript コマンドの1つを使用します。

- [ACCESSDATA](#)
- [IMPORT ODBC](#)
- [IMPORT GRCRESULTS](#)
- [IMPORT GRCPROJECT](#)
- [DEFINE TABLE DB](#)

メモ

これらのコマンドでは、アナリティクス ヘッダーで対応するアナリティクス タグは必要ありません。

他のコマンドのいずれかを使用する理由がない限り、ACCESSDATA を使用してください。DEFINE TABLE DB は、昔のスクリプトを使って下位互換性で保持されている昔のコマンドです。

手動のアップロード

手動のアップロードは、ロボットにデータをインポートするシンプルな方法を提供します。ユーザーがソース データ ファイルをローカルに保管している場合にこの方法が適切かもしれません。

Excel や区切り文字などの非 Analytics ファイルを手動でロボット アプリにアップロードできます。Analytics テーブルを利用可能にするには、異なる方法を使用する必要があります。

- **非 Analytics ファイル**- Excel または区切り文字などの非 Analytics ファイルを、ロボットで **入力/出力** タブに手動でアップロードできます。アナリティクス スクリプトでアップロードしたデータにアクセスするには、アナリティクス ヘッダーでは [FILE](#) タグを使用し、スクリプト本体では [IMPORT EXCEL](#) などの適切なインポート コマンドを使用します。
- **Analytics テーブル**- Analytics テーブルは、**入力/出力** タブに手動でアップロードすることはできません。代わりに、アナリティクス ヘッダーで [DATA](#) タグを使用し、Analytics 出力テーブルを **入力/出力** タブに保存します。後続のスクリプトで Analytics テーブルにアクセスするには、スクリプト本体で [OPEN](#) コマンドを使用します。

アナリティクス スクリプト の作成 とテスト のワークフロー

メモ

次のワークフローをアナリティクス スクリプト 開発方法として推奨しますが、ご自分に最も適した方法で自由にアナリティクス スクリプトを開発することができます。

Analytics スクリプトを作成する

ユーザー入力のためのカスタム ダイアログ ボックスや、その他スクリプトの実行中にユーザー操作を必要とする機能を一切使用せずに、Analytics でスクリプトを作成します。アナリティクス スクリプトは、スクリプトの実行に先立ってユーザー入力をできるようにしますが、スクリプトとは異なり、実行中のユーザー操作はサポートしていません。

Analytics スクリプトにテスト入力値を保存するには、スクリプトの先頭で、変数を一時的に作成します。例：

```
ASSIGN v_AnalysisTable = "Trans_May"
```

スクリプトがエラーなしで実行されるまで、スクリプトをテストしてデバッグします。

アナリティクス ヘッダーおよびアナリティクス タグの追加

スクリプトにアナリティクス ヘッダーを追加します。スクリプトの先頭からアナリティクス ヘッダー デザイナー内の対応するタグへ変数名をコピーします。

アナリティクス ヘッダーの結果タグの例：

```
//TABLE v_AnalysisTable "Table to classify"
```

詳細については、「アナリティクス ヘッダーを使った作業」ページ 2588を参照してください。

アナリティクス スクリプト 結果にログを含める

ログは、アナリティクス スクリプトの失敗の原因を診断する重要なツールです。ログは、アナリティクス スクリプトが成功しても予期しない結果となった場合にも重要となります。ログは、アナリティクス スクリプトが失敗した場合には自動的に出力されますが、アナリティクス スクリプトが成功した場合は、**RESULT** アナリティクス タグを指定した場合にのみ出力されます。

アナリティクス ヘッダー デザイナーでは、**ログファイルを保持する**]をオンにして、アナリティクス スクリプトを実行するときはいつでもログを利用できるようにします。アナリティクス ヘッダーに対応するタグを加えます。

```
//RESULT LOG
```

アナリティクス ヘッダーを検証する

アナリティクス ヘッダーを検証します。必要とする頻度でアナリティクス ヘッダーを検証することができます。詳細については、「アナリティクス ヘッダーの検証」 ページ 2594を参照してください。

アナリティクス タグに一時的なテスト値を代入する

特殊な代入演算子(`:=`)を使用して、ユーザー入力を必要とするすべての分析タグに一時的なテスト値を代入します。スクリプトの先頭にある一時的な変数代入からテスト値をコピーすることができます。例：

```
//TABLE v_AnalysisTable "分類するテーブル" := "Trans_May"
```

アナリティクス ヘッダー デザイナーを使用して一時的にテスト値を割り当てるには、ユーザーの入力に必要なすべてのアナリティクス タグに対して **テスト値** フィールドに値を入力します。

一時的なテスト値の詳細については、「Analytics でのテスト入力値の指定」 ページ 2607を参照してください。

一時変数を削除する

スクリプトの先頭から一時変数を削除するか、または、それらを再度使用するかもしれない場合は、一時変数をコメントにします。

アナリティクス スクリプトをステップ実行する

実行  をクリックするか、または F10 キーを繰り返し押して、アナリティクス スクリプトをステップ実行します。ナビゲーターの **変数** タブの内容をよく調べて、アナリティクス ヘッダー内のすべての変数が正しく作成され、テスト値が正しく割り当てられていることを確認します。

アナリティクス スクリプトがエラーなしで実行されるまで、テストしてデバッグします。

メモ

アナリティクス スクリプトが完了する前にそのアナリティクス スクリプトを終了したい場合は、Esc キーを押し、確認プロンプトで **[はい]** をクリックします。

ヒント

コマンド ラインで `DELETE ALL OK` と入力することにより、Analytics プロジェクトからすべての格納された変数と変数の代入値を削除することができます。ステップ実行する前に **変数** タブをクリアすると、アナリティクス スクリプトをクリーンに開始できます。

一時的なテスト値を削除する

テストが完了したら、すべてのアナリティクスタグから、一時的なテスト値と特殊な代入演算子を削除できます。または、追加のテストが必要な場合があると想定する場合は、これらを維持します。開発環境ではテスト値は無視されます。

アナリティクス スクリプト の展開

アナリティクス スクリプト を展開するには、スクリプト をロボット にコミット します。

詳細については、"Analytics からロボット へのスクリプト のコミット (アップロード)" ページ 2731 を参照してください。

アナリティクス ヘッダーを使った作業

アナリティクス ヘッダーは、スクリプトの始めにコメント ブロックに囲まれた一連のタグです。タグは、スクリプトの入力またはスクリプトの出力のいずれかを指定します。

アナリティクス スクリプトの入出力の視覚的な概要と関連付けられたアナリティクス タグについては、「スクリプトの入出力 - 拡大図」 ページ 2582を参照してください。

ユーザーが直面する入力パラメータを指定するタグは、ユーザーが事前にスクリプトの入力値を指定できるようにします。すなわち、即時、またはスケジュールした時間のいずれかにアナリティクス スクリプトは無人で実行できるということです。

Analytics プロジェクトでスクリプトを開発したあとは、スクリプトをロボットにコミットする前に、1 つ以上のスクリプトにアナリティクス ヘッダーを追加する必要があります。

アナリティクス ヘッダー デザイナーを使用するのが、アナリティクス ヘッダーを追加または変更する最も簡単な方法です。アナリティクス ヘッダーは、手動で追加したり、変更したりすることもできます。

メモ

バージョン 16 以降の Analytics は Analytics Exchange (AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[ロボットにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

アナリティクス ヘッダー デザイナー

アナリティクス ヘッダー デザイナーには、アナリティクス ヘッダーを構成するアナリティクス タグを進行的に追加する上で直感的なインターフェイスがあります。アナリティクス ヘッダーを構築しながら、自由にタグを追加、変更、または削除することができます。

自動エラー チェックおよびデザイナーに埋め込まれたガイダンスは、確実に、構築するヘッダーが有効で、かつ正確に動作するよう支援します。

デザイナーで **保存** をクリックすると、設定したタグは、スクリプトのトップでアナリティクス ヘッダーに自動的に解釈されます。必要に応じてアナリティクス ヘッダーを手動で編集できますが、デザイナーを再度開いて編集を行うことが推奨されます。



サンプルのアナリティクス ヘッダー

見つからない小切手を特定するアナリティクス スクリプトの、以下に示すアナリティクス ヘッダーが、上に示すアナリティクス ヘッダー デザイナーのタグを使用して作成されました。スペース節約のため、デザイナーのスクリーン キャプチャは、アナリティクス ヘッダーでタグのサブセットのみを表示するようサイズが変更されま

```
COMMENT
//ANALYTIC TYPE ANALYSIS 欠落している小切手の識別
```

```
このアナリティクス スクリプトは、欠落している小切手番号を検出します。
//TABLE v_table_payments 支払テーブル
  支払が記載され、小切手番号の列があるテーブルを選択します
//FIELD v_check_num CN 小切手番号
  小切手番号を含むフィールドを選択します
//PARAM v_start_date D OPTIONAL 開始日(任意)
  分析の開始日を入力します
//PARAM v_end_date D OPTIONAL 終了日(任意)
  分析の終了日を入力します
//PARAM v_region C MULTI SEPARATOR , QUALIFIER ' VALUES
|Northeast|Southeast|Central|Western|West Coast| Regions
  分析に含める 1 つ以上の地域を入力します
//RESULT TABLE Missing_Checks
//RESULT FILE Missing_Checks.xls
//RESULT LOG
END

COMMENT スクリプト本体はここから始まります。
SET SAFETY OFF
OPEN %v_table_payments%
.
.
.
SET SAFETY ON
```

ロボットでのスクリプトの入力の状況

上述のサンプルのアナリティクス ヘッダーにある入力タグは、ロボットでアナリティクス スクリプトをスケジュールまたは実行する際にユーザーが投入する必要のある入力パラメータを作成します。

入力パラメーターがロボット アプリにどのように表示されるかを次に示します。

スクリプトを選択する

分析

欠落している小切手の識別
 パラメーター
2省略可能, 2必須
▲

この分析は、欠落している小切手番号を識別します

開始日 (省略可能)

📅 2019/10/01

分析の開始日を入力します

終了日 (省略可能)

📅 2019/12/31

分析の終了日を入力します

地域

東北 ▼

分析に含める 1つ以上の地域を入力します

テーブル

支払テーブル

AP_2019 ▼

支払が記載され、小切手番号の列があるテーブルを選択します

小切手番号

Cheque_No ▼

小切手番号を含むフィールドを選択します

各タグの実行内容

上述のサンプルのアナリティクス ヘッダー内の各アナリティクスタグは、ロボットでユーザーが関連するアナリティクス スクリプトをスケジュールまたは実行する際に特定のタスクを実行します。

アナリティクス ヘッダーの構文	説明
<pre>COMMENT . . . END</pre>	<p>アナリティクス タグのブロックを囲みます。</p> <p>すべてのアナリティクス ヘッダーは、スクリプトの先頭行から始まる <code>COMMENT</code> コマンドによって囲まれている必要があります。</p>
<p>分析</p>	<p>アナリティクス スクリプト のタイプと名前を含む、アナリティクス ヘッダーの基本構成を作成します。</p> <p>すべてのアナリティクス ヘッダーは、 <code>//ANALYTIC</code> タグで始まる必要があります。</p>
<pre>//TABLE v_table_payments</pre>	<p>ユーザーによる支払テーブルの選択を可能にする入力パラメータを作成します。</p> <p>テーブル名は多様であるため、ユーザーが選択するテーブル名は <code>v_table_payments</code> 変数に格納されます。</p>
<pre>//FIELD v_check_num</pre>	<p>ユーザーによる支払テーブルからの小切手番号フィールドの選択を可能にする入力パラメータを作成します。</p> <p>フィールド名は多様であるため、ユーザーが選択するフィールド名は <code>v_check_num</code> 変数で並び替えられます。</p>
<pre>//PARAM v_start_date</pre>	<p>分析されるレコード範囲の開始日のユーザーによる指定を可能にする、入力パラメータを作成します。</p> <p>ユーザーはさまざまな開始日を指定するため、ユーザーが指定した実際の日付は <code>v_start_date</code> 変数に格納されます。</p>
<pre>//PARAM v_end_date</pre>	<p>分析されるレコード範囲の終了日のユーザーによる指定を可能にする、入力パラメータを作成します。</p> <p>ユーザーはさまざまな終了日を指定するために、ユーザーが指定する実際の日付は <code>v_end_date</code> 変数で並び替えられます。</p>
<pre>//PARAM v_region</pre>	<p>ユーザーによる分析に含まれる地域の指定を可能にする入力パラメータを作成します。</p> <p>ユーザーはさまざまな地域を指定するために、ユーザーが指定する実際の地域は <code>v_region</code> 変数で並び替えられます。</p>
<pre>//RESULT TABLE Missing_Checks</pre>	<p>Missing_Checks 結果テーブルをロボットでユーザーが利用できるように指定する出力パラメータを作成します。</p> <p>スクリプトからの出力結果は、存在していても、自動的に利用可能にはなりません。アナリティクス ヘッダーで可用性を指定する必要があります。</p>
<pre>//RESULT FILE Missing_Checks.xls</pre>	<p>Missing_Checks.xls 結果ファイルをロボットでユーザーが利用できるように指定する出力パラメータを作成します。</p> <p>スクリプトからの出力結果は、存在していても、自動的に利用可能にはなりません。アナリティクス ヘッダーで可用性を指定する必要があります。</p>

アナリティクス ヘッダーの構文	説明
<div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;">//RESULT LOG</div>	<p>ログファイルが正常に実行するスクリプト用の出力であることを指定します。</p> <p>ログファイルは、スクリプトが失敗した場合に自動的に出力されます。</p>

アナリティクス ヘッダーの作成

アナリティクス ヘッダーを作成するには、必要なスクリプトの入力と出力は何かを事前に知っておく必要があります。詳細については、「スクリプトの入力と出力の特定」ページ 2581を参照してください。

アナリティクス ヘッダーの基本構成の設定

1. スクリプト エディターで新規または既存のスクリプトを開きます。
2. **アナリティクス ヘッダーの編集**  をクリックします。
アナリティクス ヘッダー デザイナーが開きます。
3. **アナリティクス タイプ** を選択します。
アナリティクス スクリプトは、ロボットでタイプ別にグループ化されます。グループはスクリプト シーケンスでユーザーをガイドします。
 - **IMPORT** - データソースからデータを取得するスクリプト。
 - **PREPARE** - 生データを分析に適したものにするために必要な方法で生データを変換するスクリプト。
 - **ANALYSIS** - データで分析を実行するスクリプト。
4. **アナリティクス名** を指定します。

メモ

Analytics プロジェクトのアナリティクス スクリプト名は一意でなくてはなりません。名前はロボットでアナリティクス スクリプトを識別します。このアナリティクス スクリプト名は、スクリプトを初めに作成したときに、Analytics で指定するスクリプト名とは同じではありません。

5. 正常に終了したスクリプトのログファイルを維持するかどうかを選択します。
 - **ログファイルを保持する** をオンにする - スクリプトが正常に実行された場合にログファイルが自動的に出力されます
 - **ログファイルを保持する** をオフにする - スクリプトが正常に実行された場合にログファイルは出力されません

ログファイルを維持する]の設定にもかかわらず、ログファイルは、スクリプトが失敗した場合には自動的に出力されます。

ヒント

正常に終了したスクリプト用のログファイル名をカスタマイズする場合、`RESULT LOG` タグを使用します。

アナリティクス タグの追加

アナリティクス ヘッダーの基本構成を設定したら、必要なだけ多くのアナリティクス タグを追加できます。

タグを追加する順序に制限はありません。

1. アナリティクス ヘッダー デザイナーで、**タグを追加する**]をクリックします。
2. **タグタイプ**]を選択します。
3. タグを構成するには、タグの構成で必要なフィールドのすべて、および必要なオプションのフィールドを完了します。

タグの構成に関するガイダンスは、各タグの構成セクションに埋め込まれています。

アナリティクス ヘッダー構文の詳細と、アナリティクス タグの完全な一覧については、「アナリティクス ヘッダーおよびアナリティクス タグ」ページ 2605を参照してください。

4. アナリティクス ヘッダーで必要な各追加タグのプロセスを繰り返します。
5. 終了したら **保存**]をクリックします。

アナリティクス ヘッダーの検証

1つ以上のスクリプトにアナリティクス ヘッダーを追加したら、Analytics のツールを使用して、ヘッダー構文を検証し、それが正しいことを確認します。アナリティクス スクリプトの実行時に失敗しないように、スクリプトをロボットにコミットする前に検証を行います。

1つのツールは、スクリプト レベルで個別のアナリティクス ヘッダーを検証します。他のツールは、1度に1つのプロジェクトのすべてのアナリティクス ヘッダーを検証します。2つの種類の検証の焦点は異なります。

個別のアナリティクス ヘッダーを検証する

アナリティクス ヘッダーのスクリプト レベルの検証は、個別のアナリティクス タブの構文に焦点を置き、エラーと該当する行番号を報告します。

1. アナリティクス ヘッダーを含んでいるスクリプトを開きます。
2. スクリプト エディター ツールバーで、**アナリティクス ヘッダーの検証**]をクリックします 。

アナリティクス ヘッダーが有効であることを知らせるメッセージか、または発生したエラーと発生場所の行番号を示すメッセージが表示されます。

3. アナリティクス ヘッダーにエラーがある場合は、エラーを修正してからもう一度 **アナリティクス ヘッダーの検証**]をクリックし、これ以上エラーがないことを確認します。

ヒント

エラーの本質がエラーメッセージからは明らかでない場合は、関連するアナリティクス タグのヘルプトピックを参照してください。トピックの構文をアナリティクス ヘッダーの構文と注意して比較します。エラーは、アナリティクス ヘッダー構文のわずかな違いが原因で発生することもあります。

プロジェクトのすべてのアナリティクス ヘッダーを検証する

アナリティクス ヘッダーのプロジェクト レベルの検証は次の2つのことを確認します。

- 1つ以上のアナリティクス ヘッダーがプロジェクトに存在する
- 複数のアナリティクス スクリプトの名前が一意である

メモ

アナリティクス スクリプト名は、ナビゲーターの **概要** タブのスクリプト名ではなく、**ANALYTIC** タグで指定された名前です。

スクリプトをロボットにコミットするときに、プロジェクト レベルの検証が自動的に実行されます。 **スクリプトの確認**  ボタンを Analytics ツールバーに追加する場合に、手動で検証を実行することもできます。

1. 必要に応じて、 **スクリプトの確認** ボタンを Analytics ツールバーに追加します。
 - a. ツールバー上の何もない場所をダブルクリックすると、 **ツールバーの変更** ダイアログ ボックスが開きます。
 - b. **利用できるツールバー ボタン** リストで、 **スクリプトの確認** を選択し、 **追加** をクリックします。
 - c. **現在のツールバー ボタン** リストで、 **スクリプトの確認** ボタンを選択し、 **上へ** または **下へ** をクリックして、ボタンの位置を変更します。
上から下へのボタンの順序は、ツールバーの左から右への位置に対応しています。
 - d. **閉じる** をクリックして、変更を保存します。
2. ツールバーで、 **スクリプトの確認**  をクリックします。
プロジェクトのアナリティクス ヘッダーが有効であるというメッセージが表示されるか1つ以上のエラーを指定します。
3. アナリティクス ヘッダーにエラーがある場合は、エラーを修正してからもう一度 **スクリプトの確認**  をクリックし、これ以上エラーがないことを確認します。

アナリティクス開発のベスト プラクティス

アナリティクス スクリプトでは、標準のスクリプトで使用できるコマンドのほとんどがサポートされています。ただし、開発者としては、アナリティクス スクリプトをユーザーの操作なしに実行できるようにすると共に、HighBond のロボット アプリのアナリティクス スクリプトを処理するエンジンでサポートされていないコマンドをアナリティクス スクリプトに記述しないようにする必要があります。

アナリティクス スクリプトではすべての ACLScript 関数がサポートされます。

一般的なベストプラクティス

ロボットごとに1つの Analytics プロジェクトを使用する

ロボットごとに新しい Analytics プロジェクトを作成します。プロジェクトには、ロボットを構成するすべてのアナリティクス スクリプトと、必要となるすべてのサブスクリプトが含まれている必要があります。

ローカルでテストする

どのアナリティクス スクリプトも、ロボット アプリに展開する前に、ローカルでテストします。アナリティクス スクリプトが想定通りに実行され、ユーザー操作が必要でないことを確認します。

詳細については、「アナリティクス スクリプトの開発」ページ 2581を参照してください。

テストに安定したデータ接続を使用する

ODBC データソースを使用しているかどうかについてローカルでアナリティクス スクリプトをテストするには、お使いのローカルコンピューター上で、アナリティクス スクリプトが実行する環境の接続と一致する、ODBC 接続を構成する必要があります。

絶対ファイルパスの使用を避ける

アナリティクス スクリプトが実行される環境に同一のファイルパスが存在していることが確実な場合以外、アナリティクス スクリプトで絶対ファイルパス(`C:\results` など)を使用することは避けてください。

相対ファイルパス(`\results`など)を使用すると、アナリティクス スクリプトをローカルで開発してテストし、他の環境に同じディレクトリ構造がなくても、別の環境に展開できます。

SET を使って初期設定を行う

SET コマンドを使用して、アナリティクススクリプトで必要となるすべての優先順位の設定を指定します。アナリティクス スクリプトの初期設定をしない場合は、Analytics のデフォルトの初期設定が使用されます。アナリティクス ヘッダーの後ろ、アナリティクス スクリプト ロジックの前に SET コマンドを置きます。

結果テーブルやデータ出力テーブルでは演算フィールドは使用禁止

アナリティクス スクリプトを実行するセッション以外のセッションでも保持したい出力テーブル内では、演算フィールドを使用しないでください。

解釈内での使用目的に、あるいは後続のスクリプトの入力として保持される結果テーブルやデータテーブルが演算フィールドを含んでいる場合には、予期しない値が表示される可能性があります。演算値は初期設定ファイル(.prf)に定義されている設定によって決まるか、または SET コマンドを使って作成されるため、環境に応じて、出力される値が異なってきます。

演算フィールドの値を保持する必要がある場合は、EXTRACT コマンドを FIELDS または ALL オプションとともに使用することで、そのフィールドを結果テーブルやデータテーブルの物理フィールドに変換します。詳細については、「EXTRACT コマンド」ページ 1801を参照してください。

データ接続のパスワードを暗号化する

アナリティクス スクリプトにデータソースのパスワードをプレーンテキストで設定しないようにするには、//PASSWORD アナリティクスタグを使用します。このタグは、アナリティクス スクリプトを実行する前に、パスワードの入力をユーザーに求め、入力された値を暗号化します。

HighBond との間でインポートまたはエクスポートを行うときにパスワードを使用する

PASSWORD パラメーターは、HighBond との間でインポートまたはエクスポートを行うすべてのコマンドで必要です。

- IMPORT GRCRESULTS
- IMPORT GRCPROJECT
- EXPORT... ACLGRC
- EXPORT... HBDATA

ロボット アプリでは、コマンドは PASSWORD パラメーターを指定しないと失敗します。

コマンド内で PASSWORD パラメーターを使用するときには、アナリティクス ヘッダーにある関連の //PASSWORD タグも指定する必要があります。詳細については、「PASSWORD タグ」ページ 2626を参照してください。

メモ

`PASSWORD` パラメーターは、Analytics でインポート およびエクスポートコマンドを実行するときには不要です。Windows レジストリにローカル保存されている現在のユーザーの HighBond アクセストークンが自動的に使用されます。

ユーザー操作の回避

アナリティクス スクリプトは、ユーザーの操作なしに実行できるようにしておく必要があります。コマンドがダイアログボックスを作成しようとする、開発環境のエンジンがアナリティクス スクリプトを停止し、エラーがログに記録されます。

ユーザー操作コマンドをアナリティクス タグに置き換える

ユーザー操作を必要とする Analytics コマンドを使用しないようにします。アナリティクス ヘッダーの同等のアナリティクス タグで置換します。アナリティクス タグでは、アナリティクス スクリプトを実行する前に、ユーザーが入力値を指定できます。

使用しない	置換
<code>DIALOG</code>	<code>//PARAM</code> 、 <code>//TABLE</code> 、 <code>//FIELD</code>
<code>ACCEPT</code>	<code>//PARAM</code> 、 <code>//TABLE</code> 、 <code>//FIELD</code>
<code>PASSWORD</code>	<code>//PASSWORD</code>
<code>PAUSE</code>	相当するオプションなし

ガイドライン

- **対話型コマンド** - アナリティクス スクリプト 処理が失敗しないようにするには、すべての対話型コマンドを削除しておきます。
- **SET SAFETY** - 必要に応じてファイルを上書きできるようにする(確認ダイアログボックスの表示なし)には、アナリティクス スクリプトの冒頭に `SET SAFETY OFF` コマンドを追加します。

アナリティクス スクリプトの終わりに `SET SAFETY ON` コマンドを追加することで、デフォルトの動作に戻しておくことができます。

- **OK パラメーター** - 確認ダイアログによってアナリティクス スクリプトがクラッシュしないようにするには、通常、確認ダイアログボックスを表示する以下のコマンドの後に `OK` パラメーターを追加します。
 - `RENAME`
 - `DELETE`

スクリプト 構文を確認する

Analytics は、アナリティクス スクリプト が失敗する原因となるスクリプト 構文、またはローカル環境とアナリティクス スクリプト が展開される環境との間で調整が必要になるスクリプト 構文を検出するツールを提供しています。このツールでは警告のみが表示されます。警告のあるアナリティクス スクリプト でも、コミットまたはインポートできます。

ツールの確認機能

ツールは、次の項目について、プロジェクトのすべてのスクリプトを確認します。

- ユーザー操作を必要とするコマンド
- 絶対ファイルパス
- 外部スクリプトの呼び出し

確認が実行されるタイミング

スクリプト 構文 確認は、スクリプトをロボットにコミットするときに、自動的に実行されます。

構文の確認はデフォルトで有効です。オフにする場合は、**[オプション]** ダイアログボックス(**[ツール > オプション > インターフェイス]**) で、**[スクリプトをコミットする前に、スクリプト 構文チェックを無効にする]** を選択します。

確認を手動で実行する

スクリプト 構文の確認を手動で実行できます。Analytics ツールバーに**スクリプトの確認**  ボタンを最初に追加しなければならない場合があります。

1. 必要に応じて、**[スクリプトの確認]** ボタンを Analytics ツールバーに追加します。
 - a. ツールバー上の何もない場所をダブルクリックすると、**[ツールバーの変更]** ダイアログボックスが開きます。
 - b. **[利用できるツールバーボタン]** リストで、**[スクリプトの確認]** を選択し、**[追加]** をクリックします。
 - c. **[現在のツールバーボタン]** リストで、**[スクリプトの確認]** ボタンを選択し、**[上へ]** または **[下へ]** をクリックして、ボタンの位置を変更します。
上から下へのボタンの順序は、ツールバーの左から右への位置に対応しています。
 - d. **[閉じる]** をクリックして、変更を保存します。

2. ツールバーで、**スクリプトの確認**  をクリックします。
プロジェクトのスクリプト構文が有効であること、または1つ以上の警告を示すメッセージが表示されます。
3. 次のいずれかを実行します。
 - 警告があるスクリプト構文を修正し、**スクリプトの確認**  をもう一度クリックして、警告が表示されないことを確認します。
 - 展開環境にディレクトリ構造、またはアナリティクス スクリプトで指定されたパスまたは外部スクリプトと一致する外部スクリプトがあることを確認します。

アナリティクス スクリプト のサンプル

アナリティクス スクリプト のサンプルには、1つのスクリプトのインポート、1つの準備スクリプト、そして1つの分析スクリプトが含まれています。アナリティクス スクリプトはロボットで実行できます。

アナリティクス スクリプト の順序

3つのアナリティクス スクリプトは、合わせて作業するために設計され、次の順序で実行する必要があります。

順番検査	アナリティクスタイプ	アナリティクス スクリプト名
1	IMPORT	インポート アナリティクス Robots_AX のサンプル
2	PREPARE	準備アナリティクスのサンプル
3	ANALYSIS	分析アナリティクスのサンプル

インポート アナリティクス スクリプト のサンプル

サンプルの Excel ファイル [Trans_May.xls](#) からデータをインポートし、それを新しい Analytics テーブル [Trans_May_raw](#) (生データ テーブル) に保存します。

アナリティクス スクリプト名	使用場所	インポート ファイルの要件
インポート アナリティクス Robots_AX のサンプル	<ul style="list-style-type: none"> ロボット 	<ul style="list-style-type: none"> ロボット -の Trans_May.xls ファイルはアナリティクス スクリプトと同じロボットの 入力/出力 タブにある必要があります

ロボット用 インポート アナリティクス スクリプト のサンプル

```
COMMENT
//ANALYTIC TYPE IMPORT インポート アナリティクス ロボット AX のサンプル
  このアナリティクス スクリプトは、サンプル Excel ファイル Trans_May.xls からデータをインポート
  し、それを新しい Analytics テーブル「Trans_May_raw」(生データ テーブル)に保存します。
//FILE Trans_May.xls
//DATA Trans_May_raw
//RESULT LOG
END

SET SAFETY OFF
```

```

IMPORT EXCEL TO Trans_May_raw Trans_May_raw.fil FROM "Trans_May.xls" TABLE
"Trans2_May$" KEPTITLE FIELD "CARDNUM" C WID 22 AS "" FIELD "CODES" C WID 4
AS "" FIELD "DATE" D WID 10 PIC "YYYY-MM-DD" AS "" FIELD "CUSTNO" C WID 6 AS
"" FIELD "DESCRIPTION" C WID 95 AS "" FIELD "AMOUNT" N WID 9 DEC 2 AS ""
SET SAFETY ON

```

準備アナリティクス スクリプトのサンプル

分析対象となる生データテーブルを準備し、それを新しい Analytics テーブル **Trans_May_prepared** (分析テーブル) に保存します。分類化では最大 64 文字のフィールド長しかサポートされないため、このアナリティクス スクリプトでは "Description" フィールドの短い方のバージョンを定義します。

```

COMMENT
//ANALYTIC TYPE PREPARE 準備アナリティクスのサンプル
このアナリティクス スクリプトは、分析対象となる生データ テーブルを準備し、それを新しい
Analytics テーブル Trans_May_prepared(分析テーブル)に保存します。分類化では最大 64
文字のフィールド長しかサポートされないため、このアナリティクス スクリプトでは "Description"
フィールドの短い方のバージョンを定義します。
//TABLE v_RawTable 準備するテーブル
準備する生データ テーブルを選択します
//RESULT TABLE Trans_*_prepared
//DATA Trans_*_prepared
//RESULT LOG
END

SET SAFETY OFF
OPEN %v_RawTable%
DEFINE FIELD DESC_SHORT ASCII 43 64
EXTRACT RECORD TO "Trans_May_prepared"
SET SAFETY ON

```

サンプル分析アナリティクス スクリプト

分析テーブルを分類化し、その結果を新しい Analytics テーブル **Classified_Trans_May_prepared** (結果テーブル) に出力します。ユーザーは、テーブルを分類化するために使用するフィールドを指定できます。また、処理されるレコードを制限するために、商業カテゴリコード、顧客番号と、日付および取引金額の範囲を指定することができます。

```

COMMENT
//ANALYTIC TYPE ANALYSIS 分析アナリティクスのサンプル
このアナリティクス スクリプトは分析アナリティクスのサンプル - 分析テーブルを分類化し、その結

```

果を新しい Analytics テーブル "Classified_Trans_May_prepared" (結果テーブル) に出力します。処理されるレコードを制限するために、商業カテゴリ コード、顧客番号と、日付および取引金額の範囲を指定できます。

```
//TABLE v_AnalysisTable 分類化するテーブル
  分類化する分析テーブルを選択します
//FIELD v_FieldA C 分類化の対象とするフィールド
  分類したいフィールドを選択します
//PARAM v_codes C MULTI SEPARATOR , QUALIFIER ' VALUES |4112 Passenger
Railways|4121 Taxis/Limousines|4131 Bus travel|4215 Courier Services - Air
or Ground|4411 Cruise Lines|4457 Boat Leases and Boat Rentals|4722 Travel
Agencies and Tour Operations|4814 Local/long-distance calls|5812
Restaurants|5813 Drinking Places (Alcoholic Beverages)|5814 Fast food
restaurants|5921 Package Stores, Beer, Wine, Liquor|5993 Cigar Stores &
Stands|5994 Newsstand|7216 Dry cleaners| MC codes to include
  含める商業カテゴリ コードを 1 つ以上指定します
//PARAM v_cust_no C OPTIONAL MULTI SEPARATOR , QUALIFIER ' 除外する顧客番号
(省略可能)
  除外する顧客番号を 1 つ以上指定します。番号の指定後に Enter を押して、各番号を新しい行に移動します。番号を引用符で囲まないでください。
//PARAM v_start_date D VALUES
|05/01/2003|05/02/2003|05/03/2003|05/04/2003|05/05/2003|05/06/2003|05/07/200-
3|05/08/2003|05/09/2003|05/10/2003|05/11/2003|05/12/2003|05/13/2003|05/14/20-
03|05/15/2003|05/16/2003|05/17/2003|05/18/2003|05/19/2003|05/20/2003|05/21/2-
003|05/22/2003|05/23/2003|05/24/2003|05/25/2003|05/26/2003|05/27/2003|05/28/-
2003|05/29/2003|05/30/2003|05/31/2003|Start Date
  開始日を選択します
//PARAM v_end_date D 終了日
  終了日を入力するか、またはカレンダーから終了日を選択します
//PARAM v_min_amount N 最低額
  最低額を入力します
//PARAM v_max_amount N 最高額
  最高額を入力します
//RESULT TABLE Classified_*
//RESULT LOG
END

SET SAFETY OFF
OPEN %v_AnalysisTable%
IF NOT ISDEFINED("v_cust_no") v_cust_no = ""
GROUP IF v_cust_no = ""
  CLASSIFY ON %v_FieldA% IF MATCH(CODES, %v_codes%) AND BETWEEN(DATE, v_
start_date, v_end_date) AND BETWEEN(AMOUNT, v_min_amount, v_max_amount)
SUBTOTAL AMOUNT TO "Classified_%v_AnalysisTable%.FIL" OPEN
ELSE
  CLASSIFY ON %v_FieldA% IF MATCH(CODES, %v_codes%) AND NOT MATCH(CUSTNO,
%v_cust_no%) AND BETWEEN(DATE, v_start_date, v_end_date) AND BETWEEN(AMOUNT,
v_min_amount, v_max_amount) SUBTOTAL AMOUNT TO "Classified_%v_
```

```
AnalysisTable%.FIL" OPEN  
END  
SET SAFETY ON
```

アナリティクス ヘッダーおよびアナリティクス タグ

アナリティクス ヘッダーは、Analytics スクリプトの最初にコメント ブロックで囲まれた一連のタグです。アナリティクス タブは、アナリティクス スクリプトの実行またはスケジュール前にユーザーが入力する入力パラメーターと出力パラメーターを指定します。

アナリティクス ヘッダーは、ロボットで実行する予定の、アナリティクス スクリプトで必要です。

アナリティクス スクリプトの入出力の視覚的な概要と関連付けられたアナリティクス タグについては、「スクリプトの入出力 - 拡大図」ページ 2582を参照してください。

メモ

バージョン 16 以降の Analytics は Analytics Exchange (AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[ロボットにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

アナリティクス ヘッダーの追加または変更

アナリティクス ヘッダーを追加または変更する最も簡単な方法は、アナリティクス ヘッダー デザイナーを使用することです。デザイナーに備わる自動エラー チェックおよび埋め込まれているガイダンスを使用することで、作成するヘッダーが有効であり、かつ正確に動作するようにできます。詳細については、「アナリティクス ヘッダーを使った作業」ページ 2588を参照してください。

また、アナリティクス ヘッダーを作成する別の方法として、必要な ANALYTIC タグを手動で入力するという方法もあります。あるいは、手動入力とデザイナーを組み合わせることもできます。

基本的なアナリティクス ヘッダーの要件

アナリティクス ヘッダーは、スクリプトの先頭行から始まるコメント ブロック内に定義する必要があります。次を除き、タグは任意の順序でアナリティクス ヘッダーに配置できます。

- ANALYTIC タグは先頭に配置する必要があります
- FIELD タグは関連付けられた TABLE タグの直後に配置する必要があります

例

このアナリティクス ヘッダーは、スクリプトで使用するテーブルおよびフィールドと開始日パラメーターを特定します。

```
COMMENT
//ANALYTIC 欠落している小切手の識別
    このアナリティクス スクリプトは、欠落している小切手番号を検出します。
//TABLE v_table_payments 支払テーブル
    支払が記載され、小切手番号の列があるテーブルを選択します
//FIELD v_check_num CN 小切手番号
    小切手番号を含むフィールドを選択します
//PARAM v_start_date D OPTIONAL 開始日(任意)
    分析の開始日を入力します
END
```

タグ形式

ヘッダーの各タグは次の形式を使用します。

```
//タグ名 属性
    任意の説明テキスト
```

// タグ インジケータは、スクリプト行でスペース以外の最初の文字でなければなりません。タグ名は、タグインジケータの直後に、間にスペースも文字も入れずに続ける必要があります。

任意の説明テキストは、タグより後の次の行に入力する必要があります。説明を複数行で指定することはできますが、行を飛ばすことはできません。説明テキストをロボットで表示する場合には、改行は保持されません。

タグ表記

コンポーネント	表記
タグ名	タグ名は大文字と小文字が区別されません。 Analytics のコマンド名と関数名とは異なり、タグ名は省略できません。
タグ属性	タグの属性を指定するときには、スペースを含め、任意で引用符で値を囲むことができます。
タグ説明	説明は省略可能です。説明を指定する場合には複数行を指定できますが、ロボットでは改行が保持されません。

Analytics でのテスト入力値の指定

特殊割り当て演算子 `:=` を使用し、定義が必要なアナリティクス タグのテスト値を指定できます。

- FILE
- PARAM
- TABLE
- FIELD

次の構文を使用して、Analytics でのアナリティクス スクリプトの実行とテストを行います。

```
//TABLE v_AnalysisTable "分類するテーブル" := "Trans_May"
```

Analytics でスクリプトを実行するときには、パラメーターは割り当てで指定された値を取得します。アナリティクス スクリプトがロボットで実行されるときには、テスト値が無視され、ユーザー定義入力パラメーターが使用されます。

割り当て演算子とその前のタグ構文の間にはスペースを残す必要があります。割り当て値は、Analytics 全体に必要なデータ型に適切な修飾子を使用する必要があります。詳細については、「データ型」ページ 1457を参照してください。

使用可能なアナリティクス タグの完全な一覧

タグ	説明
"ANALYTIC タグ" ページ 2609	スクリプトをロボットで実行可能なアナリティクス スクリプトとして指定します。
入力タグ	
"FILE タグ" ページ 2613	ロボットで実行するアナリティクス スクリプトで使用する入力を提供する、Excel ファイルや区切り文字付きファイルなどの、Analyticsファイル以外のファイルを指定します。このファイルはアナリティクス スクリプトと同じロボットの 入力/出力 タブにある必要があります。
"PARAM タグ" ページ 2616	アナリティクス スクリプトのための入力パラメーターを作成し、その入力値の要件を定義します。 入力パラメーターは、ユーザーがアナリティクス スクリプトを実行またはスケジュールする際に使用する実際の値を指定できるようにするプレースホルダーです。
"PASSWORD タグ" ページ 2626	アナリティクス スクリプトのパスワード入力パラメーターを作成します。パラメーターは、ACLScript コマンドで後から使用するための暗号化されたパスワードの格納場所を提供します。 ユーザーは、アナリティクス スクリプトをスケジュールまたは開始するときに、必要なパスワード値を指定するように要求されます。これにより、アナリティクス スクリプトを実行するときにユーザーの介入が不要になります。
"TABLE タグ" ページ 2629	ユーザーがアナリティクス スクリプトの入力として選択する Analytics テーブルを定義します。 <code>TABLE</code> タグの後には、順次行に入力されたゼロ個以上の <code>FIELD</code> タグを続けることができます。

タグ	説明
"FIELD タグ" ページ 2631	ユーザーがアナリティクス スクリプトの入力として選択するフィールドを定義します。 このフィールドは、先行する TABLE タグで定義したテーブルに含まれます。最初の FIELD タグは、 TABLE タグの直後に続ける必要があります。その後続けて、追加の FIELD タグを連続した行に入力できます。
出カタグ	
"RESULT タグ" ページ 2634	エンドユーザーがロボットで、アナリティクス スクリプトで出力された結果を使用できるように指定します。 出力結果は、存在していても、自動的に利用可能にはなりません。
"DATA タグ" ページ 2641	アナリティクス スクリプトによって出力された Analytics テーブルを、ロボットの中央データ保存場所にコピーすることを指定します。 通常は、後続のアナリティクス スクリプトの入カテーブルとして利用できるよう、Analytics テーブルを格納します。

ANALYTIC タグ

スクリプトをロボットで実行可能なアナリティクススクリプトとして指定します。

構文

```
//ANALYTIC <TYPE IMPORT|PREPARE|ANALYSIS> 名前
<説明>
```

メモ

ACLScript の `COMMENT` コマンドはアナリティクススクリプトの1行目に入力し、その後2行目に `ANALYTIC` タグを続ける必要があります。ほかの場所で `ANALYTIC` タグが使用されている場合、そのタグは無視されます。

Analytics プロジェクト内の1つ以上のスクリプトに `ANALYTIC` タグを含めることができます。

パラメーター

名前	説明
TYPE 省略可能	<p>アナリティクススクリプトを次の3つのタイプのいずれかとして指定します。</p> <ul style="list-style-type: none"> <code>IMPORT</code> データソースからデータを取得します。インポート アナリティクススクリプトの出力は、生のデータテーブルです。 <code>PREPARE</code> 生データを分析に適したものにするために必要な方法で生データを変換します。準備アナリティクススクリプトの出力は、分析テーブルです。 <code>ANALYSIS</code> 分析テーブルのデータでテストを実行します。分析アナリティクススクリプトの出力は、1つまたは複数の結果テーブルです。 <p>指定したタイプは、アナリティクススクリプトの実際の内容に対する制限がありません。スクリプト作成者は、指定したタイプとスクリプトの内容がどのように一致するかを制御します。</p> <p>指定したタイプのアナリティクススクリプトは、ロボットの対応する領域でグループ化されます。この領域は次のように並べて表示されます。</p> <ul style="list-style-type: none"> インポート 準備 分析 <p><code>TYPE</code> を省略した場合は、アナリティクススクリプトが <code>分析</code> 領域に表示されます。</p> <p>詳細については、「アナリティクススクリプト タイプと名前を使用して、一連のスクリプトを順番に並べ替える」ページ 2612を参照してください。</p>
名前	<p>アナリティクススクリプトの名前。</p> <p>名前はロボットでアナリティクススクリプトを識別します。このアナリティクススクリプト名は、スク</p>

名前	説明
	<p>リプトを初めに作成したときに、Analytics で指定するスクリプト名とは同じではありません。</p> <p>メモ 同じプロジェクトのアナリティクス スクリプト名は一意でなければなりません。複数のアナリティクス スクリプトで同じ名前が使用されていると、アナリティクス スクリプトをコミットしようとしたときにエラーが発生します。</p> <h3>アナリティクス スクリプト名でサポートされている文字</h3> <p>アナリティクス スクリプト名で処理の問題が発生しないことを保証するために、名前では次の文字のみを使用することがベストプラクティスです。</p> <p>A-Z、a-z、0-9、アンダースコア(<code>_</code>)、ダッシュ(<code>-</code>)</p> <h3>アナリティクス スクリプト名でサポートされていない文字</h3> <p>名前では次の文字を使用しないでください。次の文字はサポートされていません。</p> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p>< > : " / \ ? *</p> </div> <p>名前として <code>TYPE</code> という値を使用しないでください。</p> <h3>アナリティクス スクリプト名で連続のプレフィックスを使用する</h3> <p>ロボットでは、アナリティクス スクリプト名は英数字順にリストされます。複数のアナリティクス スクリプトを正しい順序で実行するようユーザーを導くには、アナリティクス スクリプト名を順序付けるためのプレフィックスを追加します。例: <code>A_01_analyze_POs</code>、<code>A_02_analyze_invoices</code> など。詳細については、「アナリティクス スクリプト タイプと名前を使用して、一連のスクリプトを順番に並べ替える」ページ 2612を参照してください。</p>
説明 省略可能	<p>アナリティクス スクリプトの説明や、アナリティクス スクリプトを正常に実行するために必要と思われるその他の情報の説明。</p> <p>この説明は、ロボット内にアナリティクス スクリプトと一緒に表示されます。説明を複数行で指定することはできますが、行を飛ばすことはできません。説明は、関連する <code>ANALYTIC</code> タグの下に行に入力する必要があります。</p>

例

基本的なアナリティクス ヘッダー

次のアナリティクス ヘッダーには、アナリティクス スクリプトの名前と説明が表示されています。

```
COMMENT
//ANALYTIC 欠落している小切手の識別
    このアナリティクス スクリプトは、欠落している小切手番号を検出します。
END
```

アナリティクス ヘッダーおよびアナリティクス タイプ

次のアナリティクス ヘッダーは、準備アナリティクス スクリプトと、スクリプトの機能の説明を指定しています。

```
COMMENT
//ANALYTIC TYPE PREPARE 住所データの標準化
    このアナリティクス スクリプトは、重複分析の準備として、住所フィールドの不要なデータを
    クリーンアップして標準化します。
END
```

アナリティクス ヘッダーと追加のアナリティクス タグ

次のアナリティクス ヘッダーには追加のアナリティクス タグが含まれます。

```
COMMENT
//ANALYTIC TYPE IMPORT 取引データをインポート
    このアナリティクス スクリプトは、Excel ファイル Trans_May.xls からデータをインポート
    し、それを新しい Analytics テーブル「Trans_May_raw」(生データ テーブル)に保存しま
    します。
//FILE Trans_May.xls
//DATA Trans_May_raw
//RESULT LOG
END
```

備考

アナリティクス スクリプト タイプと名前を使用して、一連のスクリプトを順番に並べ替える

`TYPE`および `name` パラメーターを使用して、ロボットのタスクで一連のアナリティクス スクリプトを順番に並べ替えることができます。

メモ

`TYPE` および `name` パラメーターは、アナリティクス スクリプトの順序にのみ影響します。
`DO SCRIPT` コマンドを使用して呼び出される補助スクリプトやヘルパー スクリプトには影響しません。

<code>TYPE</code> パラメーター	次の順序で、アナリティクス スクリプトをタスクの個別の領域にグループ化します。 <ul style="list-style-type: none"> ○ インポート ○ 準備 ○ 分析
<code>name</code> パラメーター	タスクまたはタスクの領域で、アナリティクス スクリプトを名前のアルファベット順に並べ替える
アナリティクス スクリプト 実行シーケンス	実行シーケンスが適用されます。 シーケンスは、 <code>name</code> または <code>TYPE</code> と <code>name</code> で規定される順序に従います。

FILE タグ

ロボットで実行するアナリティクス スクリプトで使用する入力を提供する、Excel ファイルや区切り文字付きファイルなどの、Analyticsファイル以外のファイルを指定します。このファイルはアナリティクス スクリプトと同じロボットの **[入力/出力]** タブにある必要があります。

構文

```
//FILE ファイル名
<説明>
```

パラメーター

名前	説明
ファイル名	<p>アナリティクス スクリプトの入力ファイルとして使用する、ロボットの入力/出力タブにあるファイルの名前。</p> <p>メモ ファイル名値は、入力/出力タブのファイル名と完全に一致する必要があります。ファイル名にパスを含めることはできません。 名前一致を支援するために、ファイル名の中でワイルドカード文字を使用することができます。 ファイル名には変数を使用できません。</p> <p>ファイル名でサポートされていない文字</p> <p>ファイル名にはスペースを使用しないでください。 ファイル名では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>< > : " \ / </pre> <p>ワイルドカード文字</p> <p>ファイル名を指定する際には、ワイルドカードを使用できます。連続する0個以上の文字の代わりに単一のアスタリスク(*)を使用します。</p> <p>例:</p> <ul style="list-style-type: none"> Inv12* は、Inv12、Inv123、Inv1234 のすべてと一致します。 *.* はすべての拡張子と一致します

名前	説明
	<ul style="list-style-type: none"> ◦ <code>Inv_*.*</code> は <code>Inv_Jan.pdf</code> および <code>Inv_Feb.xls</code> と一致します。 ◦ <code>*.xlsx</code> は <code>.xlsx</code> 拡張子のすべて Excel ファイルと一致します
説明 省略可能	<p>Analytics ファイル以外のファイルやその他の情報に関する説明テキスト。説明を複数行で指定することはできませんが、行を飛ばすことはできません。</p> <p>説明は、アナリティクス ヘッダーにのみ存在するものであり、ロボットのエンド ユーザーには表示されません。</p>

例

基本的な例

特定のファイルを指定します。

```
//FILE FlaggedAccounts.csv
```

"Flagged" で始まるすべての CSV ファイルを指定します。

```
//FILE Flagged*.csv
```

すべてのファイルを指定します。

```
//FILE *.*
```

高度な例

関連ファイルからのデータのインポート

毎月ロボットの入力/出力タブに手動でアップロードされる区切り文字付きファイルに含まれる、従業員データの月次分析を実行するとします。ロボットのアナリティクス スクリプトでは、区切り文字付きファイル内のデータを Analytics テーブルにインポートします。

```
COMMENT
//ANALYTIC TYPE IMPORT employee_import
```

[入力/出力]タブに格納されている区切り文字付きファイル内の従業員レコードをインポートします。

```
//FILE Employees.csv
```

```
END
```

```
IMPORT DELIMITED TO Employees "Employees.fil" FROM "Employees.csv" 0  
SEPARATOR "," QUALIFIER "'" CONSECUTIVE STARTLINE 1 KEPTITLE FIELD  
"First_Name" C AT 1 DEC 0 WID 11 PIC "" AS "First Name" FIELD "Last_  
Name" C AT 12 DEC 0 WID 12 PIC "" AS "Last Name"
```

備考

Analytics ファイル以外のファイルをスクリプト内で使用するには、まず Analytics テーブルにインポートする必要があります。Analytics ファイル以外のファイルをスクリプト内で直接使用することはできません。

PARAM タグ

アナリティクス スクリプトのための入力パラメーターを作成し、その入力値の要件を定義します。

入力パラメーターは、ユーザーがアナリティクス スクリプトを実行またはスケジュールする際に使用する実際の値を指定できるようにするプレースホルダーです。

構文

```
//PARAM 変数名 型 <OPTIONAL> <MULTI> <SEPARATOR 値> <QUALIFIER 値> <VALUES
値リスト> ラベル
<説明>
```

パラメーター

名前	説明
変数名	<p>ユーザーによって選択または指定された入力値を格納する変数の名前。変数名をアナリティクス スクリプト内で使用することで、入力値を参照できます。</p> <p>例:</p> <ul style="list-style-type: none"> v_start_date v_regions v_input_file <p>パラメーターの一意の識別子にもなります。</p> <p>メモ</p> <p>アナリティクス スクリプトが実行されたとき、ユーザーが入力値を提供する場合にのみ、この変数は作成されます。パラメーターが省略可能で、ユーザーが入力をスキップした場合、変数は作成されません。</p> <p>アナリティクス スクリプトの以降のロジックで変数の存在が必要となる場合は、変数の存在をテストすることができるので、存在しなければ、変数を作成して初期化します。詳細については、"省略可能な入力パラメーターの設計" ページ 2622を参照してください。</p> <p>パラメーター変数名でサポートされていない文字</p> <p>変数名にはスペースを使用しないでください。</p> <p>変数名では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>` ~ ! @ # \$ % ^ & * () - + = { } [] \ : ; ' " < > , . / ?</pre>

名前	説明				
<p>種類</p>	<p>パラメーターのデータ型。これにより、入力できる入力値の種類を制御します。</p> <p>次の型を、大文字を使用して指定することができます。</p> <ul style="list-style-type: none"> ○ C- 文字データ ○ N- 数値データ ○ D- 日付時刻データの日付サブタイプ ○ DT- 日付時刻データの日付時刻サブタイプ ○ T- 日付時刻データの時刻サブタイプ ○ L- 論理データ <p>メモ アナリティクス スクリプトが正常に実行されるようにするには、文字の入力値を修飾する必要があります。</p>				
<p>OPTIONAL 省略可能</p>	<p>パラメーターは任意であり、ユーザーは値を入力しなくてもいいことを示します。</p> <p>詳細については、"省略可能な入力パラメーターの設計" ページ 2622を参照してください。</p>				
<p>MULTI 省略可能</p>	<p>パラメーターは複数の入力値を受け入れることを示します。</p> <p>メモ MULTI は、型が L(論理名) または F(ファイル) の場合には使用できません。</p> <p>MULTI と VALUES</p> <p>MULTI は、VALUES オプションの有無にかかわらず使用できます。</p> <table border="1" data-bbox="516 1066 1414 1283"> <tbody> <tr> <td data-bbox="516 1066 784 1178"> <p>MULTI ✓</p> <p>VALUES ✓</p> </td> <td data-bbox="784 1066 1414 1178"> <p>ユーザーは値のリストから1つ以上の値を選択できます。</p> </td> </tr> <tr> <td data-bbox="516 1178 784 1283"> <p>MULTI ✓</p> <p>VALUES ✗</p> </td> <td data-bbox="784 1178 1414 1283"> <p>ユーザーは1つ以上の値を手動で入力できます。</p> </td> </tr> </tbody> </table> <p>詳細については、"MULTI および VALUES オプションの要約" ページ 2623を参照してください。</p> <p>複数文字入力値</p> <p>MULTI を指定し、型が C(文字) である場合は、入力値の文字列に区切り文字とテキスト修飾子が自動的に挿入されるよう、SEPARATOR と QUALIFIER オプションを指定することもできます。</p> <p>メモ アナリティクス スクリプトが正常に実行されるようにするには、複数の文字の入力値を区切ったり修飾したりする必要があります。区切り文字および修飾子は自動的に挿入するか、ユーザーが手動で挿入できます。</p>	<p>MULTI ✓</p> <p>VALUES ✓</p>	<p>ユーザーは値のリストから1つ以上の値を選択できます。</p>	<p>MULTI ✓</p> <p>VALUES ✗</p>	<p>ユーザーは1つ以上の値を手動で入力できます。</p>
<p>MULTI ✓</p> <p>VALUES ✓</p>	<p>ユーザーは値のリストから1つ以上の値を選択できます。</p>				
<p>MULTI ✓</p> <p>VALUES ✗</p>	<p>ユーザーは1つ以上の値を手動で入力できます。</p>				
<p>SEPARATOR 値 省略可能</p>	<p>SEPARATOR は、MULTI が指定され、型が C(文字) も場合にのみ使用できます。</p> <p>処理するアナリティクス スクリプトに渡される区切り文字付きの一覧を作成するときに、複数の文字入力値の間に区切り文字を自動的に挿入することを指定します。</p> <p>値には、使用する区切り文字を指定します。よく使われる分離記号、すなわち区切り文字</p>				

名前	説明										
	<p>はカンマ,です。</p> <p>SEPARATORを省略した場合は、デフォルトで、1つのスペースが区切り文字として使用されます。スペース文字を値として指定することはできません。</p> <p>詳細については、"文字入力値の区切りと修飾" ページ 2623を参照してください。</p>										
<p>QUALIFIER 値 省略可能</p>	<p>QUALIFIER は、MULTIが指定され、型がC(文字)も場合にのみ使用できます。</p> <p>処理するアナリティクス スクリプトに渡される区切り文字付きの一覧を作成するときに、各文字入力値の始まりと終わりにテキスト修飾子の文字を自動的に挿入することを指定します。修飾子文字内に囲まれたあらゆるテキストが、プレーンテキストとして扱われます。</p> <p>値には、使用する修飾子文字を指定します。よく使われる修飾子は一重引用符'です。</p> <p>QUALIFIERを省略した場合に使用されるデフォルトの修飾子はありません。スペース文字を値として指定することはできません。</p> <p>詳細については、"文字入力値の区切りと修飾" ページ 2623を参照してください。</p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p>メモ</p> <p>アナリティクスの入力パラメーターは現在のところ、テキスト修飾子として二重引用符(")の使用をサポートしていません。代わりに、一重引用符(')を使用することができます。二重引用符を指定すると、PARAM タグが正しく動作しなくなります。</p> </div>										
<p>VALUES 値リスト 省略可能</p>	<p>ユーザーがアナリティクス スクリプトを実行するときを選択できる値のリスト。</p> <p>次の構文を使用して、値を指定します。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0; text-align: center;"> VALUES 値 1 値 2 値 3 値 n </div> <p>VALUES と MULTI</p> <p>VALUES は MULTI オプションの有無にかかわらず使用できます。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td style="padding: 5px;"> VALUES ✓ MULTI ✓ </td> <td style="padding: 5px;">ユーザーは値のリストから1つ以上の値を選択できます。</td> </tr> <tr> <td style="padding: 5px;"> VALUES ✓ MULTI ✗ </td> <td style="padding: 5px;">ユーザーは値のリストから単一の値を選択できます。</td> </tr> </tbody> </table> <p>詳細については、"MULTI および VALUES オプションの要約" ページ 2623を参照してください。</p> <p>値リストの値の形式</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td style="width: 20%; padding: 5px;">文字値</td> <td style="padding: 5px;"> <ul style="list-style-type: none"> ◦ スペースと句読点を含めることができます </td> </tr> <tr> <td style="padding: 5px;">数値</td> <td style="padding: 5px;"> <ul style="list-style-type: none"> ◦ 正数でも負数でも指定できます。 ◦ 10進表記法を用いて、桁区切り記号を付けずに指定する必要があります。 <p style="margin-left: 20px;">例: 1500.00 または -1500.00</p> </td> </tr> <tr> <td style="padding: 5px;">日付時刻値</td> <td style="padding: 5px;"> <ul style="list-style-type: none"> ◦ 日付 -は、書式 MM/DD/YYYY を使用して指定する必要があります </td> </tr> </tbody> </table>	VALUES ✓ MULTI ✓	ユーザーは値のリストから1つ以上の値を選択できます。	VALUES ✓ MULTI ✗	ユーザーは値のリストから単一の値を選択できます。	文字値	<ul style="list-style-type: none"> ◦ スペースと句読点を含めることができます 	数値	<ul style="list-style-type: none"> ◦ 正数でも負数でも指定できます。 ◦ 10進表記法を用いて、桁区切り記号を付けずに指定する必要があります。 <p style="margin-left: 20px;">例: 1500.00 または -1500.00</p>	日付時刻値	<ul style="list-style-type: none"> ◦ 日付 -は、書式 MM/DD/YYYY を使用して指定する必要があります
VALUES ✓ MULTI ✓	ユーザーは値のリストから1つ以上の値を選択できます。										
VALUES ✓ MULTI ✗	ユーザーは値のリストから単一の値を選択できます。										
文字値	<ul style="list-style-type: none"> ◦ スペースと句読点を含めることができます 										
数値	<ul style="list-style-type: none"> ◦ 正数でも負数でも指定できます。 ◦ 10進表記法を用いて、桁区切り記号を付けずに指定する必要があります。 <p style="margin-left: 20px;">例: 1500.00 または -1500.00</p>										
日付時刻値	<ul style="list-style-type: none"> ◦ 日付 -は、書式 MM/DD/YYYY を使用して指定する必要があります 										

名前	説明
	<div style="border: 1px solid black; padding: 5px;"> <p>(例: 12/31/2014)</p> <ul style="list-style-type: none"> 日付時刻 -は、書式 MM/DD/YYYY hh:mm:ss を使用して指定する必要があります <p>(例: 12/31/2014 23:59:59)</p> <ul style="list-style-type: none"> 時刻 -は、書式 hh:mm:ss を使用して指定する必要があります(例: 23:59:59) </div> <p>論理値 VALUE は、型が <code>LOCAL</code> の場合には使用できません。</p>
ラベル	<p>パラメーターのユーザー インターフェイスのラベル。</p> <p>ロボットで、ラベルは入力フィールドで表示されます。</p>
説明 省略可能	<p>パラメーターに関する詳細情報を提供する説明テキスト。</p> <p>ロボットで、説明は入力フィールドで表示されます。</p> <p>説明はユーザーを支援する手順を提供できます。たとえば、"給与支払期間の締切日を入力してください"などと指定します。</p> <p>説明は、関連する <code>PARAM</code> タグより後の次の行に入力する必要があります。説明を複数行で指定することはできますが、行を飛ばすことはできません。ロボットで表示する場合には、改行は保持されません。</p>

例

基本的な例

ユーザーが任意で日付の範囲を指定できるようにする

```
//PARAM v_start_date D OPTIONAL 開始日(任意)
    分析の開始日を入力します
//PARAM v_end_date D OPTIONAL 終了日(任意)
    分析の終了日を入力します
```

ユーザーに処理する取引の最大数を選択するように要求する

```
//PARAM v_maxTrans N VALUES |250|500|750|1000| 処理する最大取引数
```

ユーザーに商業カテゴリコードを1つ以上指定するように要求する

```
//PARAM v_codes C MULTI SEPARATOR , QUALIFIER ' 含める MC コード
商業カテゴリ コードを 1 つ以上 指定します。コードの後に Enter を押して、各コードが新しい
行に移動するようにします。コードを引用符で囲まないでください。
```

ユーザーに商業カテゴリコードを1つ以上選択するように要求する

```
//PARAM v_codes C MULTI SEPARATOR , QUALIFIER ' VALUES |4121 タクシー/リムジン
|5812 レストラン|5813 飲み屋 - アルコール飲料|5814 ファスト フード レストラン| 含める MC
コード
商業カテゴリ コードを 1 つ以上 選択します。
```

高度な例

ユーザーに金額の範囲を指定するように要求する

最低金額 ~ 最高金額の範囲内に含まれる、テーブル内のレコードを分類化する必要があります。この範囲は変更される場合があるため、スクリプト作成者は、アナリティクススクリプトを実行するユーザーがスクリプトをスケジュールまたは実行する際にこの範囲を定義できるようにする入力パラメーターを提供します。

```
COMMENT
//ANALYTIC テスト アナリティクス
//PARAM v_min_amount N 最低額
最低額を入力します
//PARAM v_max_amount N 最高額
最高額を入力します
END

CLASSIFY ON %v_FieldA% IF BETWEEN(AMOUNT, v_min_amount, v_max_amount)
SUBTOTAL AMOUNT TO "Classified_%v_AnalysisTable%.FIL"
```

ユーザーが任意で1つ以上の顧客番号を除外できるようにする

テーブル内のレコードを分類化する必要がある場合に、ユーザーに分析から特定の顧客を除外できるようにさせたいとします。

これを行うには、スクリプト作成者はオプションの文字パラメーターを提供します。作成するスクリプトでは、顧客番号の値が指定されたかどうかをテストし、指定されている場合には、指定された顧客番号を分類化コマンドの対象から除外します。

```
COMMENT
//ANALYTIC テスト アナリティクス
//PARAM v_cust_no C OPTIONAL MULTI SEPARATOR , QUALIFIER ' 除外する顧客
番号 (省略可能)
    1 つ以上の顧客番号を指定します。番号の指定後に Enter を押して、各番号を新し
い行に移動します。番号を引用符で囲まないでください。
END

IF FTYPE("v_cust_no") = "U" v_cust_no = ""
GROUP IF v_cust_no = ""
    CLASSIFY ON %v_FieldA% SUBTOTAL AMOUNT TO "Classified_%v_
AnalysisTable%.FIL"
ELSE
    CLASSIFY ON %v_FieldA% IF NOT MATCH(CUSTNO, %v_cust_no%) SUBTOTAL
AMOUNT TO "Classified_%v_AnalysisTable%.FIL"
END
```

オプションのパラメーターへのデフォルト値の使用

取引レコードを結果テーブルに抽出するアナリティクス スクリプトを作成するとします。スクリプトを実行するユーザーに対し、日付範囲を提供するだけでなく、抽出するレコードをフィルタリングするためのエンティティの一覧をも提供するオプションを用意する必要があります。

そのようにするには、以下の3つのオプション パラメーターを作成します。

- v_start_date
- v_end_date
- v_entity_list

スクリプトの冒頭の数行で、これらの値が設定されているかどうかをテストします。設定されていない場合は、最早日付および最遅日付のデフォルト値と、v_entity_list に対してテストするというデフォルト フラグを設定します。

EXTRACT コマンドでは、レコードをフィルタリングする値を使用します。

```
COMMENT
//ANALYTIC テスト
    このアナリティクス スクリプトは PARAM をテストします
//RESULT TABLE t_results
//PARAM v_start_date D OPTIONAL 開始日を入力
//PARAM v_end_date D OPTIONAL 終了日を入力
```

```
//PARAM v_entity_list C MULTI OPTIONAL |entity1|entity2|
END

IF NOT ISDEFINED("v_start_date") v_start_date = `19000101`
IF NOT ISDEFINED("v_end_date") v_end_date = `99991231`
IF NOT ISDEFINED("v_entity_list") v_entity_list = "'all'"

EXTRACT FIELDS ALL TO t_results IF BETWEEN(transaction_date v_start_
date v_end_date) AND (MATCH(entity_field %v_entity_list%) OR v_entity_
list = "'all'")
```

備考

省略可能な入力パラメーターの設計

`PARAM` タグで `OPTIONAL` を使用した場合、その入力パラメーターに関連付けられた変数は、アナリティクス スクリプトの実行時に作成されるかどうかわかりません。

- ユーザーが入力値を指定した場合は、**変数が自動的に作成** されます
- ユーザーが省略可能なパラメーターをスキップし、入力値を指定しなかった場合には、**変数は作成されません**。

パラメーター変数の存在のテスト

アナリティクス スクリプトの以降のロジックが、パラメーター変数の内容を評価できるものとしている場合は、それが空や NULL の状態を評価するものであっても、パラメーター変数の存在をテストする必要があります。パラメーター変数が存在しない場合は、作成して、ヌルに初期化する必要があります。

IF コマンドと FTYPE() 関数または ISDEFINED() 関数を使用し、テストを実行して、存在しない場合は変数を作成します。

```
IF FTYPE("変数名") = "U" 変数名 = ""
```

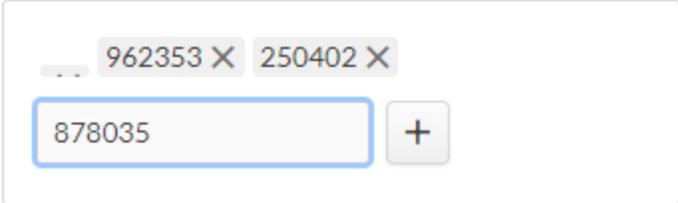
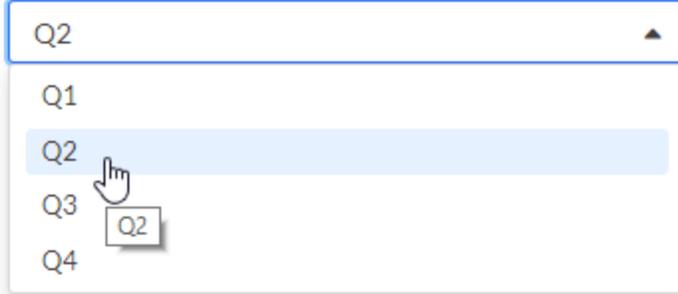
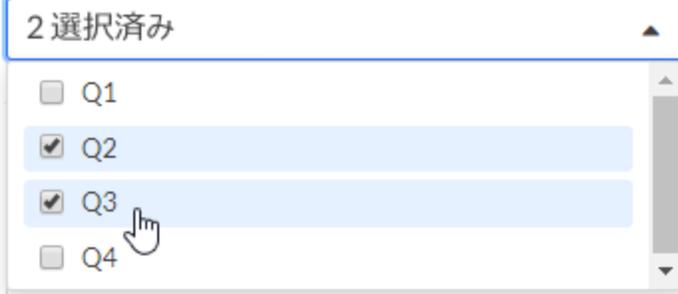
```
IF NOT ISDEFINED("var_name") var_name = ""
```

テストを実行するとき

アナリティクス ヘッダーより後、パラメーター変数の存在に依存する Analytics スクリプトのロジックより前に、テストを実行します。

MULTI および VALUES オプションの要約

以下の表は、ユーザー インターフェイスのユーザー入力コントロールに対する、MULTI および VALUE オプションの効果の要約です。

ユーザー入力コントロール(ロボット)	パラメーター設計	MULTI	VALUES
	フィールドに単一の入力値を手動で入力する	✗	✗
	フィールドに1つ以上の入力値を手動で入力する	✓	✗
	ドロップダウンリストの値から単一の入力値を選択する	✗	✓
	チェックリストの値から1つ以上の入力値を選択する	✓	✓

文字入力値の区切りと修飾

文字入力値が複数存在する場合は、値を区切り文字で区切る必要があります。また、アナリティクス スクリプトを正常に実行させるために、値を修飾する必要があります。

入れ子になったテキスト修飾子を使用しない

文字の入力パラメーターを作成するとき、またアナリティクス スクリプトを実行するユーザーに文字入力値の入力方法を指示するときには、余分なテキスト修飾子、つまり入れ子になったテキスト修飾子(修飾子の中の修飾子)を作成しないように注意する必要があります。余分なテキスト修飾子により、入力パラメーターが正しく動作しなくなります。

テキスト修飾子を挿入する方法

文字入力値の前後にテキスト修飾子を挿入するための方法は4種類あります。方法によっては、入力値の間に区切り文字も挿入されます。

アナリティクス スクリプトを開発するときには、別の方法で実験し、ユーザーが入力する文字値に適切なものを見つけなければならない場合があります。

メモ

MULTI オプションと VALUES オプションをどのように使用しているかによって、1つまたは複数の方法を適用できない場合があります。

各入力パラメーターはこれらの方法の**1つのみ**を使用する必要があります。

1	SEPARATOR および QUALIFIER を使用する	<p>PARAM タグに SEPARATOR および QUALIFIER オプションを含めます。</p> <p>例:</p> <pre>//PARAM v_regions C MULTI SEPARATOR , QUALIFIER '</pre> <p>MULTI なしで VALUES を使用する場合は適用されません。</p> <p>ヒント 可能な限り、この方法を使用します。これは、必要とする労力が最も少なく、また、エラーを起こす度合いが最も低いです。</p>
2	区切り文字と修飾子を手動で指定する	<p>アナリティクス スクリプトのユーザーに、実際の入力値に加えて区切り文字と修飾子を手動で指定するように要求します。</p> <p>例:</p> <pre>'North America', 'Europe', 'Asia'</pre> <p>MULTI の有無にかかわらず、VALUES を使用する場合は適用されません。</p>
3	値リストに修飾子を含める	<p>VALUES オプションと一緒に指定される値リストの各値に修飾子を含めます。</p> <p>例:</p> <pre>VALUES 'Asia' 'Europe' 'Middle East' 'North America' </pre>

		VALUES なしで MULTI を使用する場合は適用されません。
4	パラメーター変数を修飾子で囲む	Analytics スクリプトの構文で、パラメーター変数をテキスト修飾子で囲みます。 例： <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> IF MATCH(REGIONS, "%v_regions%") </div> この方法は、MULTI なしで VALUES を使用している場合にのみ使用します。
	メモ アナリティクスの入力パラメーターは現在のところ、テキスト修飾子として二重引用符(")の使用をサポートしていません。その代わりに、値リストで、あるいは入力値の前後に手動で修飾子を指定するときは、QUALIFIER オプションによって一重引用符(')を使用できます。二重引用符は、Analytics スクリプトの本文ではテキスト修飾子として使用できます。	

別の方法を使用するとき

以下の表には、テキスト修飾子を挿入するさまざまな方法をどのような場合に使用するかまとめてあります。

	MULTI / VALUES	MULTI / VALUES	MULTI / VALUES
方法 1 SEPARATOR および QUALIFIER オプションを使用する	使用した場合は、方法 2 を使用しない	該当なし	使用した場合は、方法 3 を使用しない
方法 2 区切り文字と修飾子を手動で指定する	使用した場合は、方法 1 を使用しない	該当なし	該当なし
方法 3 値リストに修飾子を含める	該当なし	使用した場合は、方法 4 を使用しない	使用した場合は、方法 1 を使用しない
方法 4 パラメーター変数を修飾子で囲む	使用しない	使用した場合は、方法 3 を使用しない	使用しない

PASSWORD タグ

アナリティクス スクリプト のパスワード入力パラメーターを作成します。パラメーターは、ACLScript コマンドで後から使用するための暗号化されたパスワードの格納場所を提供します。

ユーザーは、アナリティクス スクリプト をスケジュールまたは開始するときに、必要なパスワード値を指定するように要求されます。これにより、アナリティクス スクリプト を実行するときにユーザーの介入が不要になります。

構文

```
//PASSWORD 識別子 ラベル
<説明>
```

パラメーター

名前	説明
識別子	パスワードに関連付ける数値識別子。値は 1 から 10 までのいずれかでなければなりません。
ラベル	ロボットで、パスワードを入力するように指示されたときにユーザーに表示されるインターフェイスラベル。例: <code>SAP パスワード:</code>
説明 省略可能	ロボットで、ユーザーに表示される必須パスワードに関連付けられた説明テキスト。この説明により、ユーザーはパスワードを入力できます。 説明を複数行で指定することはできますが、行を飛ばすことはできません。説明は、関連する <code>PASSWORD</code> タグの下の子の入力する必要があります。

例

Direct Link SAP クエリのパスワード入力パラメーターを作成する

アナリティクス ヘッダーは、SAP パスワードを入力するようにユーザーに指示するパスワード入力パラメーターを指定します。保存されたパスワードは、スクリプトの本文の後続の RETRIEVE コマンドで使用されます。

```
COMMENT
//ANALYTIC SAP パスワードの例
//PASSWORD 1 SAP パスワード:
//DATA RSADMIN
END
SET SAFETY OFF
RETRIEVE RSADMIN PASSWORD 1
OPEN RSADMIN
SET SAFETY ON
```

メモ

RETRIEVE コマンドのパスワード入力パラメーターとパスワードパラメーターは、同じ数値 ID を使用してリンクされます。

```
//PASSWORD 1 SAP パスワード:
.
.
.
RETRIEVE RSADMIN PASSWORD 1
```

リザルトにエクスポートするためのパスワード入力パラメーターを作成する

アナリティクス ヘッダーに指定するパスワード入力パラメーターは、HighBond パスワードを入力するようにユーザーに指示するものです。保存されたパスワードは、スクリプトの本文の後続の EXPORT コマンドで使用されます。

```
COMMENT
//ANALYTIC HighBond パスワードの例
//PASSWORD 3 HighBond パスワード:
END
SET SAFETY OFF
OPEN AR_Exceptions
EXPORT FIELDS No Due Date Ref Amount Type ACLGRC PASSWORD 3 TO
"10926@us"
SET SAFETY ON
```

備考

パスワードの格納と暗号化

パスワードの値は、個々のユーザーと関連付けられ、格納時に暗号化されます。パスワードは、アナリティクススクリプト処理を通じて安全性を確保し続け、展開環境で作成されるすべての一時ファイルで暗号化されます。

Analytics でのテスト

Analytics 内に1つ以上の `PASSWORD` タグを持つアナリティクススクリプトをテストすると、Analytics により、`PASSWORD` コマンドが自動的に生成され、適切なパスワードを入力するように求められます。この自動生成されたコマンドにより、テスト目的でアナリティクススクリプトのスクリプト部で `PASSWORD` コマンドを挿入し、アナリティクススクリプトをユーザーに提供する前にそれを削除する必要がありません。

自動生成された `PASSWORD` コマンドは、パスワード値なしでログに保存されます。

パスワード値は、Analytics でアナリティクススクリプトを実行するときに保存されるため、カーソル位置からのアナリティクスの実行またはステップなど、アナリティクススクリプトを実行するたびにパスワードを指定する必要があります。

TABLE タグ

ユーザーがアナリティクス スクリプトの入力として選択する Analytics テーブルを定義します。

TABLE タグの後には、順次行に入力されたゼロ個以上の FIELD タグを続けることができます。

メモ

TABLE タグと FIELD タグを選択できるようにするには、Analytics テーブルが保存場所にあらかじめ存在している必要があります。詳細については、「DATA タグ」ページ 2641を参照してください。

ユーザーが同じアナリティクス スクリプトで使用するために別のテーブルやフィールドを指定できる変数を作成する場合は、TABLE および FIELD タグを使用します。スクリプトが常に同じテーブルやフィールドのセットで実行され、名前が変更されないという設計の場合は、テーブル名とフィールド名をスクリプトにハードコーディングして、TABLE および FIELD タグを使用しないこともできます。

構文

```
//TABLE 変数名 ラベル
<説明>
```

パラメーター

名前	説明
変数名	<p>ユーザーによって選択された入力テーブル名を格納する変数の名前。変数名をアナリティクス スクリプト内で使用することで、テーブルを参照できます。</p> <p>変数名にはスペースを使用しないでください。</p> <p>変数名では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>` ~ ! @ # \$ % ^ & * () - + = { } [] \ : ; ' " < > , . / ?</pre>
ラベル	<p>ロボットで、テーブルを選択するように指示されたときにユーザーに表示されるインターフェイスラベル。例: <code>支払テーブル</code></p>
説明 省略可能	<p>ロボットで、ユーザーに表示される入力フィールドに関連付けられた説明テキスト。説明を複数行で指定することはできませんが、行を飛ばすことはできません。</p> <p>この説明により、ユーザーは正しいテーブルを選択できます。例: <code>支払が記載されており、小切手番号の列があるテーブルを選択します。</code></p> <p>説明は、関連する TABLE タグの下に行に入力する必要があります。</p>

例

基本的な例

ユーザーが正しい入力テーブルを選択できるようにするための説明がある `TABLE` タグ:

```
//TABLE v_table_payments 支払テーブル  
支払が記載されており、小切手番号の列があるテーブルを選択します。
```

高度な例

TABLE タグで定義したテーブルのスクリプトでの使用

次のスクリプトは、プロジェクト内のデータテーブルからユーザーが選択したテーブルに対して AGE コマンドを実行します。

```
COMMENT  
//ANALYTIC スクリプト例  
//TABLE v_table_payments 支払テーブル  
 支払が記載されており、小切手番号の列があるテーブルを選択します。  
END  
  
OPEN %v_table_payments%  
AGE ON payment_date CUTOFF 20141231 INTERVAL 0,30,60,90,120,10000  
SUBTOTAL Payment_Amount TO r_output  
CLOSE %v_table_payments%
```

FIELD タグ

ユーザーがアナリティクス スクリプトの入力として選択するフィールドを定義します。

このフィールドは、先行する `TABLE` タグで定義したテーブルに含まれます。最初の `FIELD` タグは、`TABLE` タグの直後に続ける必要があります。その後続けて、追加の `FIELD` タグを連続した行に入力できます。

メモ

`TABLE` タグと `FIELD` タグを選択できるようにするには、Analytics テーブルが保存場所にあらかじめ存在している必要があります。詳細については、「DATA タグ」ページ 2641を参照してください。

ユーザーが同じアナリティクス スクリプトで使用するために別のテーブルやフィールドを指定できる変数を作成する場合は、`TABLE` および `FIELD` タグを使用します。スクリプトが常に同じテーブルやフィールドのセットで実行され、名前が変更されないという設計の場合は、テーブル名とフィールド名をスクリプトにハードコーディングして、`TABLE` および `FIELD` タグを使用しないこともできます。

構文

```
//FIELD 変数名 型 ラベル
<説明>
```

パラメーター

名前	説明
変数名	<p>ユーザーによって選択された入力フィールド名を格納する変数。変数名をアナリティクス スクリプト内で使用することで、フィールドを参照できます。</p> <p>変数名にはスペースを使用しないでください。</p> <p>変数名では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>` ~ ! @ # \$ % ^ & * () - + = { } [] \ : ; ' " < > , . / ?</pre>
種類	<p>選択できるフィールドの型。次の一覧から、任意の型または型の組み合わせを選択できます。</p> <ul style="list-style-type: none"> <code>C</code> - 文字データ <code>N</code> - 数値データ <code>D</code> - 日付時刻データの日付、日付時刻、または時刻サブタイプ <code>L</code> - 論理データ <p>テーブル内の演算フィールドは、指定された型に関係なく選択できます。</p>

名前	説明
ラベル	ロボットで、フィールドを選択するように指示されたときにユーザーに表示されるインターフェイスラベル。例: <code>支払日フィールド</code>
説明 省略可能	<p>ロボットで、ユーザーに表示される入力フィールドに関連付けられた説明テキスト。説明を複数行で指定することはできますが、行を飛ばすことはできません。</p> <p>この説明により、ユーザーは正しいフィールドを選択できます。例: <code>小切手の支払日を含んでいる列を選択します。</code></p> <p>説明は、関連する <code>FIELD</code> タグの下に行に入力する必要があります。</p>

例

基本的な例

文字フィールドを指定します。

```
//FIELD v_last_name C Last Name Field
```

文字または数値フィールドを指定します。

```
//FIELD v_inv_num CN Invoice Number
```

高度な例

2つのFIELDタグを指定したTABLE

次のアナリティクスヘッダーは、スクリプトが実行される際にユーザーが `v_table_payments` テーブル内の2つの入力フィールドを指定できるようにしています。

```
COMMENT
//ANALYTIC テスト アナリティクス
//TABLE v_table_payments 支払テーブル
  支払が記載されており、小切手番号の列があるテーブルを選択します。
//FIELD v_check_num CN 小切手番号フィールド
//FIELD v_payment_date D 支払日フィールド
  小切手の支払日を含んでいる列を選択します。
END
```

アナリティクス スクリプト

```
OPEN %v_table_payments%  
EXTRACT FIELDS %v_check_num%, %v_payment_date% TO t_analyze
```

RESULT タグ

エンドユーザーがロボットで、アナリティクス スクリプトで出力された結果を使用できるように指定します。出力結果は、存在していても、自動的に利用可能にはなりません。

メモ

組織でオンプレミスのロボット エージェントを使用している場合は、`RESULT` タグを指定すると、エージェントから HighBond のクラウド ロボット アプリにデータをアップロードできます。詳細については、「クラウドベースのロボット アプリへのアップロード」ページ 2638を参照してください。

構文

```
//RESULT 種類 名前
<説明>
```

パラメーター

名前	説明
種類	<p>結果項目の種類：</p> <ul style="list-style-type: none"> ◦ <code>TABLE</code> - Analytics テーブルと関連付けられたデータ ファイル (.fil) ◦ <code>Log</code> - アナリティクスのログ ファイル ◦ <code>FILE</code> - Analytics ファイル以外のファイル <p>メモ</p> <p>オンプレミスのロボット エージェントを使用していて、ロボット へのファイルアップロードが無効にされている場合は、<code>//RESULT LOG</code> や <code>//RESULT FILE</code> を使用しないでください。詳細については、「クラウドベースのロボット アプリへのアップロード」ページ 2638を参照してください。</p> <p>ログの詳細については、「ログファイルの出力方法」ページ 2639を参照してください。</p>
名前	結果項目の名前。

名前	説明
	<p>メモ</p> <p>名前値はアナリティクス スクリプトの結果項目の名前と正確に一致する必要があります。名前で項目名を指定していません。スクリプトで指定された名前と一致しています。</p> <p>スクリプトを使った名前一致を支援するために、名前の中でワイルドカード文字を使用することができます。</p> <p>名前には変数を使用できません。</p> <h3>テーブル名</h3> <p>名前値には、Analytics テーブル名を指定します。ソース データ ファイルの名前ではなく、テーブルの名前である必要があります。</p> <p>正しい記述：</p> <pre>//RESULT TABLE Missing_Checks</pre> <p>間違った記述：</p> <pre>//RESULT TABLE Missing_Checks.fil</pre> <p>名前にはスペースを使用しないでください。</p> <p>名前では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>! @ # \$ % ^ & () - + = { } [] \ : ; ' " < > , / . ` ~</pre> <h3>ログ名</h3> <p>省略可能。名前値はアナリティクス ログ ファイル名を指定します。名前を指定しない場合は、デフォルト ログ名 <code>analytic_name.log</code> が使用されます。</p> <p>メモ</p> <p>ログ名を指定する場合は、<code>SET LOG TO ログ名</code> がスクリプトに表示される必要があります。</p> <p>名前にはスペースを使用しないでください。</p> <p>名前では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre>< > : " \ / </pre> <h3>ファイル名</h3>

名前	説明
	<p>名前値には、Analytics ファイル以外のファイルの名前を指定します。</p> <p>出力される非 Analytics ファイルの種類には、適切なファイル拡張子を指定する必要があります。</p> <p>正しい記述：</p> <pre data-bbox="565 447 1344 514">//RESULT FILE Missing_Checks.xlsx</pre> <p>間違った記述：</p> <pre data-bbox="565 615 1344 682">//RESULT FILE Missing_Checks</pre> <p>名前にはスペースを使用しないでください。</p> <p>名前では次の文字を使用しないでください。次の文字はサポートされていません。</p> <pre data-bbox="565 825 1344 892">< > : " \ / </pre> <h3 data-bbox="516 953 834 995">ワイルドカード文字</h3> <p>スクリプトを使ってテーブル名、ログ名、またはファイル名との一致を支援するために、名前の中で1つまたは複数のワイルドカード文字を使用することができます。連続する0個以上の文字の代わりに単一のアスタリスク(*)を使用します。</p> <p>ワイルドカード文字とリテラル文字を混在させて作成したパターンを使用することで、特定のファイル種類の全項目と一致させたり(例: *.xlsx)、スクリプト内の変数定義によって名前の一部が変化する項目と一致させたりすることができます。</p>
<p>説明 省略可能</p>	<p>結果やその他の情報に関する説明テキスト。説明を複数行で指定することはできますが、行を飛ばすことはできません。</p> <p>説明は、アナリティクス ヘッダーにのみ存在するものであり、ロボットのエンド ユーザーには表示されません。</p>

例

基本的な例

Analytics テーブルを示す `RESULT` タグ:

```
//RESULT TABLE Missing_Checks
```

アナリティクスのログを示す `RESULT` タグ(名前はデフォルト値):

```
//RESULT LOG
```

アナリティクスのログを示す `RESULT` タグ(名前は指定された値) :

```
//RESULT LOG My_log_name  
.  
.  
.  
SET LOG TO My_log_name
```

特定の Excel ファイルを示す `RESULT` タグ:

```
//RESULT FILE Missing_Checks.xlsx
```

すべての Excel ファイルを示す `RESULT` タグ:

```
//RESULT FILE *.xlsx
```

高度な例

変化する月名を含んだテーブル名

出力テーブル名に月名が含まれている(invoices-jan、invoices-feb など) ため、「`invoices-*`」と指定すれば、月の接尾辞に限定されないテーブル名を結果内で利用できるようになります。

```
//RESULT TABLE invoices-*
```

変化する日付が含まれたログ名

ログファイル名には日付スタンプが含まれている(prepare_invoice_table_31072019 など) ため、日付スタンプに限定されないログファイル名を結果内で利用できるように、「`prepare_invoice_table_*`」を指定します。

```
//RESULT LOG prepare_invoice_table_*
```

変化する月名を含んだファイル名

出力ファイル名に月名が含まれている(invoices-jan.xlsx、invoices-feb.xlsx など)ため、「invoices-*.xlsx」と指定すれば、月の接尾辞に限定されないテーブル名を結果内で利用できるようになります。

```
//RESULT FILE invoices-*.xlsx
```

変化する月名とファイル形式名を含んだファイル名

出力ファイル名に月名が含まれており、さまざまなファイル形式で出力される(invoices-jan.xlsx、invoices-jan.del など)ため、「invoices-*.*/」と指定すれば、月の接尾辞やファイルの種類に限定されないテーブル名を結果内で利用できるようになります。

```
//RESULT TABLE invoices-*/
```

備考

クラウドベースのロボット アプリへのアップロード

組織でオンプレミスのロボット エージェントを使用している場合は、アナリティクス ヘッダーの `RESULT` タグを指定すると、エージェントから HighBond のクラウド ロボット アプリにデータをアップロードできます。転送中のデータとロボットに保存されているデータはすべて暗号化されます。

ロボットの `許可されたファイルアップロード` 構成設定は、`RESULT` タグで指定された出力結果が次の状態であるかどうかを制御します。

- ロボットにアップロードされる
- ロボット エージェントがインストールされているサーバーでのローカル出力に制限されている

構成設定の詳細については、[ロボット エージェントの構成](#)を参照してください。

アナリティクス タグ	許可されたファイル アップロード] 設定:		
	結果ファイルとログのみ	結果テーブル、ファイル、ログ	ファイル アップロードは許可されていません
//RESULT TABLE	Analytics 結果テーブルレイアウトのみがアップロードされます(フィールド名、データ型、フィールド長)。結果のテーブルデータはネットワークのサーバーに残りません。	Analytics 結果テーブル(レイアウトとデータ)がアップロードされます。	Analytics 結果テーブルレイアウトのみがアップロードされます(フィールド名、データ型、フィールド長)。結果のテーブルデータはネットワークのサーバーに残ります。
//RESULT LOG	Analytics ログファイル(成功したタスクと失敗したタスクの両方)がアップロードされます。	Analytics ログファイル(成功したタスクと失敗したタスクの両方)がアップロードされます。	指定しない]はアナリティクス スクリプトが失敗します。
//RESULT FILE	Excel のような非 Analytics の結果ファイルがアップロードされます。	Excel のような非 Analytics の結果ファイルがアップロードされます。	指定しない]はアナリティクス スクリプトが失敗します。

ログファイルの出力方法

アナリティクス スクリプトのログファイルの出力方法は、次の条件によって異なります。

- スクリプトが成功したか失敗したかどうか
- ネットワークまたはクラウドでスクリプトを実行しているかどうか
- **許可されたファイル アップロード] 構成設定 (オンプレミスのロボット エージェントのみ)**

構成設定の詳細については、[ロボット エージェントの構成](#)を参照してください。

アナリティクス スクリプト 成功

オンプレミスロボット エージェント	クラウドベースのロボット エージェント
<ul style="list-style-type: none"> ◦ RESULT LOG が指定される 許可されたファイル アップロード] 設定が [ファイル アップロードは許可されていません] ではない場合、ログファイルはクラウドベースのロボット アプリにアップロードされます。 ◦ RESULT LOG が指定されない ログファイルなし 	<ul style="list-style-type: none"> ◦ RESULT LOG が指定される ログファイルはクラウドベースのロボット アプリにアップロードされます。 ◦ RESULT LOG が指定されない ログファイルなし

アナリティクス スクリプト 失敗

オンプレミスロボット エージェント	クラウドベースのロボット エージェント
<ul style="list-style-type: none">◦ RESULT LOG タグが考慮されない 許可されたファイルアップロード]設定が [ファイルアップロードは許可されていません]ではない場合、ログファイルは自動的にクラウドベースのロボット アプリにアップロードされます。	<ul style="list-style-type: none">◦ RESULT LOG タグが考慮されない ログファイルは自動的にクラウドベースのロボット アプリにアップロードされます。

DATA タグ

アナリティクス スクリプトによって出力された Analytics テーブルを、ロボットの中央データ保存場所にコピーすることを指定します。

通常は、後続のアナリティクス スクリプトの入力テーブルとして利用できるよう、Analytics テーブルを格納します。

メモ

Analytics テーブルの中央データ保管場所への保存は、オンプレミスのロボット エージェントを持つ ACL ロボットでのみサポートされています。クラウドベースのロボット エージェントを使用している場合、この機能は使用できません。DATA タグは、クラウドベースのエージェントで実行されるアナリティクス スクリプトでは無視されます。

構文

```
//DATA テーブル名
<説明>
```

パラメーター

名前	説明
テーブル名	<p>格納する Analytics テーブルの名前。</p> <p>メモ テーブル名値はアナリティクス スクリプトの Analytics 出力テーブルの名前と正確に一致する必要があります。テーブル名でテーブル名を指定していません。スクリプトで指定されたテーブル名と一致しています。 スクリプトを使ったテーブル名一致を支援するために、テーブル名の中でワイルドカード文字を使用することができます。 テーブル名には変数を使用できません。</p> <p>ソースデータファイルの名前ではなく、テーブルの名前である必要があります。</p> <p>正しい記述：</p> <pre>//DATA Missing_Checks</pre> <p>間違った記述：</p>

名前	説明
	<div data-bbox="565 270 1344 338" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;">//DATA Missing_Checks.fil</div> <div data-bbox="565 375 1344 478"> <p>メモ</p> <p>データ サブフォルダーの既存の Analytics の名前が指定した値と同じ場合、既存のテーブルが上書きされます。</p> </div> <div data-bbox="516 533 1289 579"> <h2>データ テーブル名でサポートされていない文字</h2> </div> <div data-bbox="516 604 1354 678"> <p>テーブル名にはスペースを使用しないでください。 テーブル名では次の文字を使用しないでください。次の文字はサポートされていません。</p> </div> <div data-bbox="565 709 1344 774" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;">! @ # \$ % ^ & () - + = { } [] \ : ; ' " < > , / . ` ~</div> <div data-bbox="516 835 834 882"> <h2>ワイルドカード文字</h2> </div> <div data-bbox="516 907 1419 1024"> <p>テーブル名の一部が変化する可能性がある場合は、テーブルの名前にワイルドカード文字を使用することができます。たとえば、テーブル名が月によって異なる場合 (invoices-jan、invoices-feb など) は、「invoices-*」と指定すれば、月の接尾辞に関係なくテーブルがデータサブフォルダーにコピーされます。</p> </div> <div data-bbox="516 1041 1395 1098"> <p>分析スクリプト内のすべての Analytics 出力テーブルをデータサブフォルダーにコピーするために、単一のワイルドカード文字を指定できます:</p> </div> <div data-bbox="565 1129 1344 1194" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;">//DATA *</div> <div data-bbox="565 1234 1344 1430"> <p>注意</p> <p>ワイルドカード文字を使用するときには注意してください。指定するワイルドカード パターンが意図していないテーブルと一致する場合、既存のデータテーブルが意図せず上書きされる場合があります。</p> <p>ベストプラクティスとして、テーブル名値をできるかぎり具体的にします。必要な場合にのみワイルドカード文字を使用してください。</p> </div> <div data-bbox="516 1491 906 1537"> <h2>ロボットへのアップロード</h2> </div> <div data-bbox="516 1562 1386 1619"> <p>ロボットへのアップロードについては、「クラウドベースのロボット アプリへのアップロード」ページ 2644を参照してください。</p> </div>
<p>説明 省略可能</p>	<p>出力テーブルやその他の情報に関する説明テキスト。説明を複数行で指定することはできませんが、行を飛ばすことはできません。</p> <p>説明は、アナリティクス ヘッダーにのみ存在するものであり、ロボットのエンド ユーザーには表示されません。</p>

例

Analytics テーブルを格納場所にコピーする

次のアナリティクス ヘッダーは、関連付けられたスクリプトで出力されるインボイステーブルが格納場所にコピーされることを指定します：

```
COMMENT
//ANALYTIC テーブルのインポート
//DATA Invoices
END
IMPORT DELIMITED TO Invoices "Invoices.fil" FROM "Invoices.csv" 0
SEPARATOR "," QUALIFIER "'" CONSECUTIVE STARTLINE 1 KEPTITLE ALLCHAR
ALLFIELDS
```

備考

出力テーブルの格納

出力テーブルは自動的に格納場所にコピーされません。格納したいテーブルごとに `DATA` タグを使用する必要があります。必要に応じて、1つのアナリティクス ヘッダーに複数の `DATA` タグを含めることができます。

DATA タグを使用するとき

次の2つの状況では、`DATA` タグを使用して、Analytics テーブルを格納する必要があります。

- 出力テーブルが後続のアナリティクス スクリプトの入力として使用される
- ユーザーがアドホックでアナリティクス スクリプトをスケジュールまたは実行するときに、入力テーブルまたはフィールドを選択できる

メモ

データ分析プロセス全体が単一のアナリティクス スクリプトを使用して完了する場合、`DATA` タグを使用する必要はありません。

`DATA` タグは、結果テーブルを指定するために使用するものではありません。代わりに `RESULT` タグを使用します。詳細については、「RESULT タグ」ページ 2634を参照してください。

出力テーブルが後続のアナリティクス スクリプトの入力として使用される

展開環境および関連付けられたスクリプトの構造によっては、`DATA` タグを使用して、後続のアナリティクス スクリプトで使用するための Analytics 出力テーブルを格納しなければならない場合があります。

アナリティクス スクリプト 処理中に、ロボットは一時ディレクトリを使用して、Analytics 出力テーブルを格納してアクセスするため、`DATA` タグを使用する必要がない場合があります。

以下の表は指針です。

展開環境	DATA タグを使用する場合	DATA タグを使用する必要がない場合
ロボット (Enterprise Edition のみ)	<ul style="list-style-type: none"> 1 つのロボット タスク内で出力された Analytics テーブルが、別のロボット タスクの入力として必要である 	<ul style="list-style-type: none"> Analytics テーブルが出力され、単一のロボット タスクでの一連のアナリティクス スクリプト実行中に後から入力される データ分析プロセス全体が単一のアナリティクス スクリプトを使用して完了する

ユーザーが入力テーブルまたはフィールドを選択できる

`TABLE` および `FIELD` アナリティクス タグは、ユーザーが Analytics テーブルを選択し、テーブルからフィールドを選択できる入力パラメーターを作成し、アナリティクス スクリプトの入力として使用します。ただし、選択できるようにするために、テーブルが格納場所に予め存在する必要があります。

ユーザーが 1 つ以上の入力テーブルとフィールドを選択できるアナリティクス スクリプトを開発している場合は、`DATA` タグの前のアナリティクス スクリプトを実行し、適切なテーブルを保存場所に保存する必要があります。

ロボットのソース テーブル セクションで出力テーブルを見つける

任意で、`src_` プレフィックスを出力テーブル名に追加し、ロボットの `[入力/出力]` タブの `[ソース テーブル]` セクションで出力テーブルを検索することができます。

```
//DATA src_Invoices
```

`DATA` タグと付属するスクリプトの両方で、プレフィックスをテーブルに追加する必要があります。

`[ソース テーブル]` セクションでは、後続のスクリプトの入力を提供するテーブルを視覚的に区切ることができます。出力テーブル名に `src_` プレフィックスがない場合、`[ソース テーブル]` セクションは `[入力/出力]` タブに表示されず、すべてのテーブルは `[他のテーブル]` セクションでデフォルトで検索されます。

クラウド ベースのロボット アプリへのアップロード

ロボット インストールでアナリティクス スクリプトを実行し、`DATA` 指定すると、オンプレミスロボット エージェントから HighBond のクラウド ベースのロボット アプリにテーブルレイアウトのみ(ファイル名、データ型、ファイ

アナリティクス スクリプト

ル長さ) をアップロードします。テーブルデータは、ロボット エージェント ディレクトリ内の組織のネットワークにあります。

すべての情報は転送中に暗号化されます。

アナリティクス スクリプトの Unicode への変換

非 Unicode の既存のアナリティクス スクリプトおよび標準スクリプトを Unicode 版の Analytics に移行する場合、これらは自動的に Unicode に変換されます。ただし、これらを Unicode データに適用した場合でもスクリプトのロジックが変わらないことを確認する必要があります。

Unicode とは

Unicode は、各文字を表すために 2 バイト以上を使用するテキストのエンコードの標準であり、すべての言語の文字が単一の文字セットに格納されています。Unicode 版の Diligent 製品を使用すると、現在使用されている全言語で、Unicode でエンコードされたデータを含んでいるファイルやデータベースの表示および操作が行えます。

メモ

Analytics およびロボット エージェントは、リトルエンディアン(LE)でエンコードされた Unicode データをサポートしています。これらの製品を使用して、ビッグエンディアン(BE)でエンコードされたデータを分析することはできません。

必要なアナリティクス スクリプトの変更

バイト数で値を指定するすべてのパラメーターの更新

非 Unicode 版 Analytics での文字の長さは 1 バイトです。Unicode データの場合、Unicode エディションでの文字の長さは 2 バイトです。非 Unicode 版 Analytics で、フィールド長または開始位置をバイトで指定するときには、バイト数は文字数と同じです。これは Unicode 版 Analytics における Unicode データには当てはまりません。

スクリプトを Unicode 版 Analytics で使用できるように変換するには、バイト数でフィールドの長さや開始位置を指定したすべてのパラメーターの数値を調整する必要があります。たとえば、非 Unicode 版 Analytics で 7 の WID 値を指定する `IMPORT DELIMITED` コマンドの場合、Unicode 版 Analytics で同じ結果を出力するには、WID の値を 2 倍の 14 にする必要があります。

また、Unicode データの場合、フィールドの奇数の開始バイト位置とフィールド長の偶数のバイト数を指定します。偶数の開始位置または奇数の長さを指定すると、文字が正しく表示されない可能性があります。

IMPORT PRINT および IMPORT DELIMITED の全 インスタンスを再作成する

IMPORT PRINT コマンド および IMPORT DELIMITED コマンド の全 インスタンスを再作成する必要があります。

1. Unicode 版の Analytics でデータ定義 ウィザードを使用してソースデータファイルをインポートします。
データ定義 ウィザードを使用すると、すべての構文が有効であることが保証されます。
2. ログからインポート構文を使用してスクリプトを更新します。
3. アナリティクス スクリプトの場合、スクリプトをロボットに再コミットします。

ZONED() および EBCDIC() 関数の全 インスタンスの変更

次のように ZONED() および EBCDIC() 関数の全 インスタンスを変更し、関数で返される ASCII 値が正しく Unicode データに変換されるようにする必要があります。

- **演算フィールド** -は、ZONED() や EBCDIC() インスタンスを BINTOSTR() 関数で囲みます
- **静的な式** -では ZONED() インスタンスを BINTOSTR() 関数で囲みます

```
BINTOSTR(ZONED(%result%, 5), 'A')
```

OPEN FORMAT コマンドのすべてのインスタンスを変更します

OPEN FORMAT コマンドのすべてのインスタンスを変更する必要があります。SKIP パラメーターを使用して、開いている Unicode ファイルの最初の 2 バイトをスキップする必要があります。これは、UTF-16 エンコード ファイルの最初の 2 バイトはバイト オーダー マークとして予約されており、ファイル内のテキストとは異なるものであるため、必要となります。

非 Unicode

```
OPEN "ascii_test.txt" FORMAT template_table CRLF  
DEFINE FIELD full_rec ASCII 1 10
```

Unicode

```
OPEN "utf-16_test.txt" FORMAT template_table CRLF SKIP 2  
DEFINE FIELD full_rec UNICODE 1 20
```

変換されたアナリティクス スクリプト の検証

Unicode 版のアナリティクス スクリプトによって生成される結果が、非 Unicode 版のアナリティクス スクリプトによって生成された結果とまったく同じになるかどうかを検証します。これを行う最良の方法は、Diff ツールを使用して、分析で生成されたログ ファイルを比較することです。Diff ツールは、ファイル間のあらゆる相違点を識別します。

結果がまったく同じでなかった場合の処置

Unicode 版のアナリティクス スクリプトを使用しても非 Unicode 版と同じ結果を生成できなかった場合には、スクリプトのログ出力をアナリティクス スクリプトのステップごとに比較することで、問題の切り分けができる可能性があります。

Unicode 互換性チェック

Unicode 版にアップグレードする際は、スクリプトに追加したすべてのカスタム ロジックが、Unicode データに対して実行されたときに同じ結果を生じるかどうかを検証する必要があります。スクリプトが Unicode データに対して実行されたときに、スクリプトが影響を受ける可能性があることが予測できる領域がありません。

ビットおよび文字関数

以下に挙げる関数はそれぞれ、バイト位置やバイト数に基づいて値を返します。非 Unicode 版で使用される 1 バイト文字の表現から、Unicode データに使用される 2 バイト文字エンコードへ移動しても、これらの関数が正しく使用されているかどうかをチェックする必要があります。

- ASCII()
- BIT()
- BYTE()
- CHR()
- DIGIT()
- HEX()
- MASK()
- SHIFT()

バイト長と文字長は等しくない

以下の関数について、データ内の文字数とバイト数が一対一対応でと見なす方法で使用されていないことを保証するために、これらの関数がスクリプトで使用されている方法を確認する必要があります。

文字とバイト間を一対一対応と見なすロジックがあるインスタンスを見つけた場合は、そのロジックが Unicode データで正常に動作するよう、つまり、1 文字を表すために 2 バイトを使用するように調整する必要があります。`STRING(1000, 4)` 内の 4 など、文字列関数のパラメーターとして指定された数値は文字数を表すため、このような関数を標準的な方法で使用しても問題は発生しません。

変換関数

- PACKED()
- STRING()
- UNSIGNED()

- VALUE()
- ZONED()

文字列関数

- AT()
- BLANKS()
- INSERT()
- LAST()
- LENGTH()
- REPEAT()
- SUBSTRING()

その他の関数

- FILESIZE()
- LEADING()
- OFFSET()
- RECLLEN()

Unicode 固有の関数を代用する

Diligent Unicode 製品は、非 Unicode データと Unicode データ間の変換をサポートする、6 個の Unicode 固有の関数をサポートしています。次の関数を Diligent Unicode 製品で使用できます。

- **BINTOSTR()** - 関数は、ZONED または EBCDIC データを対応する Unicode 文字列に変換します。これにより、ZONED または EBCDIC のデータとしてエンコードされた値を正しく表示できるようになります。
- **DHEX()** - は、指定された Unicode のフィールド値に相当する 16 進数の値を返します。これは HTOU() の逆関数です。
- **DBYTE()** - は、レコード内の指定された位置にある 2 バイト文字に相当する Unicode 文字を返します。
- **DTOU()** - は、指定されたロケール設定に基づいて、日付値を正しい Unicode 文字列表示に変換します。
- **HTOU()** - は、指定された 16 進数の文字列に相当する Unicode 文字列を返します。これは DHEX() の逆関数です。
- **UTOD()** - は、ロケール固有の Unicode 文字列を Analytics の日付値に変換します。

スクリプト エンジンエラーコード

以下の表は、アナリティクス スクリプトを実行するときに発生する可能性のあるエラーコードを示します。

スクリプト エンジンスタート アップエラー

エラーコード	説明
202	システムエラーです。
203	評価期間が終了しました。
204	評価期間が終了しました。
205	アクティブ化に失敗しました。
206	セッションの最大数に達しました。
207	メモリの初期化に問題があります。
209	不明なスクリプト エラーです。
210	データベース プロファイルのパスワードがありません。
211	サーバー接続に失敗しました。
212	サポートされていないコマンドが検出されました。
213	ダイアログ ボックスがスクリプトによって生成されました。
256	スクリプトエンジンの開始に失敗しました。

スクリプト エンジンエラーコード

エラーコード	説明
1000	初期設定ファイル(*.prf)が見つかりませんでした。新しいデフォルト初期設定ファイルが作成されました。
1001	初期設定ファイル(*.prf)に必要な情報がありません。新しいデフォルト初期設定ファイルが作成されました。
1002	初期設定ファイル(*.prf)が見つからず、必要なフォルダーアクセス許可が存在しないので新しいデフォルト初期設定ファイルを作成できないため、スクリプトエンジンを開始できません。

エラーコード	説明
1003	プロジェクトは以前のバージョンからアップグレードされました。コピーが拡張子 .old で保存されました。
1004	プロジェクト ファイルを処理できませんでした。前回保存したプロジェクトが使用されます。
1005	プロジェクト ファイルが指定されていません。
1006	指定されたプロジェクト ファイルは存在しません。
1007	指定されたプロジェクト ファイルは読み取り専用です。
1008	指定されたプロジェクトは現在、別のアプリケーションで使用されています。
1009	指定された .old プロジェクト ファイルは使用できません。拡張子が .ACL のプロジェクト ファイルを指定する必要があります。
1010	指定されたプロジェクト ファイルは Analytics プロジェクト ファイルではありません。
1011	プロジェクトは、Unicode に変換できませんでした。
1012	指定されたプロジェクト ファイルは、以前のバージョンとして保存することはできません。
1013	ログファイルを書き込み用に開けません。
1014	スクリプトが指定されていません。
1015	指定されたスクリプトは存在しません。
1016	Analytics ライセンスが見つからなかったか無効です。
1017	必要なライブラリファイル(*.dll)が見つかりませんでした。
1018	書き込みアクセス許可がないため、作業ディレクトリにデータを保存できませんでした。
1019	Analytics サブスクリプションを検証できませんでした。
1020	Analytics のライセンスはリリースできません。
1021	不明なエラーが発生しました。

コマンド エラー

以下の表は、ACLScript コマンドが無効であることが原因でアナリティクス スクリプトが失敗した場合に返されるエラーコードを示します。返されるエラーコード番号によって、失敗したコマンドが特定されます。

アナリティクス スクリプト

エラー コード	コマンド
1	SAMPLE
2	EXTRACT
3	LIST
4	TOTAL
5	DEFINE
6	COMMENT
7	QUIT
8	STOP
9	BYE
10	USE
11	OPEN
12	SAVE
13	DISPLAY
14	ACTIVATE
15	CLOSE
16	HELP
17	COUNT
18	STATISTICS
19	HISTOGRAM
20	STRATIFY
21	SUMMARIZE
22	EXPLAIN
23	GROUP
24	ELSE

エラーコード	コマンド
25	END
26	CANCEL
27	SUBMIT
28	DELETE
29	RANDOM
30	SORT
31	FIND
32	DIRECTORY
33	TYPE
34	DUMP
35	INDEX
37	SET
40	DO
41	TOP
42	EXPORT
43	VERIFY
44	SEEK
45	JOIN
46	MERGE
47	SEQUENCE
48	CALCULATE
49	PRINT
50	LOCATE
51	RENAME

アナリティクス スクリプト

エラーコード	コマンド
54	COPY
55	REPORT
56	EJECT
58	LET
59	ACCUMULATE
63	ACCEPT
64	ASSIGN
65	AGE
66	CLASSIFY
67	PROFILE
68	DO REPORT
69	LOOP
70	PAUSE
71	SIZE
72	EVALUATE
73	DIALOG
74	IF
75	GAPS
76	DUPS
77	SQLOPEN
78	PASSWORD
79	IMPORT
80	REFRESH
81	NOTIFY

エラーコード	コマンド
82	CONNECT
83	RETRIEVE
84	FIELDSHIFT
85	BENFORD
86	CROSSTAB
87	(未使用)
88	ESCAPE
89	NOTES
90	FUZZY DUPLICATE
91	EXECUTE
92	ACCESSDATA32
93	ACCESSDATA64
94	APPEND
95	RCOMMAND
96	CVSPREPARE
97	CVSSAMPLE
98	CVSEVALUATE
99	OUTLIER
100	FUZZYJOIN
101	CLUSTER
102	TRAIN
103	PREDICT

アナリティクス ジョブ処理エラー

メモ

これらのエラーの中には、レガシー製品である Analytics Exchange と AX Exception に特に関連するものがあります。

エラーコード	エラーメッセージ
-10	アナリティクスの実行開始後に出力先の結果フォルダーが削除されたため、アナリティクスの結果を保存できませんでした。
-11	ジョブが停止されました。
-12	サーバーがシャットダウンしたため、停止されました。
-13	結果を作成できませんでした。
-16	サーバープロパティの構成エラーのため、実行できませんでした。
-17	一意の名前の結果ディレクトリを作成できません。
-19	ジョブがスキップされました。
-20	結果テーブルを公開する準備ができませんでした。
-21	結果を AX Exception に公開できませんでした。
-22	公開に失敗しました。テーブル名が無効です。
-23	公開に失敗しました。テーブルの1つ以上の列の名前が長すぎます。
-24	公開に失敗しました。Analytics テーブルのデータセル内の値が無効です。
-25	公開に失敗しました。テーブルのフィールド内にサポートされないデータ型があります。
-26	公開に失敗しました。AX Exception サーバーに接続できませんでした。
-27	ジョブは実行されませんでした。ユーザーが削除された、またはユーザーにアクセス許可がありません。
-28	ジョブは実行されませんでした。予期しないエラーです。詳細については、サーバーログおよび Analytics ログを調べてください。
-29	データファイルをコピーできませんでした。必要なデータファイルをジョブディレクトリにコピーできなかったため、アナリティクスは失敗しました。
-30	ジョブは実行されませんでした。分析のリンクが機能していません。

エラーコード	エラー メッセージ
-31	公開に失敗しました。例外マッピングファイルが見つかりませんでした。
-32	公開に失敗しました。例外マッピングファイルを解析できませんでした。
-34	ジョブの結果を保存できませんでした。ジョブフォルダーを保管するドライブ上に十分なスペースがあることと、データファイルがロックされていないことを確認します。

このページは意図的に空白のままにしています

ACL for Windows インストールおよび アクティベーション ガイド

ACL for Windows インストールおよびアクティベーション ガイド

このガイドでは、ACL for Windows のインストールまたはアップグレードについて詳しく説明します。次のコンポーネントが含まれています。

- Analytics
- オフライン プロジェクト

前のバージョンからのアップグレード

以下の表は、別のバージョンの Analytics のアップグレード パスについて説明します。

現在インストールされている Analytics バージョン	アクション	結果
Analytics 16.0	バージョン 16.1 インストーラー (ACLforWindows161.exe) を実行する	Analytics がバージョン 16.1 にアップグレードされます
16.0 より前の ACL Analytics または ACL Desktop	バージョン 16.1 インストーラー (ACLforWindows161.exe) を実行する	Analytics 16.1 が前のバージョンと共存インストールされます アップグレードはありません。以前のバージョンの Analytics または ACL Desktop は、インストールしたままにすることもアンインストールすることもできます。

ACL for Windows インストールクイックスタート

(詳細なインストール手順については、"ACL for Windows をインストールする" ページ 2679を参照してください。)

メモ

ACL for Windows をインストールまたはアップグレードする場合、インストールまたはアップグレード中に指定した Analytics 作業ディレクトリにある既存の Analytics サンプルデータファイルは上書きされます。

サンプルプロジェクトやデータファイルに対して行ったような変更も保持したい場合は、インストールまたはアップグレードを実行する前に、他の場所にファイルを保存するか、それらが含まれるフォルダーの名前を変更してください。関連付けられたコマンドログファイルを維持したい場合は、ログと同じ操作を実行します。

Analytics が認証局に接続できることを確認する

Analytics はデジタル証明書を使用して、その信頼性と完全性を検証します。証明書は、検証できるように、サードパーティの認証局である DigiCert へのインターネット接続が必要です。ネットワークファイアウォールの背後に Analytics をインストールする場合、ファイアウォールは、証明書が DigiCert に接続できるように設定されている必要があります。必要な接続がないと、Analytics を使用することはできません。

詳細については、「Analytics が認証局に接続できることを確認する」ページ 2680を参照してください。

インストーラーのダウンロード

Launchpad (www.highbond.com) にサインインし、ACL for Windows インストールパッケージ ([ACLforWindows161.exe](#)) をダウンロードします。

Launchpad サインイン資格情報がない場合は、「ACL for Windows をインストールする」ページ 2679を参照してください。

インストールファイルの展開

1. ACL for Windows インストールパッケージ ([ACLforWindows161.exe](#)) をダブルクリックします。
2. セキュリティ警告のダイアログボックスが表示された場合は、リストされた情報を確認して、**[はい]** をクリックします。

ヒント

[セキュリティ警告] ダイアログボックスは開いたウィンドウの後ろに配置して非表示にできます。

3. インストールに使用したい言語を選択して、**[OK]** をクリックします。
4. **[展開場所の設定]** ページで、**[展開]** をクリックします。

ファイルが展開された後、インストーラーが自動的に開始されます。

必須コンポーネントのインストール(必要な場合)

必須コンポーネントをインストールするよう求められた場合は、**[インストール]** をクリックします。

必須コンポーネントがインストールされた後、インストーラーは自動的に進みます。

ACL for Windows インストールまたはアップグレードの実行

画面の手順に従い、ACL for Windows をインストールまたはアップグレードします。インストーラーによって、Analytics 16.1 の新規インストールが実行されます。バージョン 16.0 がインストールされている場合は、アップグレードが実行されます。

ACL エディションの選択] ページで、インストールするエディションを選択します。

- 非 Unicode
- Unicode

注意

インストールするエディションが、御社に合ったエディションであることを確認してください。詳細については以下を参照 "Analytics の非 Unicode 版または Unicode 版をインストールすべきですか?" ページ 2666

メモ

旧バージョンの Analytics との共存インストールを実行する場合、または Analytics バージョン 16.0 からアップグレードする場合は、インストーラーにより、現在インストールされているバージョンと同じエディションが強制的に選択されます。

非 Unicode 版と Unicode 版を並べてインストールすることはできません。詳細については、"異なるバージョンまたはエディションの共存インストール" ページ 2668を参照してください。

任意の Analytics データ コネクタールと Python エンジンを選択 - する

Analytics は、サポートされているさまざまなデータソースのデータにアクセスするためのデータコネクタールと、Analytics 機械学習コマンドを有効にするための Python エンジンを提供します。このようなコネクタールの一部と Python エンジンは任意です。ACL for Windows のインストール中にインストールするかどうかを選択できます。

メモ

任意のデータコネクタールや Python エンジンを選択せず、後からこれらのいずれかが必要になった場合は、ACL for Windows をアンインストールしてから再インストールする必要があります。

バージョン 16.0 からアップグレードする場合は、これらのオプションに対して以前に選択した内容がインストーラーにより強制的に適用されます。

Microsoft Access データベースエンジンを除外するオプション

ACL for Windows のインストールまたはアップグレードでは、Analytics のインストールから Microsoft Access データベースエンジンを除外するオプションが用意されています。データベースエンジンは、Analytics と古い Excel ファイル、および Microsoft Access ファイルとの間の接続を提供します。

- Analytics で現行世代の Excel ファイル(*.xlsx)のみを使用し、古い Excel ファイル(*.xls)や Microsoft Access ファイル(*.mdb)を使用しない場合は、**データベースエンジンを除外**します。データベースエンジンは、他の新しい Microsoft Office 製品との互換性の問題を引き起こす可能性があります。そのため、データベースエンジンが必要ない場合は、除外してください。
- 古い Excel ファイル(*.xls)や Microsoft Access ファイル(*.mdb)をインポートしたり、それらにエクスポートしたりする必要がある場合は、**データベースエンジンをインストール**します。

メモ

Microsoft Access データベースエンジンが以前の ACL for Windows の一部としてすでにコンピューターにインストールされている場合、データベースエンジンを除外するオプションは表示されません。

Analytics 16.0 から 16.1 にアップグレードする場合、Microsoft Access データベースエンジンが不要であれば、手動で削除することができます。旧バージョンの Analytics をバージョン 16.1 と共存インストールする場合、旧バージョンのすべての機能を保持するために、データベースエンジンはインストールしたままにしておきます。

データベースエンジンを除外したものの、後で必要になった場合は、ACL for Windows をアンインストールして再インストールすることができます。

Analytics のアクティブ化

Analytics のアクティブ化の方法は、御社の認証方法によって異なります。

- **標準認証** - 単一の Analytics サブスクリプションをご利用の場合は、高い確実性で標準認証を使用します。
- **カスタムドメインを使用する認証** - 御社がシングルサインオン(SSO)をご利用の場合は、カスタムドメインを使用して認証します。

メモ

御社がカスタムドメインをご利用の場合は、Launchpad サインイン画面の最初の画面で電子メールアドレスとパスワードを入力しないでください。

標準認証

1. デスクトップにある **ACL for Windows 16** ショートカットをダブルクリックします。
2. ACL Launchpad の資格情報を入力し、**[サインイン]**をクリックします。
3. HighBond インスタンスを選択するように指示されたら、そのようにし、**[Analytics のアクティブ化]**を

クリックします。

ACL for Windows が表示されます。

カスタムドメインを使用した認証

1. デスクトップにある **ACL for Windows 16** ショートカットをダブルクリックします。
2. Launchpad サインイン画面の下で、**カスタムドメインにサインイン**]をクリックします。
3. 御社のカスタムドメインを入力し、**続行**]をクリックします。

自分のカスタムドメインが分からない場合は、会社の Analytics アカウント管理者にお問い合わせください。

4. SSO 資格情報を入力します。
5. HighBond インスタンスが選択されていることを確認し、**Analytics のアクティブ化**]をクリックします。

ACL for Windows が表示されます。

Analytics の開始

Analytics の作業を開始するには、ACL for Windows のメイン画面でオプションを選択します。

オプション	アクション
新しい空の Analytics プロジェクトを作成する	作成する]から、 Analytics プロジェクト]をクリックします
既存の Analytics プロジェクトを開く	開く]から、 Analytics プロジェクト]をクリックします
最近使用した Analytics プロジェクトまたはサンプルの Analytics プロジェクト (*.acl) を開く	最近のアナリティクスファイル]、または サンプルファイル]から、プロジェクト名をクリックします

ACL for Windows インストールおよびアクティベーションの概要

このセクションでは、非 Unicode または Unicode 版をインストールするかどうかに関するガイダンスを含む、ACL for Windows のインストールまたはアクティベーションに関する一般的な情報を提供します。

ACL for Windows には次のコンポーネントがあります。

- Analytics
- オフライン プロジェクト

Analytics の非 Unicode 版または Unicode 版をインストールすべきですか？

Analytics には Unicode 版と非 Unicode 版があります。いずれのエディションも同じインストールパッケージに含まれ、インストール時にインストールするエディションを指定します。

Unicode データの表示または分析のための要件がある場合を除き、非 Unicode 版をインストールする必要があります。Unicode データは、Analytics の Unicode 版でのみ開くことができます。

グローバル情報システムのある環境で作業する、または複数の言語を含むデータを分析する場合に、Unicode データを見る可能性が高くなります。

Unicode 版が必要なとき

次のデータを表示または分析するには、Unicode 版をインストールする必要があります。

- アジア文字
- 非 Unicode、または従来の文字エンコーディングの組み合わせ

たとえば、次の文字エンコーディングの 2 つ以上から成る言語の組み合わせ

- ラテン 1(英語および西欧)
- ラテン 2(中欧)
- キリル文字
- ギリシャ文字
- アラビア文字

メモ

中国語または日本語の Analytics ユーザー インターフェイスを使用するための唯一の選択肢は、Unicode エディションをインストールすることです。Unicode エディションが必要な理由は、データの言語でなくユーザー インターフェイスの言語に関連しています。

現在使用中の Analytics のエディションを特定するにはどうすればよいですか？

現在使用中の Analytics のエディションを特定するには、**[ヘルプ > バージョン情報]** を選択し、製品およびサブスクリプション情報が記載されているダイアログ ボックスを開きます。バージョン番号の後に **Unicode** または **非 Unicode** が表示されます。

Analytics の非 Unicode 版と Unicode 版の間での変更

ACL for Windows インストーラーを使用して、非 Unicode 版の Analytics を Unicode 版の Analytics へアップグレードすることはできません。逆も同様です。

バージョンのアップグレードと同時に Analytics のエディションも変更するには、まず既存の Analytics をアンインストールし、その後、新しいバージョンの新規インストールを実行します。

言語サポート

ACL for Windows は英語とその他 6 言語で用意されています。

以下の表に、利用可能な言語と、それらの言語における非 Unicode 版および Unicode 版のサポートをまとめてあります。

言語	サポートされているエディション
中国語	Unicode
英語	非 Unicode、Unicode
フランス語	非 Unicode、Unicode
ドイツ語	非 Unicode、Unicode
日本語	Unicode
ポルトガル語	非 Unicode、Unicode
スペイン語	非 Unicode、Unicode

インストール-作業の概要

ACL for Windows をインストールするには、次の作業を実行します。

1. **要件の確認** - ACL for Windows をインストールするコンピューターの構成が、ソフトウェアとハードウェアの最小要件を満たしているか確認します。

詳細については、"ACL for Windows のシステム要件" ページ 2707を参照してください。

2. **管理者権限** - インストールを行うコンピューターの管理者権限を持っていることを確認します。
3. **ダウンロード** - Diligent からのウェルカム メール、または HighBond からのメール通知の情報を使用して、Launchpad(www.highbond.com) から、ACL for Windows インストールパッケージをダウンロードします。
4. **インストール** - このガイドの説明を使用して ACL for Windows をインストールします。
5. **アクティブ化** - Analytics をアクティブ化します。

Analytics のインストールとアクティベーションの詳細については、"ACL for Windows をインストールする" ページ 2679のインストールを参照してください。

アクティベーションとライセンスの詳細については、[Launchpad ヘルプ](#)を参照してください。

前のバージョンからのアップグレード

以下の表は、別のバージョンの Analytics のアップグレード パスについて説明します。

現在インストールされている Analytics バージョン	アクション	結果
Analytics 16.0	バージョン 16.1 インストーラー (ACLforWindows161.exe) を実行する	Analytics がバージョン 16.1 にアップグレードされます
16.0 より前の ACL Analytics または ACL Desktop	バージョン 16.1 インストーラー (ACLforWindows161.exe) を実行する	Analytics 16.1 が前のバージョンと共存インストールされます アップグレードはありません。以前のバージョンの Analytics または ACL Desktop は、インストールしたままにすることもアンインストールすることもできます。

異なるバージョンまたはエディションの共存インストール

Analytics 16.1 は、以前のバージョンの Analytics または 16.0 より前のバージョンの ACL Desktop と共存インストールできます。

Unicode 版と非 Unicode 版の Analytics または ACL Desktop は、バージョン番号にかかわらず共存することはできません。

注意

Analytics 16.1 を以前のバージョンと共存インストールした場合、以前のバージョンの Analytics のデータコネクタが機能しなくなることがあります。旧バージョンのデータコネクタを引き続き動作させるには、これら 2 つのバージョンを同じコンピューターに共存インストールしないでください。

これらのデータコネクタはデータアクセス ウィンドウに表示されます。

将来のアップグレード

Analytics 16.1 をインストールまたはアップグレードすると、新しいバージョンが利用可能になったときに自動的に通知され、アップグレードをダウンロードしてインストールするオプションが提供されます。

サイレント インストール

IT 管理者は ACL for Windows をサイレントに、ユーザー操作を要せずにインストールすることもできます。詳細については、"サイレント インストールによる ACL for Windows のインストール" ページ 2687 を参照してください。

アクティブ化 およびアカウント 管理

ACL for Windows はサブスクリプション型のソフトウェアです。インストール時に、シリアル番号の入力は要求されません。代わりに、初めて使用する前にソフトウェアをアクティブ化する必要があります。

ソフトウェアのアクティブ化は、Launchpad(www.highbond.com) にサインインすることによって行います。これは、すべての Diligent オンプレミス ソフトウェアをダウンロードしたり、カスタマー サービスおよびリソースへアクセスしたりすることのできるポータル サイトです。Launchpad にサインインする方法は、Diligent からすべてのライセンス ユーザーへ送信されたウェルカム メールに記載されています。

Analytics アカウント 管理者

Analytics アカウント 管理者は通常、弊社に登録されている御社のご担当者です。アカウント 管理者は社内のサブスクリプションを管理します。必要に応じて、別のアカウント 管理者、または追加のアカウント 管理者を指名することができます。詳細については、サポートにお問い合わせください。

デフォルトでは、単一の Analytics ライセンスを持つ個々のお客様が Analytics アカウント 管理者です。

Launchpad を使用したアカウントの集中管理

Launchpad は集中型のアカウント管理を提供し、Analytics アカウント 管理者が次のタスクを実行できるようにしています。

- 社内の Analytics ユーザーを 1 つ以上の HighBond インスタンスに招待する
ユーザーをインスタンスに招待すると、自動的にそのユーザーが Launchpad に追加されます。Launchpad に追加されたら、ユーザーは使用可能な ACL for Windows ライセンスをアクティブ化することができます。
- インスタンスからユーザーを削除する
- 個々のユーザーの ACL for Windows ライセンスを取り消す
- サブスクリプション情報を表示する
- 組織の設定を更新する

HighBond インスタンスは、御社と直接関連付けることができます。つまり、御社は、さまざまな部門や作業単位を反映した2つ以上のHighBond インスタンスを作成することができます。

詳細については、[HighBond ヘルプ](#)を参照してください。

ACL for Windows インストーラーの配布

Diligent ソフトウェアはサブスクリプションベースであり、Analytics アカウント管理者は ACL for Windows インストールパッケージをダウンロードして、自社内の Analytics ユーザーに配布できるようになりました。各ユーザーに個別にダウンロードさせるよりも、インストールパッケージを社内で配布する方が簡単です。

ソフトウェアをインストールできるユーザー数に制限はありません。ただし、貴社のソフトウェアサブスクリプションの種類の利用規約により、一度に割り当てることができる最大ライセンス数が規定されています。詳細については、[サブスクリプションとライセンス](#)を参照してください。

インストールされていても、使用可能なライセンスでアクティブ化されていないソフトウェアは機能しません。

複数デバイスでのインストールとアクティベーション

ACL for Windows サブスクリプションの各ユーザーは、本ソフトウェアを複数のコンピューターにインストールできます。ただし、これは本ソフトウェアがインストールされるすべてのコンピューターで本ソフトウェアを個人で使用する場合に限ります。たとえば、次の複数デバイスへのインストールはいずれも許可されています。

- 会社のコンピューターと自宅のコンピューター
- デスクトップコンピューターと、移動時に使用するノートブックコンピューター
- 2台の会社のコンピューター。1台はスクリプトを実行するために使用し、もう1台はスクリプトを開発したり、一時的な分析を実行したりするために使用する場合。

本ソフトウェアは、インストールされている各コンピューターでアクティブ化する必要があります。ソフトウェアのアクティベーションは Diligent によって追跡されます。2人の異なるユーザーがソフトウェアを使用する場合は、2つの個別のライセンスが必要です。Galvanize マスターサブスクリプション契約にある以下の規定に従ってください。

指名ユーザーの ID とパスワードを他の個人と共有することは禁止されています。複数の個人間で指名ユーザーのアクセスを共有したりプールしたりすることで、部署や組織の複数のユーザーによる一時的な使用を許可することは、固く禁じられています。

構成可能なオプションの設定

構成可能なオプションは、Analytics の **オプション** ダイアログボックスの設定です。

グローバル設定

バージョン 10 以降の Analytics の構成可能なオプションのグローバル設定に加えた変更は、バージョン 16.1 のインストールに複製されます。

ACL Desktop 9.3 以前のバージョンでグローバル設定に加えた変更は、バージョン 16.1 では複製されません。バージョン 16.1 のインストールを完了した後、**オプション**]ダイアログボックスでカスタマイズした設定を再作成することができます。

プロジェクト固有の設定

構成可能なオプションにプロジェクト固有の設定がされている Analytics プロジェクトがある場合、Analytics 16.1 をインストールすると、これらの設定は保持されます。

Analytics の共存インストール

Analytics 16.1 が以前のバージョンの Analytics と共存インストールされている場合は、それぞれのバージョンで後から指定されたグローバルまたはプロジェクト固有の初期設定は別々に扱われ、お互いに影響を与えません。

詳しい情報

構成可能なオプションのグローバルおよびプロジェクト固有の設定の詳細については、"Analytics の初期設定ファイルの動作" ページ 148を参照してください。

ACL for Windows インストーラーに含まれるオフラインプロジェクト

オフラインプロジェクトは、ACL for Windows インストールパッケージに同梱されており、Analytics をインストールすると自動的にインストールされます。

オフラインプロジェクトはデスクトップベースのアプリケーションであり、HighBond ユーザーは、HighBond から切断中にプロジェクトから実地調査セクションで引き続き作業することが可能です。

HighBond ユーザーではない方には、オフラインプロジェクトはお使いのコンピューターでが無効になります。

詳細については、[オフラインプロジェクトの使用](#)を参照してください。

Diligent Unicode 製品

Diligent 製品の Unicode エディションでは、Unicode データが含まれるファイルを表示したり操作したりできます。

Unicode とは、ほとんどの世界の言語をサポートする文字暗号化を行う業界標準の方法です。

Analytics の非 Unicode 版または Unicode 版をインストールすべきですか？

Analytics には Unicode 版と非 Unicode 版があります。いずれのエディションも同じインストールパッケージに含まれ、インストール時にインストールするエディションを指定します。

Unicode データの表示または分析のための要件がある場合を除き、非 Unicode 版をインストールする必要があります。Unicode データは、Analytics の Unicode 版でのみ開くことができます。

グローバル情報システムのある環境で作業する、または複数の言語を含むデータを分析する場合に、Unicode データを見る可能性が高くなります。

Unicode 版が必要なとき

次のデータを表示または分析するには、Unicode 版をインストールする必要があります。

- アジア文字
- 非 Unicode、または従来の文字エンコーディングの組み合わせ
たとえば、次の文字エンコーディングの 2 つ以上から成る言語の組み合わせ
 - ラテン 1(英語および西欧)
 - ラテン 2(中欧)
 - キリル文字
 - ギリシャ文字
 - アラビア文字

メモ

中国語または日本語の Analytics ユーザー インターフェイスを使用するための唯一の選択肢は、Unicode エディションをインストールすることです。Unicode エディションが必要な理由は、データの言語でなくユーザー インターフェイスの言語に関連しています。

一言語だけで書かれたデータ

作業するデータが英語のみである、または西欧言語の 1 つのみを使用する場合は、高い可能性で非 Unicode 版をインストールする必要があります。ただし、英語のみのファイルを Unicode とすることが可能であることを認識しておいてください。

メモ

操作する御社データでどのような文字エンコードが使用されているかが不明な場合は、IT 部門にお問い合わせください。

Unicode データに対する非 Unicode 版 Analytics の使用

Unicode データに対して非 Unicode 版の Analytics を使用することが可能である場合と望ましい場合があります。

作業する Unicode データのすべての文字が、たとえば英語のみのデータなど、従来の文字暗号化の 1 つでサポートされている場合は、Unicode 版の Analytics を使用する必要はありません。このデータを非 Unicode 版の Analytics にインポートする場合、テキスト フィールドは、データを損傷したり破損することなく、Unicode から ASCII に自動的に変換されます。

このアプローチが望ましい理由については、「Unicode 版の欠点」見開きページを参照してください。

メモ

Unicode データを非 Unicode 版の Analytics にインポートし、そのデータに拡張 ASCII 文字セットでサポートされていない文字が含まれている場合は、データの破損が生じます。

重要なのはデータの言語

作業するデータの言語とは、通常、インストールすべき Analytics のエディションを指示する言語であり、Analytics のユーザー インターフェイスの言語ではありません。

たとえば、組織では Analytics のスペイン語 インターフェイスを使用しているかもしれませんが、非 Unicode 版と Unicode 版のどちらをインストールするかは、データ内で見る可能性がある言語に基づいて選択します。

ただし、Analytics の中国語および日本語のユーザー インターフェイスは、Analytics のエディションの選択に関する一般的なガイドラインにおける例外です。いずれのインターフェイスにも Unicode エディションしかありません。Analytics のローカライズされているインターフェイスと Unicode のサポートの詳細については、「言語サポート」ページ 2667を参照してください。

現在使用中の Analytics のエディションを特定するにはどうすればよいですか？

現在使用中の Analytics のエディションを特定するには、**[ヘルプ > バージョン情報]**を選択し、製品およびサブスクリプション情報が記載されているダイアログ ボックスを開きます。バージョン番号の後に **Unicode** または **非 Unicode** が表示されます。

ロボット ユーザー

組織で使用するロボット エージェントのエディションと一致する Analytics のエディションをインストールする必要があります。エディションが一致しない場合、Analyticsはロボットまたはロボット エージェントと連携することができません。

Unicode 版の欠点

Unicode 版 Analytics には以下の欠点があります。

- **データファイルのサイズが大きい** - Unicode データは、非 Unicode データの約 2 倍の記憶域が必要になります。これは、各文字が 1 バイトではなく 2 バイトで表されるためです。
- **パフォーマンスが低下する可能性** - データファイルが大きくなるため一部の Analytics コマンドは明らかに実行時間が長くなる場合があります。これは、Unicode 版では処理されるデータ量が 2 倍になるためです。

これらの欠点があるため、実際に Unicode データで作業する必要がある場合のみ Unicode 版をインストールしてください。

Analytics の 1 バイト文字と 2 バイト文字

非 Unicode 版の Analytics

データファイルの読み取りと書き込み時には、非 Unicode 版の Analytics は 1 バイト文字セット (SBCS) でのみ動作します。1 バイト文字セットでは、1 バイトのデータを使用して各文字を表し、最大 256 の異なる文字がサポートされます。

非 Unicode 版 Analytics で使用される 1 バイト文字セットは、コンピューターのシステム ロケール設定で指定された言語によって異なります。システム ロケールが英語または西欧言語のいずれかを指定する場合は、Windows-1252 文字セットが使用されます。Windows-1252 は「Windows Latin 1」とも呼ばれます。Windows コントロールパネルでシステム ロケールを設定できます。

1 バイト文字セットを参照する他の一般的な方法は、「ANSI」、「ANSI 文字セット」、「拡張 ASCII」です。

メモ

非 Unicode 版 Analytics がデータを処理するために使用する文字セットは、必ずしも、Analytics ユーザー インターフェイスのテキストで使用される文字セットと同じではありません。

Unicode 版 Analytics

データの読み込み

Unicode 版 Analytics は 2 バイトまたは 1 バイト文字セットを読み取ることができます。2 バイト Unicode 文字は 2 バイトのデータを使用して、各文字を表します。2 バイト (以上) のデータを使用して文字をエンコーディングすると、Unicode は、1 文字セットの世界の言語すべてを表すことができます。

データの書き込み

出力ファイルを作成する書き込み処理では、一般的に、Unicode 版 Analytics は 2 バイト UTF-16 文字エンコーディングを使用します。一部の処理については、出力ファイルはソースファイルにある 1 バイト文字エンコーディングを維持します。

型の数と文字数の比較

2 バイト Unicode データを操作しているときは、**テーブルレイアウト** ダイアログボックスに示されるバイト単位のフィールドの長さ、文字単位のフィールドの長さの違いに留意してください。

たとえば、Unicode フィールドの長さが **テーブルレイアウト** ダイアログボックスで 44 バイトの場合、フィールドには実際に 22 文字が含まれます。

ACLスクリプトでバイトと文字が問題になる理由

STRING() や SUBSTRING() などの、フィールド長パラメータを持つ関数を使用する場合には、長さはバイトではなく文字で指定します。逆に、DEFINE FIELD などの一部のコマンドでは、文字ではなく、バイトでフィールド長を指定する必要があります。

非 Unicode 版 Analytics では、1 バイトが 1 文字に相当します。このため、バイトと文字の違いは問題になりません。Unicode 版 Analytics では、ダブルバイトの Unicode データを操作するときには、2 バイトが 1 バイトに相当するため、違いが問題になります。

特定の[コマンド](#)と[関数](#)で使用する単位のタイプの詳細については、ACLScript ドキュメントを参照してください。

Unicode 版 Analytics へのテキスト ファイルのインポート

テキスト ファイルの文字エンコードは、テキスト ファイルが Unicode 版 Analytics にどのようにインポートされるかと、インポート後の Analytics テーブル内の文字フィールドで使用されるデータ型に影響します。

ASCII および EBCDIC ファイルを Unicode 版 Analytics にインポートするときには、次の 2 つのオプションがあります。

- 文字データ型を UNICODE に変換し、Analytics データ ファイルに変換する
後から UNICODE データ型を ASCII または EBCDIC に変更する場合、フィールドの文字が正しく表示されません。
- ASCII または EBCDIC 文字エンコーディングを維持し、Analytics テーブルレイアウトを Analytics データ ファイルを使用せずに作成する
Analytics テーブルレイアウトはソーステキスト ファイルにリンクされ続けます。

テキスト ファイル文字エンコーディング	データ定義ウィザード オプション	Analytics テーブルの文字データ型	文字長
UTF-16 LE (Unicode)	Unicode テキスト	UNICODE	2 バイト文字
UTF-8 (Unicode)	エンコードされたテキスト + データファイルの適切な文字セット (コード ページ)	UNICODE	2 バイト文字
拡張 ASCII (ANSI 文字セット)	ASCII > 区切り文字付きテキスト ファイル ASCII > 印刷イメージ (レポート) ファイル	UNICODE	2 バイト文字
	ASCII > その他のファイル形式	ASCII	1 バイト文字
EBCDIC	EBCDIC > 印刷イメージ (レポート) ファイル	UNICODE	2 バイト文字
	EBCDIC > その他のファイル形式	EBCDIC	1 バイト文字

リトルエンディアンおよびビッグエンディアン データ

「リトルエンディアン」(LE) および「ビッグエンディアン」(BE) は、Unicode データをエンコードするための 2 つの異なる方法を示す用語です。Microsoft Windows コンピューターから作成された Unicode データは通常、リトルエンディアンとしてエンコードされます。Windows コンピューターで Analytics を使用している場合は、ビッグエンディアン データを解析できません。

非 Unicode 版 Analytics プロジェクトの Unicode への変換

Unicode 版 Analytics で非 Unicode 版 Analytics プロジェクトを開くことはできますが、その逆の、非 Unicode 版 Analytics で Unicode 版 Analytics プロジェクトを開くことはできません。

	非 Unicode 版 Analytics で開く	Unicode 版 Analytics で開く
非 Unicode プロジェクト	はい	はい
Unicode プロジェクト	いいえ	はい

プロジェクトの変換

Unicode 版 Analytics で非 Unicode 版 Analytics プロジェクトを開く場合、プロジェクトと関連ログファイルを Unicode に自動的に変換する旨プロンプト表示されます。プロジェクトの変換を進める場合、元の非 Unicode プロジェクトとログファイルのコピーが .OLD というファイル拡張子を付けて保存され、それらが変更されることはありません。

メモ

非 Unicode 版 Analytics プロジェクトを Unicode に変換した場合、そのプロジェクトを非 Unicode 版 Analytics で開くことはできなくなります。そのプロジェクトを非 Unicode に再変換することはできません。必要に応じて、.OLD ファイルを使用すると、プロジェクトの非 Unicode 版を復元できます。

Analytics データ ファイル

非 Unicode 版 Analytics プロジェクトを Unicode に変換しても、関連する Analytics データファイル(.fill)は Unicode に変換されません。Unicode プロジェクトでは、1 バイトの ASCII (ANSI) データのままです。

メモ

Unicode 版 Analytics では、変換されていない 1 バイト データのフィールドのバイト位置とバイト長は、非 Unicode 版 Analytics と同じ方法で動作します。1 バイトは 1 文字に相当します。バイト位置またはバイト長を参照する変換されていないデータに対してコマンドを実行する場合は、この違いに注意してください。

Analytics の Unicode 用関数

Analytics には、データの解析および変換を支援する、6 つの Unicode 固有の関数があります。次の表ではこれらの関数の概要を説明しています。これらの関数は Unicode 版 Analytics にのみ搭載されています。

関数	目的
BINTOSTR()	ZONED または EBCDIC 文字データから変換された Unicode 文字データを返します。"Binary to String" の省略形です。 この変換により、確実に ZONED または EBCDIC でエンコードされた値を正しく表示できるようになります。

関数	目的
DBYTE()	レコード内の指定されたバイト位置にある Unicode 文字を返します。
DHEX()	Unicode 文字列を 16 進数の文字列に変換します。 HTOU() の逆です。
HTOU()	16 進数の文字列を Unicode 文字列に変換します。"Hexadecimal to Unicode" の省略形です。 DHEX() の逆です。
DTOU()	Analytics 日付値を指定された言語およびロケール書式の Unicode 文字列に変換します。"Date to Unicode" の省略形です。 UTOD() の逆です。
UTOD()	書式設定された日付が含まれる Unicode 文字列を Analytics 日付値に変換します。"Unicode to Date" の省略形です。 DTOU() の逆です。

ACL for Windows をインストールする

ここでは、ACL for Windows をインストールまたはアップグレードする手順を示します。

メモ

IT 管理者は ACL for Windows をサイレントに、ユーザー操作を要せずにインストールすることもできます。詳細については、「サイレント インストールによる ACL for Windows のインストール」 ページ 2687を参照してください。

お客様が所属する御社の窓口担当者である場合：

御社の窓口担当者本人の場合は、Launchpad アカウントにサインインする方法が記載されている、Diligent からのウェルカム電子メールをお受け取りになっていることと思います。

ウェルカム電子メールをお受け取りになっていない場合は、サポートにお問い合わせください。

メモ

Launchpad アカウントにサイン インして ACL for Windows を使用し、御社の他の ACL for Windows ユーザーにライセンスを供与できる必要があります。

お客様が窓口担当者でない場合

窓口担当者のご本人ではない方は、窓口担当者、または会社の Analytics アカウント管理者があなたを Launchpad に追加した場合は、電子メール通知が送信されているはずです。電子メール通知に Launchpad へのサイン インの説明が記載されています。**ACL for Windows を使用するには、Launchpad にサイン インできる必要があります。**

電子メール通知をお受け取りになっていない場合は、貴社の Analytics アカウント管理者にお問い合わせください。スパムフィルターもチェックしてください。電子メール通知は notifications@highbond.com から送信されます。

Microsoft .NET Framework 前提条件に関する重要な情報

Windows 8.1 更新 KB2919355 は Microsoft .NET Framework 4.6.x で必要であり、ACL for Windows 16 で必要です。

Windows 8.1 を使用し、.NET 4.6.x をインストールしてなく、更新 KB2919355 を実行していない場合は、ACL for Windows インストーラーが終了し、.NET 4.6.2 前提条件のインストール中にエラーメッセージが表示されます。

ACL for Windows インストールを続行する前に、更新 KB2919355 をダウンロードしてインストールする必要があります。

あるいは、ACL for Windows インストールを開始する前に更新 KB2919355 をインストールし、エラーメッセージを回避できます。

Analytics プロジェクト およびデータ ファイル

Analytics をインストールまたはアップグレードしても、以前に作成した Analytics プロジェクト、およびデータ ファイルへの影響はなく、すべて新しいバージョンで使用できます。

Analytics サンプル データ ファイル

Analytics をインストールまたはアップグレードする場合、インストールまたはアップグレード中に指定した Analytics 作業ディレクトリにある既存の Analytics サンプル データ ファイルは上書きされます。

作業ディレクトリのデフォルトの場所は次のとおりです。

`C:\Users\ユーザーアカウント名\Documents\ACL Data\Sample Data Files\`

バージョン 11 より前の作業ディレクトリは次のとおりです。

`C:\ACL DATA\Sample Data Files`

注意

サンプルプロジェクトやデータファイルに対して行ったどのような変更も保持したい場合は、インストールまたはアップグレードを実行する前に、他の場所にファイルを保存するか、それらが含まれるフォルダーの名前を変更してください。関連付けられたコマンド ログ ファイルを維持したい場合は、ログと同じ操作を実行します。

Analytics が認証局に接続できることを確認する

Analytics はデジタル証明書を使用して、その信頼性と完全性を検証します。証明書は、検証できるように、サードパーティの認証局である DigiCert へのインターネット接続が必要です。ネットワークファイアウォールの背後に Analytics をインストールする場合、ファイアウォールは、証明書が DigiCert に接続できるように設定されている必要があります。必要な接続がないと、Analytics を使用することはできません。

必要な場合は、組織の IT 部門と協力して、証明書と DigiCert の接続を許可するネットワークファイアウォールまたはプロキシサーバーを設定してください。Analytics のデジタル証明書は、以下の URL を使用して DigiCert の証明書取り消しリストと OCSP サーバーに接続します。

- <http://cacerts.digicert.com>
- <http://crl3.digicert.com>
- <http://crl4.digicert.com>
- <http://ocsp.digicert.com>

より詳細な URL 情報については、インストール済みの証明書を検査できます。

インストール済みの証明書を確認する

Analytics をインストールした後、インストール済みの証明書を検査すると、認証局の URL を確認できません。

1. Analytics インストールディレクトリで、**ACLWin.exe** を右クリックして、**プロパティ**]を選択します。
デフォルトのインストールディレクトリは **C:\Program Files (x86)\ACL Software\ACL for Windows 16** です。
2. **ACLWin.exe プロパティ**]ダイアログボックスで **デジタル署名**]タブを選択します。
3. **署名リスト**]リストで **Diligent Corporation** をダブルクリックし、**証明書の表示**]をクリックします。
4. **証明書**]ダイアログボックスの **詳細**]タブで、次の各フィールドを選択します。

- **CRL 配布ポイント**
- **認証局情報アクセス**

各フィールドで、URL は、ダイアログボックスの下部にある表示領域に表示されます。

5. **証明書**]ダイアログボックスの **証明書パス**]タブで、以下を実行します。
 - a. 次の証明書パスの各項目を選択します。
 - **DigiCert Trusted Root G4**
 - **DigiCert Trusted G4 コードが署名中です...**
 - b. 各項目について、**証明書の表示**]をクリックします。
 - c. **詳細**]タブで、これらの各フィールドを選択します。

- **認証局情報アクセス**
- **CRL 配布ポイント**

各フィールドで、URL は、ダイアログボックスの下部にある表示領域に表示されます。

インストール手順

注意

インストール処理中に特定の時点でコンピューターを再起動するように指示された場合は、すぐに再起動してください。コンピューターの再起動のメッセージは無視しないでください。

指示されたときにコンピューターを再起動しないと、.NET、他の前提条件、または ACL for Windows のインストールで問題が生じるおそれがあります。

インストーラーのダウンロードとインストールの開始

1. Launchpad から、ACL for Windows のインストールパッケージ([ACLforWindows161.exe](#)) をダウンロードします。
 - a. Launchpad(www.highbond.com) にサインインします。
 - b. [ソース]の下にある **ダウンロード**]をクリックします。
 - c. 最上部のメニューにある **ACL for Windows**]をクリックします。
 - d. **[バージョン 16.1 のダウンロード]**をクリックします。

会社の Analytics アカウント 管理者は他のユーザーにインストール パッケージを提供することもできます。

インストール パッケージには、Analytics の非 Unicode 版と Unicode 版の両方が含まれています。
2. 他のすべての Windows アプリケーションを閉じます。
3. インストール パッケージをダブルクリックします。
4. セキュリティ警告のダイアログ ボックスが表示された場合は、リストされた情報を確認して、**[はい]** をクリックします。

ヒント

[セキュリティ警告]ダイアログ ボックスは開いたウィンドウの後ろに配置して非表示にできます。

5. インストールに使用したい言語を選択して、**[OK]**をクリックします。

インストールファイルの展開

[セットアップの展開場所] ページで、インストール ファイルを展開するフォルダーを指定して、**[展開]** をクリックします。

[参照] をクリックしてフォルダーを選択するか、または次のデフォルトの場所をそのまま使用します。

C:\Users\ユーザーアカウント名\Downloads\ACL Installers\ACLforWindows161

セットアップ ファイルは指定したインストール先フォルダーに展開されます。ファイルが展開された後、インストーラーが自動的に開始されます。

メモ

インストーラーが何らかの理由で起動されなかった場合は、Windows エクスプローラーを使ってセットアップ ファイルの展開先フォルダーに移動し、**setup.exe** をダブルクリックします。

必須コンポーネントのインストール(必要な場合)

必須コンポーネントをインストールするよう求められた場合は、**[インストール]**をクリックします。必須コンポーネントがインストールされた後、インストーラーは自動的に進みます。

ソフトウェア必須コンポーネントの一覧については、"自動的にインストールされる前提条件" ページ 2710 を参照してください。

メモ

ACL for Windows インストーラーが .NET 4.6.2 必須コンポーネントのインストール中にエラーメッセージで終了した場合は、[Windows 8.1 更新 KB2919355](#) をダウンロードしてインストールし、コンピューターを再起動してから、ACL for Windows インストーラーを再起動します。

Windows のインストールまたはアップグレードのための主要な ACL を実行する

1. アップグレードを確認するよう求められたら、**[はい]**をクリックします。
2. **[よろこ]** ページで、**[次へ]**をクリックします。
3. **[使用許諾契約]** ページで、**[上記の条項に同意します]** を選択して、**[次へ]**をクリックします。
4. **[ACL エディションの選択]** ページで、インストールするエディションを選択し、**[次へ]**をクリックします。

- 非 Unicode
- Unicode

注意

インストールするエディションが、組織に合ったエディションであることを確認してください。

詳細については以下を参照 "Analytics の非 Unicode 版または Unicode 版をインストールすべきですか?" ページ 2666

メモ

Analytics を旧バージョンと共存インストールするか、Analytics をアップグレードしようとするか、インストーラーにより、その現在インストールされている旧バージョンと同じエディションが強制的に選択されます。

非 Unicode 版と Unicode 版を並べてインストールすることはできません。

5. **[任意の Analytics データコネクタ]** ページで **[次へ]** をクリックして、Analytics の任意のデータコネクタのインストールを許可します。
ページの一覧に表示されている任意のデータコネクタをインストールしない場合は、**[任意の Analytics データコネクタをインストールする]** をオフにしてから、**[次へ]** をクリックします。将来、任

意のデータコネクタをインストールする場合は、ACL for Windows をアンインストールしてから再インストールする必要があります。

Analytics をアップグレードする場合、インストーラーはバージョン 16.0 に対して行った選択を強制します。

6. **機械学習を有効にする]** ページで **次へ]** をクリックして、Analytics 機械学習コマンドを有効にするために Python エンジンのインストールを許可します。

任意の Python エンジンをインストールしない場合は、**機械学習を有効にする(Analytics 版 Python エンジンをインストールする)**] をオフにしてから、**次へ]** をクリックします。将来、Python エンジンをインストールする場合は、ACL for Windows をアンインストールしてから再インストールする必要があります。

Analytics をアップグレードする場合、インストーラーはバージョン 16.0 に対して行った選択を強制します。

7. **転送先フォルダー]** ページで、ACL for Windows アプリケーションファイル、および Analytics のサンプルデータファイルがインストールされる場所を指定し、次に **次へ]** をクリックします。

必要に応じて、**変更]** をクリックしてデフォルトの場所の一方または両方を変更します。

サンプルデータファイルをインストールする場所がデフォルトの Analytics の作業ディレクトリーになります。

デフォルトの場所:

ACL for Windows アプリケーションファイル (64ビットオペレーティングシステム)	C:\Program Files (x86)\ACL Software\ACL for Windows 16\
ACL for Windows アプリケーションファイル (32ビットオペレーティングシステム)	C:\Program Files\ACL Software\ACL for Windows 16\
Analytics サンプルデータファイル	C:\Users\ユーザーアカウント名\Documents\ACL Data\Sample Data Files\

8. **プログラムをインストールする準備ができました]** ページでは、**[インストール]** をクリックします。ACL for Windows がインストールまたはアップグレードされます。
9. Microsoft Access データベースエンジンのインストールを選択するプロンプトが表示された場合は、次のいずれかを実行してください。
- 古い Excel ファイル(*.xls) や Microsoft Access ファイル(*.mdb) をインポートしたり、それらにエクスポートしたりする必要がある場合は、
 - **[はい]** をクリックしてデータベースエンジンをインストールします。Analytics で現行世代の Excel ファイル(*.xlsx) のみを使用し、古い Excel ファイルや Microsoft Access ファイルを使用しない場合は、
 - **[いいえ]** をクリックしてデータベースエンジンを除外します。データベースエンジンは、他の新しい Microsoft Office 製品との互換性の問題を引き起こす可能性があります。このため、データベースエンジンは、実際に必要な場合を除き、除外する必要があります。
10. インストールまたはアップグレード処理が完了したら、**完了]** をクリックします。

Analytics のアクティブ化

メモ

以前のバージョンからアップグレードする場合、または以前のバージョンと共存インストールする場合、Analytics は事前にアクティブ化されている場合があります。

Analytics のアクティブ化の方法は、御社の認証方法によって異なります。

- **標準認証** - 単一の Analytics サブスクリプションをご利用の場合は、高い確実性で標準認証を使用します。
- **カスタムドメインを使用する認証** - 御社がシングルサインオン(SSO)をご利用の場合は、カスタムドメインを使用して認証します。

メモ

御社がカスタムドメインをご利用の場合は、Launchpad サインイン画面の最初の画面で電子メールアドレスとパスワードを入力しないでください。

標準認証

1. デスクトップにある **ACL for Windows 16** ショートカットをダブルクリックします。
Launchpad サイン イン画面が開きます。
2. ユーザー名(電子メール)とパスワードを入力して、**[サインイン]**をクリックします。
これらの資格情報は、ウェルカムメールまたは電子メール通知の説明に従って手続きされていれば、この時点で既にお持ちのはずです。
3. HighBond インスタンスを選択するように指示されたら、そのようにし、**[Analytics のアクティブ化]**をクリックします。
ACL for Windows が表示されます。
オフラインプロジェクトのみを使用する必要がある場合は、**[ライセンスをアクティブ化せずに続行する]**をクリックすることもできます。

カスタムドメインを使用した認証

1. デスクトップにある **ACL for Windows 16** ショートカットをダブルクリックします。
Launchpad サイン イン画面が開きます。
2. Launchpad サインイン画面の下で、**[カスタムドメインにサインイン]**をクリックします。

メモ

このサインイン方法は御社で設定されている場合のみ利用できます。

3. 御社のカスタムドメインを入力し、**続行]**をクリックします。

自分のカスタムドメインが分からない場合は、会社の Analytics アカウント管理者にお問い合わせください。

4. SSO 資格情報を入力します。
5. HighBond インスタンスが選択されていることを確認し、**Analytics のアクティブ化]**をクリックします。

ACL for Windows が表示されます。

オフラインプロジェクトのみを使用する必要がある場合は、**ライセンスをアクティブ化せずに続行する]**をクリックすることもできます。

Analytics の開始

Analytics の作業を開始するには、ACL for Windows のメイン画面でオプションを選択します。

オプション	アクション
新しい空の Analytics プロジェクトを作成する	作成する] から、 Analytics プロジェクト] をクリックします
既存の Analytics プロジェクトを開く	開く] から、 Analytics プロジェクト] をクリックします
最近使用した Analytics プロジェクトまたはサンプルの Analytics プロジェクト (*.acl) を開く	最近のアナリティクスファイル] 、または サンプルファイル] から、プロジェクト名をクリックします

ツールバーとコマンドラインの表示

Analyticsで、ツールバーまたはコマンドラインを表示するには、**ウィンドウ]> ツールバー]**または **ウィンドウ]> コマンドライン]**を選択します。

サイレント インストールによる ACL for Windows のインストール

重要

Windows 8.1 オペレーティングシステムのコンピューターに ACL for Windows をサイレントインストールする場合は、まず、Windows Update KB2919355 をインストールする必要があります。

この要件は、Microsoft .NET 4.6.x 必須コンポーネントがまだインストールされていないコンピューターにのみ適用されます。

詳細については、「ACL for Windows のシステム要件」ページ 2707を参照してください。

メモ

ACL for Windows のサイレント インストールでは、Python エンジンまたは任意のデータコネクタの除外がサポートされていません。サイレント インストールでは、このような任意のコンポーネントが自動的にインストールされます。

任意のコンポーネントを除外する場合は、インストールウィザードを使用してインストールを実行する必要があります。

詳細については、「Windows のインストールまたはアップグレードのための主要な ACL を実行する」ページ 2683を参照してください。

多くのワークステーションに ACL for Windows を展開する必要がある場合は、サイレント インストールを使用して、ユーザー操作を要せずにインストーラーを実行することができます。

サイレントインストールを使用して、ACL for Windows 16.1 の新規インストールを実行できます。

サイレント インストールを実行した後に、ACL for Windows をアクティブ化する必要があります。ACL for Windows をアクティブ化する手順については、「Analytics のアクティブ化」ページ 2685を参照してください。

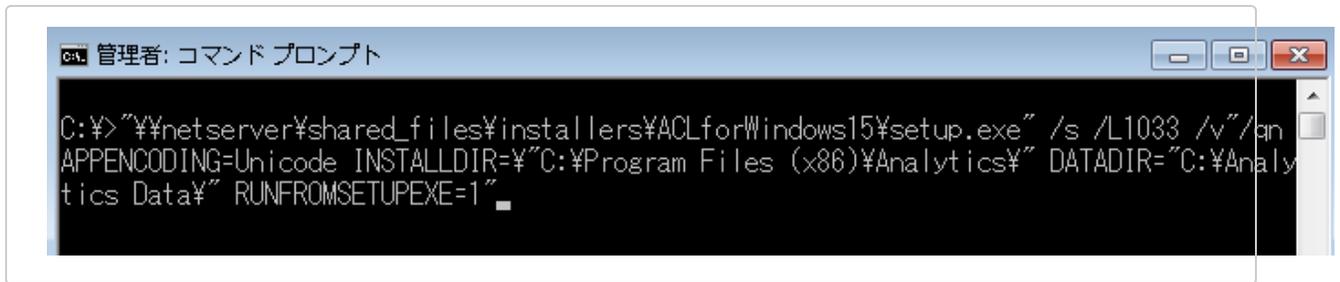
機能の仕組み

サイレント インストールは、Windows コマンド ラインを使用し、事前を選択されたオプションと一緒に、ユーザー インターフェイスなしで、ACL for Windows インストーラーを実行します。

このコマンドは、インストールパッケージからのインストーラー ファイルのいずれかを使用します。どれを使用すべきか判断するには、「2 つのインストーラー ファイル」見開きページを参照してください。

例

以下の図に、Windows コマンド ラインで入力できるサイレント インストール構文例を提供します。



```
管理: コマンド プロンプト
C:\>"%netserver%shared_files%installers%ACLforWindows15%setup.exe" /s /L1033 /v"/qn
APPENCODING=Unicode INSTALLDIR="%C:\Program Files (x86)%Analytics%" DATADIR="%C:\Analy
tics Data%" RUNFROMSETUPEXE=1"
```

インストーラーファイルの展開

インストーラーファイルにアクセスするには、Launchpad(www.highbond.com) から ACL for Windows インストールパッケージ([ACLforWindows161.exe](#)) をダウンロードします。インストールパッケージを実行して 2 つのインストーラーファイルを抽出します。

インストールパッケージは、ファイルが展開された後、インストール処理を自動的に開始するように設計されています。サイレントインストールオプションを使用している場合は、ファイルが展開されたらすぐに [キャンセル] をクリックします。

展開されたファイルのデフォルトの場所は次のとおりです。

`C:\Users\ユーザーアカウント名\Downloads\ACL Installers\ACLforWindows161\`

2 つのインストーラーファイル

ACL for Windows インストールパッケージ([ACLforWindows161.exe](#)) には 2 つのインストーラーファイルがあります。

- [setup.exe](#)
- [ACL for Windows.msi](#)

サイレントインストールの実行に使用するインストーラーファイルは、必要なソフトウェア必須コンポーネントが対象コンピューターにインストール済みであるかどうかによって変わります。

ソフトウェア必須コンポーネントの一覧については、「ACL for Windows のシステム要件」ページ 2707 を参照してください。

setup.exe

[setup.exe](#) 必要なソフトウェア要件が対象コンピューターにインストールされていない場合は、それをインストールします

ACL for Windows.msi

[ACL for Windows.msi](#) を使用するには、対象コンピューターにすべてのソフトウェア要件をインストールする必要があります。

サイレント インストールを実行

1. インストールパッケージ `ACLforWindows161.exe` をダブルクリックし、インストーラーが抽出されたらすぐに **[キャンセル]** をクリックします。
2. 管理者として Windows コマンド プロンプトを開きます。

Windows のバージョンによりますが、管理者としてコマンド プロンプトを開くにはさまざまな方法があります。

この方法は Windows のすべてのバージョンで動作します。

Windows Explorer を使って、`C:\windows\system32` サブフォルダーで `cmd.exe` を右クリックし、**管理者として実行]** を選択します。

3. 以下のサイレント インストールコマンドのいずれかを実行します。

メモ

インストーラー ファイルの絶対パスを指定してください。パスにスペースが含まれている場合は、パスを引用符で囲ってください。

下記の例は一般的な構文のみを示しています。詳細な構文のガイドライン、およびサイレント インストールに関する追加情報が、後続のセクションで確認できます。

アクション	インストーラーファイル	サイレントインストールコマンド
ACL for Windows 16.1 をインストールする	<code>setup.exe</code>	<pre>"setup_exe のパスとファイル名" /s /L<言語 ID> /v"/qn APPENCODING=<Analytics エディション> RUNFROMSETUPEXE=1"</pre>
	<code>ACL for Windows.msi</code>	<pre>msiexec /i "msi のパスとファイル名" TRANSFORMS=<言語 ID>.mst APPENCODING=<Analytics エディション> /qn</pre>

サイレント インストールのガイドライン

ガイドラインを確認し、ガイドラインに従って、Windows のコマンド ラインでサイレント インストールのコマンド構文を入力してください。

注意

ガイドラインのいずれかに従わないと、サイレント インストールが失敗する原因となります。

ガイドライン	詳細
管理者として実行	.msi ファイルを使用して ACL for Windows をサイレント インストールするためには、Windows コマンドラインを管理者として実行する必要があります。cmd.exe を右クリックして、 管理者として実行]を選択します。
構文の改行	構文中に改行を入れないでください。Windows のコマンドラインは、構文を次の行へ自動的に折り返すことができます。
パラメーターの区切り	<p>パラメーターはスペースで区切ります。</p> <p>例：</p> <pre>TRANSFORMS=1033.mst INSTALLDIR="C:\Program Files\Analytics"</pre> <p>唯一の例外は、次のような構文の要素です。これは、スペースを入れないで入力する必要があります。/v"/qn"</p>
スペースを含む値	<p>スペースを含むパラメーター値やネットワークパスは、引用符で囲む必要があります。</p> <p>例：</p> <pre>DATADIR="C:\Analytics Data"</pre> <pre>"\\nas-server-2\installers\ACLforWindows161\ACL for Windows.msi"</pre>
/v パラメーターに続く複数のパラメーター	<p>/v パラメーターの後に続く複数のパラメーターは、引用符で囲む必要があります。</p> <p>例：</p> <pre>/v"/qn INSTALLDIR=C:\Analytics DATADIR=C:\Analytics_Data"</pre>
引用符内の引用符	<p>引用符の内側で引用符を記述する場合は、エスケープ文字として円記号を使用する必要があります (\")。</p> <p>例：</p> <pre>/v"/qn DATADIR=\"C:\Analytics Data\""</pre> <p>上記の例では、2つのパラメーター(/qn と DATADIR)を引用符で囲む必要があり、またスペースを含むパラメーター(C:\Analytics Data)も引用符で囲む必要があります。内側にある引用符のセットの各引用符の前に、エスケープ文字を置く必要があります。</p>
引用符の形式	構文をコピーして Windows のコマンドラインに貼り付ける場合は、すべての引用符が、波状の引用符ではなく直線状の引用符になっていることを確認してください。

コマンド ラインのパラメーター

setup.exe パラメーター

パラメーター	詳細
"setup_exe のパスとファイル名"	<p>setup.exe インストーラー ファイルへのネットワーク パスまたはローカル パスを指定します。パスにはファイル名とファイル拡張子を含める必要があります。</p> <p>エンド ユーザーのコンピューターに必須コンポーネントがまだインストールされていない場合は、このパラメーターを使用します。</p>
/s	setup.exe がサイレント モードで実行するよう指定します。
/L<言語 ID>	ACL for Windows ユーザー インターフェイスの言語を指定します。
/v	<p>パラメーター値がインストーラーに渡されることを指定します。</p> <ul style="list-style-type: none"> このとき、/v と最初のパラメーターの間にスペースを入れてはいけません。 /v に1 つ以上のパラメーターが続く場合、パラメーター リストを引用符で囲む必要があります。
SKIP_MS_ACCESS=TRUE	32 ビット Microsoft Access データベースエンジンを ACL for Windows のインストールの一部としてインストールしないことを指定します。
/RUNFROMSETUPEXE=1	このパラメーターと値 1 を使用すると、サイレント インストールが setup.exe インストーラー ファイルを使用することを指定します。

ACL for Windows.msi パラメーター

パラメーター	詳細
msiexec /i "msi のパスとファイル名"	<p>ACL for Windows.msi インストーラー ファイルへのネットワーク パスまたはローカル パスを指定します。パスにはファイル名とファイル拡張子を含める必要があります。</p> <p>エンド ユーザーのコンピューターに必須コンポーネントがすべてインストールされている場合は、このパラメーターを使用します。</p>
TRANSFORMS=言語 ID	ACL for Windows ユーザー インターフェイスの言語を指定します。

一般パラメーター - setup.exe と ACL for Windows.msi

パラメーター	詳細
/qn	ユーザー インターフェイスを表示しないで、インストーラーがサイレント モードで実行するよう指定します。

パラメーター	詳細
INSTALLDIR=ACL for Windows アプリケーション ファイル フォルダー へのパス	<p>アプリケーション ファイルのインストール先 フォルダを指定します。</p> <p>パラメーターを省略した場合は、デフォルト 値が使用されます。</p> <ul style="list-style-type: none"> 64 ビットオペレーティングシステム: -C:\Program Files (x86)\ACL Software\ACL for Windows 16\ 32 ビットオペレーティングシステム: -C:\Program Files\ACL Software\ACL for Windows 16\
DATADIR= サンプル データ ファイル フォルダーへのパス	<p>Analytics サンプル データ ファイルのインストール先 フォルダを指定します。このフォルダは、Analytics 作業 ディレクトリにもなります。</p> <p>パラメーターを省略した場合は、デフォルト 値が使用されます。</p> <p>C:\Users\ユーザーアカウント名\Documents\ACL Data\Sample Data Files\</p> <p>メモ エンド ユーザーに対して、指定したデータ ファイル フォルダの読み取りおよび書き込みが許可されていることを確認してください。</p>
APPENCODING=Analytics のエ ディション	<p>インストールする Analytics エディションを指定します。</p> <ul style="list-style-type: none"> APPENCODING=NonUnicode Analytics の非 Unicode エディションをインストールするように指定します。パラメーターを省略しても同様のことを行います。 APPENCODING=Unicode は、Analytics の Unicode エディションをインストールするように指定します。

ユーザー インターフェイスの言語の指定

ACL for Windows ユーザー インターフェイスの言語の指定

- /L パラメーターを `setup.exe` で使用する
- TRANSFORMS パラメーターを `ACL for Windows.msi` で使用する

各パラメーターを使用して、適切な言語 ID を指定する必要があります。

例：

- `setup.exe` を使用している場合に、英語のユーザー インターフェイスを指定するには、/L1033 を使用します。
- `setup.exe` を使用している場合に、スペイン語のユーザー インターフェイスを指定するには、/L1034 を使用します。
- `ACLfor Windows.msi` を使用している場合に、スペイン語のユーザー インターフェイスを指定するには、TRANSFORMS=1034.mst を使用します。

メモ

英語のオペレーティングシステムに英語版の ACL for Windows をインストールする場合でも、常に言語 パラメーターを使用することをお勧めします。

言語パラメーターの省略

言語パラメーターを省略した場合、インストールでは使用するインストーラーファイルに応じて、コンピューターのオペレーティングシステムの言語、または英語をデフォルトとして設定します。

言語パラメーターとインストーラーファイルの動作

使用されるパラメーター	setup.exe	ACL for Windows.msi
/L	ACL for Windows は指定言語を使用する	該当なし
TRANSFORMS	該当なし	ACL for Windows は指定言語を使用する
パラメーターを省略	ACL for Windows はオペレーティングシステムの言語を使用します。ACL for Windows がオペレーティングシステムの言語をサポートしていない場合は、英語を使用する	ACL for Windows は英語を使用する

ACL for Windows でサポートされている言語のサイレントインストールの構文

言語	setup.exe の構文	ACL for Windows.msi の構文
中国語	/L2052	TRANSFORMS=2052.mst
英語	/L1033	TRANSFORMS=1033.mst
フランス語	/L1036	TRANSFORMS=1036.mst
ドイツ語	/L1031	TRANSFORMS=1031.mst
日本語	/L1041	TRANSFORMS=1041.mst
ポルトガル語	/L1046	TRANSFORMS=1046.mst
スペイン語	/L1034	TRANSFORMS=1034.mst

Microsoft Access データベースエンジンの除外

サイレントインストールに **setup.exe** を使用する場合には、ACL for Windows のインストールから Microsoft Access データベースエンジンを除外するオプションがあります。

組織が Analytics で現行世代の Excel ファイル (*.xlsx) のみを使用し、古い Excel ファイルタイプ (*.xls) を使用しなくなっている場合は、データベースエンジンを除外することができます。詳細については、「Microsoft Access データベースエンジンを除外するオプション」ページ 2664 を参照してください。

- データベースエンジンを除外するには、SKIP_MS_ACCESS=TRUE を使用します。
- データベースエンジンを含めるには、SKIP_MS_ACCESS パラメーターを使用しないでください。

データベースエンジンが以前の ACL for Windows インストールの一部としてコンピューターに既にインストール済みの場合、SKIP_MS_ACCESS パラメーターはそれをアンインストールしません。

SKIP_MS_ACCESS パラメーターは、ACL for Windows.msi インストーラーファイルで使用するためのものではありません

非 Unicode または Unicode 版の指定

メモ

- インストーラーを使用して、非 Unicode エディションの Analytics を Unicode エディションの Analytics へアップグレードすることはできません。逆も同様です。
- Analytics または ACL Desktop の Unicode エディションと非 Unicode エディションは、共存させることができません。

インストールする Analytics のエディションを指定するには、`setup.exe` または `ACL for Windows.msi` で `APPENCODING` パラメーターを使用します。非 Unicode 版をインストールする場合は、このパラメーターは任意です。

- `APPENCODING=NonUnicode` を使用するか、パラメーターを使用せずに、非 Unicode エディションの Analytics をインストールします。
- `APPENCODING=Unicode` を使用して、Unicode エディションの Analytics をインストールします。

メモ

中国語および日本語版の Analytics は Unicode のみです。

構文例

32ビット オペレーティングシステムを実行中のコンピューターに ACL for Windows をインストールする場合、デフォルトの場所を使用しないでインストールディレクトリを指定するときは、以下の例の `C:\Program Files (x86)\` 部分を `C:\Program Files\` に置き換えてください。

setup.exe を使用したデフォルト設定での非 Unicode

次の例では、英語の非 Unicode 版と必要なソフトウェア前提条件をインストールします。

```
\\nas-server-2\shared_files\installers\ACLforWindows161\setup.exe /s /L1033  
/v"/qn RUNFROMSETUPEXE=1"
```

setup.exe を使用したデフォルト設定での Unicode

以下の例では、オペレーティングシステム言語の Unicode エディションを、32 ビット Microsoft Access データベースエンジンを除く、ソフトウェア必須コンポーネントとともにインストールします。

```
"\\nas-server-2\shared_files\installers\ACLforWindows161\setup.exe" /s  
/v"/qn SKIP_MS_ACCESS=TRUE APPENCODING=Unicode RUNFROMSETUPEXE=1"
```

setup.exe を使用した 2 つのカスタム設定での非 Unicode

次の例では、英語の非 Unicode 版と必要なソフトウェア前提条件をインストールします。

```
\\nas-server-2\shared_files\installers\ACLforWindows161\setup.exe /s /L1033  
/v"/qn APPENCODING=NonUnicode INSTALLDIR="C:\Program Files  
(x86)\Analytics\" DATADIR="C:\Analytics Data\" RUNFROMSETUPEXE=1"
```

setup.exe を使用したデフォルト設定での Unicode

次の例では、ドイツ語の Unicode 版と必要なソフトウェア前提条件をインストールします。

```
\\nas-server-2\shared_files\installers\ACLforWindows161\setup.exe /s /L1031  
/v"/qn RUNFROMSETUPEXE=1"
```

ACL for Windows.msi を使用した 1 つのカスタム設定での非 Unicode

次の例では、英語の非 Unicode 版(必要なソフトウェア前提条件はインストール済みである必要があります)をインストールします。

```
msiexec /i "\\nas-server-2\shared_files\installers\ACLforWindows161\ACL for  
Windows.msi" INSTALLDIR="C:\Program Files (x86)\Analytics"  
TRANSFORMS=1033.mst APPENCODING=Unicode /qn
```

ACL for Windows.msi を使用したデフォルト設定での非 Unicode

次の例では、中国語の Unicode 版(必要なソフトウェア前提条件はインストール済みである必要があります)をインストールします。

```
msiexec /i "\\nas-server-2\shared_files\installers\ACLforWindows161\ACL for Windows.msi" TRANSFORMS=2052.mst APPENCODING=Unicode /qn
```

ACL for Windows のアンインストール

ACL for Windows をアンインストールすると、以下をアンインストールすることになります。

- Analytics
- オフライン プロジェクト

すべての ACL for Windows アプリケーション ファイルはコンピューターから削除されますが、Analytics プロジェクト ファイル、データ ファイル、ログ、およびあらゆるプロジェクト固有の設定ファイルは Analytics プロジェクト フォルダーに残っています。

メモ

アプリケーションをアンインストールするには、管理者権限減を持つ Windows ユーザーとしてログインしている必要があります。

1. Windows コントロール パネルで **プログラムと機能** を開きます。
 2. **ACL for Windows** を選択し、**アンインストール** をクリックします。
 3. 確認ダイアログ ボックスで、**[はい]** をクリックします。
 4. セキュリティ警告のダイアログ ボックスが表示された場合は、リストされた情報を確認して、**[はい]** をクリックします。
 5. これらのプロンプトのいずれかまたは両方が表示されたら、**[OK]** をクリックします。
 - システムの実行中に更新できないファイルまたはサービスを、セットアップで更新する必要があります。
 - インストールを続ける前に、次のアプリケーションを終了してください。
- ACL for Windowsがアンインストールされます。
6. アンインストールを完了するには、コンピューターを再起動します。

Python の Analytics 連携用設定

メモ

これらの設定のインスタンスは、Analytics Python 関数を使用するために必要な顧客がインストールする Python インスタンスについて言及しています。Python のこのインスタンスは、機械学習コマンドをサポートするための Analytics のインストールの一環としてインストール可能な Python のインスタンスと同じものではありません。

Analytics で動作するように Python を設定するには、互換バージョンの Python をインストールし、Python 実行可能ファイルをコンピューターの PATH 環境変数に追加する必要があります。ACLPYTHONDLL および PYTHONPATH システム環境変数を設定する必要があります。

機能の仕組み

Python スクリプトを実行するには、Analytics では Python 実行可能ファイルを呼び出し、実行を指示するスクリプトを見つけられる必要があります。Analytics では、Python を探すには PATH 環境変数を使用し、スクリプトを探すには PYTHONPATH 環境変数を使用します。

Python をインストール (32ビット)

1. [Python ダウンロード ページ](#)] から、次のバージョンの Python の1つをお使いのコンピューターまたはサーバーにダウンロードします。
 - 3.4.x
 - 3.5.x
 - 3.6.x

メモ

リストにある Python のバージョンは、テスト済みであり、Analytics またはロボットエージェントで動作することを確認済みです。

バージョン 3.4.x 以上の Python は動作するはずですが、ただし、リストにないバージョンが動作するかは保証できません。

2. コンピューターまたはサーバーで、インストーラーをダブルクリックします。
3. インストーラーで、**Python バージョン番号の PATH への追加**] を選択します。
4. **インストール**] をクリックし、画面の説明に従います。
5. Analytics スクリプトで呼び出される Python スクリプトを実行する前に、コンピューターまたはサーバーを再起動します。

ACLPYTHONDLL および PYTHONPATH 環境変数の設定

1. お使いの Windows オペレーティングシステムの **C:** に、Python スクリプトを格納する 1 つまたは複数のフォルダーを作成します。
例 **-C:\python_scripts**。
2. オペレーティングシステムから、**システムのプロパティ**] ダイアログ ボックスを開き、**環境変数**] をクリックします。
3. **システム変数**] セクションで、**新規**] をクリックし、次の変数を入力します。

変数名	変数値
PYTHONPATH	Python スクリプトを格納するために作成したフォルダーへの絶対パス。複数のフォルダーのパスをセミコロンで区切ります。 例：

変数名	変数値
	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 0 auto;">C:\python_scripts;C:\dev;C:\tmp</div>
ACLPYTHONDLL	<p>Python インストールフォルダー内の Python DLL ファイルのフルパスとファイル名。</p> <p>例:</p> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;">c:\python_install\python35.dll</div> <p>メモ</p> <p>インストール時に、Python は DLL をインストールフォルダーではなく、システムフォルダー (<code>c:\windows\system32\python35.dll</code>) に追加します。システムフォルダーからインストールフォルダーに DLL をコピーし、変数値としてインストールフォルダーのロケーションを使用します。Analytics またはロボット エージェントは、Python インストールフォルダーの DLL を検索します。</p> <p>また、インストールフォルダーに読み取り専用設定があれば、その解除も必要です。</p>

- 変数を保存するには、**[OK]** をクリックして、**[システムのプロパティ]** ダイアログ ボックスで **[OK]** をクリックします。

Analytics の Python 関数における Python の使用

Analytics から、お使いの PYTHONPATH に存在するスクリプトの関数を呼び出すには、Analytics の Python 関数を使用します。

詳細については、「Python」 ページ 2153 を参照してください。

メモ

Python スクリプトに編集を行う場合は、Analytics プロジェクト でビューを更新し、最新バージョンの Python スクリプトを使用する必要があります。ビューを更新する最も簡単な方法は、操作中のテーブルを閉じてから、もう一度開くことです。

インストールのトラブルシューティングとアクティブ化

ACL for Windows をインストールできない場合、またはインストールした後にソフトウェアをアクティブ化できない場合は、このセクションのトラブルシューティング情報を使用して解決できる可能性があります。

メモ

サポートに問い合わせる前に、ここで情報を確認してください。

インストールのトラブルシューティング

さまざまな問題が ACL for Windows のインストールを妨げる可能性があります。一般的に、このような問題は次の3つのカテゴリのいずれかに該当します。

- **管理者の権利** - コンピューターの管理者の権利がなく、ソフトウェアをインストールできない。
- **IT の承認** - ソフトウェアが IT 部門によって承認されるまでは、会社によって、新しいソフトウェアまたは既存のソフトウェアの新しいバージョンのインストールが禁止されている。
- **技術要件** - コンピューターが ACL for Windows をインストールするために必要なソフトウェアまたはハードウェア要件を満たしていない。

詳細については、「ACL for Windows のシステム要件」ページ 2707を参照してください。

上記の状況のいずれにおいても、通常は、IT 部門に問い合わせて問題を解決する必要があります。

アクティブ化のトラブルシューティング

さまざまな問題が ACL for Windows のアクティブ化を妨げる可能性があります。アクティブ化しないとソフトウェアを使用できません。一般的に、このような問題は次の2つのカテゴリのいずれかに該当します。

- **接続またはサインイン** - Launchpad への接続またはサインインができません。
ACL for Windows をアクティブ化するには、Launchpad(www.highbond.com) にサインインする必要があります。これは、Diligent 製品とサービスへのアクセスを管理するための Diligent のクラウドベースのポータルです。
- **ライセンス** - Launchpad にサインインできても、ACL for Windows ライセンスを取得できません。

メモ

ACL for Windows は、アクティブ化の問題をトラブルシューティングする際の助けとなるアクティベーション ログを作成します。

C:\Users\ユーザーアカウント名\AppData\Local\ACL\activation.log

詳細については、「ACL for Windows のアクティブ化 ログ」見開きページを参照してください。

以下の表にはアクティブ化に関連するトラブルシューティング情報があります。

問題	考えられる解決策
<p>ACL for Windows] サインイン ダイアログ ボックスにサインイン フィールドがなく、エラー メッセージが表示されます。</p> <p>または</p> <p>ACL for Windows] サインイン ダイアログ ボックスでサインイン認証情報を入力すると、エラー メッセージは表示されませんがソフトウェアをアクティブ化できないか、または「アクティブ化エラーが発生しました」というメッセージが表示されます。</p>	<ul style="list-style-type: none"> コンピューターがインターネットに接続されていることを確認してください。 お使いのコンピューターがファイアウォール内の環境にある、またはプロキシ サーバーを経由してインターネットに接続している場合、IT 部門は、インターネットにアクセスするために、Analytics (ACLWin.exe) および Internet Explorer 実行可能ファイルを許可されたアプリケーション一覧に追加する必要がある場合があります。 <p>Analytics は、ポート 443 を使用して、Launchpad (*highbond.com) にアクセスする必要があります。</p> <p>Analytics は、サイレント認証を提供するプロキシ サーバーとファイアウォールとのみ互換性があります。プロキシ サーバーまたはファイアウォールが送信接続を認証するように要求する場合は、Analytics の有効化が失敗する可能性があります。</p> <ul style="list-style-type: none"> プロキシ サーバーがセッション接続を許可する場合は、ブラウザを開き、www.highbond.com に接続してから、アクティブ化処理を開始してください。 <p>メモ</p> <p>詳細については、「プロキシ サーバーを介した HighBond への接続」ページ 2705を参照してください。</p>
<p>ACL for Windows] サインイン ダイアログ ボックスの内容が正しく表示されない(たとえば、内容が多すぎてダイアログ ボックスに収まらない)か、ポップアップ ウィンドウにスクリーン エラーが表示されます。</p>	<p>Internet Explorer のセキュリティ設定のレベルを高くすると、この問題が発生する可能性があります。</p> <p>この問題を解決するには、https://*.highbond.com を、Internet Explorer 設定の信頼できるサイトのリストに追加します。あるいは、Internet Explorer のデフォルトのセキュリティ設定を復元できます。</p>
<p>ACL for Windows] サインイン ダイアログ ボックスにサインイン認証情報を入力すると、次のメッセージのいずれかが表示されます。</p> <ul style="list-style-type: none"> "無効な認証情報" "組織に所属していません" "サブスクリプションの有効期限が切れました" 	<ol style="list-style-type: none"> コンピューターでサポートされているブラウザ(IE 9 以上、Chrome、Firefox、または Safari)を開き、Launchpad(www.highbond.com)に移動します。 Launchpad 認証情報を使用してサインインします。 <p>サインインが失敗したら、次の理由のいずれかが考えられます。</p> <ul style="list-style-type: none"> 認証情報が無効である。 <p>正しい電子メールアドレスを使用していることを確認してください。gmail.com や hotmail.com などのドメインではなく、社内の電子メールアドレス(たとえば、my_company.com)を使用してください。パスワードを忘れた場合は、[パスワードの再設定] オプションを使用します。 Launchpad アカウントがありません。 <p>社内の Analytics アカウント管理者に連絡し、ユーザー アカウントを作成してもらるか、サポートまでお問い合わせください。</p> <ol style="list-style-type: none"> Launchpad にサインインできる場合は、ウィンドウの右上隅にある オプション > 組 </p>

問題	考えられる解決策
	<p>織]をクリックします。</p> <ul style="list-style-type: none"> 注文したサブスクリプションの種類]の下のエントリを確認し、ACL for Windows サブスクリプションが購入済みであることを確認します。 <p>この一覧には、ACL for Windows が、購入済みのライセンス数、および現在割り当てられている数と共に表示されます。</p> <p>例: 5/8</p> <p>すべてのライセンスが現在割り当てられている場合は、社内の Analytics アカウント管理者は、Launchpad にサインインして、ライセンスの割当先担当者は誰か、および使用できるようライセンスをリリースできるかどうかを確認できるはずで</p> <p>ACL for Windows サブスクリプションのエントリが表示されない場合は、社内で ACL for Windows が購入されていないか、注文がまだ処理されていないこととなります。サポートについては、社内の Analytics アカウント管理者、あるいは Diligent の御社アカウント担当者またはサポートまでお問い合わせください。</p> <p>d. 問題が解決しない場合は、コンピューターのレジストリの次の場所のいずれかにあるエントリの画面をキャプチャして、サポートまでお問い合わせください。</p> <ul style="list-style-type: none"> 32ビット Windows を使用している場合 : HKEY_LOCAL_MACHINE\SOFTWARE\ACL Software 64ビット Windows を使用している場合 : HKEY_LOCAL_MACHINE\SOFTWARE\WoW6432Node\ACL Software

ACL for Windows のアクティブ化 ログ

Analytics を使用できない原因となっているアクティブ化の問題をお客様や IT 部門がトラブルシューティングするために、ACL for Windows のアクティブ化ログを使用することができます。

場所

アクティブ化ログは、お使いのコンピューターの **AppData** フォルダーにあります。

`C:\Users\ユーザーアカウント名\AppData\Local\ACL\activation.log`

アクティブ化ログをテキスト エディターで開いて末尾までスクロールすることで、最新の情報を確認することができます。

アクティブ化ログの設定

アクティブ化ログは、デフォルトでは最小限の情報量のみを記録するように設定されています。効率的にトラブルシューティングを行うには、それよりも多い情報量を記録するようにアクティブ化ログを設定する必要があります。

アクティブ化ログの設定	ログを設定する場所	詳細
0	Windows レジストリ	すべてのログ記録をオフにする 設定は手動で変更するまで保持される
1	Windows レジストリ	最小限のログ記録(デフォルト) 設定は手動で変更するまで保持される
<pre>start aclwin.exe /debugactivation</pre>	Windows コマンドプロンプト	より詳細なログ記録 設定は現在の Analytics セッション中のみ保持される
2	Windows レジストリ	完全なログ記録(含む: Analytics の全ネットワークトラフィック) 設定は手動で変更するまで保持される

Windows レジストリを使ってアクティブ化ログの設定を変更する

手動で変更するまで保持されるアクティブ化ログの設定を指定するには、Windows レジストリエディターを使用します。

メモ

この変更を行うには、コンピューターの管理者権限が必要です。

1. レジストリエディターを開きます。

レジストリエディターを開くには、いくつかの方法があります。次の方法はお使いの Windows のバージョンで使用できます。

Windows エクスプローラーを使って、**C:\windows\system32** サブフォルダーにある **regedt32.exe** を右クリックし、**管理者として実行**]を選択します。

2. 次に移動します。**Computer\HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\ACL Software**
3. **LogSubscription** をダブルクリックします。
4. **Value data(値のデータ)** フィールドにアクティブ化ログの設定(0、1、または2)を入力し、**OK**]をクリックします。
5. レジストリエディターを終了します。

次回 Analytics を使用すると、アクティブ化ログによって、指定した設定が使用されます。

Windows コマンド プロンプト を使ってアクティブ化 ログの設定を変更する

Analytics セッションの間により詳細なログを記録するように指定するには、Windows コマンド プロンプトを使用します。

メモ

この変更を行うには、コンピューターの管理者権限が必要です。

1. 管理者として Windows コマンド プロンプトを開きます。

Windows のバージョンによりますが、管理者としてコマンド プロンプトを開くにはさまざまな方法があります。

この方法は Windows のあらゆるバージョンで使用できます。

Windows Explorer を使って、**C:\windows\system32** サブフォルダーにある **cmd.exe** を右クリックし、**管理者として実行**を選択します。

2. 次のコマンド構文を入力し、Enter キーを押します。

```
start aclwin.exe /debugactivation
```

Analytics が開始され、セッション中にアクティブ化 ログによって、指定した設定が使用されます。

プロキシ サーバーを介した HighBond への接続

Analytics と Analytics Exchange をアクティブ化するには、HighBond への HTTPS 接続と、データのインポートおよびエクスポートが必要です。プロキシ サーバーを介してインターネットに接続するには、HighBond の Web サイトが許可リストに登録されるようにプロキシ サーバーを構成する必要があります。

HighBond Web サイト 接続の許可

Analytics と Analytics Exchange では、HighBond ドメイン上のいくつかのサイトが使用されます。プロキシ サーバーを介してこれらの接続を許可する最も簡単な方法は、ドメイン名全体、*.highbond.com を許可リストに登録することです。

ポート 443 での接続を許可する必要があります。

ACL for Windows の構成

ACL for Windows は 2 つの目的で HighBond に接続します。

- **アクティブ化** -では、Internet Explorer の設定を使って接続する
- **データの共有 (アップロードとダウンロード)** -では、統合 Windows 認証を使用しないプロキシ認証など、プロキシ設定が透過でない場合、追加の構成が必要になります。

Analytics Exchange の構成

サーバーのプロキシ設定は通常厳重であり、アナリティクス スクリプトを実行する ACL Analytics Exchange Service のアカウントは権限が制限されているため、AX Server からプロキシへのアクセスは困難な場合があります。プロキシ サーバーを介した HighBond への接続を可能にするには、以下のいずれかのアプローチを使用します。

- **IP/ホスト ルール**-。これは、AX Server の IP アドレスから発信された要求を *.highbond.com で受信できるようにし、ACL Analytics Exchange Service アカウントの権限を変更しないというものです
- **サービス アカウントの権限の変更** -
 1. プロキシ サーバーへのアクセス権限を ACL Analytics Exchange Service アカウントに与えます。
 2. ACL Analytics Exchange Service アカウントに対し、**[インターネット オプション]**のプロキシ設定を構成します。
 3. "ACL for Windows の構成" 上で概説されている残りの構成を使用します。

追加リソース

defaultProxy エLEMENTの詳細については、Microsoft Developer Network の[system.Net Element](#)の記事を参照してください。

ACL for Windows のシステム要件

ACL for Windows のインストールに進む前に、インストール先のコンピューターが以下に概要を示した要件を満たしていることを必ず確認してください。

ソフトウェア要件

ユーザーによって確認またはインストールされる必要がある要件

メモ

ACL for Windows は、一覧に記載されているバージョンのオペレーティングシステムとサードパーティ製ソフトウェアで動作することがテスト、検証されています。ACL for Windows は一覧に記載されていないバージョンでも動作する場合がありますが、保証は行われません。

一覧に記載されていないバージョンで ACL for Windows を使用することは「サポートされていないインストール」と見なされます。問題が発生した場合に、サポート チームが解決できない可能性があります。

要件	追加情報
Windows オペレーティングシステム 次のうち、いずれかのオペレーティングシステム: <ul style="list-style-type: none">Microsoft Windows 11(64ビット)Microsoft Windows 10(64ビット)Microsoft Windows 8.1(64ビット)	ACL for Windows は Windows の 64 ビット バージョンで実行される 32 ビット アプリケーションです。 メモ Windows XP および Windows 7 オペレーティングシステムは、ACL for Windows インストールでサポートされていません。

要件	追加情報
<p>Windows パッチ</p> <p>Microsoft Windows 8.1 を使用している場合</p> <p>Windows 8.1 更新 KB2919355</p>	<p>重要</p> <p>Windows 8.1 更新 KB2919355 は Microsoft .NET Framework 4.6.x で必要であり、ACL for Windows 16 で必要です。</p> <p>Windows 8.1 を使用し、.NET 4.6.x をインストールしてなく、更新 KB2919355 を実行していない場合は、ACL for Windows インストーラーが終了し、.NET 4.6.2 前提条件のインストール中にエラーメッセージが表示されます。</p> <p>ACL for Windows インストールを続行する前に、更新 KB2919355 をダウンロードしてインストールする必要があります。</p> <p>あるいは、ACL for Windows インストールを開始する前に更新 KB2919355 をインストールし、エラーメッセージを回避できます。</p> <p>注意</p> <p>インストール処理中に特定の時点でコンピューターを再起動するように指示された場合は、すぐに再起動してください。コンピューターの再起動のメッセージは無視しないでください。</p> <p>指示されたときにコンピューターを再起動しないと、.NET、他の前提条件、または ACL for Windows のインストールで問題が生じるおそれがあります。</p>
<p>R、64ビット(オプション)</p> <p>R プログラム言語と統合する Analytics 関数を使用するには、次のものをインストールし、構成する必要があります。</p> <p>以下のバージョンの R はテスト済みであり、Analytics で動作します。</p> <ul style="list-style-type: none"> ○ 4.0.3 ○ 3.4.4 ○ 3.3.2 ○ 3.3.1 ○ 3.2.5 ○ 3.2.3 <p>CRAN R または Microsoft R を使用できます。</p> <p>メモ</p> <p>その他のバージョンの R も同様に動作するはずですが、保証の限りではありません。</p>	<p>CRAN R パッケージの 1 つを使用している場合は、R バイナリフォルダーへのパスをご利用のコンピューターの PATH 環境変数に追加する必要があります。</p> <p>例: <code>C:\Program Files\R\R-<version>\bin\x64</code></p> <p>メモ</p> <p>R 関数または RCOMMAND を使用しない場合は、R をインストールする必要はありません。</p>

要件	追加情報
<p>メモ</p> <p>Analytics の2つのオプションの機能では、Python の別個の2つのインスタンスをインストールする必要があります:</p> <ul style="list-style-type: none"> • Analytics Python 関数 -では、顧客がインストールした Python のインスタンスが必要になります。 • Analytics 機械学習コマンド -では、Analytics のインストールの一環として、インストール可能な Python のインスタンスが必要となります。 <p>その他の詳細は以下に表示されます。</p>	
<p>Python、32ビット (オプション)</p> <p>Python プログラミング言語と統合できる Analytics 関数を使用するには、次の項目をインストールし、構成する必要があります。</p> <ul style="list-style-type: none"> ◦ Python ◦ PYTHONPATH 環境変数 ◦ ACLPYTHONDLL 環境変数 <p>以下のバージョンの Python はテスト済みであり、Analytics で動作します。</p> <ul style="list-style-type: none"> ◦ 3.4 ◦ 3.5 ◦ 3.6 <p>メモ</p> <p>バージョン 3.4.x 以上の Python は Analytics で動作します。ただし、リストにならないバージョンが動作するかは保証できません。</p>	<p>Python をインストールする場合は、お使いのコンピューターでも実行できるよう Python を設定する必要があります。詳細については、"Python の Analytics 連携用設定" ページ 2697を参照してください。</p> <p>メモ</p> <p>Python 言語と Analytics 関数を統合して使用しない場合は、この言語をインストールする必要はありません。</p>
<p>Python、32ビット (オプション)</p> <p>Analytics 機械学習コマンドを有効にするには、次の項目をインストールする必要があります。</p> <ul style="list-style-type: none"> ◦ Python エンジン(Analytics 版) (3.7.9) 	<p>ACL for Windows のインストール中に 機械学習を有効にする(Analytics 版 Python エンジンをインストールする)] チェックボックスをオンにした場合は、Analytics 版 Python エンジンがインストールされます。</p> <p>メモ</p> <p>Python のこのインスタンスは、Analytics の Python 関数との併用、または一般的な Python の使用を目的としたものではありません。これらの用途には、Python の別のインスタンスをインストールする必要があります。</p>
<p>Oracle Instant Client (オプション)</p> <p>ACL Connector for Oracle を使用するには、次の項目をインストールする必要があります。</p> <ul style="list-style-type: none"> ◦ Oracle Instant Client 11g または 12c 	<ul style="list-style-type: none"> ◦ ACL Connector for Oracle を使用しない場合は、Oracle Instant Client をインストールする必要はありません。 ◦ Oracle Instant Client が何ビット版であるかと、お使いのオペレーティングシステムが何ビット版であるかが一致する必要があります。32 ビット版の Instant Client を 64 ビット マシンにインストールしても、接続が失敗するためです。

自動的にインストールされる前提条件

次のソフトウェア必須コンポーネントがコンピューターにインストールされていない場合、それらは ACL for Windows インストーラーによって自動的にインストールされます。

- Microsoft .NET Framework 4.6.2

メモ

お使いのコンピューターに既に .NET 4.6.0 または 4.6.1 がインストールされている場合、ACL for Windows ではそのインストールされている .NET のバージョンが使用されるため、4.6.2 バージョンはインストールされません。

- Microsoft Visual C++ 2015-2019 再配布可能パッケージ (x64 および x86)
- Microsoft Visual C++ 2013 再配布可能パッケージ (x64) (任意のデータコネクタがインストールされている場合のみ)
- Microsoft Visual C++ 2012 再配布可能パッケージ (x64) (任意のデータコネクタがインストールされている場合のみ)
- Microsoft Access データベースエンジン 2016 (32 ビット)、オプション

Analytics のインストール ディレクトリに格納される、必要なソフトウェア

以下のコンポーネントのローカルコピーは、Analytics とともにインストールされ、Analytics のインストール ディレクトリに格納されます。

- Chromium Embedded Framework (CEF) 106.0.290.0 - Analytics の埋め込み Chromium ブラウザ機能で使用されます。
- Python エンジン 3.7.9 および TPOT 0.10.2 - Analytics の機械学習コマンドで使用されます。

サポートされているオペレーティングシステムの一環としてインストールされる要件

次のコンポーネントも Analytics の必須コンポーネントですが、サポートされているオペレーティングシステムの一部としてインストールされています。

- Microsoft Data Access Components (MDAC) 2.8
- Microsoft Jet 4.0 Database Engine (MSJet)
- Microsoft XML Core Services (MSXML) 6.0
- Internet Explorer 9 以上 (画面に出力されるコマンド結果を XML ベースで書式設定するために必要)

Analytics データ コネクタ

Analytics インストールには多数のデータ コネクタが含まれます。ほとんどのコネクタは ODBC ドライバーで、Analytics インストール ディレクトリにインストールされます。

標準 データ コネクタ

ACL for Windows をインストールするときには、デフォルトで次のデータ コネクタがインストールされます。

カテゴリ	コネクタ
ビッグデータとNoSQL	<ul style="list-style-type: none"> ◦ Amazon Athena ◦ Apache Phoenix ◦ Azure Data Catalog ◦ Azure Data Lake Storage ◦ Azure Table ◦ CockroachDB ◦ Elasticsearch ◦ IBM Cloudant ◦ MarkLogic ◦ Parquet ◦ Presto ◦ Snowflake
ERP および CRM システム	<ul style="list-style-type: none"> ◦ Dynamics 365 Business Central ◦ Dynamics 365 Finance and Operations ◦ Dynamics 365 Sales ◦ Dynamics CRM ◦ Epicor ERP ◦ Exact Online ◦ NetSuite ◦ Odoo ◦ Oracle HCM Cloud ◦ SAP(機能させるには、追加のサブスクリプション資格が必要です) ◦ SAP ByDesign ◦ ServiceNow ◦ SugarCRM ◦ SuiteCRM ◦ Workday
財務会計ツール	<ul style="list-style-type: none"> ◦ ADP ◦ Dynamics GP ◦ Dynamics NAV ◦ QuickBooks ◦ QuickBooks Online ◦ QuickBooks POS ◦ Sage 50 UK ◦ Sage Cloud Accounting ◦ Sage Intacct ◦ SAP Concur
マーケティングおよびアナリティクス	<ul style="list-style-type: none"> ◦ Google Analytics ◦ LinkedIn ◦ Marketo ◦ Oracle Eloqua

カテゴリ	コネクター
	<ul style="list-style-type: none"> ◦ Oracle Sales Cloud ◦ Splunk ◦ SurveyMonkey ◦ Twitter
コラボレーション ソリューション	<ul style="list-style-type: none"> ◦ Active Directory ◦ Airtable ◦ AWS Data Management ◦ Azure Management ◦ Basecamp ◦ DocuSign ◦ 電子メール ◦ Excel ◦ Excel Online ◦ Exchange ◦ Google Contacts ◦ Google Sheets ◦ Jira ◦ Kintone ◦ Microsoft Teams ◦ SAP SuccessFactors ◦ SharePoint ◦ Slack ◦ Zendesk
ファイルおよび API 統合	<ul style="list-style-type: none"> ◦ Amazon S3 ◦ Box ◦ CSV ◦ DigitalOcean ◦ Dropbox ◦ Google Cloud Storage ◦ Google Drive ◦ JSON ◦ LDAP ◦ Microsoft OneDrive ◦ OData ◦ REST ◦ RSS/ATOM ◦ SFTP
eコマース ソリューション	<ul style="list-style-type: none"> ◦ Edgar Online ◦ Open Exchange Rates ◦ ShipStation ◦ Square ◦ Stripe ◦ UPS ◦ USPS
関係 データベース	<ul style="list-style-type: none"> ◦ Azure Analysis Services ◦ Microsoft Access ◦ MySQL ◦ SAP Hybris Cloud for Customer ◦ Sybase ◦ Sybase IQ ◦ xBase

カテゴリ	コネクタ
セキュリティ ツール	<ul style="list-style-type: none"> ○ Qualys ○ Tenable.sc
その他のコネクタ	<ul style="list-style-type: none"> ○ Rsam

任意のデータ コネクタ

次のデータ コネクタは任意です。ACL for Windows をインストールするときにインストールしないことを選択できます。

- Amazon Redshift
- Cassandra
- Concur
- Couchbase
- Drill
- DynamoDB
- Google BigQuery
- HBase
- Hive
- Impala
- MongoDB
- Oracle
- Salesforce
- Spark
- SQL Server
- Teradata

ACL Connector for Analytics と Tableau を使用する

ACL Connector for Analytics を使用して、Analytics プロジェクト から Tableau にデータを抽出できます。

Tableau と Analytics との統合を最適化するには、次の手順に従います。

1. ACL for Windows をインストールします。
2. **ACL Connector for Analytics.tdc** を

C:\Program Files (x86)\ACL Software\ACL for Windows 15\ACL ODBC

から

..\Documents\My Tableau Repository\Datasources

にコピーしますメモ

これらのファイルパスはデフォルト の場所です。Analytics のインストールと Tableau のインストールとの間で、異なる場所が使用されている可能性があります。

3. Tableau を再起動します。

ハードウェア要件

メモ

本番環境での最善の Analytics のパフォーマンスには、一覧の最低仕様よりも多くのリソースを必要とする場合があります。

コンポーネント	最小	推奨
プロセッサ	1.8 GHz	
メモリ (RAM)	2 GB	<ul style="list-style-type: none"> 64ビットのオペレーティングシステム: 8 GB 以上を推奨(特に大容量ファイルの並べ替えをする場合) 32ビットのオペレーティングシステム: 4 GB(特に大容量ファイルの並べ替えをする場合)
ハード ディスク領域 (Analytics アプリケーション ファイル)	1.1 GB	
ハード ディスク領域 (ソフトウェア必須コンポーネント)	8 GB	
ハード ディスク領域 (データ記憶領域)		100 GB 以上 Analytics アプリケーション ファイルと前提条件をインストールするために必要なハード ディスク領域の他に、コンピューターがデータの抽出、フラットファイル、および結果を保存するために使用される場合は、追加の大きい領域が必要です。

接続

TCP/IP の接続は次の目的で必要となります。

- インストール後、ACL for Windows をアクティブ化するために必要となります
- コンテキスト オンライン ヘルプにアクセスするために必要となります
- 進行中のソフトウェア サブスクリプションを定期的に確認するために必要となります
- 自動ソフトウェア更新のために定期的に必要となります

接続要件

メモ

御社がプロキシ サーバー、ファイアウォール、または他のネットワークセキュリティ対策を使用して、ACL for Windows がインターネットに接続できないようにしている場合は、"プロキシサーバーを介した HighBond への接続" ページ 2705 および "インストールのトラブルシューティングとアクティブ化" ページ 2700 を参照してください。

ACL for Windows は次の機能を実行するためにインターネット 接続が必要となります。

- 初めて使用する前にソフトウェアをアクティブ化する
- 継続的にソフトウェアのサブスクリプションを検証する
- HighBond を使用してデータを共有する
- Analytics ソフトウェアコンポーネント間でアプリケーションレベルの通信を提供する
- ソフトウェア更新の通知を送信する
- 状況依存のオンラインヘルプにアクセスする

ACL for Windows 内のさまざまな実行ファイルに必要な特定の接続は次のとおりです。

アプリケーション名 (実行ファイル)	必要な接続	接続の理由
Analytics (ACLWin.exe)	https://*.highbond.com、 ポート 443	<ul style="list-style-type: none"> ◦ Analytics の初期アクティブ化 ◦ 進行中のソフトウェアサブスクリプション確認 ◦ HighBond を使用したデータの共有
ソフトウェア更新の通知 (ACL-service.exe)	https://*.highbond.com、 ポート 443 localhost(動的ポート)	<ul style="list-style-type: none"> ◦ ソフトウェア更新の自動通知 ◦ Analytics ソフトウェアコンポーネント間でのアプリケーションレベルの通信
Internet Explorer (Iexplore.exe)	https://*.highbond.com、 ポート 443	<ul style="list-style-type: none"> ◦ Analytics の初期アクティブ化

重大なセキュリティ脆弱性 (CVE-2021-44832、-45105、-45046、-44228) のパッチ

メモ

以下で説明するセキュリティの脆弱性は、バージョン 15.1.0 まですべてのバージョンの Analytics に影響します。バージョン 16.0 の Analytics では脆弱性のあるコンポーネントが含まれなくなったため、この脆弱性はありません。

Analytics の 分析アプリ] ウィンドウは [Apache Log4j](#) を使用しています。結果として、次の一覧の Log4j 重大なセキュリティ脆弱性の影響を受けますが、Analytics にとっての実際の脆弱性は低レベルです。

- [CVE-2021-44832](#)
- [CVE-2021-45105](#)
- [CVE-2021-45046](#)
- [CVE-2021-44228](#)

この脆弱性の一覧は、以前のパッチが不完全であることが判明したときに大きくなりました。Analytics の上記の脆弱性をすべて軽減するパッチを用意しました。

オプション

Analytics ユーザーにとってのリスクは高くありませんが、次のオプションのいずれかを使用して脆弱性の状況を解決することをお勧めします。

- **Analytics 16.1 へのアップグレード** - 詳細については、"ACL for Windows インストールおよびアクティベーションガイド" ページ 2661を参照してください。
- **既存のバージョンの Analytics に手動でパッチを適用する** - 詳細と手順については、以下を参照してください。

パッチについて

このパッチは Log4j のプロバイダーである Apache からの正式リリースです。Analytics の [CVE-2021-44832](#)、[CVE-2021-45105](#)、[CVE-2021-45046](#)、[CVE-2021-44228](#) が軽減されます。

脆弱な Analytics のバージョン

リスクは低いものの、15.1.0 以下のすべてのバージョンの Analytics が影響を受けます。インストールしたバージョンに関係なく、最新バージョンの Analytics(16.1) にアップグレードするか、今すぐ既存のバージョンの Analytics に手動でパッチを適用することをお勧めします。

メモ

15.1.0 より前のバージョンの Analytics は他の脆弱性や不具合の影響を受けます。このような問題はこの手動パッチでは修正されません。15.1.0 より前のバージョンの Analytics を使用している場合は、Analytics 16.1 にアップグレードすることをお勧めします。これにより、以前に特定されたすべての脆弱性や不具合が解決されます。

パッチをインストールする方法

メモ

ACL for Windows が **Program Files** フォルダーにインストールされている場合は、このパッチを実行するためにコンピューターで管理者権限が必要です。パッチを自分で実行できない場合は、システム管理者にサポートを依頼してください。

このパッチは、Analytics がインストールされている各コンピューターにインストールする必要があります。手動でインストールするか、パッチスクリプトを使用してインストールできます。

複数のバージョンの Analytics をサイドバイサイドインストールしている場合は、すべてのインストール済みのバージョンに個別でパッチを適用する必要があります。

バッチスクリプト方式

Analytics がデフォルトの場所 (**C:\Program Files (x86)\ACL Software\ACL for Windows <<<version>>>**) にインストールされている場合にのみ、この方法を使用してください。Analytics が別の場所にインストールされている場合は、次に説明する [手動の方法](#) を使用してください。

1. 開いている場合は、Analytics と ACL for Windows を終了します。
2. [この zip 圧縮されたフォルダーをコンピューターにダウンロード](#) します。
3. **C:/temp/patch** などの簡単な場所でフォルダーを解凍します。
4. 管理者モードで Windows コマンド プロンプトを開きます。
5. 作成したフォルダーに移動します。たとえば、**C:/temp/patch** に解凍した場合は、次のように入力します。

```
cd c:\temp\patch
```

6. 解凍したフォルダーからバッチ スクリプトを実行します。

```
.\do_replace.bat
```

手動の方法

1. 開いている場合は、Analytics と ACL for Windows を終了します。
2. [この zip 圧縮されたフォルダーをコンピューターにダウンロード](#)します。
3. フォルダーを解凍します。
4. ACL for Windows がインストールされているフォルダーに移動します。例：`C:\Program Files (x86)\ACL Software\ACL for Windows 15`。
5. ACL アプリ フォルダーを開きます。
6. `acl-service.ini` をダウンロードした新しいバージョンのファイルで置換します。
7. `lib` フォルダーを開きます。
8. 次の一覧の 4 つの `*.jar` ファイルを、ダウンロードした新しいバージョンのファイルで置き換えます。新しいバージョンはファイル名に `2.17.1` が含まれています。
 - `log4j-1.2-api-2.8.2.jar`
 - `log4j-api-2.8.2.jar`
 - `log4j-core-2.8.2.jar`
 - `log4j-slf4j-impl-2.8.2.jar`

Analytics Exchange にもパッチを提供する必要がある場合があります

組織で Analytics Exchange を使用する場合は、システム管理者がパッチを実行するかアップグレードしてください。[Analytics Exchange にパッチを適用する手順](#)を参照してください。

このページは意図的に空白のままにしています

自動化と共有

自動化と共有

Analytics でデータ分析作業の価値を高めるには、幅広いワークフローの一部として他のユーザーが使用できるようにします。Analytics プロジェクトとそこに含まれる作業を共有および拡張するための複数のオプションがあります。

下記の機能や製品は、Analytics に統合される別々のソフトウェアです。

メモ

バージョン 16 以降の Analytics は Analytics Exchange(AX) と連携しません。2023 年 1 月 1 日をもって Analytics Exchange のサポートが終了します。[詳細を確認](#)するか、[ロボットにアップグレード](#)してください。

Analytics と Analytics Exchange の連携の詳細については、[Analytics および ACLScript 15.1 のヘルプ](#)を参照してください。

データをリザルトに公開する	リザルトは、追加の処理および問題改善のためのレコードのアップロード先となる HighBond アプリです。リザルトには、トリガー、アンケート、およびメトリクスなどのワークフローの自動化ツールがあります。データを視覚化することもできます。
"データをストーリーボードに公開する" ページ 2724	ストーリーボードは、単一のプレゼンテーションで複数の視覚化とリッチテキストコンテンツを表示する HighBond アプリおよびコミュニケーションプラットフォームです。ストーリーボードは、簡単に管理職や他の関係者と共有できます。
ロボットを使用した自動化	ロボットは、Analytics で作成されたアナリティクス スクリプトを使用した自動的な定期タスクに使用できる HighBond アプリです。ロボットは、構成に応じて定期のタスクを処理します。

ACL Robotics のインフォグラフィック



1



アナリティクススクリプトを作成し、これらをロボットにアップロードする

Analytics



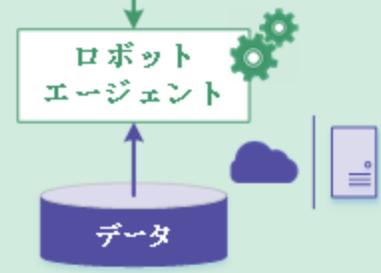
2



タスクをロボットでスケジュールし実行する

ロボットエージェントがデータを処理

ロボット



3

アナリティクスまたはリザルトでタスクの結果を表示

ストーリーボードで結果を提示

Analytics



リザルト

ストーリーボード

データをリザルトに公開する

リザルトは [HighBond](#) の一部です。リザルトは、改善およびワークフローを自動化するアプリであり、例外データの管理、およびアンケートによる人間の背景の追加を行い、監視をトリガーとメトリクスで継続します。

リザルトは、視覚化を構築して、活動を自動化するために使用できるため、Analytics や ACLScript での作業を公開する場所として最適です。

特徴と機能

- **他のユーザーとのコラボレーション** - チームを招待し、データアナリティクスで特定された問題を調査、改善、および追跡します。
- **自動化ワークフロー** - 会社の改善プロセスを自動化するトリガーを作成します。
- **アンケートの回答をデータアナリティクスに統合** - アンケートを送信し、追加の証拠を収集し、回答を分析します。
- **データの視覚化** - 視覚化と解釈を作成してデータに対する洞察を伝え、ストーリーボードで表示します。
- **監視の提供** - プロセスまたはプログラムに対する完全な監視のための予防統制を検出統制を組み合わせます。
- **データの保管** - 参照テーブルを使い、データを他のソースから HighBond に取り込み、このプラットフォームで参照します。

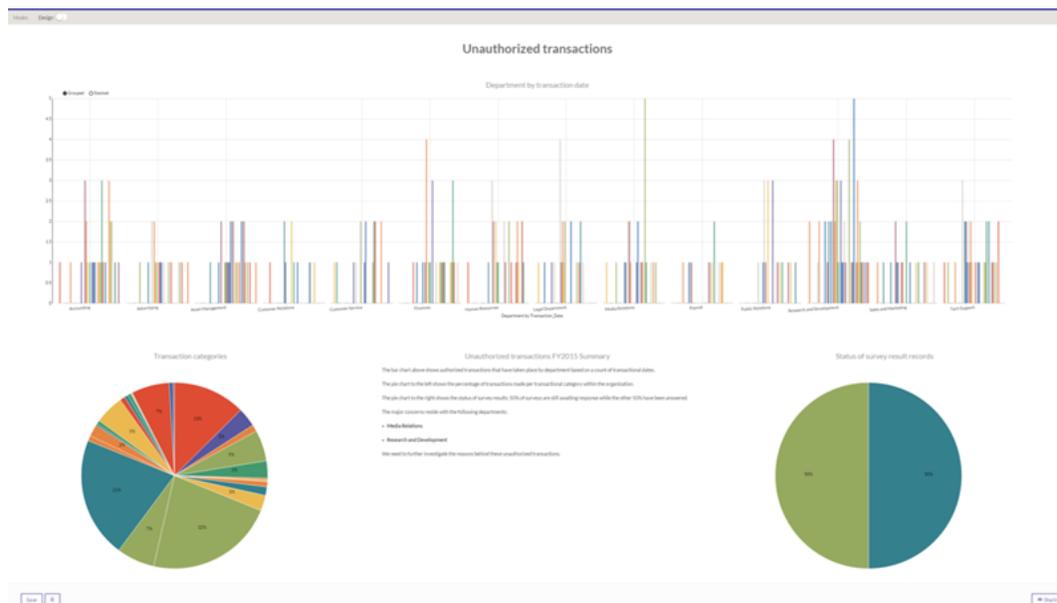
リザルトの基本操作

すべての Analytics ユーザーは HighBond にアクセスできます。また、拡張機能を使用すると、リザルトにもアクセスできます。

- [リザルトへの例外のエクスポート](#)
- [EXPORT を使用してリザルトへのエクスポートをスクリプト化する](#)
- [HighBond へのログイン](#)
- [リザルトの使用に関する一般的なヘルプ](#)

データをストーリーボードに公開する

ストーリーボードは [HighBond](#) のアプリです。ストーリーボードは、単一のプレゼンテーションで複数の視覚化とリッチテキストコンテンツを表示するコミュニケーションプラットフォームです。ストーリーボードは、簡単に管理職や他の関係者と共有できます。



ストーリーボードの仕組み

ストーリーボードは、行と列に視覚化とリッチテキストコンテンツを表示します。[リザルトでデータを入力](#)すると、そのデータを使用して、視覚化とメトリクスを作成できます。次に、ストーリーボードでこれらの視覚化やメトリクスを追加します。

- [HighBond へのログイン](#)
- [リザルトの使用に関する一般的なヘルプ](#)
- [ストーリーボードの使用に関する一般的なヘルプ](#)

ロボットを使用した自動化

ロボットは、Analytics で構築されたスクリプトを使用して、定期タスクの自動化に使用する [HighBond](#) アプリです。スクリプトを作成したら、これをロボットにアップロードし、必要なタスクの自動化を構成できます。ロボットは、構成に応じて定期のタスクを処理します。

ロボット アプリの詳細については、HighBond ヘルプの[ロボットを使用した作業の自動化](#)を参照してください。HighBond へのログインについては、[アカウントへのアクセス](#)を参照してください。

ロボットで自動化する方法

ロボットを使って定期タスクを自動化するには、まず、1 つ以上の **アナリティクス スクリプト** を含む Analytics でプロジェクトを作成する必要があります。アナリティクス スクリプトは、それを実行するために、アナリティクス ヘッダーを使っていくつかの命令とプロパティを宣言した、標準の Analytics スクリプトです。

アナリティクス スクリプトとアナリティクス ヘッダーの詳細については、「[アナリティクス スクリプトの概要](#)」ページ 2579を参照してください。

スクリプトのコミットとロボットの作成

アナリティクススクリプトを作成した後、ロボットアプリにアップロードします。スクリプトを Analytics からロボットにアップロードするアクションは、**スクリプトのコミット**と呼ばれます。初めてスクリプトをコミットすると、**ロボット**がロボット アプリで作成されます。ロボットは、コミットされたアナリティクス スクリプト、補助スクリプト、関連するファイルを格納するコンテナです。ロボットは、スケジュールされた、自動化されたタスクを実行するように構成するオブジェクトです。

開発モードと本番モード

スクリプトは、ロボットでのみ開発モードにコミットされ、本番モードにコミットされることはありません。この設計により、本番スクリプトは開発モードとは完全に分離されて保護されます。

スクリプトの最終バージョンを開発モードにコミットしたら、そのバージョンは本番環境で明示的にアクティブ化する必要があります。

代替アプローチ

ロボットアプリでは、空のロボットを作成してから、Analytics のスクリプトを空のロボットにコミットすることができます。ロボットアプリで手動でロボットを作成するか、初めてスクリプトをコミットするときに自動的にロボットを作成するかどうかに関係なく、結果は同じです。

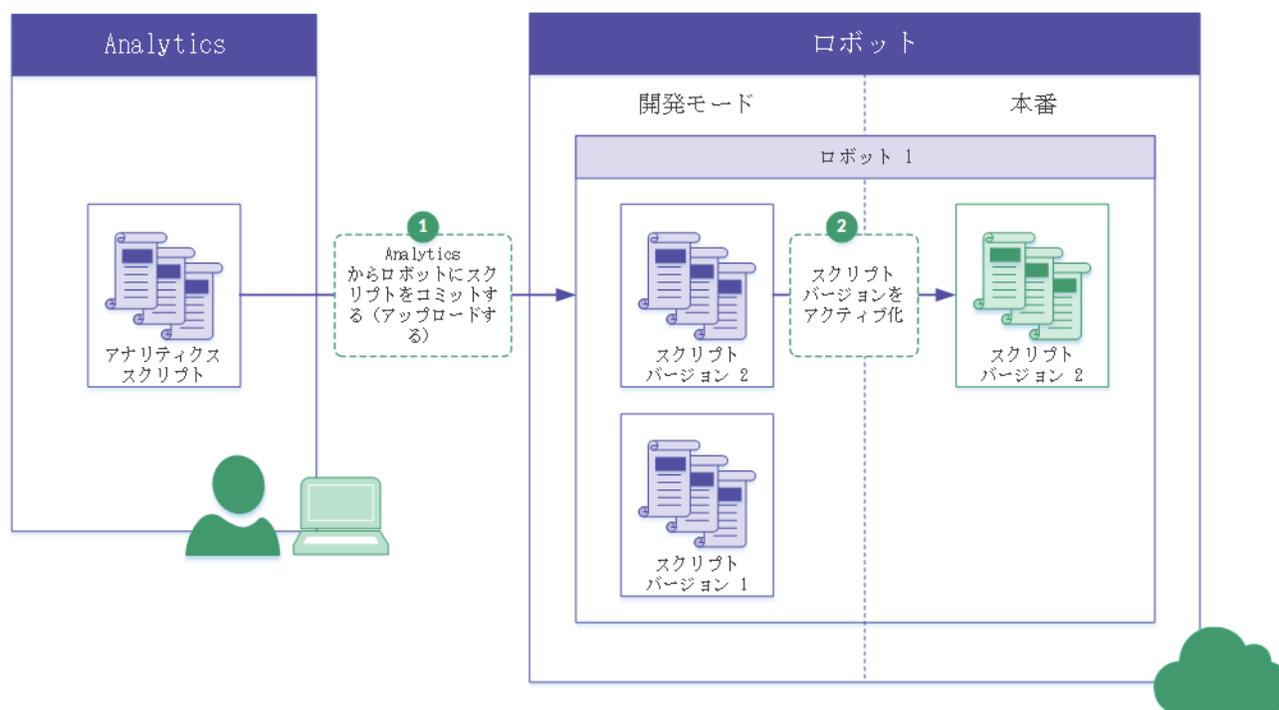
Analytics およびロボットにおけるスクリプトの開発ワークフロー

アナリティクス スクリプト (ACL スクリプト) のオーサリングが Analytics アプリで実施されます。Analytics でアナリティクス スクリプトを作成し、必要に応じて Analytics で更新します。

Analytics からロボット アプリにアナリティクス スクリプトをアップロードする準備ができたなら、特定の ACL ロボットにスクリプトをコミットします。詳細については、"Analytics からロボットへのスクリプトのコミット (アップロード)" ページ 2731を参照してください。

スクリプトをコミットする際に、新バージョンのスクリプトが ACL ロボットに追加され、そのバージョンは開発モードでアクセスできます。新しいバージョンには、コミットするスクリプトの正確な内容が含まれます。バージョンは自己完結型で、前のバージョンのスクリプトとマージされません。Analytics でスクリプトを削除した場合、ロボットでは新しいバージョンのスクリプトは使用できません。

開発モードでスクリプトバージョンが正常に動作していることを確認できたら、本番モードで使用するためにアクティブ化できます。



反復ワークフロー

1つ以上のスクリプトを編集し、ロボットに再コミットできます。スクリプトを再びコミットするたびに、新バージョンのスクリプトを作成します。これらの方法のいずれかを使用して、スクリプトの編集および再コミットができます。

- ロボットに関連付けられた Analytics プロジェクトで既存のスクリプトを編集し、もう一度スクリプトをコミットします。
- ロボットから Analytics にスクリプトをダウンロードし、1 つ以上のスクリプトを編集し、スクリプトをコミットします。

スクリプト バージョンの仕組み

- **バージョンは連続して採番される**

1 つ以上のスクリプトを同じロボットにコミットするたびに、新しい、連番のバージョンのスクリプト (バージョン 1、バージョン 2 など) が作成されます。

連続したスクリプトのバージョンを保存することで、ロボット アプリは、スクリプト作業が一切失われないことを保証し、必要に応じて、古いバージョンに簡単にアクセスすることができます。

- **バージョンは自己完結している**

各コミットされたバージョンは、完全に自己完結しています。前のバージョンのスクリプトは上書きされませんが、スクリプトがバージョン間でマージされることはありません。

プロジェクトからスクリプトを削除すると、コミットする後続のすべてのバージョンにはスクリプトが含まれません。

- **バージョンにはすべてのスクリプトが含まれる**

バージョンは、スクリプトをロボットにコミットするときに、プロジェクトにあるすべてのスクリプト (すべてのアナリティクス スクリプト、およびすべての補助スクリプト) を含みます。プロジェクトからスクリプトを選択してコミットすることはできません。

- **バージョン変更が記録されます**

開発モードの [スクリプト バージョン] タブでは、スクリプト バージョンを選択し、そのバージョンの個別のスクリプトの名前とカテゴリを表示できます。新規追加、削除、または修正されたスクリプトの名前はハイライト表示されます。

スクリプト バージョンの例

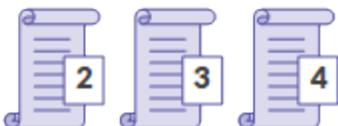
シナリオ

現在、部署で隔週の給与ファイルに対して手動で実行する分析テストのセットを自動化したいと考えています。テストは、従業員が正しく支払われていること、適切な給与控除が行われていること、存在しない従業員に支払われていないことなどを確認します。

スクリプト 開発 ワークフロー

- Analytics で、給与ファイルをインポートし、一部の事前データ準備タスクを実行してから、すべての分析テストを実行するスクリプトを開発します。

- スクリプトをロボットにコミットします。ロボットでは、開発モードで、実際のデータのコピーに対してスクリプトを実行し、正常に動作することを確認します。
- スクリプトが正常に動作したら、本番モードで使用するためにアクティブ化し、2週間ごとに自動実行されるようにスケジュールします。

Analytics	アクション/結果	ロボット
 <p>スクリプトのコミット >></p>	<ul style="list-style-type: none"> ◦ コミット - スクリプト 1 をコミットします。これは、隔週の給与データをインポート、準備、分析します。 ◦ テスト - ロボット開発モードでスクリプト 1 をテストします。正常に動作しているように思われます。 ◦ 問題 - スクリプト 1 に含まれている機能が多すぎるため、他の人が簡単に理解したり編集したりすることが難しいかもしれないということに気がきました。 	 <p>>> バージョン 1</p>
 <p>スクリプトのコミット >></p>	<ul style="list-style-type: none"> ◦ 編集 - スクリプト 1 を3つの個別のスクリプト(2、3、4)に分割します。それぞれ、インポート、準備、分析フェーズです。スクリプト 1 を削除します。 ◦ コミット - プロジェクトでスクリプトをコミットします。 ◦ テスト - 3つのスクリプトをテストします。出力に結果の一部が含まれていないことに気がきました。 ◦ 問題 - スクリプト 3 によって実行されるデータクレンジングが十分に包括的でないため、一部の結果が出力されません。 	 <p>>> バージョン 2</p>
 <p>スクリプトのコミット >></p>	<ul style="list-style-type: none"> ◦ 編集 - スクリプト 3 で、Analytics 機能を使用して、追加のデータクレンジングを実行する演算フィールドを作成します。スクリプト 2 および 4 は変更しません。 ◦ コミット - プロジェクトでスクリプトをコミットします。 ◦ テスト - 3つのスクリプトをテストします。出力にすべての結果が含まれています。 	 <p>>> バージョン 3</p>

Analytics	アクション/結果	ロボット
	<ul style="list-style-type: none"> 成功 - すべての3つのスクリプトとプロセス全体が正常に動作しています。本番モードで使用するために、スクリプトのバージョン3をアクティブ化します。 	
<div style="display: flex; justify-content: space-around; align-items: center;"> <div data-bbox="250 512 500 579">  編集されたスクリプト </div> <div data-bbox="537 512 850 579">  変更されていないスクリプト </div> </div>		

スクリプトを編集、コミットする際のベスト プラクティス

ロボットのスクリプトをマスターバージョンとして取り扱う

スクリプトはいったんロボットにコミットされると、変更されないよう保護されます。ベスト プラクティスとして、ロボットのスクリプトをマスターバージョンとして取り扱う必要があります。スクリプトを編集する場合は、ローカルで保存されたコピーを使用するのではなく、ロボットからスクリプトをまずダウンロードする必要があります。

ロボットの最新バージョンのスクリプトのダウンロードに制限されていません。作業する任意のバージョンのスクリプトをダウンロードできます。

ローカルコピーを使用して作業を始めるリスク

スクリプトのローカルコピーを使って作業の編集を始めるリスクは、そのコピーがロボットのバージョンと一致しない場合があることです。

- ローカルコピーを不注意に変更してしまった、またはそれを変更したことを忘れてしまった
- 自分がスクリプトを最後にコミットした後に、誰かがスクリプトのバージョンをロボットにコミットした

ダウンロードをスキップしても安全なのはいつか?

スクリプトバージョンをダウンロードした後の、単一のスクリプト作成セッション中に、反復間でダウンロードせずに、Analytics プロジェクトから直接スクリプトの反復バージョンをコミットすることが一般的に安全です。

他の誰かが自分と同時にスクリプトで作業をしている可能性がある場合は、反復間でダウンロードします。一度に一人のみがスクリプトバージョンで作業することが推奨されます。

スケジュール済みタスクの一部である編集されたスクリプトをテストする

ロボット アプリのスケジュール済みタスクに含まれている 1 つまたは複数の編集済みスクリプトを必ずテストすることで、編集でタスクが破壊されていないことを確認する必要があります。

たとえば、スクリプトのアナリティクス ヘッダーに次の編集を行うと、関連タスクが破壊される場合があります、タスクを再び作成する必要があります。

- 入力パラメーターの追加または削除
- 入力ファイル、テーブル、またはフィールドの追加または削除
- ハードコーディングされた入力ファイル、テーブル、またはフィールド名の変更

構文検証

スクリプトをロボットにコミットするたびに、Analytics は自動的に 2 つのタイプの構文検証または確認を実行します。

- アナリティクス ヘッダー検証
- スクリプト 構文確認

スクリプトが検証または確認を渡さない場合は、スクリプトをコミットできず、問題の場所を示すメッセージが表示されます。

アナリティクス ヘッダー検証

アナリティクス スクリプトのアナリティクス ヘッダーは特定の要件に適合する必要があります。適合しない場合は、実行時にアナリティクス スクリプトが失敗します。

アナリティクス ヘッダー検証は無効にできません。

詳細については、"アナリティクス ヘッダーを使った作業" ページ 2588を参照してください。

スクリプト 構文確認

ランタイム ユーザー対話コマンドや絶対ファイルパスなどのアナリティクス スクリプトの特定の要素はサポートされないか、推奨されません。

スクリプト 構文の確認はデフォルトで有効です。オフにする場合は、**[オプション]**ダイアログボックス(**ツール > オプション > インターフェイス**)で、**[スクリプトをコミットする前に、スクリプト 構文チェックを無効にする]**を選択します。

詳細については、"アナリティクス開発のベスト プラクティス" ページ 2596を参照してください。

Analytics からロボット へのスクリプトのコミット (アップロード)

Analytics のスクリプトをロボット アプリの ACL ロボットにコミットまたはアップロードするときには、複数の方法があります。

さまざまな可能性を理解するための最も簡単な方法は、Analytics プロジェクトと ACL ロボットを、関連付けることができる、スクリプトを格納するための 2 つのコンテナであると考えことです。関連付けると、プロジェクトの連続したバージョンのスクリプトをロボットにコミットできます。

機能の仕組み

スクリプトを ACL ロボット アプリにコミットするときには、Analytics プロジェクトのすべてのスクリプトがコミットされ、関連付けられたロボットで「バージョン」になります。プロジェクトからスクリプトを選択してコミットすることはできません。

アナリティクス スクリプトを新しく作成された ACL ロボットに対し始めてインポートする場合、ロボット アプリを使用できます。その後、Analytics を使用し、スクリプトをコミットするか、アップロードする必要があります。

スクリプトをロボットにコミットするには、ロボット アプリの該当するルールに割り当てる必要があります。詳細については、[ロボット アプリのアクセス許可](#)を参照してください。

アクション	コミットの結果
"スクリプトを新しい ACL ロボットにコミットする" 下	コミットされたスクリプトのバージョン 1 を含む、新しい ACL ロボット が作成されます。
"スクリプトを既存の ACL ロボットにコミットする" 見開きページ	既存の ACL ロボットにはコミットされたスクリプトが含まれます。バージョン番号は、ロボットに既にスクリプトが含まれているかどうかによって異なります。
"編集されたスクリプトを ACL ロボットにコミットする" ページ 2733	新しいバージョンのスクリプトは、プロジェクトに関連付けられた ACL ロボットにコミットされます。
"スクリプトを別の ACL ロボットにコミットする" ページ 2735	スクリプトは、新しく作成された ACL ロボットまたは既存の ACL ロボットにコミットされます。 プロジェクトと前のロボットの間に関連付けは削除されます。

スクリプトを新しい ACL ロボットにコミットする

初めてスクリプトをロボット アプリにコミットし、コミットされたスクリプトを含む新しい ACL ロボットを作成します。

1. お使いのコンピューターから、Analytics デスクトップ アプリケーションを開きます。
2. Analytics メイン メニューから、**[ファイル > スクリプトのコミット]**を選択します。
エラーメッセージが表示される場合は、プロジェクトの1つ以上のスクリプトで、アナリティクス ヘッダーまたはスクリプト構文に問題がある場合があります。
詳細については、[Analytics およびロボットにおけるスクリプトの開発 ワークフロー](#)を参照してください。
3. 必要に応じて、**[送信先の選択]**ダイアログボックスで、該当する HighBond インスタンスをダブルクリックします。
[ロボット コレクション]と既存のロボットの一覧が表示されます。
4. **[新しいロボット]**フィールドにロボット名を入力し、**[作成]**をクリックします。
ロボットが作成され、ロボット ID が自動的に生成されます。
スクリプトを含む Analytics プロジェクトと既存ロボットが関連付けられ、次回以降のコミットでロボットを手動で検索する必要がなくなります。

メモ

ロボット名には "\$"、"€" の文字を使用しないでください。

5. コミットされたスクリプトを説明する簡潔なコミット メッセージを入力し、**[OK]**をクリックします。
バージョン 1 のスクリプトが新しく作成されたロボットにコミットされます。このスクリプトは、この時点では開発モードでのみ存在します。
6. 省略可能。 **[スクリプトのコミットが成功しました]**ダイアログボックスで、新しく作成されたロボット、あるいはコミットされたスクリプトを検査するためのリンクのいずれかをクリックします。
7. **[OK]**をクリックして、ダイアログボックスを閉じます。

スクリプトを既存の ACL ロボットにコミットする

ACL ロボットにスクリプトをコミットし、ロボットを入力します。既存のロボットはスクリプトが含まれていても、空であってもかまいません。

メモ

プロジェクトがまだロボットに関連付けられていない場合に、この方法を使用します。プロジェクトがロボットに関連付けられている場合は、"スクリプトを別の ACL ロボットにコミットする" ページ 2735を参照してください。

1. お使いのコンピューターから、Analytics デスクトップ アプリケーションを開きます。
2. Analytics メイン メニューから、**[ファイル > スクリプトのコミット]**を選択します。
エラーメッセージが表示される場合は、プロジェクトの1つ以上のスクリプトで、アナリティクス ヘッダーまたはスクリプト構文に問題がある場合があります。

詳細については、[Analytics およびロボットにおけるスクリプトの開発 ワークフロー](#)を参照してください。

- 必要に応じて、**送信先の選択**]ダイアログボックスで、該当する HighBond インスタンスをダブルクリックします。
ロボット コレクション]と既存のロボットの 一覧が表示されます。
- ロボットのリストで、スクリプトをコミットするロボットを選択し、**OK**]をクリックします。
スクリプトを含む Analytics プロジェクトと既存ロボットが関連付けられ、次回以降のコミットでロボットを手動で検索する必要がなくなります。
- コミットされたスクリプトを説明する簡潔なコミット メッセージを入力し、**OK**]をクリックします。
スクリプトは、既存のロボットにコミットされます。スクリプトのバージョン番号は、ロボットに既にスクリプトがあるか、以前に空であったかどうかによって異なります。
このスクリプトは、この時点では開発モードでのみ存在します。
- 省略可能。**スクリプトのコミットが成功しました**]ダイアログボックスで、既存のロボット、あるいはコミットされたスクリプトを検査するためのリンクのいずれかをクリックします。
- OK**]をクリックして、ダイアログボックスを閉じます。

編集されたスクリプトを ACL ロボットにコミットする

編集されたスクリプトをコミットし、関連付けられた ACL ロボットで新しいバージョンのスクリプトを作成します。

コミットする前にスクリプトを編集するには、2 つの方法があります。関連付けられたプロジェクトでスクリプトを編集するか、ロボットから新しい Analytics プロジェクトにスクリプトをダウンロードして、新しいプロジェクトでスクリプトを編集できます。

メモ

編集作業を開始する前にロボットからスクリプトをダウンロードすることをお勧めします。ローカル保存されたスクリプトとは異なり、ダウンロードしたスクリプトが変更されないようにすることができます。

単一のスクリプト作成セッション中に、自分だけがスクリプトで作業している場合は、一般的に、反復間でダウンロードせずに、プロジェクトから直接、反復するスクリプトのバージョンを安全にコミットできます。

ロボットからスクリプトをダウンロードする

既にローカルコンピューターにあるスクリプトではなく、ロボットに含まれるスクリプトで作業する場合は、この手順の一部を実行します。

- [ロボット アプリを開きます](#)。
- ロボットのダッシュボードから、**ACL ロボット**]を選択します。

3. ダウンロードしたいスクリプトを含むロボットをクリックします。
4. ロボットの右上隅にある **開発モード**]トグルを使って、使用する環境を選択します。
5. **スクリプト バージョン**]タブで、編集するスクリプトのバージョンを選択します。
6. **バージョンの詳細**]パネルで、**スクリプトをダウンロード**]をクリックします。

選択したスクリプトのバージョンは、コンピューターのデフォルトのダウンロードフォルダーにダウンロードされます。スクリプトは、ダウンロードしたロボットと同じ名前で、新しく作成された Analytics プロジェクトに含まれます。プロジェクトとロボットは自動的に関連付けられます。

メモ

ロボットは2つのプロジェクトに関連付けられます。ダウンロードによって作成されたプロジェクトと、スクリプトをコミットするために以前に使用されたプロジェクトです。1つのロボットは複数のプロジェクトに関連付けることができます。

7. 省略可能。別のフォルダーで作業する場合は、ダウンロードしたスクリプトを含むプロジェクトを移動します。

Analytics でスクリプトを編集してコミットする

1. お使いのコンピューターから、Analytics デスクトップアプリケーションを開きます。
2. スクリプトを編集し、変更を保存します。

メモ

必要に応じて、スクリプトを追加したり、削除したりすることもできます。

3. Analytics メインメニューから、**ファイル > スクリプトのコミット**]を選択します。

エラーメッセージが表示される場合は、プロジェクトの1つ以上のスクリプトで、アナリティクスヘッダーまたはスクリプト構文に問題がある場合があります。

詳細については、[Analytics およびロボットにおけるスクリプトの開発ワークフロー](#)を参照してください。

4. コミットされたスクリプトの変更を説明する簡潔なコミットメッセージを入力し、**OK**]をクリックします。

スクリプトは関連付けられたロボットにコミットされ、新しいバージョンとして保存されます。ロボットの既存のバージョンのスクリプトは上書きされません。

関連付けられたロボットが見つからないというエラーメッセージが表示される場合は、ロボットがロボットに存在することと、ロールがスクリプトのコミットを許可していることを確認します。

5. 省略可能。 **スクリプトのコミットが成功しました**]ダイアログボックスで、新しいバージョンのスクリプトを検査するためのリンクのいずれかをクリックします。
6. **OK**]をクリックして、ダイアログボックスを閉じます。

スクリプトを別のACL ロボットにコミットする

スクリプトを別のACL ロボットにコミットし、スクリプトのバージョンをロボットに追加して、プロジェクトに関連付けられたロボットを変更します。スクリプトは編集または編集解除できます。

スクリプトを別のロボットにコミットするときには、2つの方法があります。

- スクリプトを新しいロボットにコミットする
 - スクリプトを既存のロボットにコミットする
1. お使いのコンピューターから、Analytics デスクトップ アプリケーションを開きます。
 2. Analytics メイン メニューから、**[ファイル > スクリプトのコミット]**を選択します。
エラー メッセージが表示される場合は、プロジェクトの1つ以上のスクリプトで、アナリティクス ヘッダーまたはスクリプト 構文に問題がある場合があります。
詳細については、[Analytics およびロボットにおけるスクリプトの開発 ワークフロー](#)を参照してください。
 3. 必要に応じて、**[送信先の選択]** ダイアログボックスで、該当する HighBond インスタンスに移動します。
[ロボット コレクション]と既存のロボットの一覧が表示されます。
 4. 次のいずれかを実行します。
 - **スクリプトを新しいロボットにコミットする - 新しいロボット** フィールドにロボット名を入力し、**[作成]**をクリックします。
ロボットが作成され、ロボット ID が自動的に生成されます。スクリプトを含む Analytics プロジェクトと新しいロボットが関連付けられます。

メモ

ロボット名には "\$"、"€" の文字を使用しないでください。

- **スクリプトを既存のロボットにコミットする** - ロボットのリストで、スクリプトをコミットするロボットを選択し、**[OK]**をクリックします。
スクリプトを含む Analytics プロジェクトと既存のロボットが関連付けられます。
プロジェクトと前のロボットの間に関連付けは削除されます。
5. コミットされたスクリプトを説明する簡潔なコミット メッセージを入力し、**[OK]**をクリックします。
スクリプトは、新しく作成されたロボットまたは既存のロボットにコミットされます。スクリプトのバージョン番号は、ロボットに既にスクリプトがあるか、以前に空であったかどうかによって異なります。
このスクリプトは、この時点では開発モードでのみ存在します。
 6. 省略可能。 **[スクリプトのコミットが成功しました]** ダイアログボックスで、新しく作成されたロボットまたは既存のロボット、あるいはコミットされたスクリプトを検査するためのリンクのいずれかをクリックします。
 7. **[OK]**をクリックして、ダイアログボックスを閉じます。

ACL ロボットのテーブル、ファイル、ログの表示

ACL ロボットに含まれる Analytics テーブルまたは結果ファイルを表示するには、まずそれらをローカルコンピュータにダウンロードする必要があります。結果ログはロボットから直接閲覧することもできますし、ダウンロードすることもできます。テーブル、ファイル、ログのダウンロードと表示に関するさまざまなオプションは、以下のとおりです。

テーブルまたはファイルの種類	結果パッケージまたは Analytics プロジェクトでダウンロード	個別にダウンロード	ロボットの表示
Analytics データ テーブル	✔	✘	✘
Analytics 結果 テーブル	✔	✘	✘
Analytics 結果 ファイル	✔	✔	✘
Analytics 結果 ログ	✔	✔	✔
非 Analytics 関連ファイル	✘	✔	✘

Analytics でのテーブルの表示

ダウンロードした Analytics テーブルがどのように表示されるかは、使用しているロボット エージェントの種類によって異なります。ダウンロードしたテーブルは、ダウンロードした結果パッケージ (*.zip) または Analytics プロジェクト (*.acl) に含まれています。

ロボット エージェントの詳細については、[HighBond ヘルプ](#)を参照してください。

エージェントの種類	詳細をダウンロードおよび表示
オンプレミスロボット エージェント	<ul style="list-style-type: none"> ダウンロード - ダウンロード済みパッケージまたはプロジェクトには、テーブルレイアウトのみが含まれています。テーブルデータは組織のネットワークに残りますが Analytics で分析用データを使用できます。 表示 - サーバー プロファイルを使用して、Analytics から組織のロボット エージェント内のテーブルデータに接続します。 サポートされるテーブル - Analytics 結果テーブルと Analytics データ テーブル。
クラウドベースのロボット エージェント	<ul style="list-style-type: none"> ダウンロード - ダウンロード済みパッケージまたはプロジェクトには、テーブルレイアウトとテーブルデータの両方が含まれています。 表示 - Analytics のみを使用してローカルのテーブルデータにアクセスします。 サポートされているテーブル - Analytics 結果テーブルのみ。

タスク結果のダウンロード

タスク結果には、Analytics 結果テーブル、他のファイルタイプ、およびログファイルを含めることができます。

タスク結果は //RESULT アナリティクス タグを使用して指定されます。詳細については、「RESULT タグ」ページ 2634を参照してください。

1. [ロボットアプリを開きます](#)。
2. ロボットのダッシュボードから、**ACL ロボット**]を選択します。
3. 結果を含むロボットをクリックします。
4. ロボットの右上隅にある **開発モード**]トグルを使って、使用する環境を選択します。
5. **タスクの実行**]タブを選択します。
6. ダウンロードする結果のタスク実行を選択します。
7. **タスク実行の詳細**パネルで、次のいずれかを実行します。

- **結果ファイルを個別にダウンロードする**

Excel などの Analytics 結果ファイルや、Analytics ログファイルをダウンロードするには、ファイル名の横にある**ダウンロード**をクリックします。

結果 - ファイルはローカルファイルシステムにダウンロードされ、ネイティブアプリケーションで開くことができます。Analytics ログファイルは任意のテキスト エディターで開けます。また、ログファイル名の横の **表示**]をクリックすると、ログを直接表示できます。

- **パッケージ内の全結果をダウンロードする**

タスクの実行のすべての結果 (Analytics の結果テーブルを含む)をダウンロードするには、次を実行します。

- a. **結果パッケージを生成**をクリックし、結果パッケージが生成されるまで待ちます。
- b. **結果パッケージをダウンロード**をクリックします。

結果 - **<ロボット名>.zip** という名の Analytics プロジェクトがローカルファイルシステムにダウンロードされます。

ここでは、圧縮ファイルの内容を抽出し、Analytics で Analytics プロジェクトを開き、Analytics の結果テーブルを表示することができます。Excel など、ネイティブアプリケーションで、他の結果ファイルタイプを開くことができます。

タスク実行が失敗した場合、**失敗したパッケージのダウンロード**]をクリックして、ログファイルをダウンロードします。また、ログファイル名の横の **表示**]をクリックすると、ログを直接表示できます。ログファイルを使用すると、タスクが失敗した理由を特定できます。

Analytics データ テーブルのダウンロード

メモ

Analytics データ テーブルは、オンプレミスロボット エージェントでのみサポートされます。これらは、クラウドベースのロボット エージェントではサポートされません。

データテーブルは、//DATA アナリティクス タグを使用して指定されます。詳細については、「DATA タグ」ページ 2641を参照してください。

1. [ロボットアプリを開きます](#)。
2. ロボットのダッシュボードから、[ACL ロボット]を選択します。
3. データテーブルを含むロボットをクリックします。
4. ロボットの右上隅にある **開発モード**]トグルを使って、使用する環境を選択します。
5. **入力/出力**]タブを選択します。
6. テーブルを含むセクションの1つで、ダウンロードするデータテーブルを選択します。
7. **テーブル詳細**]パネルで、**AN でテーブルを表示**]をクリックします。

結果 - Analytics プロジェクト名 <ロボット名>.acl がローカルコンピューターにダウンロードされます。プロジェクトには、選択したテーブルだけではなく、**入力/出力**]タブのすべてのデータテーブルが含まれます。

Analytics でプロジェクトを開き、データテーブルを表示できます。

ダウンロードした Analytics テーブルを開く(オンプレミス ロボット エージェント)

作業を開始する前に

Analytics からロボット エージェント上のデータに接続するには、次の要件を満たす必要があります。

- Analytics ユーザーは、ロボット エージェントがインストールされているサーバーで、適切な Windows ログオン権限とフォルダー アクセス権が必要です。

詳細については、[オンプレミス ロボット エージェント セキュリティ](#)を参照してください。

- Analytics の **サーバー統合を有効にする**]オプション(**ツール > オプション > インターフェイス**])を選択する必要があります。
- Analytics で **RobotsProfile** サーバー プロファイルを構成する必要があります。
- Analytics のエディション(Unicode または非 Unicode) がロボット エージェントのエディションと一致する必要があります。

テーブルを開く

1. お使いのコンピューターで、ダウンロードまたは解凍した Analytics プロジェクト(*.zip または *.acl) に移動します。
2. 圧縮されているプロジェクトの場合は、それを右クリックし、該当するオプションを選択して解凍します。
3. ダウンロードまたは解凍されたプロジェクトをダブルクリックします。
プロジェクトが Analytics によって開かれます。
4. ロボット テーブルをダブルクリックして開きます。
5. ネットワーク パスワードを入力し、**OK**]をクリックします。

トラブルシューティング

Analytics でロボット テーブルを開こうとしているときに接続エラーが発生する場合は、次の修正策を実行してください。

- **ロボット データ サービス**-ロボット エージェントがインストールされているサーバーで、**ロボット データ サービス**が実行中であることを確認します。停止している場合は、サービスを起動します。
- **ロボット サーバー プロファイル**-サーバー プロファイル、**RobotsProfile** を削除し、Analytics プロジェクトを閉じてからもう一度開き、更新されたプロファイルを自動的に作成します。

RobotsProfile を削除するには、Analytics メイン メニューで、**サーバー > サーバー プロファイル**]に移動します。

ダウンロードした Analytics テーブルを開く(クラウド ロボット エージェント)

1. お使いのコンピューターで、圧縮されている Analytics プロジェクト (*.zip) に移動します。
2. このプロジェクトを右クリックし、該当するオプションを選択して解凍します。
3. 解凍されたプロジェクトをダブルクリックします。

プロジェクトが Analytics によって開かれます。

4. テーブルをダブルクリックして開きます。