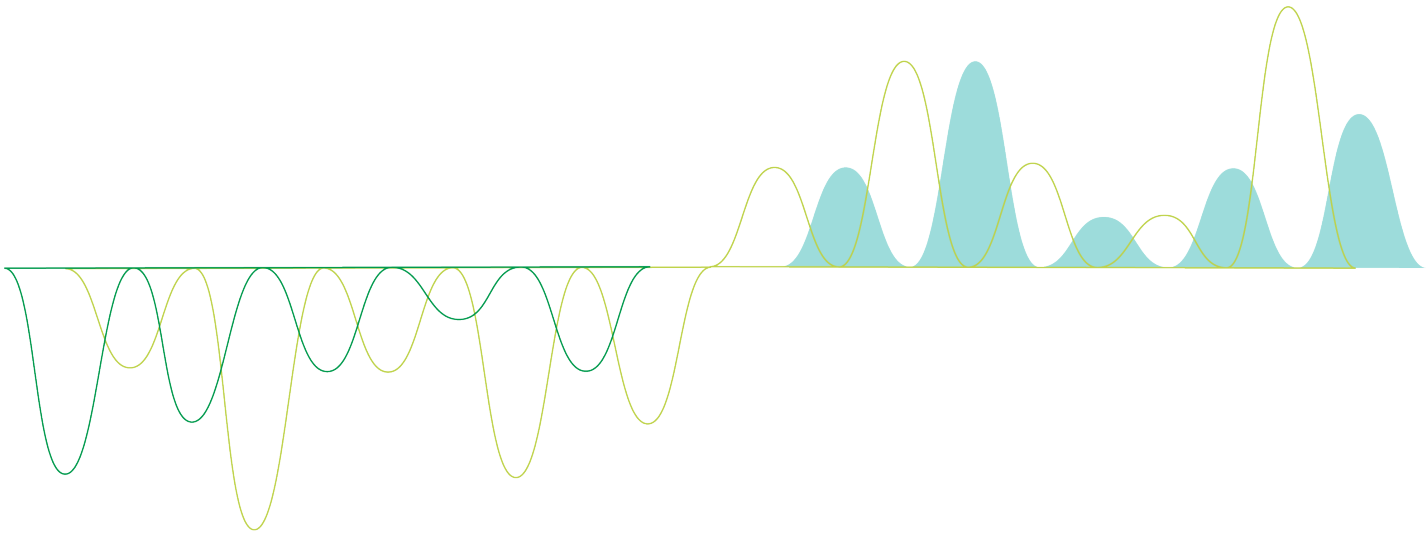
데이터 관리

Qlik Sense®
August 2022
Copyright © 1993-2022 QlikTech International AB. 무단 전재 및 복제를 금합니다.



© 2022 QlikTech International AB. All rights reserved. All company and/or product names may be trade names, trademarks and/or registered trademarks of the respective owners with which they are associated.

Contents

1 이 문서 정보 **10**
2 데이터 관리 **11**
3 데이터 관리자를 사용하여 데이터 로드 및 관리 **12**
3.1 데이터 테이블 미리 보기 12
3.2 새 데이터 테이블 추가 12
3.3 데이터 테이블 편집 13
3.4 데이터 테이블 삭제 13
3.5 데이터 테이블 연결 관리 13
3.6 변경 적용 및 데이터 다시 로드 14
3.7 데이터 관리자에서 작업 실행 취소 및 다시 실행 15
3.8 데이터 관리자에서 테이블 변환 세부 정보 보기 15
3.9 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용 15
3.10 데이터 관리자에서 테이블 연결 16
3.11 앱에 데이터 추가 16
앱 내 17
파일 위치 17
데이터 연결 17
데이터 콘텐츠 17
이 앱에 파일 첨부 18
새 데이터 소스에 연결 18
데이터 추가 18
사용 가능한 데이터 소스 18
기존 데이터 소스의 데이터 추가 18
새 데이터 소스의 데이터 추가 20
데이터 파일 첨부 및 앱에 데이터 추가 22
에서 수동으로 데이터 추가Qlik Sense 25
데이터 필드 선택 26
파일에서 데이터 필터링 32
데이터 커넥터에서 데이터 필터링 33
3.12 테이블 편집 34
테이블 이름 바꾸기 35
필드 이름 바꾸기 35
다른 테이블에 대한 연결 관리 35
필드 유형 및 표시 형식 변경 35
분석에서 필드 숨기기 36
데이터를 로드하기 전에 테이블 필드 데이터 평가 36
테이블의 필드 값 바꾸기 36
테이블에서 필드 값을 Null로 설정 36
필드 값에 사용자 지정 순서 설정 37
테이블에서 필드 분할 37
측정값 데이터를 범위로 그룹화 37
필드 변환 세부 정보 보기 37
크로스탭 데이터 피벗 해제 37
데이터 소스에서 테이블 업데이트 38
계산 필드 추가 38
테이블 정렬 38
작업 실행 취소 및 재실행 38

테이블 편집기에서 데이터 연결 ... 39
계산 필드 사용 ... 41
필드 유형 변경 ... 53
분석에서 필드 숨기기 ... 55
데이터를 로드하기 전에 테이블 필드 데이터 평가 ... 55
테이블의 필드 값 바꾸기 ... 57
테이블에서 필드 값을 Null로 설정 ... 59
차원 값 순서 사용자 지정 ... 60
테이블에서 필드 분할 ... 61
측정값 데이터를 범위로 그룹화 ... 63
데이터 관리자에서 크로스탭 데이터 피벗 해제 ... 66
3.13 데이터 관리자에서 테이블 연결 ... 68
테이블 자동연결 ... 68
테이블 간 강제 연결 ... 69
연결된 테이블 분할 ... 71
3.14 데이터 관리자에서 테이블 조인 ... 72
조인 연산자 ... 72
테이블 조인 ... 75
조인된 테이블 분할 ... 77
3.15 데이터 관리자에서 테이블 및 필드 변환 세부 정보 보기 ... 77
테이블 세부 정보 보기 ... 78
필드 세부 정보 보기 ... 78
3.16 단계별 연습 - 강제 연결을 사용하여 테이블 결합 ... 78
연결 개요 ... 78
연습 - 강제 연결 ... 78
추가 단계 - 새 테이블 추가 및 데이터 필드 연결 ... 84
3.17 데이터 관리자에서 스크립팅된 테이블 동기화 ... 86
스크립팅된 테이블 동기화 ... 87
관리되는 스크립팅된 테이블 제거 ... 87
3.18 데이터 연결 관리 ... 88
권장 연결 패널을 사용하는 테이블 연결 ... 89
수동으로 테이블 연결 ... 90
연결 해제 ... 91
연결 편집 ... 91
데이터 미리 보기 ... 92
가상 키 ... 92
제한 사항 ... 92
변경 적용 및 데이터 로드 ... 93
4 스크립트를 사용하여 데이터 로드 및 변환 ... 94
4.1 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용 ... 94
4.2 데이터 로드 편집기 사용 ... 95
도구 모음 ... 95
데이터 연결 ... 95
텍스트 편집기 ... 95
섹션 ... 95
출력 ... 96
데이터 로드 편집기에서 데이터 소스에 연결 ... 96

데이터 로드 편집기에서 데이터 선택 98
데이터 로드 스크립트 편집 105
스크립트 코드 분류 108
데이터 로드 스크립트 디버그 109
로드 스크립트 저장 112
데이터를 로드하는 스크립트 실행 112
데이터 로드 편집기의 키보드 바로 가기 112
4.3 스크립트 구문 및 데이터 구조에 대한 이해 114
추출, 변환 및 로드 114
데이터 로드 문 115
스크립트의 실행 115
필드 116
논리 테이블 120
Qlik Sense의 데이터 유형 130
달러 기호 확장 134
스크립트에서 인용 부호 사용 145
데이터의 와일드카드 149
NULL 값 처리 150
4.4 데이터 및 필드에 대한 지침 152
로드되는 데이터의 양에 대한 지침 152
데이터 테이블 및 필드의 상한 152
로드 스크립트 섹션의 권장 한계 153
숫자 및 시간 형식에 대한 설명 153
4.5 QVD 파일 작업 156
QVD 파일의 용도 157
QVD 파일 만들기 157
QVD 파일에서 데이터 읽기 158
QVD 형식 158
4.6 에서 분석 연결 구성Qlik Sense Desktop 158
Qlik 오픈 소스 SSE 리포지토리 159
요소 설명 159
5 Section Access를 사용하여 데이터 보안 관리 161
5.1 로드 스크립트의 섹션 161
Section Access 시스템 필드 161
5.2 앱에 대한 사용자 액세스 관리 163
5.3 앱의 특정 데이터에 대한 사용자 액세스 관리 164
행 수준 데이터에 대한 액세스 관리 164
열 수준 데이터에 대한 액세스 관리 165
사용자 그룹에 대한 액세스 관리 166
5.4 가장을 사용하여 데이터 다시 로드 166
5.5 Multi-Cloud 환경에서 사용자 액세스 관리 167
5.6 Section Access 및 Insight Advisor Chat 사용 168
5.7 Section Access 사용에 대한 지침 및 팁 168
6 On-demand 앱으로 빅 데이터 관리 170
6.1 On-demand 앱 구성 요소 170
6.2 On-demand 앱 구성 171
6.3 On-demand 앱 게시 172

6.4 On-demand 앱의 장점 ... 172
6.5 제한 사항 ... 172
6.6 On-demand 선택 앱 만들기 ... 173
6.7 On-demand 템플릿 앱 만들기 ... 173
템플릿 앱의 구조 ... 174
단일 사인온(SSO) ... 175
템플릿 앱에 대한 노드 다시 로드 ... 176
On-demand 템플릿 앱의 바인딩 표현식 ... 176
6.8 On-demand 앱 만들기 ... 182
7 동적 보기를 사용한 데이터 관리 ... 185
7.1 동적 보기 개요 ... 185
동적 보기 ... 186
동적 보기 템플릿 앱 ... 187
동적 차트 ... 187
7.2 동적 보기 제한 ... 187
7.3 스트림 및 동적 보기 ... 188
7.4 동적 보기 및 차트 만들기 ... 188
동적 보기 만들기 ... 188
시트에 동적 차트 추가 ... 189
동적 보기 편집 ... 189
7.5 동적 보기 및 차트 사용 ... 189
동적 보기에서 선택 ... 189
동적 보기 세부 정보 보기 ... 191
동적 보기 새로 고침 ... 194
8 데이터 소스에 연결 ... 195
8.1 연결 만들기 ... 195
8.2 데이터 연결 유형 ... 195
첨부된 파일 ... 195
데이터베이스 커넥터 ... 195
Essbase ... 196
로컬 또는 네트워크 파일 ... 196
DSN을 통한 ODBC 연결 ... 196
Qlik 웹 커넥터 ... 196
REST ... 197
Salesforce ... 198
SAP ... 198
웹 파일 ... 198
웹 저장소 공급자 커넥터 ... 198
타사 커넥터 ... 198
8.3 데이터 연결 저장 위치 ... 198
8.4 파일에서 데이터 로드 ... 199
파일 형식 ... 199
연결 유형 ... 199
파일에서 데이터를 로드하는 방법 ... 199
로컬 및 네트워크 파일 폴더에서 파일 로드 ... 200
웹 리소스에서 파일 로드 ... 200

Microsoft Excel 스프레드시트에서 데이터 로드 ... 201
8.5 데이터베이스에서 데이터 로드 ... 204
ODBC 데이터베이스에서 데이터 로드 ... 204
ODBC ... 205
ODBC 데이터 소스에서 데이터 로드 ... 205
OLE DB ... 208
데이터베이스 내의 논리 ... 209
8.6 를 사용하여 대규모 데이터 셋에 액세스Direct Discovery ... 210
Direct Discovery와 인 메모리 데이터의 차이점 ... 210
Direct Discovery 필드 유형 ... 216
에서 지원하는 데이터 소스Direct Discovery ... 217
를 사용할 때의 제한 사항Direct Discovery ... 219
에서 다중 테이블 지원Direct Discovery ... 221
에서 하위 쿼리 사용하기Direct Discovery ... 222
Direct Discovery 액세스 로깅 ... 225
9 데이터 모델 보기 및 변환 ... 226
9.1 테이블 이동 ... 226
9.2 테이블 크기 조정 ... 227
9.3 데이터 모델 성능 ... 227
9.4 데이터 모델 뷰어에서 테이블 및 필드 미리 보기 ... 230
테이블의 미리 보기 표시 ... 230
필드의 미리 보기 표시 ... 231
9.5 데이터 모델 뷰어에서 마스터 차원 만들기 ... 231
9.6 데이터 모델 뷰어에서 마스터 측정값 만들기 ... 232
10 데이터 모델링의 모범 사례 ... 233
10.1 데이터 열을 행으로 변환 ... 233
10.2 데이터 행을 필드로 변환 ... 233
10.3 계층 수준으로 구성된 데이터(예: 조직 체계) 로드 ... 234
10.4 대규모 데이터베이스에서 신규 또는 업데이트된 레코드만 로드 ... 235
10.5 공통 필드를 사용하여 두 테이블의 데이터 결합 ... 235
10.6 간격에 불연속 값 일치 ... 235
10.7 일관성 없는 필드 값 처리 ... 236
10.8 일관성 없는 필드 값 대소문자 처리 ... 237
10.9 특정 지역 관련 데이터를 로드하여 맵으로 데이터 시각화 ... 238
10.10 증분 로드를 통해 새 레코드 및 업데이트된 레코드 로드 ... 238
추가만 ... 239
삽입만(업데이트 또는 삭제는 제외) ... 239
삽입 및 업데이트(삭제는 제외) ... 240
삽입, 업데이트 및 삭제 ... 240
10.11 Join 및 Keep을 사용한 테이블 결합 ... 241
SQL SELECT 문 내의 조인 ... 242
Join ... 242
Keep ... 243
Inner ... 243
Left ... 244
Right ... 245

10.12 조인에 대한 대안으로 매핑 사용 ... 246
10.13 데이터 로드 스크립트에서 교차 표 작업 ... 248
하나의 한정하는 열이 있는 크로스탭 피벗 해제 ... 248
두 개의 한정하는 열이 있는 크로스탭 피벗 해제 ... 249
10.14 일반 데이터베이스 ... 250
10.15 간격과 불연속 데이터의 연결 ... 252
Intervalmatch 예 ... 252
확장된 intervalmatch 구문을 사용하여 느리게 변화하는 차원 문제 해결 ... 253
10.16 단일 날짜에서 날짜 간격 만들기 ... 255
10.17 계층 구조 데이터 로드 ... 258
10.18 자체 맵 데이터 로드 ... 259
맵 시각화의 필드에 대해 지원되는 이름 데이터 ... 259
KML 파일에서 포인트 및 영역 데이터 로드 ... 260
데이터 프로파일링을 사용하여 맵 데이터 로드 ... 260
포인트 데이터 로드 및 서식 지정 ... 261
10.19 데이터 정리 ... 263
매핑 테이블 ... 263
매핑 테이블 사용 ... 264
11 Insight Advisor의 논리 모델 사용자 지정 ... 265
11.1 비즈니스 논리를 사용하여 Insight Advisor의 논리 모델 구축 ... 265
논리 모델 이해하기 ... 266
논리 모델 사용자 지정 ... 266
사용자 지정 비즈니스 논리 활성화 ... 267
비즈니스 논리 재설정 ... 267
비즈니스 논리 비활성화 ... 267
필드 및 그룹 정의 ... 267
패키지로 논리 모델 범위 설정 ... 271
계층 구조로 드릴다운 분석 만들기 ... 272
논리 모델에 동작 적용 ... 272
캘린더 기간으로 분석 기간 정의 ... 275
11.2 Insight Advisor용 어휘 만들기 ... 281
제한 사항 ... 282
Insight Advisor에 동의어 추가 ... 282
Insight Advisor에 사용자 지정 분석 추가 ... 283
11.3 자습서 - Insight Advisor에서 데이터 해석 방법 사용자 지정 ... 287
학습 내용 ... 287
이 자습서의 대상 ... 287
준비 사항 ... 287
이 자습서의 단원 ... 288
추가 자료 및 리소스 ... 288
Insight Advisor 및 비즈니스 논리란? ... 288
사용자 지정 논리 모델 활성화 ... 291
필드 및 그룹 사용자 지정 ... 293
패키지 구성 ... 300
계층 구조 정의 ... 301
캘린더 기간 구성 ... 304
동작 구성 ... 308

어휘 만들기 311
12 문제 해결 - 데이터 로드 313
12.1 파일을 데이터 추가에 끌어서 첨부하는 방식이 작동하지 않습니다. 313
12.2 ANSI 이외로 인코딩된 데이터 파일을 사용하는 경우의 문자 집합 문제 313
12.3 데이터 로드 시 순환 참조 경고 313
12.4 고정된 레코드 파일에서 데이터를 선택하면 열이 예상대로 정렬되지 않습니다. 314
12.5 커넥터가 작동하지 않습니다. 314
커넥터가 제대로 설치되지 않았습니다. 314
커넥터가 Qlik Sense에 맞게 조정되어 있지 않습니다. 314
12.6 SQL Server를 다시 시작한 후 데이터 연결이 작동을 멈춥니다. 315
12.7 데이터 로드 편집기에 스크립트가 표시되지 않습니다. 315
12.8 데이터 로드 스크립트가 오류 없이 실행되었으나 데이터가 로드되지 않습니다. 316
문이 세미콜론으로 종결되지 않았습니다. 316
문자열 내의 작은따옴표 316
12.9 데이터가 포함된 앱의 테이블이 데이터 관리자에 표시되지 않습니다. 316
12.10 서버에서 앱을 만드는 모든 사용자에 대해 데이터 관리자 워크플로가 손상되었습니다. 317
12.11 OLE DB 데이터 소스를 사용하는 경우의 데이터 선택 문제 317
12.12 날짜 필드가 시트 보기에서 날짜 필드로 인식되지 않습니다. 317
테이블 추가 시 데이터 프로파일링이 비활성화되었습니다. 317
날짜 서식이 인식되지 않습니다. 318
12.13 파일 첨부 시 "잘못된 경로" 오류 메시지 발생 318
12.14 QlikView 문서에서 변환된 앱 로드 시 오류 발생 318
스크립트에서 절대 파일 경로 참조가 사용되었습니다. 318
스크립트에서 지원되지 않는 함수 또는 문이 사용되었습니다. 319
12.15 Microsoft Excel: 데이터 관리자 또는 데이터 로드 편집기에서 파일의 데이터를 로드할 수 없습니다. 319
12.16 Microsoft Excel: ODBC를 통해 파일에 연결하거나 해당 파일에서 데이터를 로드할 때 문제 발생 319
12.17 디스크 공간이 부족합니 다. 319
12.18 데이터 로드 시 가상 키 경고 320
12.19 공통 필드가 있는 테이블이 필드 이름에 의해 자동으로 연결되지 않습니다. 320

1 이 문서 정보

이 문서에서는 데이터 추가 및 관리 방법, 고급 데이터 모델을 위한 데이터 로드 스크립트 작성 방법, 데이터 모델 뷰어에서 결과 데이터 모델을 보는 방법을 설명하고 Qlik Sense에서 데이터를 모델링하는 모범 사례를 제공합니다.

스크립트 함수 및 차트 함수에 대한 더 자세한 내용은 스크립트 구문 및 차트 함수을 참조하십시오.

이 문서는 Qlik Sense에 대한 온라인 도움말에서 파생되었습니다. 도움말의 일부를 오프라인으로 읽거나 페이지를 간편하게 인쇄하려는 사용자를 위한 것이며, 온라인 도움말에 비해 다른 추가 정보는 포함되어 있지 않습니다.

help.qlik.com/sense에서 온라인 도움말, 추가 가이드 등을 찾을 수 있습니다.

2 데이터 관리

Qlik Sense 앱을 만든 후 첫 번째 단계는 탐색 및 분석할 수 있는 데이터를 추가하는 것입니다. 이 섹션에서는 데이터 추가 및 관리 방법, 고급 데이터 모델을 위한 데이터 로드 스크립트 작성 방법, 데이터 모델 뷰어에서 결과 데이터 모델을 보는 방법을 설명하고 Qlik Sense에서 데이터를 모델링하는 모범 사례를 제공합니다.

앱에 데이터를 추가하는 방법은 두 가지가 있습니다.

- **데이터 관리자**

  스크립트 언어를 배울 필요 없이 자체 데이터 소스 또는 Qlik DataMarket 등의 다른 소스에서 데이터를 추가할 수 있습니다. 데이터 선택 항목을 편집할 수 있으며, 데이터 모델에서 데이터 연결을 만드는 데 도움이 됩니다.

- **데이터 로드 편집기**

  Qlik Sense 데이터 로드 스크립트 언어를 사용하여 ETL (Extract, Transform & Load) 프로세스를 통해 데이터 모델을 만들 수 있습니다. 강력한 스크립트 언어를 사용하면 복잡한 변환 및 확장 가능한 데이터 모델 생성을 수행할 수 있습니다.

데이터 관리자*에서 기본 제공되는 데이터 모델을* ***데이터 로드 편집기****에서 추가로 개발할 수 있는 데이터 로드 스크립트로 변환할 수 있지만, 데이터 로드 스크립트를* ***데이터 관리자*** *데이터 모델로 변환할 수는 없습니다. 데이터 로드 스크립트에서 정의한* ***데이터 관리자*** *데이터 모델 및 데이터 테이블은 계속 공존할 수 있지만 이로 인해 데이터 모델에서 문제를 해결하기 어렵게 될 수 있습니다.*

3 데이터 관리자를 사용하여 데이터 로드 및 관리

앱에서 데이터를 사용할 수 있도록 **데이터 관리자**에서 데이터를 추가하고 관리합니다.

데이터 관리자에는 두 가지 보기가 있습니다.

- **연결**
  테이블 간의 연결을 만들고 편집할 수 있습니다.
- **테이블**
  데이터 추가를 사용하여 추가하든 또는 데이터 로드 스크립트를 사용하여 로드하든 상관없이 앱의 모든 데이터 테이블에 대한 개요를 볼 수 있습니다. 각 테이블이 테이블 이름, 데이터 필드 수, 데이터 소스 이름과 함께 표시됩니다.

3.1 데이터 테이블 미리 보기

테이블을 미리 보고 테이블이 포함하고 있는 열 및 데이터의 샘플 집합을 확인할 수 있습니다.

다음과 같이 하십시오.

- 미리 볼 데이터 테이블을 선택합니다.

테이블 데이터 셋의 미리 보기가 표시됩니다.

3.2 새 데이터 테이블 추가

앱에 데이터 테이블을 신속하게 추가할 수 있습니다. **데이터 관리자**를 연 다음 ＋을 클릭합니다. ☰에서 **데이터 추가**를 클릭할 수도 있습니다. 또한 새 앱을 만들 때 데이터를 추가할 것인지 묻는 메시지가 표시됩니다.

다음 데이터 소스에서 데이터를 추가할 수 있습니다.

데이터 소스

데이터 소스	설명
앱 내	앱에서 사용할 수 있는 데이터 소스에서 선택합니다. 이 데이터 소스는 앱에 첨부된 파일이 될 수 있습니다. 데이터 소스를 만들고 **수동 입력**을 사용하여 수동으로 데이터를 추가할 수도 있습니다.
파일 위치	관리자에 의해 정의된 드라이브 등 네트워크 드라이브에 있는 파일에서 선택합니다.
데이터 연결	사용자 또는 관리자에 의해 정의된 기존 데이터 연결에서 선택합니다.

3.3 데이터 테이블 편집

데이터 추가를 사용하여 추가한 모든 데이터 테이블을 편집할 수 있습니다. 데이터 테이블에서 테이블과 필드의 이름을 변경하고 데이터 소스의 필드를 업데이트할 수 있습니다. 또한 계산 필드를 추가하고 날짜 및 시간 서식을 조정할 수도 있습니다.

다음과 같이 하십시오.

1. 편집할 데이터 테이블에서 을 클릭합니다.
   데이터 테이블 편집기가 열리고 원하는 편집 및 변환 작업을 수행할 수 있습니다.
2. **닫기**를 클릭하여 돌아갑니다.

이제 테이블이 **업데이트 보류**로 표시되고 다음 번에 데이터를 다시 로드할 때 변경 내용이 앱 데이터에 적용됩니다.

> ***데이터 추가**를 사용하여 추가한 데이터 테이블만 편집할 수 있습니다. 로드 스크립트를 사용하여 로드한 테이블에서 을 클릭하면 데이터 로드 편집기가 열립니다. 자세한 내용은 데이터 로드 편집기 사용 (page 95)을 참조하십시오.*

3.4 데이터 테이블 삭제

데이터 추가를 사용하여 추가한 데이터 테이블만 삭제할 수 있습니다. 로드 스크립트를 사용하여 로드한 데이터 테이블만 데이터 로드 편집기에서 스크립트를 편집하여 제거할 수 있습니다.
다음과 같이 하십시오.

- 삭제할 데이터 테이블에서 을 클릭합니다.

이제 해당 테이블은 **삭제 보류**로 표시되며 다음 번에 데이터를 다시 로드할 때 제거됩니다.

및 를 클릭하여 삭제 작업을 실행 취소 및 다시 실행할 수 있습니다.

> *시각화에서 해당 데이터 테이블의 필드를 사용한 경우 데이터 테이블을 제거하면 앱에서 오류가 표시됩니다.*

3.5 데이터 테이블 연결 관리

연결해야 하는 여러 테이블을 추가할 때, 가장 좋은 방법은 서로 다른 테이블에서 동일한 이름을 갖는 키 필드를 사용하여 테이블을 연결하는 것입니다. 이 경우 **데이터 추가**의 데이터 프로파일링 옵션을 비활성화한 상태로 Qlik Sense에 키 필드를 추가할 수 있으며, 그 결과로 테이블이 올바르게 연결된 데이터 구조가 생성됩니다.

이상적인 데이터 소스에 미치지 못하면 여러 가지 연결 문제가 발생할 수 있습니다.

- 동일한 데이터가 있지만 필드 이름이 서로 다른 두 필드를 서로 다른 두 테이블에서 로드하는 경우 테이블이 연결되도록 필드 이름을 동일하게 지정하는 것이 좋습니다.
- 서로 다른 데이터가 있지만 필드 이름이 동일한 두 필드를 서로 다른 두 테이블에서 로드하는 경우 각각 별도의 필드로 로드되도록 필드 중 하나 또는 둘 모두 이름을 변경해야 합니다.
- 둘 이상의 공통 필드를 포함하는 두 개의 테이블을 로드하는 경우

데이터를 연결하려면 데이터 프로파일링이 활성화된 상태에서 **데이터 추가** 옵션을 사용하는 것이 좋습니다. 이 옵션이 기본 옵션입니다. 앱 데이터 페이지의 오른쪽 아래 모서리에 있는 **데이터 추가** 버튼 옆에 있는 •••을 클릭하여 이 설정을 확인할 수 있습니다.

Qlik Sense에서는 로드할 데이터의 프로파일링을 수행하여 테이블 연결을 정정하도록 지원합니다. 기존의 잘못된 연결 및 잠재적으로 올바른 연결이 강조 표시되므로 데이터 분석 결과에 기반하여 연결할 필드를 선택하는 데 도움이 됩니다.

다음 두 가지 방법으로 테이블 연결을 관리할 수 있습니다.

- 데이터 관리자의 **연결** 보기에서
  Insight Advisor 권장 사항에 기반하여 연결을 만들거나 하나 이상의 필드에 기반하여 사용자 지정 연결을 만들 수 있습니다.
- 테이블 편집기의 **연결** 옵션 사용
  사용자 지정 연결 및 여러 필드에 기반한 복합 키 연결을 만들 수 있습니다.

> *데이터 추가 시 데이터 프로파일링을 비활성화하면 Qlik Sense에서 공통 필드 이름에 기반하여 자동으로 테이블을 연결합니다.*

3.6 변경 적용 및 데이터 다시 로드

데이터 관리자에서 수행한 변경은 데이터를 로드할 때까지 앱에서 사용할 수 없습니다. 데이터를 로드하면 변경 내용이 적용되고 추가한 새 데이터는 모두 외부 데이터 소스에서 로드됩니다. 이전에 로드한 데이터는 다시 로드되지 않습니다.

데이터 관리자 바닥글의 버튼을 사용하여 외부 데이터 소스에서 모든 데이터를 다시 로드할 수 있습니다.

버튼은 선택한 테이블에 대한 모든 데이터를 다시 로드합니다. 앱의 모든 테이블에 대한 모든 데이터를 로드하지는 않습니다.

데이터 관리자의 데이터가 앱 데이터와 동기화되어 있지 않으면 **데이터 로드** 버튼이 녹색으로 표시됩니다. **연결** 보기에서 모든 새 테이블 또는 업데이트된 테이블은 *로 표시되고, 삭제된 테이블은 밝은 회색으로 표시됩니다. **테이블** 보기에서 모든 새 테이블, 업데이트된 테이블 또는 삭제된 테이블은 파란색으로 강조 표시되고 테이블 상태를 보여 주는 아이콘이 표시됩니다.

- **삭제 보류** 로 표시된 테이블이 삭제됩니다.
- **업데이트 보류** 로 표시된 테이블이 추가, 이름 변경 또는 제거된 필드로 업데이트되거나 테이블의 이름이 변경됩니다.
- **추가 보류** 로 표시된 테이블이 추가됩니다.

다음과 같이 하십시오.

- **데이터 로드**를 클릭하여 앱의 변경 내용을 로드합니다.

이제 **데이터 관리자**에서 수행한 변경 사항으로 앱 데이터가 업데이트됩니다.

변경 내용을 적용하고 외부 데이터 소스에서 선택된 테이블의 모든 데이터를 다시 로드하는 방법:

다음과 같이 하십시오.

- **데이터 관리자** 바닥글의 버튼을 클릭합니다.

3.7 데이터 관리자에서 작업 실행 취소 및 다시 실행

데이터 관리자에서 편집하는 경우, 및 를 클릭하거나 키보드 바로 가기 Ctrl + Z 및 Ctrl + Y를 사용하여 일부 작업을 실행 취소 또는 다시 실행할 수 있습니다.

다음 경우에는 작업 로그가 지워집니다.

- 테이블 개요에서 **연결**로 이동하는 등 보기를 변경하는 경우
- 데이터를 로드하는 경우
- **데이터 관리자**를 닫는 경우

3.8 데이터 관리자에서 테이블 변환 세부 정보 보기

데이터 관리자에서 **세부 정보** 대화 상자를 사용하여 테이블에서 수행되는 작업 및 변환을 볼 수 있습니다. **세부 정보** 대화 상자는 **연결** 및 **테이블** 보기에서 제공됩니다.

세부 정보에는 선택한 테이블에 수행되는 현재 작업 및 변환이 표시됩니다. 세부 정보에서는 테이블 소스, 현재 변경된 사항 및 변경 적용 순서를 보여 줍니다. **세부 정보**를 통해 테이블이 어떻게 현재 상태가 되었는지 보다 쉽게 이해할 수 있습니다. 예를 들어 **세부 정보**를 사용하여 테이블 연결 순서를 쉽게 확인할 수 있습니다.

3.9 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용

데이터 관리자에서 데이터 테이블을 추가하면 데이터 로드 스크립트 코드가 생성됩니다. 데이터 로드 편집기의 **자동 생성 섹션**에서 스크립트 코드를 볼 수 있습니다. 또한 생성된 스크립트 코드를 잠금 해제하여 편집할 수 있지만, 그렇게 하면 더 이상 데이터 테이블을 **데이터 관리자**에서 관리할 수 없습니다.

기본적으로 로드 스크립트에서 정의한 데이터 테이블은 **데이터 관리자**에서 관리되지 않습니다. 즉, 데이터 개요에서 테이블을 볼 수 있지만 **데이터 관리자**에서 테이블을 삭제하거나 편집할 수는 없으며, 스크립트를 사용하여 로드된 테이블에는 연결 권장 사항이 제공되지 않습니다. 그러나 스크립팅된 테이블을 **데이터 관리자**와 동기화하면 스크립팅된 테이블이 관리되는 스크립팅된 테이블로 **데이터 관리자**에 추가됩니다.

> *테이블을 동기화한 경우 다른 탭에서 **데이터 관리자**를 연 상태로 데이터 로드 편집기에서 변경해서는 안 됩니다.*

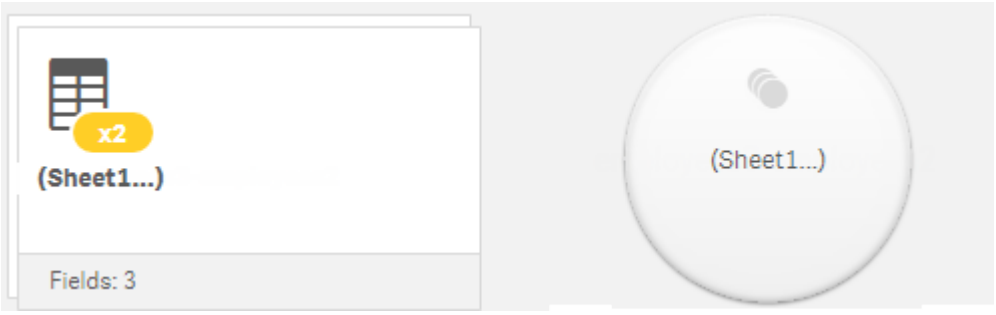
스크립트 섹션을 추가하고 **데이터 관리자**에서 만든 데이터 모델을 개선하고 이 데이터 모델과 상호 작용하는 코드를 개발할 수 있지만, 일부 주의해야 할 부분이 있습니다. 사용자가 작성한 스크립트 코드는 **데이터 관리자** 데이터 모델에 방해가 될 수 있으며, 다음과 같은 경우에 문제가 발생합니다.

- 스크립트에서 **데이터 관리자**를 사용하여 추가된 테이블의 이름을 바꾸거나 삭제하는 경우.
- **데이터 관리자**를 사용하여 추가된 테이블에서 필드를 삭제하는 경우.
- **데이터 관리자**를 사용하여 추가된 테이블과 스크립트에서 로드된 테이블을 연결하는 경우.
- **데이터 관리자**를 사용하여 추가된 테이블의 필드에서 **Qualify** 문을 사용하는 경우
- 스크립트에서 **Resident**를 사용하여 **데이터 관리자**를 통해 추가된 테이블을 로드하는 경우
- 생성된 코드 섹션 뒤에 스크립트 코드를 추가하는 경우. 데이터 모델에서의 결과 변경이 **데이터 관리자**에는 반영되지 않습니다.

3.10 **데이터 관리자**에서 테이블 연결

연결은 두 개의 테이블을 결합된 필드가 있는 단일 테이블로 결합합니다. 콘텐츠를 통합하여 콘텐츠를 공유하는 별도의 테이블 및 필드 수를 줄입니다. **데이터 관리자**에서 테이블을 자동 또는 강제로 연결할 수 있습니다.

테이블 보기 및 연결 보기의 연결된 테이블

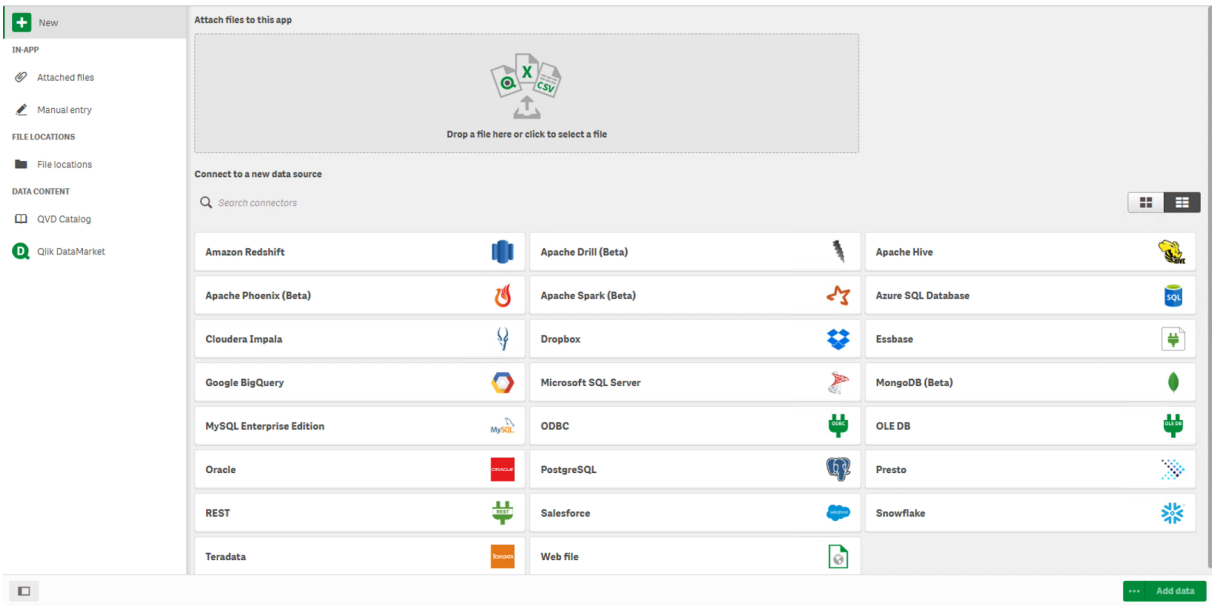


3.11 앱에 데이터 추가

앱에 데이터를 신속하게 추가할 수 있습니다. **데이터 관리자**를 연 다음 ＋을 클릭합니다. ☰에서 **데이터 추가**를 클릭할 수도 있습니다. 또한 새 앱을 만들 때 데이터를 추가할 것인지 묻는 메시지가 표시됩니다. 시트를 편집하는 경우 **필드** 패널에서 **추가**를 클릭하여 데이터를 추가할 수도 있습니다.

사용할 수 있는 데이터 추가 옵션 및 데이터 소스는 Qlik Sense 플랫폼 및 구성에 따라 달라집니다.

데이터 보기 추가



앱 내

첨부된 파일 배포: Qlik Sense Enterprise on Windows. 앱에 파일을 첨부하거나 앱에 첨부한 파일을 보려면 클릭합니다. 이러한 파일에서 데이터를 로드할 수 있습니다.

수동 입력. 앱 내 테이블을 만들고 앱에 데이터를 추가하려면 클릭합니다.

파일 위치

파일 위치. 배포: Qlik Sense Enterprise on Windows. 관리자가 정의한 폴더 위치에 대한 액세스를 제공합니다.

내 컴퓨터. 배포: Qlik Sense Desktop.

데이터 파일을 업로드하거나 이미 업로드된 파일의 데이터를 추가하려면 클릭합니다.

데이터 연결

배포: 모두.

외부 데이터 소스에 대해 만들어진 연결을 표시합니다. 연결은 **새 데이터 소스에 연결**에서 만들어진 후에 표시됩니다.

앱에 데이터를 추가하려면 연결을 클릭합니다.

데이터 콘텐츠

Qlik Catalog Service. 플랫폼: Qlik Sense Enterprise on Windows.

Qlik Catalog Service QVD 소스에 대한 액세스를 제공합니다. 관리자가 Qlik Catalog에 대한 연결을 만든 경우에만 사용할 수 있습니다.

앱에 데이터를 추가하려면 클릭합니다.

이 앱에 파일 첨부

앱에 파일을 첨부하려면 클릭합니다.

새 데이터 소스에 연결

새 데이터 소스에 대한 연결을 만들려면 클릭합니다.

데이터 추가

앱에 데이터를 추가하려면 클릭합니다. 연결을 만들고 로드할 데이터를 선택하면 버튼이 활성화됩니다. 앱에 데이터를 추가하려면 클릭합니다. 프로파일링이 활성화 또는 비활성화된 데이터를 추가할 수 있습니다. 데이터 프로파일링은 권장되며 기본적으로 활성화됩니다. 데이터 프로파일링을 비활성화하려면 •••을 클릭합니다.

사용 가능한 데이터 소스

사용 가능한 데이터 소스 유형은 다음과 같이 다양한 요인에 따라 달라집니다.

- 액세스 설정
  관리자 설정을 통해 사용자가 연결할 수 있는 데이터 소스 유형을 결정합니다.
- 설치된 커넥터
  Qlik Sense에는 많은 데이터 소스에 대한 지원 기능이 기본 제공됩니다. 기본 제공된 커넥터는 Qlik Sense에서 자동으로 설치됩니다. 추가적인 데이터 소스에 연결하려면 해당 데이터 소스에 맞는 커넥터를 별도로 설치해야 할 수 있습니다. 이러한 별도로 설치되는 커넥터는 Qlik 또는 타사에서 공급합니다.
- 로컬 파일의 가용성
  사용자의 데스크톱 컴퓨터에 있는 로컬 파일은 Qlik Sense Desktop에서만 사용할 수 있습니다. Qlik Sense의 서버 설치에서는 이러한 파일을 사용할 수 없습니다.

  Qlik Sense의 서버 설치에 로드하려는 로컬 파일이 있는 경우 해당 파일을 앱에 첨부하거나 Qlik Sense 서버에서 사용할 수 있는 폴더(폴더 데이터 연결로 이미 정의되어 있는 폴더)로 파일을 전송해야 합니다.

기존 데이터 소스의 데이터 추가

사용자 또는 관리자에 의해 이미 정의된 연결에서 데이터를 앱에 추가할 수 있습니다. 여기에는 데이터베이스, 데이터 파일이 포함된 폴더 또는 Salesforce.com 등과 같은 외부 데이터 소스를 위한 커넥터가 포함될 수 있습니다.

***데이터 로드 편집기**에 동일한 이름과 동일한 열을 가진 스크립트된 테이블로 이미 추가된 테이블은 **데이터 관리자**에서 추가하지 마십시오.*

데이터 추가에서 연결을 삭제하려면 연결을 마우스 오른쪽 단추로 클릭하고 을 선택합니다.

> *연결을 삭제하는 경우 **데이터 관리자**에서 데이터를 로드하기 전에 해당 연결을 사용한 모든 테이블을 삭제해야 합니다.*

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 을 클릭합니다. 에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **데이터 연결**에서 기존 연결을 선택합니다.
   일부 연결은 로드할 테이블 및 필드를 선택할 수 있는 데이터 소스에 직접 연결됩니다. 예를 들어 Salesforce.com이나 데이터베이스에 연결하면 데이터를 선택할 수 있는 소스에 직접 연결됩니다.
4. 연결에서 선택 옵션을 제공하는 경우 데이터를 추가할 특정 데이터 소스를 선택합니다.
   이러한 옵션은 데이터 소스 유형에 따라 달라집니다.
   - 파일 기반 데이터 소스: 파일을 선택합니다.
   - 데이터베이스: 사용할 데이터베이스를 설정합니다.
   - 웹 파일: 웹 파일의 URL을 입력합니다.
   - 기타 데이터 소스: 커넥터에서 지정합니다.
5. 로드할 테이블과 필드를 선택합니다.
6. 필요한 경우 선택한 필드에 포함된 데이터의 하위 집합을 선택하려면 데이터 필터를 적용하도록 선택합니다.
   데이터 소스가 파일이면 **필터**를 선택합니다. 필터를 추가할 테이블 옆의 **필터 추가**를 클릭하고 필드를 선택하고 기준을 선택한 다음 필터링할 값을 입력합니다.

   > *Qlik Sense는 QVD 파일에서 날짜 필드에 대한 필터를 지원하지 않습니다.*

   다음에 주의하십시오.
   - 동일한 필드에 여러 필터를 적용할 수 있습니다.
   - **데이터 관리자**의 **연결** 보기 또는 **소스에서 데이터 선택**에서 필터를 제거할 수 있습니다. 변경 사항을 적용하려면 **데이터 로드** 버튼을 클릭하여 데이터를 다시 로드합니다.

   데이터베이스 및 커넥터의 경우 **데이터 필터링**을 선택하면 필터 기준에 대한 텍스트 상자가 열립니다.
   다음에 주의하십시오.
   - 필터는 데이터베이스에서 필드 이름에 적용됩니다. **데이터 관리자**에서 필드 이름을 바꾸려면 데이터베이스에서 원래 필드 이름에 필터를 적용해야 합니다. 예를 들어 데이터베이스에서 EMP로 지정된 필드 이름을 **데이터 관리자**에서 EMPLOYEE로 바꾸려면 필터 EMP = '*filter_value*'를 적용해야 합니다.
   - **데이터 관리자**의 **연결** 보기에서 데이터 필터를 지울 수 있습니다. 변경 사항을 적용하려면 **데이터 로드** 버튼을 클릭하여 데이터를 다시 로드합니다. 필터를 지우기 전에 연결된 테이블을 분할해야 합니다.
   - 데이터 필터링은 현재 일부 커넥터 또는 첨부된 파일에 사용할 수 없습니다.

7. **데이터 추가**를 클릭하여 데이터 관리자의 **연결** 보기에서 데이터를 엽니다. 그러면 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결할 수 있습니다.
   데이터 프로파일링은 **데이터 추가**를 클릭하면 기본적으로 활성화됩니다. 데이터 프로파일링에서는 다음을 수행합니다.
   - 데이터 연결을 권장합니다.
   - 테이블 간의 공통 필드를 자동으로 한정합니다. 이렇게 하면 테이블 이름을 기준으로 고유한 접두사를 추가합니다.
   - 날짜 및 시간 필드를 autoCalendar에 매핑합니다.

   테이블은 공통 필드 이름을 기준으로 자동으로 연결되지 않습니다. **연결** 보기에서 테이블을 연결할 수 있습니다.

   > *앱으로 직접 데이터를 로드하려면•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 이렇게 하면 데이터를 추가할 때 외부 데이터 소스에서 새로 선택된 데이터를 로드합니다. 테이블이 공통 필드 이름을 기준으로 자동으로 연결됩니다. 날짜 및 시간 필드는 만들어지지 않습니다.*

   자세한 내용은 *데이터 연결 관리 (page 88)*를 참조하십시오.
8. 데이터 준비가 완료되면 **데이터 로드**를 클릭합니다. 심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.
   자세한 내용은 *문제 해결 - 데이터 로드 (page 313)*를 참조하십시오.

외부 소스에서 선택한 모든 데이터를 다시 로드하려면 **데이터 관리자** 바닥글에 있는 ⟳ 버튼을 사용합니다. 이렇게 하면 선택 내용에 대한 소스에서 현재 데이터를 모두 가져올 수 있습니다. 모든 데이터를 로드하면 새 데이터만 로드하는 것보다 더 오래 걸립니다. 이전에 로드한 데이터가 데이터 소스에서 변경되지 않은 경우 모든 데이터를 로드할 필요는 없습니다.

새 데이터 소스의 데이터 추가

새 데이터 소스에서 앱에 데이터를 추가할 수 있습니다. 새 데이터 소스의 데이터를 추가하는 경우 해당 데이터 소스에 대한 연결이 데이터 연결에서 생성되므로 동일한 데이터 소스에서 데이터를 보다 쉽게 추가할 수 있습니다.

> ***데이터 로드 편집기**에 동일한 이름과 동일한 열을 가진 스크립트된 테이블로 이미 추가된 테이블은 **데이터 관리자**에서 추가하지 마십시오.*

데이터 추가에서 연결을 삭제하려면 연결을 마우스 오른쪽 단추로 클릭하고 🗑을 선택합니다.

> *연결을 삭제하는 경우 **데이터 관리자**에서 데이터를 로드하기 전에 해당 연결을 사용한 모든 테이블을 삭제해야 합니다.*

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 ＋을 클릭합니다. ☰에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **새 데이터 소스에 연결**에서 소스를 선택합니다.
4. 데이터 소스에 필요한 연결 매개 변수를 입력합니다.
   예:
   - 파일 기반 데이터 소스는 파일에 대한 경로를 지정하고 파일 형식을 선택해야 합니다.
   - Oracle 및 IBM DB2와 같은 데이터베이스는 데이터베이스 속성 및 액세스 자격 증명이 필요합니다.
   - 웹 파일은 해당 웹 파일의 URL이 필요합니다.
   - ODBC 연결은 DSN 자격 증명이 필요합니다.
5. 로드할 테이블과 필드를 선택합니다.
6. 필요한 경우 선택한 필드에 포함된 데이터의 하위 집합을 선택하려면 데이터 필터를 적용하도록 선택합니다.
   데이터 소스가 파일이면 **필터**를 선택합니다. 필터를 추가할 테이블 옆의 **필터 추가**를 클릭하고 필드를 선택하고 기준을 선택한 다음 필터링할 값을 입력합니다.

   > *Qlik Sense는 QVD 파일에서 날짜 필드에 대한 필터를 지원하지 않습니다.*

   다음에 주의하십시오.
   - 동일한 필드에 여러 필터를 적용할 수 있습니다.
   - **데이터 관리자**의 **연결** 보기 또는 **소스에서 데이터 선택**에서 필터를 제거할 수 있습니다. 변경 사항을 적용하려면 **데이터 로드** 버튼을 클릭하여 데이터를 다시 로드합니다.

   데이터베이스 및 커넥터의 경우 **데이터 필터링**을 선택하면 필터 기준에 대한 텍스트 상자가 열립니다.
   다음에 주의하십시오.
   - 필터는 데이터베이스에서 필드 이름에 적용됩니다. **데이터 관리자**에서 필드 이름을 바꾸려면 데이터베이스에서 원래 필드 이름에 필터를 적용해야 합니다. 예를 들어 데이터베이스에서 EMP로 지정된 필드 이름을 **데이터 관리자**에서 EMPLOYEE로 바꾸려면 필터 EMP = '*filter_value*'를 적용해야 합니다.
   - **데이터 관리자**의 **연결** 보기에서 데이터 필터를 지울 수 있습니다. 변경 사항을 적용하려면 **데이터 로드** 버튼을 클릭하여 데이터를 다시 로드합니다. 필터를 지우기 전에 연결된 테이블을 분할해야 합니다.
   - 데이터 필터링은 현재 일부 커넥터 또는 첨부된 파일에 사용할 수 없습니다.
7. **데이터 추가**를 클릭하여 데이터 관리자의 **연결** 보기에서 데이터를 엽니다. 그러면 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결할 수 있습니다.
   데이터 프로파일링은 **데이터 추가**를 클릭하면 기본적으로 활성화됩니다. 데이터 프로파일링에서는 다음을 수행합니다.
   - 데이터 연결을 권장합니다.
   - 테이블 간의 공통 필드를 자동으로 한정합니다. 이렇게 하면 테이블 이름을 기준으로 고유한 접두사를 추가합니다.
   - 날짜 및 시간 필드를 autoCalendar에 매핑합니다.

테이블은 공통 필드 이름을 기준으로 자동으로 연결되지 않습니다. **연결** 보기에서 테이블을 연결할 수 있습니다.

데이터를 앱으로 직접 로드하려면•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 또한 이렇게 하면 데이터를 추가할 때 데이터 소스에서 기존의 모든 데이터를 다시 로드합니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다. 날짜 및 시간 필드는 만들어지지 않습니다.

자세한 내용은 *데이터 연결 관리 (page 88)*를 참조하십시오.

8. 데이터 준비가 완료되면 **데이터 로드**를 클릭합니다. 심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.
자세한 내용은 *문제 해결 - 데이터 로드 (page 313)*를 참조하십시오.

데이터 파일 첨부 및 앱에 데이터 추가

앱에 데이터 파일을 첨부한 다음 앱에서 데이터를 사용할 수 있습니다.

첨부된 파일은 해당 파일을 첨부한 앱에서만 사용할 수 있습니다. 원본 데이터 파일에 대한 연결이 없으므로 원본 파일을 업데이트한 경우에는 첨부된 파일을 새로 고쳐야 합니다.

제한된 데이터를 노출하지 않으려면 앱을 게시하기 전에 섹션 액세스 설정이 있는 모든 첨부 파일을 제거합니다. 앱이 게시될 때 첨부 파일이 포함됩니다. 게시된 앱을 복사하는 경우 복사본에 첨부 파일이 포함됩니다. 하지만 첨부된 데이터 파일에 섹션 액세스 제한이 적용된 경우, 파일을 복사할 때 섹션 액세스 설정이 유지되지 않으므로 복사된 앱 사용자는 첨부된 파일에서 모든 데이터를 볼 수 있습니다.

제한 사항

- 앱에 첨부된 최대 파일 크기는 50MB입니다.
- 앱에 첨부된 최대 총 파일 크기는 미디어 라이브러리에 업로드된 이미지 파일을 포함하여, 200MB입니다.
- Qlik Sense Desktop에서는 파일을 첨부할 수 없습니다.

여러 데이터 파일을 신속하게 첨부

일련의 데이터 파일을 앱에 첨부하고 추가하는 가장 빠르고 대개의 경우 가장 쉬운 방법이 앱에 파일을 끌어서 놓는 방법입니다.

다음과 같이 하십시오.

- 하나 이상의 데이터 파일을 앱에 끌어서 놓습니다.
파일이 업로드되어 앱에 첨부되고 데이터 모델에 추가됩니다.

이 방법으로 파일을 첨부하는 경우 Qlik Sense에서는 포함된 필드 이름, 필드 구분 기호 또는 문자 집합 인식 등과 같이 데이터를 로드하는 데 필요한 최적의 설정을 선택하려고 시도합니다. 테이블이 최적의 설정으로 추가되지 않은 경우는 테이블 편집기에서 테이블을 열고 **소스에서 데이터 선택**을 클릭하여 설정을 수정할 수 있습니다.

데이터 로드 편집기나 데이터 모델 뷰어에서는 파일을 끌어서 놓을 수 없습니다.

단일 데이터 파일 첨부

데이터 파일은 하나씩 첨부할 수 있습니다. 이렇게 하면 포함된 필드 이름, 필드 구분 기호 또는 사용된 문자 집합 등과 같은 파일 가져오기 설정을 더 상세하게 제어할 수 있습니다.

데이터 로드 편집기*에 동일한 이름과 동일한 열을 가진 스크립트된 테이블로 이미 추가된 테이블은* ***데이터 관리자****에서 추가하지 마십시오.*

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 ＋을 클릭합니다.☰에서 **데이터 추가**를 클릭할 수도 있습니다.
3. 데이터 파일을 끌어서 놓거나 로드할 파일을 컴퓨터에서 클릭하여 선택합니다.
   이미 첨부된 파일과 동일한 이름의 파일을 첨부하려고 하면 해당 첨부된 파일을 새 파일로 바꾸는 옵션이 표시됩니다.

   각 첨부된 파일은 고유한 파일 이름을 가져야 합니다.
4. 로드할 테이블과 필드를 선택합니다.
5. 필요한 경우 선택한 필드에 포함된 데이터의 하위 집합을 선택하려면 데이터 필터를 적용하도록 선택합니다.
   데이터 소스가 파일이면 **필터**를 선택합니다. 필터를 추가할 테이블 옆의 **필터 추가**를 클릭하고 필드를 선택하고 기준을 선택한 다음 필터링할 값을 입력합니다.

   Qlik Sense는 QVD 파일에서 날짜 필드에 대한 필터를 지원하지 않습니다.

   다음에 주의하십시오.
   - 동일한 필드에 여러 필터를 적용할 수 있습니다.
   - **데이터 관리자**의 **연결** 보기 또는 **소스에서 데이터 선택**에서 필터를 제거할 수 있습니다. 변경 사항을 적용하려면 **데이터 로드** 버튼을 클릭하여 데이터를 다시 로드합니다.
6. **데이터 추가**를 클릭하여 데이터 관리자의 **연결** 보기에서 데이터를 엽니다. 그러면 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결할 수 있습니다.
   데이터 프로파일링은 **데이터 추가**를 클릭하면 기본적으로 활성화됩니다. 데이터 프로파일링에서는 다음을 수행합니다.
   - 데이터 연결을 권장합니다.
   - 테이블 간의 공통 필드를 자동으로 한정합니다. 이렇게 하면 테이블 이름을 기준으로 고유한 접두사를 추가합니다.
   - 날짜 및 시간 필드를 autoCalendar에 매핑합니다.

   테이블은 공통 필드 이름을 기준으로 자동으로 연결되지 않습니다. **연결** 보기에서 테이블을 연결할 수 있습니다.

> *데이터를 앱으로 직접 로드하려면•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 또한 이렇게 하면 데이터를 추가할 때 데이터 소스에서 기존의 모든 데이터를 다시 로드합니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다. 날짜 및 시간 필드는 만들어지지 않습니다.*

7. 데이터 준비가 완료되면 **데이터 로드**를 클릭합니다. 심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.

첨부된 파일 삭제

데이터 관리자에서 첨부된 파일에 기반한 테이블을 삭제하면 해당 테이블이 데이터 모델에서 삭제되지만 첨부된 데이터 파일은 앱에 그대로 유지됩니다. 앱에서 데이터 파일을 영구적으로 삭제할 수 있습니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 ＋을 클릭합니다.
3. **첨부된 파일**을 클릭합니다.
4. 해당 파일을 삭제합니다.

이제 데이터 파일이 앱에서 영구적으로 삭제됩니다.

> *앱에서 사용되는 첨부된 파일을 삭제하면 **데이터 관리자** 또는 로드 스크립트에서 해당 파일에 대한 참조를 제거하기 전까지 앱을 다시 로드할 수 없습니다. **데이터 로드 편집기**에서 로드 스크립트를 편집합니다.*

첨부된 파일에서 데이터 다시 로드

앱을 위해 업로드되는 파일은 해당 앱에 첨부되며 그 앱에서만 사용할 수 있습니다.

원래 데이터 파일에 대한 연결은 없습니다. 원본 파일을 업데이트하는 경우 앱에 첨부된 파일을 새로 고쳐야 합니다. 그런 다음 업데이트된 데이터를 앱에 로드할 수 있습니다. **데이터 관리자**에서 데이터를 다시 로드한 후 (**소스에서 데이터 새로 고침**)을 클릭하면 테이블 보기에서 업데이트된 데이터를 볼 수 있습니다.

> ***데이터 로드 편집기**에 동일한 이름과 동일한 열을 가진 스크립트된 테이블로 이미 추가된 테이블은 **데이터 관리자**에서 추가하지 마십시오.*

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 ＋을 클릭합니다.
3. **첨부된 파일**을 클릭합니다.

4. 기존 파일을 바꿉니다. 업데이트된 파일은 원래 파일과 이름이 같아야 합니다. 데이터 파일 내용이 새로 고쳐집니다.
5. **데이터 추가**를 클릭합니다. •••를 클릭하여 데이터 프로파일링이 활성화되도록 합니다.
6. **연결** 보기 또는 **테이블** 보기에서 테이블을 클릭합니다.
7. ↻를 클릭하여 데이터를 업데이트합니다.
8. **데이터 로드**를 클릭하여 데이터를 앱으로 다시 로드합니다.

> ⚠ *데이터 파일의 필드 구조를 변경한 경우 즉, 필드를 제거하거나 이름을 변경한 경우에는 앱의 데이터 모델에 영향을 줄 수 있으며, 특히 테이블을 연결하는 데 사용된 필드가 포함된 경우에는 더 큰 영향이 있습니다.*

에서 수동으로 데이터 추가Qlik Sense

데이터 추가에서 **수동 입력**을 통해 편집기에 데이터를 입력하고 **데이터 관리자**에 테이블로 추가할 수 있습니다.

수동 데이터 입력은 다른 소스에서 제한된 데이터 집합을 사용하려는 경우에 유용합니다. 예를 들어 Excel 스프레드시트 또는 웹 페이지의 테이블에서 행만 선택하여 **데이터 관리자**에 테이블로 로드하고 싶은 경우, 선택한 데이터를 복사하여 **수동 입력**에 붙여넣은 다음 **데이터 관리자**의 테이블로 추가할 수 있습니다. 데이터 양이 적은 경우 다른 데이터 소스에서 가져오는 것보다 수동으로 추가하는 것이 더 빠를 수 있는 경우에도 수동 데이터 입력이 유용합니다.

데이터를 수동으로 추가하려면 **데이터 추가**를 열고 **수동 입력**을 선택하고 데이터를 테이블에 입력한 다음 테이블을 **데이터 관리자**에 추가합니다. 테이블 편집기는 하나의 행과 두 개의 열로 시작하지만 테이블에 데이터를 추가하면 테이블에 빈 행과 열이 자동으로 추가됩니다.

> ⚠ ***수동 입력**은 데이터를 입력할 때 자동으로 저장하지 않습니다. 데이터가 **데이터 관리자**에 추가되기 전에 연결이 끊어지거나 화면을 새로 고치거나 세션이 만료되면 입력한 데이터가 손실될 수 있습니다.*

데이터 입력 외에도 다른 소스에서 데이터를 복사하여 붙여넣을 수 있습니다. **수동 입력**은 Excel 테이블에서 복사한 데이터의 행과 열을 유지합니다.

수동 입력에서 쉽고 효과적으로 작업하는 데 사용할 수 있는 여러 가지 키보드 바로 가기가 있습니다. 바로 가기 동작은 셀, 행 또는 열을 선택하는지 또는 테이블에서 셀을 편집하는지에 따라 다릅니다. 다음 표에 선택 바로 가기가 나와 있습니다.

선택을 위한 키보드 바로 가기

바로 가기	설명
화살표 키	선택 셀 간에 이동합니다.
Tab	선택 셀의 오른쪽으로 이동합니다. 오른쪽에 셀이 없는 경우 다음 행의 첫 번째 셀로 이동합니다.

바로 가기	설명
Shift+Tab	선택 셀의 왼쪽으로 이동합니다. 왼쪽에 셀이 없는 경우 이전 행의 첫 번째 셀로 이동합니다.
Enter	편집 모드로 전환합니다.
Del	현재 선택을 삭제합니다.

다음 표에는 편집 바로 가기가 나와 있습니다.

편집을 위한 키보드 바로 가기

바로 가기	설명
화살표 키	셀에서 커서를 이동합니다.
Tab	편집 내용을 적용하고 오른쪽의 다음 셀로 이동합니다.
Shift+Tab	편집 내용을 적용하고 왼쪽의 이전 셀로 이동합니다.
Enter	편집 내용을 적용하고 아래쪽의 다음 셀로 이동합니다.
Shift+Enter	편집 내용을 적용하고 위쪽의 이전 셀로 이동합니다.
Esc	편집 내용을 취소하고 선택 모드로 전환합니다.

수동 입력을 사용하여 만든 테이블은 나중에 편집하여 내용을 추가하거나 제거할 수 있습니다. 자세한 내용은 *데이터 소스에서 테이블 업데이트 (page 38)*를 참조하십시오.

수동으로 데이터 추가

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 ＋을 클릭합니다.
   ☰ 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **앱 내**에서 **수동 입력**을 클릭합니다.
4. 테이블의 이름을 입력합니다.
5. 테이블 편집기에서 데이터를 입력합니다.
   셀을 두 번 클릭하여 셀에 데이터를 입력합니다.
   셀을 편집하는 중에 테이블의 다른 셀을 클릭하면 편집 내용을 적용하고 다른 셀을 선택합니다.
6. 데이터가 완료되면 **데이터 추가**를 클릭합니다.

데이터 관리자에 테이블이 추가됩니다.

데이터 필드 선택

데이터를 추가할 때 또는 테이블을 편집할 때 사용할 테이블과 필드를 선택할 수 있습니다.

CSV 파일 등의 일부 데이터 소스에는 단일 테이블이 포함되어 있는 반면, Microsoft Excel 스프레드시트 또는 데이터베이스 등과 같은 데이터 소스에는 여러 테이블이 포함될 수 있습니다.

테이블에 머리글 행이 포함되어 있는 경우, 일반적으로 필드 이름이 자동으로 감지되지만 경우에 따라 **필드 이름** 설정을 변경해야 할 수 있습니다. 또한 데이터를 정확하게 해석하기 위해 **머리글 크기** 또는 **문자 집합** 등의 테이블 옵션을 변경해야 할 수도 있습니다. 테이블 옵션은 데이터 소스 유형마다 다릅니다.

데이터베이스에서 데이터 선택

데이터베이스에서 데이터를 선택하는 단계는 데이터베이스 연결 방법에 따라 달라집니다. Qlik Sense와 함께 설치된 Qlik ODBC Connector Package의 일부인 Qlik Database 커넥터를 통해 직접 연결하거나 DSN 소스로 ODBC 드라이버에 연결할 수 있습니다.

자세한 내용은 *ODBC (page 205)*를 참조하십시오.

데이터베이스에서 데이터를 추가하는 경우 데이터 소스에 여러 개의 테이블이 포함될 수 있습니다.

다음과 같이 하십시오.

1. 드롭다운 목록에서 **데이터베이스**를 선택합니다.
   연결을 구성할 때 데이터베이스 이름이 입력되므로 일부 선택 대화 상자에는 **데이터베이스** 드롭다운 목록이 없습니다.
2. 데이터베이스 **소유자**를 선택합니다.
   테이블 목록은 선택된 데이터베이스에서 사용할 수 있는 보기와 테이블로 채워집니다. 일부 데이터베이스는 데이터 선택 프로세스에서 소유자를 지정할 필요가 없습니다.
3. 테이블을 선택합니다.
4. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다.
   테이블 이름 옆의 상자를 선택하면 테이블 내의 모든 필드를 선택할 수 있습니다.

   기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드를 기준으로 연결되기 때문입니다.
5. 추가 테이블에서 데이터를 추가하려면 이를 선택합니다.
6. 데이터 선택이 완료되면 **데이터 추가**를 클릭하여 데이터 프로파일링을 진행하고 테이블 관계에 대한 권장 사항을 확인합니다.
   앱으로 직접 데이터를 로드하려면 **데이터 추가** 옆에 있는•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 이렇게 하면 선택한 데이터가 그대로 로드되어 데이터 프로파일링 단계를 통과하고 시각화 만들기를 시작할 수 있습니다. 테이블은 자연 연결, 즉 공통으로 이름이 지정된 필드를 사용하여 연결됩니다.

Microsoft Excel 스프레드시트에서 데이터 선택

Microsoft Excel 스프레드시트에서 데이터를 추가하면 파일에 여러 개의 시트가 포함될 수 있습니다. 각 시트는 개별 테이블로 로드됩니다. 시트에 다른 시트 또는 로드된 테이블과 동일한 필드/열 구조가 있는 경우는 예외인데, 이런 경우는 테이블이 컨케트네이트됩니다.

다음과 같이 하십시오.

1. 해당 시트에 대한 설정이 적절한지 확인하십시오.
   테이블 데이터를 올바르게 해석하도록 도와주는 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다. Excel 스프레드시트에서는 일반적으로 첫 행에 포함된 필드 이름이 있습니다. **필드 이름 없음**을 선택하면 필드 이름은 A,B,C...로 지정됩니다.
헤더 크기	테이블 헤더(일반적으로 일반 정보를 포함하며 열 서식으로 표시되지 않는 행)로서 생략할 행 수를 설정합니다.

   예
   스프레드시트가 다음과 같습니다.
   스프레드시트 예

컴퓨터:	AEJ12B	-	-
날짜:	2015-10-05 09	-	-
타임스탬프	**순서**	**연산자**	**산출값**
2015-10-05 09:22	00122344	A	52
2015-10-05 10:31	00153534	A	67
2015-10-05 13:46	00747899	B	86

   이 경우, 처음 두 줄을 무시하고 Timestamp, Order, Operator 및 Yield 필드가 있는 테이블을 로드하고자 할 수 있습니다. 이를 위해서는 다음 설정을 사용하십시오.
   처음 두 줄을 무시하고 필드를 로드하도록 설정합니다.

UI 항목	설명
헤더 크기	2 처음 두 줄을 헤더 데이터로 간주하고 파일 로드 시 무시합니다. 이 경우, Machine: 및 Date:로 시작하는 두 줄은 테이블 데이터의 일부가 아니므로 무시됩니다.
필드 이름	**포함된 필드 이름** 읽은 첫 줄을 해당 열에 대한 필드 이름으로 사용합니다. 이 경우, 처음 두 줄이 헤더 데이터이므로 읽을 첫 줄은 세 번째 줄입니다.

2. 데이터를 선택할 첫 번째 시트를 선택합니다. 시트 이름 옆의 상자를 선택하면 시트 내의 모든 필드를 선택할 수 있습니다.
3. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다.

기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드에 의해 연결되기 때문입니다.

4. 데이터 선택이 완료되면 **데이터 추가**를 클릭하여 데이터 프로파일링을 진행하고 테이블 관계에 대한 권장 사항을 확인합니다.
   앱으로 직접 데이터를 로드하려면 **데이터 추가** 옆에 있는•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 이렇게 하면 선택한 데이터가 그대로 로드되어 데이터 프로파일링 단계를 통과하고 시각화 만들기를 시작할 수 있습니다. 테이블은 자연 연결, 즉 공통으로 이름이 지정된 필드를 사용하여 연결됩니다.

테이블 파일에서 데이터 선택

다양한 데이터 파일에서 데이터를 추가할 수 있습니다.

다음과 같이 하십시오.

1. **파일 형식**에 적절한 파일 형식이 선택되어 있는지 확인하십시오.
2. 해당 파일에 대한 설정이 적절한지 확인하십시오. 파일 형식이 다르면 파일 설정이 달라집니다.
3. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다. 또한 시트 이름 옆의 상자를 선택하여 파일 내의 모든 필드를 선택할 수도 있습니다.

   > *기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드에 의해 연결되기 때문입니다.*

   데이터 선택이 완료되면 **데이터 추가**를 클릭하여 데이터 프로파일링을 진행하고 테이블 관계에 대한 권장 사항을 확인합니다.

   > *앱으로 직접 데이터를 로드하려면 **데이터 추가** 옆에 있는•••을 클릭한 다음 데이터 프로파일링을 비활성화합니다. 이렇게 하면 선택한 데이터가 그대로 로드되어 데이터 프로파일링 단계를 통과하고 시각화 만들기를 시작할 수 있습니다. 테이블은 자연 연결, 즉 공통으로 이름이 지정된 필드를 사용하여 연결됩니다.*

4.

파일 형식에 맞는 설정 선택

구분된 테이블 파일

이러한 설정은 각 레코드가 줄 바꿈으로 구분되어 있고 각 필드는 CSV 파일처럼 구분 문자를 사용하여 구분되어 있는 단일 테이블이 포함되어 있는 구분된 테이블 파일에 대해 검증됩니다.

파일 형식 설정

구분된 테이블 파일의 파일 서식 설정

UI 항목	설명
구분된 테이블 파일의 파일 서식	**구분** 또는 **고정 레코드**로 설정합니다. 선택하면, 데이터 선택 대화 상자가 선택한 파일 형식에 맞춰 조정됩니다.
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.

UI 항목	설명
구분 기호	테이블 파일에서 사용되는 **구분 기호**를 설정합니다.
인용	인용 부호를 처리하는 방법을 지정하기 위해 설정합니다. **없음** = 인용 문자가 허용되지 않음 **표준** = 표준 인용 부호(필드 값의 첫 문자 및 마지막 문자로 인용 부호 사용 가능) **MSQ** = 최신 인용 부호(필드에서 복수 행의 콘텐츠 허용)
헤더 크기	테이블 헤더로 생략할 행 수를 설정합니다.
문자 집합	테이블 파일에서 사용되는 문자 집합을 설정합니다.
주석	데이터 파일은 레코드 사이에 주석이 포함될 수 있으며, 이 주석은 행 시작 부분에 하나 이상의 특수 문자(예: //)로 표시됩니다. 주석 행을 나타내는 하나 이상의 문자를 지정합니다. Qlik Sense에서는 여기에 지정된 문자로 시작하는 행을 로드하지 않습니다.
EOF 무시	데이터에 필드 값의 일부로 파일 끝 문자가 포함되는 경우, **EOF 무시**를 선택합니다.

고정 레코드 데이터 파일

고정 레코드 데이터 파일에는 각 레코드(데이터 행)에 고정된 필드 크기의 열이 여러 개 있고 일반적으로 공백 또는 탭 문자가 추가된 단일 테이블이 포함되어 있습니다.

필드 나누기 위치 설정

두 가지 방법으로 필드 나누기 위치를 설정할 수 있습니다.

- **필드 나누기 위치**에 쉼표로 구분된 필드 나누기 위치를 수동으로 입력합니다. 각 위치는 필드의 시작 부분을 나타냅니다.

  1,12,24

- **필드 나누기**를 활성화하여 필드 데이터 미리 보기에서 대화형으로 필드 나누기 위치를 편집합니다. **필드 나누기 위치**가 선택한 위치로 업데이트됩니다. 다음을 수행할 수 있습니다.
  - 필드 데이터 미리 보기를 클릭하여 필드 구분선을 삽입합니다.
  - 필드 구분선을 클릭하여 삭제합니다.
  - 필드 구분선을 끌어서 이동합니다.

파일 형식 설정

고정 레코드 데이터 파일의 파일 서식 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.
헤더 크기	**헤더 크기**를 테이블 헤더로서 생략할 행 수로 설정합니다.

UI 항목	설명
문자 집합	테이블 파일에서 사용하는 문자 집합으로 설정됩니다.
탭 크기	테이블 파일에서 하나의 탭 문자가 나타내는 공백 수로 설정합니다.
레코드 줄 크기	테이블 파일에서 하나의 레코드 범위에 해당하는 행 수로 설정합니다. 기본값은 1입니다.

HTML 파일

HTML 파일에는 여러 개의 테이블을 포함할 수 있습니다. Qlik Sense에서는 <TABLE> 태그가 있는 모든 요소를 테이블로 해석합니다.

파일 형식 설정

HTML 파일의 파일 서식 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.
문자 집합	테이블 파일에서 사용하는 문자 집합을 설정합니다.

XML 파일

XML 형식으로 저장된 데이터를 로드할 수 있습니다.

XML 파일에 대한 특정 파일 형식 설정은 없습니다.

QVD 파일

QVD 형식으로 저장된 데이터를 로드할 수 있습니다. QVD는 네이티브 Qlik 형식이며 Qlik Sense 또는 QlikView에서만 읽고 쓸 수 있습니다. 파일 형식은 Qlik Sense 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다.

QVD 파일에 대한 특정 파일 형식 설정은 없습니다.

QVX 파일

Qlik data eXchange(QVX) 형식으로 저장된 데이터를 로드할 수 있습니다. QVX 파일은 Qlik QVX SDK를 사용하여 개발한 사용자 지정 커넥터를 통해 생성됩니다.

QVX 파일에 대한 특정 파일 형식 설정은 없습니다.

KML 파일

KML 형식으로 저장된 맵 파일을 로드하여 맵 시각화에 사용할 수 있습니다.

KML 파일에 대한 특정 파일 형식 설정은 없습니다.

이전 단계로 돌아가기(**데이터 추가**)

데이터를 추가하는 중에 이전 단계로 돌아갈 수 있습니다.

다음과 같이 하십시오.

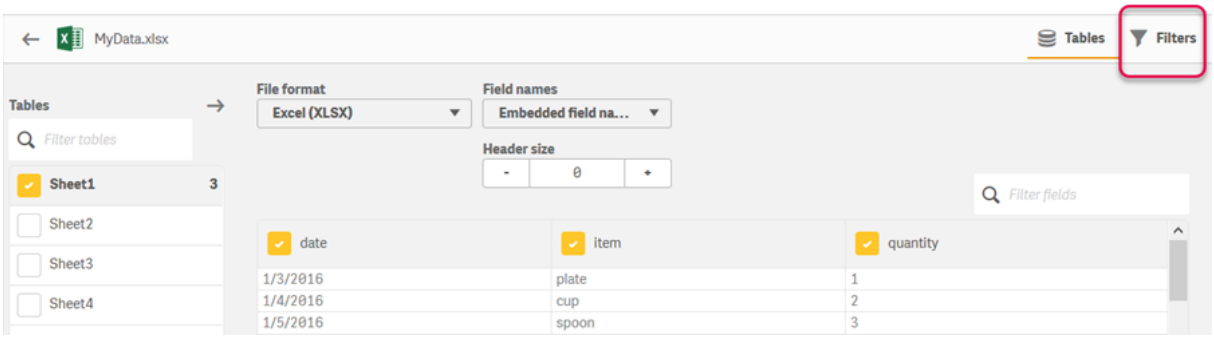
- **데이터 추가**의 이전 단계로 돌아가려면 뒤로 화살표를 클릭합니다.

파일에서 데이터 필터링

파일에서 데이터를 추가할 때 필터 조건을 만들 수 있습니다. 이를 통해 로드할 데이터 하위 집합을 선택할 수 있습니다. 이는 $40,000 이상의 매출과 같이 로드되는 데이터 양을 줄이거나 특정 데이터만 사용하려는 경우에 유용할 수 있습니다.

데이터 추가 단계에서 처음으로 파일에서 데이터를 추가하는 경우 **필터**를 클릭하여 필터 조건을 적용할 수 있습니다.

***필터** 버튼*



이후에 **데이터 관리자**에서 테이블을 클릭한 다음 **이 테이블 편집**을 클릭하여 조건을 변경할 수 있습니다. **소스에서 데이터 선택**을 클릭한 다음 **필터**를 클릭합니다.

앱에 데이터를 추가한 후 필터 조건을 적용한 다음 앱으로 데이터를 로드할 수 있습니다.

> *고급 데이터 조작이 필요한 경우 **데이터 로드 편집기**를 사용합니다. 자세한 내용은 초보자용 스크립팅 및 스크립팅 튜토리얼의 다음 단계를 참조하십시오.*

사용 가능한 필터 조건은 다음과 같습니다.

- =
- >
- <
- >=
- <=

데이터를 필터링할 때 다음을 고려합니다. 아래에 예가 나와 있습니다.

- 숫자, 날짜 또는 텍스트에 필터 조건을 적용할 수 있습니다.
- 와일드카드 문자는 지원되지 않습니다.
- 여러 조건을 적용할 수 있습니다. 그러나 동일한 필드에서 조건이 충돌하면 데이터가 반환되지 않을 수 있습니다.
- 조건은 텍스트 데이터에 사전순으로 적용됩니다. 조건은 대소문자를 구분합니다.

- 텍스트 데이터에 둘 이상의 문자를 사용할 수 있습니다. 예를 들어 *>ct*는 *>=cu*에서와 마찬가지로 단어 *cup*을 반환합니다. *>c*도 역시 *cup*을 반환한다는 점을 주목합니다.
- 둘 이상의 = 조건을 사용하는 경우 모두 true로 평가되어야 값이 반환됩니다. 그러나 동일한 필드에서 둘 이상의 = 조건을 사용하는 경우 true로 평가되는 모든 값이 반환됩니다.
- < 및 > 조건이 결합되면 모두 true로 평가되어야 값이 반환됩니다. 이들 조건이 =과 결합되는 경우 모든 조건이 true로 평가되어야 합니다.
- <= 및 >= 조건이 결합되면 모두 true로 평가되어야 값이 반환됩니다. 이들 조건이 =과 결합되는 경우 모든 조건이 true로 평가되어야 합니다.
- QVD 파일에서 날짜 필드에 대한 필터는 지원되지 않습니다.

예

이러한 예는 단일 필드(테이블의 한 열)에서 다음 값을 사용합니다. *cup*, *fork* 및 *knife*.

- 조건:
  - *=cup*
  - *=fork*
  - *=knife*
  - 반환: *cup*, *fork*, *knife*
  - 이 등호 조건은 true인 모든 값을 반환합니다.
- 조건:
  - *>b*
  - *<d*
  - 반환: *cup*
  - 문자 *c*는 *b*보다 크고 *d*보다 작습니다.
- 조건:
  - *<b*
  - *>d*
  - 반환: 값 없음
  - *b*보다 작고 *d*보다 큰 값이 없을 수 있습니다.
- 조건:
  - *=fork*
  - *>g*1
  - 반환: 값 없음
  - *fork*와 같고 *g*보다 큰 값이 없을 수 있습니다.

데이터 커넥터에서 데이터 필터링

데이터 필터링은 모든 커넥터에서 사용할 수 없습니다.

로드할 데이터 선택 단계에서 **데이터 필터링**을 선택하여 데이터 필터 표현식을 입력합니다. **데이터 필터링**을 선택하면 필터 표현식을 입력하는 텍스트 상자가 열립니다. 예:

매출 >= 40000

데이터 필터링은 *판매*와 같은 개별 필드에서 선택합니다. 이는 SQL WHERE 절로 작동합니다. WHERE 절에 사용된 연산자와 키워드는 대부분 **데이터 필터링**과 함께 사용할 수 있습니다. 유효한 연산자는 다음과 같습니다.

- =
- >
- >=
- <
- <=
- IN
- BETWEEN
- LIKE
- IS NULL
- IS NOT NULL

Qlik Sense는 **데이터 필터링**에 입력된 표현식으로부터 데이터 로드 스크립트에 WHERE 절을 작성합니다.

둘 이상의 필드에서 필터링하려는 경우와 같이 연산자를 결합하는 데 AND 연산자를 사용할 수 있습니다. 예:

Sales <= 30000 AND RegionID = 45

OR 연산자를 사용하여 두 조건 중 하나와 일치하는 데이터를 필터링할 수 있습니다. 예:

Name = 'Smith' OR Name = 'Jones'

IN 연산자에서 동일한 결과를 얻을 수 있습니다. IN 연산자는 여러 OR 조건을 사용하기 위한 약식 방법입니다. 예:

Name IN ('Smith', 'Jones')

3.12 테이블 편집

데이터 관리자 개요에서 앱에 추가한 테이블을 편집하여 테이블 이름을 변경하거나, 테이블을 다른 테이블에 연결하거나, 필드를 변환할 수 있습니다.

테이블을 편집하려면 **데이터 관리자**에서 테이블을 선택하고 를 클릭합니다. 테이블의 데이터에 대한 미리 보기와 함께 테이블 편집기가 표시됩니다. 각 필드에는 변환 옵션이 포함된 필드 메뉴가 있습니다. 를 클릭하여 필드 메뉴를 열 수 있습니다. 필드를 선택하면 데이터 프로파일링 카드 창이 표시되고 여기에 추가 변환 옵션뿐만 아니라 필드 데이터의 요약도 포함되어 있습니다.

로드된 모든 필드에 동일한 데이터가 있는 레코드가 데이터에 포함되어 있는 경우, 해당 데이터는 미리 보기 테이블에서 단일 레코드로 표시됩니다.

테이블 이름 바꾸기

데이터 관리자에서 테이블을 추가하면 데이터베이스 테이블, 데이터 파일 또는 Excel 워크시트 등의 이름에 기반하여 테이블에 기본 이름이 할당됩니다. 이름이 알기 어렵거나 적합하지 않은 경우 변경할 수 있습니다.

다음과 같이 하십시오.

1. 테이블 이름을 클릭합니다.
2. 테이블 이름을 편집합니다.
3. Enter를 누르거나 테이블 이름 바깥쪽을 클릭합니다.

이제 테이블 이름이 변경됩니다.

필드 이름 바꾸기

테이블의 필드 이름을 더 알기 쉬운 이름으로 바꿀 수 있습니다.

다음과 같이 하십시오.

1. 이름을 변경할 필드 이름을 클릭하거나 필드 메뉴에서 **이름 바꾸기**를 선택합니다.
2. 새 이름을 입력합니다.

   필드 이름은 고유해야 합니다. 여러 테이블에 동일한 이름의 필드가 있는 경우 Qlik Sense에서는 데이터 추가 시 필드 이름을 한정합니다. 즉, 테이블 이름이 접두사로 추가됩니다.

3. Enter 키를 누르거나 필드 바깥쪽을 클릭합니다.

이제 필드 이름이 변경되었습니다.

다른 테이블에 대한 연결 관리

필드 메뉴의 **연결**을 사용하여 다른 테이블의 필드에 대한 사용자 지정 연결을 만들 수 있습니다.

일반적으로 권장 사항을 따르는 대신 사용자 지정 연결을 만들어야 하는 가장 일반적인 경우는 다음과 같습니다.

- 테이블과 연결할 필드를 알지만 이 테이블 쌍에 대한 점수가 너무 낮아서 권장 사항 목록에 표시할 수 없습니다.
  각 테이블의 단일 필드에 기반하여 연결을 만듭니다.
- 테이블에 둘 이상의 공통 필드가 포함되어 있으며 연결을 형성하는 데 이들 필드를 사용해야 합니다.
  복합 키를 만듭니다.

많은 경우, **연결** 보기에서 연결을 관리하는 것이 훨씬 간편합니다.

필드 유형 및 표시 형식 변경

데이터가 추가되면 Qlik Sense에서 각 필드의 필드 유형을 해석합니다. 현재 다음과 같은 필드 유형이 지원됩니다.

- **일반**
- **날짜**
- **타임스탬프**
- **지리 데이터**

데이터가 올바르게 해석되지 않으면 필드 유형을 조정할 수 있습니다. 입력 및 날짜 또는 타임스탬프 필드의 표시 형식도 변경할 수 있습니다.

우편 지역과 같이 이름 또는 코드 형식의 지리 정보가 포함된 필드는 **지리 데이터** 필드로 지정되지 않는 경우 매핑에 사용할 수 없습니다.

분석에서 필드 숨기기

시트 보기 또는 통찰력에서 필드를 숨겨서 **데이터 관리자** 및 **데이터 로드 편집기**에서만 사용 가능하도록 할 수 있습니다. 예를 들어 다른 필드를 계산하는 데만 사용되는 필드가 있을 수 있습니다. 이러한 필드를 숨기면 시트의 자산 패널이나 통찰력에서는 사용할 수 없지만 **데이터 관리자**에서는 계속 사용할 수 있습니다.

자세한 내용은 *분석에서 필드 숨기기 (page 55)*을 참조하십시오.

데이터를 로드하기 전에 테이블 필드 데이터 평가

요약 데이터 프로파일링 카드를 사용하여 데이터를 로드하기 전에 null 값 및 이상값 등의 잠재적인 품질 문제에 대해 테이블의 데이터를 검사할 수 있습니다. **요약** 카드는 필드를 차원, 측정값 또는 임시 필드로 분류하여 각각의 경우에 대해 서로 다른 데이터 요약을 제공하고 다른 데이터 프로파일링 카드에서 서로 다른 변환 옵션을 사용합니다. **요약** 카드에서 측정값으로 설정된 필드는 **버킷** 카드를 사용하여 그룹화할 수 있습니다. **요약** 카드에서 차원으로 설정된 필드에는 **순서** 카드에서 적용된 사용자 지정 순서가 있을 수 있습니다. 여러 범주로 분류될 수 있는 필드의 경우 필드에 대한 각각의 가능한 범주의 요약 간을 전환할 수 있습니다.

자세한 내용은 *데이터를 로드하기 전에 테이블 필드 데이터 평가 (page 55)*를 참조하십시오.

테이블의 필드 값 바꾸기

바꾸기 데이터 프로파일링 카드를 사용하여 필드의 값을 바꿀 수 있습니다. **바꾸기** 카드를 사용하면 필드에서 하나 이상의 값을 선택한 다음 다른 값으로 바꿀 수 있습니다. 예를 들어 전체 및 약어 형식의 국가 이름이 포함된 데이터 셋에서 해당 이름을 단일 공유 값으로 대체할 수 있습니다.

자세한 내용은 *테이블의 필드 값 바꾸기 (page 57)*를 참조하십시오.

테이블에서 필드 값을 Null로 설정

Null 설정 데이터 프로파일링 카드를 사용하여 차원 필드에서 고유 값을 Null 값으로 처리하도록 설정할 수 있습니다. **Null 설정** 카드를 사용하면 테이블 필드에서 값을 선택한 다음 수동으로 Null로 설정할 수 있습니다. 예를 들어 데이터 세트가 X와 같은 문자를 사용하여 Null을 나타내는 경우 **Null 설정** 카드를 사용하여 Qlik Sense을 활성화하면 해당 값을 Null로 처리할 수 있습니다. **Null 설정** 카드를 사용하여 원하지 않는 값을 Null로 설정함으로써 데이터 셋을 정리할 수도 있습니다.

자세한 내용은 *테이블에서 필드 값을 Null로 설정 (page 59)*를 참조하십시오.

필드 값에 사용자 지정 순서 설정

데이터에 따라 사전순이나 숫자순이 아닌 다른 순서로 차원 값을 표시하면 훨씬 의미가 있을 수 있습니다. **요약** 데이터 프로파일링 카드에서 차원으로 설정된 필드에는 **순서** 데이터 프로파일링 카드를 사용하여 적용된 데이터의 사용자 지정 순서가 있을 수 있어 시각화에서 필드 데이터의 기본 구성을 설정할 수 있습니다.

자세한 내용은 *차원 값 순서 사용자 지정 (page 60)*를 참조하십시오.

테이블에서 필드 분할

분할 데이터 프로파일링 카드를 사용하여 기존 필드의 정보를 새로운 필드로 추출할 수 있습니다. **분할** 카드를 사용하면 필드의 콘텐츠를 여러 필드로 분할할 수 있습니다. 예를 들어 주소가 포함된 필드를 분할하기 위해 우편 번호를 추출할 수 있습니다. 이를 통해 기존 데이터의 관련 섹션이 포함된 새 필드를 빠르게 만들 수 있습니다.

자세한 내용은 *테이블에서 필드 분할 (page 61)*를 참조하십시오.

측정값 데이터를 범위로 그룹화

버킷 데이터 프로파일링 카드를 사용하여 테이블 측정값 필드의 값을 범위로 그룹화할 수 있습니다. **버킷** 카드를 사용하면 사용자 정의 버킷에서 필드의 값을 그룹화하여 테이블에 추가되는 새 필드를 만들 수 있습니다. 예를 들어 나이를 연령 범위로 그룹화하여 시각화에서 차원으로 사용할 수 있습니다.

필드 변환 세부 정보 보기

필드에서 수행되는 현재 작업 및 변환과 해당 순서를 **세부 정보** 대화 상자에서 볼 수 있습니다. **세부 정보**를 사용하여 필드를 만든 곳, 필드의 변경 사항 및 변환이 적용된 순서를 파악할 수 있습니다.

자세한 내용은 *데이터 관리자에서 테이블 및 필드 변환 세부 정보 보기 (page 77)*를 참조하십시오.

크로스탭 데이터 피벗 해제

크로스탭 형식의 데이터를 로드한 경우 일반적으로 테이블의 피벗을 해제하는 것이 가장 좋습니다. 즉, 테이블의 일부를 행으로 바꿉니다. 그러면 데이터로 작업하고 다른 데이터 테이블에 대한 연결을 만드는 것이 용이해집니다.

Year	Europe	RoW
2007	234	567
2008	345	534

Year	Region	Sales
2007	Europe	234
2007	RoW	567
2008	Europe	345
2008	RoW	534

자세한 내용은 *데이터 관리자에서 크로스탭 데이터 피벗 해제 (page 66)*을 참조하십시오.

데이터 소스에서 테이블 업데이트

데이터 소스의 필드 선택을 변경하고 싶을 수 있습니다. 예를 들어, 제외되었던 필드를 추가하거나 추가된 필드로 데이터 소스가 업데이트되었을 수 있습니다. 이런 경우 데이터 소스에서 테이블을 업데이트할 수 있습니다. 테이블을 **수동 입력**으로 만든 경우 새 행 및 열을 추가하고 테이블 데이터를 추가, 편집 또는 삭제할 수 있습니다. 자세한 내용은 *에서 수동으로 데이터 추가Qlik Sense (page 25)*를 참조하십시오.

다음과 같이 하십시오.

1. **소스에서 데이터 선택**을 클릭합니다.
   현재 선택이 포함된 데이터 선택 마법사가 열립니다.
2. 선택 사항에서 필요한 변경을 수행합니다.
3. 데이터 프로파일링이 활성화된 상태에서 **데이터 추가**를 클릭합니다.

이제 테이블이 선택 사항에 따라 필드로 업데이트되었습니다.

계산 필드 추가

로드된 필드 데이터를 조정하거나 변환해야 하는 경우가 많이 있습니다. 예를 들어, 이름과 성을 전체 이름으로 연결하거나, 제품 번호의 일부를 추출하거나, 데이터 형식을 변환하거나, 두 숫자를 곱해야 할 수 있습니다.

계산 필드를 추가하여 이러한 다양한 경우를 관리할 수 있습니다. 계산 필드는 표현식을 사용하여 필드의 결과를 정의합니다. 표현식에서 함수, 필드 및 연산자를 사용할 수 있습니다. 편집 중인 테이블의 필드만 참조할 수 있습니다.

테이블 정렬

테이블 편집 시 데이터의 개요를 더 잘 파악할 수 있도록 특정 필드에 기반하여 테이블을 정렬할 수 있습니다. 한 번에 한 필드에서만 정렬할 수 있습니다.

다음과 같이 하십시오.

- 필드 메뉴에서 **정렬**을 선택합니다.

이제 테이블 데이터가 이 필드에 따라 오름차순으로 정렬됩니다. 내림차순으로 정렬하고 싶으면 **정렬**을 다시 선택합니다.

정렬 순서는 로드된 앱 데이터에서 유지되지 않습니다.

작업 실행 취소 및 재실행

↷ 및 ↶을 클릭하여 테이블 편집 작업을 실행 취소 및 재실행할 수 있습니다.

테이블 편집기를 닫으면 실행 취소/재실행 기록이 지워집니다.

테이블 편집기에서 데이터 연결

데이터 관리자 테이블 편집기의 필드 메뉴에서 **연결**을 사용하여 다른 테이블의 필드에 대한 사용자 지정 연결을 만들 수 있습니다.

많은 경우, **연결** 보기에서 연결을 관리하는 것이 훨씬 간편합니다.

일반적으로 권장 사항을 따르는 대신 사용자 지정 연결을 만들어야 하는 가장 일반적인 경우는 다음과 같습니다.

- 테이블과 연결할 필드를 알지만 이 테이블 쌍에 대한 점수가 너무 낮아서 권장 사항 목록에 표시할 수 없습니다.
  각 테이블의 단일 필드에 기반하여 연결을 만듭니다.
- 테이블에 둘 이상의 공통 필드가 포함되어 있으며 연결을 형성하는 데 이들 필드를 사용해야 합니다.
  복합 키를 만듭니다.

단일 필드를 사용하여 연결 만들기

두 테이블이 관련 데이터를 포함하고 있지만 연결이 권장되는 것으로 표시되지 않는 경우, 테이블 편집기에서 사용자 지정 연결을 정의할 수 있습니다. 이렇게 하여 테이블을 연결하는 키 필드를 만듭니다.

다음과 같이 하십시오.

1. 데이터 관리자 개요에서 연결하려는 테이블 중 하나에 있는 을 클릭합니다.
   테이블 편집기가 열립니다.
2. 키 필드에서 사용할 필드의 필드 메뉴에서 **연결**을 선택합니다.
   왼쪽 테이블에서 선택한 필드의 미리 보기와 함께 **테이블 연결** 편집기가 열립니다. 이제 오른쪽 테이블에서 이와 연결할 필드를 선택해야 합니다.
3. **테이블 선택**을 클릭하고 연결할 테이블을 선택합니다.
4. 아이콘 을 클릭하고 연결할 필드를 선택합니다.
   선택한 필드의 미리 보기 데이터가 오른쪽 테이블에 표시됩니다. 이제 왼쪽 테이블과 오른쪽 테이블을 비교하여 일치 데이터가 포함되어 있는지 확인할 수 있습니다. 보다 쉽게 비교할 수 있도록 을 사용하여 테이블에서 검색할 수 있습니다.
5. 만들 키 필드의 이름을 **이름**에 입력합니다.
   두 테이블의 기존 필드와 동일한 이름을 사용할 수는 없습니다.
6. **연결**을 클릭합니다.

이제 테이블이 키 필드를 사용하여, 선택된 두 필드에 의해 연결됩니다. 이는 으로 표시됩니다. 아이콘 을 클릭하여 연결을 편집하거나 해제하는 옵션을 표시합니다.

복합 키 만들기

두 테이블에 연결을 생성할 공통 필드가 두 개 이상 있으면 Qlik Sense에서 가상 키를 만들어 연결을 처리합니다. 이 문제를 해결하는 가장 좋은 방법은 복합 키를 만드는 것입니다. 이 작업은 연결되어야 하는 모든 필드가 포함된 사용자 지정 연결을 만들어 수행할 수 있습니다.

다음과 같이 하십시오.

1. 데이터 관리자 개요에서 연결하려는 테이블 중 하나에 있는 을 클릭합니다.
   테이블 편집기가 열립니다.
2. 복합 키 필드에 포함할 필드 중 하나의 필드 메뉴에서 **연결**을 선택합니다.
   왼쪽 테이블에서 선택한 필드의 미리 보기와 함께 **테이블 연결** 편집기가 열립니다.
3. 아이콘 을 클릭하여 복합 키 필드에 포함할 다른 필드를 추가합니다.
   미리 보기가 복합 키 데이터로 업데이트됩니다.
   이제 오른쪽 테이블에서 이와 연결할 필드를 선택해야 합니다.
4. **테이블 선택**을 클릭하고 복합 키 필드에 포함할 필드를 선택합니다.
5. 아이콘 을 클릭하고 연결할 필드를 선택합니다. 왼쪽 테이블과 같은 순서로 선택해야 합니다.
   키의 데이터를 보다 쉽게 해석할 수 있도록 구분 기호를 추가할 수도 있습니다.
   선택한 필드의 미리 보기 데이터가 오른쪽 테이블에 표시됩니다.
   이제 왼쪽 테이블과 오른쪽 테이블을 비교하여 일치 데이터가 포함되어 있는지 확인할 수 있습니다.
   보다 쉽게 비교할 수 있도록 을 사용하여 테이블에서 검색할 수 있습니다.
6. 만들 키 필드의 이름을 **이름**에 입력합니다.
7. **연결**을 클릭합니다.

이제 테이블이 복합 키 필드를 사용하여, 선택된 필드에 의해 연결됩니다.

제한 사항

복합 키를 사용하는 데에는 일부 제한 사항이 있습니다.

- 컨케트네이트된 테이블에서는 복합 키를 만들 수 없습니다.
- 복합 키에서 계산 필드를 사용하면 계산 필드 표현식이 복합 키 표현식에서 확장됩니다. 계산 필드에 대한 참조는 없으며, 이는 계산 필드를 편집하는 경우 복합 키가 업데이트되지 않음을 의미합니다.

연결 편집

연결을 편집하여 이름을 바꾸거나 연결 필드를 변경할 수 있습니다.

다음과 같이 하십시오.

1. 아이콘 을 클릭하여 연결 메뉴를 표시합니다.
2. 아이콘 을 클릭하여 연결을 편집합니다.

테이블 연결 편집기가 열리면 연결의 이름을 바꾸거나 연결 필드를 변경할 수 있습니다.

연결 해제

필요 없는 두 테이블 사이의 연결을 만들었으면 연결 해제할 수 있습니다.

다음과 같이 하십시오.

1. 아이콘 을 클릭하여 연결 메뉴를 표시합니다.
2. 아이콘 을 클릭하여 연결을 해제합니다.

이제 테이블이 더 이상 연결되어 있지 않습니다.

계산 필드 사용

로드된 필드 데이터를 조정하거나 변환해야 하는 경우가 많이 있습니다. 예를 들어, 이름과 성을 전체 이름으로 연결하거나, 제품 번호의 일부를 추출하거나, 데이터 형식을 변환하거나, 두 숫자를 곱해야 할 수 있습니다.

계산 필드를 추가하여 이러한 다양한 경우를 관리할 수 있습니다. 계산 필드는 표현식을 사용하여 필드의 결과를 정의합니다. 표현식에서 함수, 필드 및 연산자를 사용할 수 있습니다. 편집 중인 테이블의 필드만 참조할 수 있습니다. 계산 필드에서 다른 계산 필드를 참조할 수 있습니다.

데이터 관리자의 테이블 편집기에서 계산 필드를 추가 및 편집할 수 있습니다.

계산 필드 추가

다음과 같이 하십시오.

1. **필드 추가**를 클릭하고 **계산 필드**를 선택합니다.
   계산 필드 추가를 위한 편집기가 열립니다.
2. **이름**에 계산 필드의 이름을 입력합니다.
3. **표현식**에서 계산 필드의 표현식을 정의합니다. 이 작업은 두 가지 방법으로 수행할 수 있습니다.
   - *fx*(**함수**), 目(**필드**) 및 ⊞(**연산자**) 목록을 사용하여 항목을 선택하고 표현식에 삽입합니다.
     선택한 항목은 **표현식**의 커서 위치에 삽입됩니다.
   - **표현식**에 계산 필드에 대한 표현식을 입력합니다.
     입력 시 함수 구문뿐만 아니라 제안된 함수 및 필드에 대해 지원을 받을 수 있습니다.

   계산 필드의 결과 예가 **미리 보기**에 표시됩니다.
4. **만들기**를 클릭하여 계산 필드를 만들고 계산 필드 편집기를 닫습니다.

계산 필드 편집

계산 필드의 이름을 변경하거나 표현식을 편집할 수 있습니다.

다음과 같이 하십시오.

1. 필드 이름 옆의 드롭다운 메뉴에서 **편집**을 선택합니다.
   계산 필드 업데이트를 위한 편집기가 열립니다.
2. 변경하려는 경우, **이름**에서 계산 필드의 이름을 편집합니다.
3. 계산 필드의 표현식을 편집합니다.
4. **업데이트**를 클릭하여 계산 필드를 업데이트하고 계산 필드 편집기를 닫습니다.

계산 필드 식에 사용할 수 있는 함수

계산 필드 식을 만들 때는 여기에 나열된 함수를 사용할 수 있습니다. 이는 데이터 로드 스크립트에서 사용할 수 있는 표현식의 하위 집합입니다. 표현식은 여러 레코드에서 데이터의 어떠한 집계 결과도 될 수 없으며, 다른 레코드의 데이터를 참조하기 위해 인터 레코드 함수를 사용할 수도 없습니다.

계산 필드 식에 사용할 수 있는 문자열 함수

이 함수는 텍스트 문자열 서식에서 데이터를 수정하거나 추출하는 데 사용할 수 있습니다.

문자열 함수

함수	설명
Capitalize	**Capitalize()**는 모든 단어의 첫 글자가 대문자인 문자열을 반환합니다.
Chr	**Chr()**은 입력된 정수에 해당하는 Unicode 문자를 반환합니다.
FindOneOf	**FindOneOf()**는 제공된 문자 집합의 문자 발견 위치를 찾기 위해 문자열을 검색합니다. 세 번째 인수(1보다 큰 값으로)를 지정하지 않으면 검색 집합의 문자에 대한 첫 번째 발견 위치가 반환됩니다. 일치 항목이 발견되지 않으면 **0**이 반환됩니다.
Index	**Index()**는 제공된 부분 문자열의 n번째 항목의 시작 위치를 찾기 위해 문자열을 검색합니다. 세 번째 선택적 인수가 n 값을 제공하며, 생략 시 1입니다. 음수 값은 문자열 끝부터 검색합니다. 문자열 내 위치는 **1**부터 번호가 매겨집니다.
KeepChar	**KeepChar()**은 첫 번째 문자열 'text'로 이루어진 문자열 중 두 번째 문자열 "keep_chars"에 포함되지 않는 문자를 뺀 문자열을 반환합니다.
왼쪽	**Left()**는 입력 문자열의 첫 번째(가장 왼쪽) 문자로 구성된 문자열을 반환하며 문자 수는 두 번째 인수로 결정됩니다.
Len	**Len()**은 입력 문자열의 길이를 반환합니다.
Lower	**Lower()**는 입력 문자열의 모든 문자를 소문자로 변환합니다.
LTrim	**LTrim()**은 모든 선행 공백이 제거된 입력 문자열을 반환합니다.
Mid	**Mid()**는 두 번째 인수 'start'로 정의된 문자의 위치에서 시작되고 세 번째 인수 'count'로 정의된 문자 수를 반환하는 입력 문자열의 일부를 반환합니다. 'count'를 생략하면 입력 문자열의 나머지가 반환됩니다. 입력 문자열의 첫 번째 문자는 번호가 1로 지정됩니다.
Ord	**Ord()**는 입력 문자열의 첫 번째 문자에 대한 Unicode 코드 포인트 번호를 반환합니다.
PurgeChar	**PurgeChar()**은 두 번째 인수('remove_chars')에 나타나는 문자를 제외하고, 입력 문자열('text')에 포함된 문자로 구성된 문자열을 반환합니다.
Repeat	**Repeat()**는 두 번째 인수로 정의된 횟수 만큼 반복된 입력 문자열로 구성된 문자열을 만듭니다.
Replace	**Replace()**는 입력 문자열 내에서 특정 부분 문자열의 모든 발견 항목을 다른 부분 문자열로 대체한 후의 문자열을 반환합니다. 이 함수는 비재귀적이며 왼쪽에서 오른쪽으로 작동합니다.
오른쪽	**Right()**는 입력 문자열의 마지막(가장 오른쪽) 문자로 구성된 문자열을 반환합니다. 문자 수는 두 번째 인수로 결정됩니다.
RTrim	**RTrim()**은 모든 후행 공백이 제거된 입력 문자열을 반환합니다.
SubStringCount	**SubStringCount()**는 입력 문자열 텍스트에 지정된 부분 문자열의 발생 횟수를 반환합니다. 일치하는 항목이 없으면 **0**이 반환됩니다.
TextBetween	**TextBetween()**은 구분 기호로 지정된 문자 사이에서 발견되는 입력 문자열의 텍스트를 반환합니다.

함수	설명
Trim	**Trim()**은 모든 선행 및 후행 공백이 제거된 입력 문자열을 반환합니다.
Upper	**Upper()**는 표현식의 모든 텍스트 문자에 대해 입력 문자열의 모든 문자를 대문자로 변환합니다. 숫자와 기호는 무시됩니다.

계산 필드 식에 사용할 수 있는 날짜 함수

Qlik Sense 날짜 및 시간 함수는 날짜와 시간 값을 변환하는 데 사용됩니다.

이 함수는 1899년 12월 30일 이후의 일 수와 동일한 날짜-시간 일련 번호를 기준으로 합니다. 정수 값은 날짜를 나타내고 소수 값은 해당 날짜의 시간을 나타냅니다.

Qlik Sense는 인수의 숫자 값을 사용하므로, 숫자는 날짜 또는 시간으로 서식이 지정되지 않았을 때도 인수로 유효합니다. 인수가 문자열인 경우와 같이 숫자 값에 해당하지 않는 경우 Qlik Sense는 날짜 및 시간 환경 변수에 따라 문자열을 해석하려고 시도합니다.

인수에서 사용된 날짜 서식이 **DateFormat** 시스템 변수로 설정된 서식과 일치하지 않을 경우 Qlik Sense에서 날짜를 올바르게 해석할 수 없게 됩니다. 이 문제를 해결하려면 설정을 변경하거나 해석 함수를 사용하십시오.

날짜 함수

함수	설명
addmonths	이 함수는 **startdate**를 기준으로 **n**개월 후의 날짜, 또는 **n**이 음수일 경우는 **startdate**를 기준으로 **n**개월 전의 날짜를 반환합니다.
addyears	이 함수는 **startdate**를 기준으로 **n**년 후의 날짜, 또는 **n**이 음수일 경우는 **startdate**를 기준으로 **n**년 전의 날짜를 반환합니다.
age	**age** 함수는 **date_of_birth**에 태어난 사람이 **timestamp**의 시간에 (만으로) 몇 살인지 반환합니다.
converttolocaltime	UTC 또는 GMT 타임스탬프를 이중 값 형태의 현지 시간으로 변환합니다. place는 전 세계 여러 도시, 장소, 표준 시간대로 지정할 수 있습니다.
day	이 함수는 **expression**의 분위수가 표준 숫자 해석에 따라 날짜로 해석될 경우 일을 나타내는 정수를 반환합니다.
dayend	이 함수는 **time**에 포함된 날의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **TimestampFormat**입니다.
daylightsaving	UTC 또는 GMT 타임스탬프를 이중 값 형태의 현지 시간으로 변환합니다. place는 전 세계 여러 도시, 장소, 표준 시간대로 지정할 수 있습니다.
dayname	이 함수는 **time**을 포함한 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 날짜를 표시하는 값을 반환합니다.
daynumberofquarter	UTC 또는 GMT 타임스탬프를 이중 값 형태의 현지 시간으로 변환합니다. place는 전 세계 여러 도시, 장소, 표준 시간대로 지정할 수 있습니다.

함수	설명
daynumberofyear	이 함수는 타임스탬프가 속하는 연도의 일수를 계산합니다. 해당 연도의 첫 날의 첫 번째 밀리초로부터 계산되지만 첫 번째 월은 오프셋 지정할 수 있습니다.
daystart	이 함수는 **time** 인수에 포함된 날의 첫 번째 밀리초가 있는 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **TimestampFormat**입니다.
firstworkdate	**firstworkdate** 함수는 선택적으로 나열된 공휴일을 고려하여 **no_of_workdays**(월요일 ~ 금요일)가 **end_date** 이전에 끝나게 되는 가장 최근의 시작 날짜를 반환합니다. **end_date** 및 **holiday**은 유효한 날짜 또는 타임스태프여야 합니다.
GMT	이 함수는 지역 설정에서 유추한 현재 Greenwich Mean Time을 반환합니다.
hour	이 함수는 **expression**의 분위수가 표준 숫자 해석에 따라 시간으로 해석될 경우 시간을 나타내는 정수를 반환합니다.
inday	**timestamp**가 **base_timestamp**를 포함하는 날에 속할 경우 이 함수는 True를 반환합니다.
indaytotime	이 함수는 **timestamp**가 **base_timestamp**의 정확한 밀리초까지 포함하여 **base_timestamp**를 포함한 날의 일부에 속할 경우 True를 반환합니다.
inlunarweek	This function finds if **timestamp** lies inside the lunar week containing **base_date**. Lunar weeks in Qlik Sense are defined by counting January 1 as the first day of the week and, apart from the final week of the year, will contain exactly seven days.이 함수는 **timestamp**가 **base_date**를 포함하는 날에 속하는지 확인합니다. Qlik Sense에서 음력 주는 1월 1일을 주의 첫 날로 계산하여 정의됩니다.
inlunarweektodate	This function finds if **timestamp** lies inside the part of the lunar week up to and including the last millisecond of **base_date**. Lunar weeks in Qlik Sense are defined by counting January 1 as the first day of the week and, apart from the final week of the year, will contain exactly seven days.이 함수는 **timestamp**가 **base_date**의 마지막 밀리초까지 포함하여 음력 주의 일부에 속하는지 확인합니다. Qlik Sense에서 음력 주는 1월 1일을 주의 첫 날로 계산하여 정의됩니다.
inmonth	**timestamp**가 **base_date**를 포함하는 월에 속할 경우 이 함수는 True를 반환합니다.
inmonths	이 함수는 타임스탬프가 기준일로 동일한 월, 2개월, 분기, 3개월 또는 6개월 내에 속하는지 파악합니다. 또한 타임스탬프가 이전 기간 또는 다음 기간 내에 속하는지도 알아낼 수 있습니다.
inmonthstodate	이 함수는 타임스탬프가 `base_date`의 마지막 밀리초까지 포함하여 월, 2개월, 분기, 3개월 또는 6개월 기간의 일부에 속하는지 파악합니다. 또한 타임스탬프가 이전 기간 또는 다음 기간 내에 속하는지도 알아낼 수 있습니다.

함수	설명
inmonthtodate	이 함수는 **date**가 **basedate**의 마지막 밀리초까지 포함하여 **basedate**를 포함한 월의 일부에 속할 경우True를 반환합니다.
inquarter	**timestamp**가 **base_date**를 포함하는 분기에 속할 경우 이 함수는 True를 반환합니다.
inquartertodate	이 함수는 **timestamp**가 **base_date**의 마지막 밀리초까지 포함하여 **base_date**를 포함한 분기의 일부에 속할 경우 True를 반환합니다.
inweek	**timestamp**가 **base_date**를 포함하는 주에 속할 경우 이 함수는 True를 반환합니다.
inweektodate	이 함수는 **timestamp**가 **base_date**의 마지막 밀리초까지 포함하여 **base_date**를 포함한 주의 일부에 속할 경우 True를 반환합니다.
inyear	**timestamp**가 **base_date**를 포함하는 연도에 속할 경우 이 함수는 True를 반환합니다.
inyeartodate	이 함수는 **timestamp**가 **base_date**의 마지막 밀리초까지 포함하여 **base_date**를 포함한 연도의 일부에 속할 경우 True를 반환합니다.
lastworkdate	**lastworkdate** 함수는 선택적으로 나열된 **holiday**를 고려하여 **start_date**에 시작하는 경우 **no_of_workdays**(월요일-금요일)가 끝나는 가장 빠른 끝 날짜를 반환합니다. **start_date** 및 **holiday**는 유효한 날짜 또는 타임스탬프여야 합니다.
localtime	이 함수는 지정된 표준 시간대에 대한 현재 시간의 타임스탬프를 반환합니다.
lunarweekend	이 함수는 **date**를 포함하는 음력 주의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. Qlik Sense에서 음력 주는 1월 1일을 주의 첫 날로 계산하여 정의됩니다. This function returns a value corresponding to a timestamp of the last millisecond of the last day of the lunar week containing **date**. Lunar weeks in Qlik Sense are defined by counting January 1 as the first day of the week and, apart from the final week of the year, will contain exactly seven days.
lunarweekname	이 함수는 **date**를 포함한 음력 주의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 음력 주차와 연도를 보여주는 표시 값을 반환합니다. Qlik Sense에서 음력 주는 1월 1일을 주의 첫 날로 계산하여 정의됩니다.This function returns a display value showing the year and lunar week number corresponding to a timestamp of the first millisecond of the first day of the lunar week containing **date**. Lunar weeks in Qlik Sense are defined by counting January 1as the first day of the week and, apart from the final week of the year, will contain exactly seven days.

함수	설명
lunarweekstart	이 함수는 **date**를 포함하는 음력 주의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. Qlik Sense에서 음력 주는 1월 1일을 주의 첫 날로 계산하여 정의됩니다.This function returns a value corresponding to a timestamp of the first millisecond of the first day of the lunar week containing **date**. Lunar weeks in Qlik Sense are defined by counting January 1 as the first day of the week and, apart from the final week of the year, will contain exactly seven days.
makedate	이 함수는 연도 **YYYY**, 월 **MM**, 일 **DD**에서 계산한 날짜를 반환합니다.
maketime	이 함수는 시간 **hh**, 분 **mm**, 초 **ss**에서 계산한 시간을 반환합니다.
makeweekdate	이 함수는 연도 **YYYY**, 주 **WW**, 요일 **D**에서 계산한 날짜를 반환합니다.
minute	이 함수는 **expression**의 분위수가 표준 숫자 해석에 따라 시간으로 해석될 경우 분을 나타내는 정수를 반환합니다.
month	이 함수는 환경 변수 **MonthNames**로 정의되고 1~12 사이의 정수인 월 이름이 포함된 이중 값을 반환합니다. 월은 표준 숫자 해석에 따라 표현식의 날짜 해석을 통해 계산됩니다.
monthend	이 함수는 `date`를 포함하는 월의 마지막 날의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 `DateFormat` 입니다.
monthname	이 함수는 해당 월의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 월(**MonthNames** 스크립트 변수에 따라 서식 지정) 및 연도를 보여주는 표시 값을 반환합니다.
monthsend	이 함수는 기준일을 포함하는 월, 2개월, 분기, 3개월 또는 6개월의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 또한 이전 기간 또는 다음 기간에 대한 타임스탬프를 찾을 수도 있습니다.
monthsname	이 함수는 연도뿐 아니라 기간의 월 범위(**MonthNames** 스크립트 변수에 따라 서식 지정)를 나타내는 표시 값도 반환합니다. 기본 숫자 값은 기준일을 포함하는 월, 2개월, 분기, 3개월 또는 6개월의 첫 번째 밀리초의 타임스탬프에 해당합니다.
monthsstart	이 함수는 기준일을 포함하는 월, 2개월, 분기, 3개월 또는 6개월의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 또한 이전 기간 또는 다음 기간에 대한 타임스탬프를 찾을 수도 있습니다.
monthstart	이 함수는 **date**를 포함하는 월의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat** 입니다.
networkdays	**networkdays** 함수는 선택적으로 나열된 **holiday**를 고려하여 **start_date**(포함)와 **end_date**(포함) 사이의 근무일(월요일 ~ 금요일)의 수를 반환합니다.

함수	설명
now	This function returns a timestamp of the current time. The function returns values in the **TimeStamp** system variable format. The default **timer_mode** value is 1.이 함수는 시스템 시계에서 현재 시간의 타임스탬프를 반환합니다. 기본값은 1입니다.
quarterend	이 함수는 **date**를 포함하는 분기의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.
quartername	이 함수는 해당 분기의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 분기의 월(**MonthNames** 스크립트 변수에 따라 서식 지정) 및 연도를 보여주는 표시 값을 반환합니다.
quarterstart	이 함수는 **date**를 포함하는 분기의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.
second	이 함수는 **expression**의 분위수가 표준 숫자 해석에 따라 시간으로 해석될 경우 초를 나타내는 정수를 반환합니다.
timezone	이 함수는 Windows에 정의된 현재 표준 시간대의 이름을 반환합니다.
today	이 함수는 시스템 시계에서 현재 날짜를 반환합니다.This function returns the current date. The function returns values in the `DateFormat` system variable format.
UTC	현재 Coordinated Universal Time을 반환합니다.
week	이 함수는 ISO 8601에 따른 주 번호를 나타내는 정수를 반환합니다. 주 번호는 표준 숫자 해석에 따라 표현식의 날짜 해석을 통해 계산됩니다.
weekday	이 함수는 다음을 포함하는 이중 값을 반환합니다. 환경 변수 **DayNames**로 정의한 날짜 이름. 주의 명목상 이름(0~6)에 해당하는 0~6 사이의 정수.
weekend	이 함수는 **date**를 포함한 주의 마지막 날짜(일요일)의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.
weekname	이 함수는 **date**를 포함한 주의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 연도 및 주차를 보여주는 값을 반환합니다.
weekstart	이 함수는 **date**를 포함하는 주의 첫 번째 날(월요일)의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.
weekyear	이 함수는 ISO 8601에 따라 주 번호가 속하는 연도를 반환합니다. 주차 범위는 1과 약 52 사이입니다.
year	이 함수는 **expression**이 표준 숫자 해석에 따라 날짜로 해석될 경우 연도를 나타내는 정수를 반환합니다.
yearend	이 함수는 **date**를 포함하는 연도의 마지막 날의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.

함수	설명
yearname	이 함수는 **date**를 포함한 연도의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값을 사용하여 네 자리 연도를 표시 값으로 반환합니다.
yearstart	이 함수는 **date**를 포함하는 연도의 첫 번째 날의 시작에 해당하는 타임스탬프를 반환합니다. 기본 출력 형식은 스크립트에 설정된 **DateFormat**입니다.
yeartodate	이 함수는 스크립트를 마지막으로 로드한 날짜의 연도 내에 입력 타임스탬프가 포함되는지 파악하고, 포함되는 경우 True, 포함되지 않는 경우 False를 반환합니다.

계산 필드 식에 사용할 수 있는 형식 지정 및 해석 함수

서식 지정 함수는 입력 표현식의 숫자 값을 사용하여 이를 텍스트 값으로 변환합니다. 반면, 해석 함수는 반대로 작업을 수행합니다. 결과 숫자의 서식을 지정하는 문자열 표현식을 숫자로 평가합니다. 두 경우 모두 출력 값은 텍스트 값과 숫자 값의 이중 값입니다.

예를 들어, **Date**와 **Date#** 함수 간의 출력 차이를 생각해 보십시오.

Date 및 Date# 함수

함수	출력(텍스트)	출력(숫자)
`Date#('20140831', 'YYYYMMDD')`	20140831	41882
`Date(41882, 'YYYY.MM.DD')`	2014.08.31	41882

이 함수는 서식이 Qlik Sense의 날짜 서식 설정과 일치하지 않으므로 날짜로 해석되지 않는 날짜 필드가 데이터에 포함되어 있을 때 유용합니다. 이 경우, 함수를 중첩하는 데 유용할 수 있습니다.

```
Date(Date#(DateInput, 'YYYYMMDD'),'YYYY.MM.DD')
```

입력 서식 YYYYMMDD에 따라 DateInput 필드를 해석한 후, 사용하려는 서식인 YYYY.MM.DD로 그 결과를 반환합니다.

서식 지정 및 해석 함수

함수	설명
Date	**Date()**는 데이터 로드 스크립트, 운영 체제 또는 서식 문자열(지정된 경우)로 설정된 서식을 사용하여 표현식의 서식을 날짜로 지정합니다.
Date#	**Date#**는 두 번째 인수(지정된 경우)로 지정한 서식을 사용하여 표현식을 날짜로 평가합니다.
Dual	**Dual()**은 숫자와 문자열을 단일 레코드로 결합합니다. 이러한 레코드의 숫자 표현은 정렬 및 계산 목적에 사용할 수 있으며, 문자열 값은 표시 목적으로 사용할 수 있습니다.
Interval	**Interval()**은 데이터 로드 스크립트, 운영 체제 또는 서식 문자열(지정된 경우)에서 시스템 변수의 서식을 사용하여 숫자를 시간 간격으로 서식 지정합니다.
Interval#	**Interval#()**은 운영 체제에서 기본 설정된 서식 또는 두 번째 인수(지정된 경우)로 지정된 서식을 사용하여 텍스트 표현식을 시간 간격으로 평가합니다.

함수	설명
Money	**Money()**는 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)에 설정된 시스템 변수에 지정된 서식 및 선택적 소수점 구분 기호와 천 단위 구분 기호를 사용하여 표현식을 숫자 화폐 값으로 지정합니다.
Money#	**Money#()**는 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)에 설정된 서식을 사용하여 텍스트 문자열을 화폐 값으로 변환합니다. 사용자 지정 소수점 구분 기호 및 천 단위 구분 기호는 옵션 파라메타입니다.
Num	**Num()**은 숫자의 형식을 지정합니다. 즉, 입력의 숫자 값을 변환하여 두 번째 매개 변수에 지정된 형식을 사용하여 텍스트를 표시합니다. 두 번째 매개 변수를 생략하면 데이터 로드 스크립트에 설정된 소수 및 천 단위 구분 기호를 사용합니다. 사용자 지정 소수점 구분 기호 및 천 단위 구분 기호는 옵션 파라메타입니다.
Num#	**Num#()**은 텍스트 문자열을 숫자 값으로 해석합니다. 즉, 두 번째 매개 변수에 지정된 형식을 사용하여 입력 문자열을 숫자로 변환합니다. 두 번째 매개 변수를 생략하면 데이터 로드 스크립트에 설정된 소수 및 천 단위 구분 기호를 사용합니다. 사용자 지정 소수점 구분 기호 및 천 단위 구분 기호는 옵션 파라메타입니다.
Text	**Text()**는 표현식을 숫자로 해석 가능한 경우에도 텍스트로 취급하도록 지정합니다.
Time	**Time()**은 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)의 시스템 변수에 설정된 시간 서식을 사용하여 표현식을 시간 값으로 서식 지정합니다.
Time#	**Time#()**은 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)에 설정된 시간 서식을 사용하여 표현식을 시간 값으로 평가합니다.
Timestamp	**TimeStamp()**는 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)의 시스템 변수에 설정된 타임스탬프 서식을 사용하여 표현식을 날짜 및 시간 값으로 서식 지정합니다.
Timestamp#	**Timestamp#()**는 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)에 설정된 타임스탬프 서식을 사용하여 표현식을 날짜 및 시간 값으로 평가합니다.

계산 필드 식에 사용할 수 있는 숫자 함수

다음 함수를 사용하여 숫자 값을 반올림할 수 있습니다.

숫자 함수

함수	설명
ceil	**Ceil()**은 **offset** 숫자에 의해 시프트된 **step**의 가장 가까운 배수로 숫자를 올림합니다.
div	**Div()**는 첫 번째 인수를 두 번째 인수로 산술적으로 나눈 정수 부분을 반환합니다. 두 파라메타는 모두 실수로 해석되므로 정수일 필요가 없습니다.
evens	**Even()**은 **integer_number**가 짝수 정수 또는 0일 경우 True(-1)를 반환합니다. **integer_number**가 홀수 정수일 경우는 False(0)를, **integer_number**가 정수가 아닐 경우는 NULL을 반환합니다.
fabs	**Fabs()**는 **x**의 절대값을 반환합니다. 결과는 양수입니다.

함수	설명
fact	**Fact()**는 양의 정수 **x**의 계승을 반환합니다.
floor	**Floor()**는 **offset** 숫자에 의해 시프트된 **step**의 가장 가까운 배수로 숫자를 내림합니다.
fmod	**fmod()**는 첫 번째 인수(피제수)를 두 번째 인수(제수)로 나눈 정수의 나머지 부분을 반환하는 일반화된 모듈 함수입니다. 결과는 실수입니다. 두 인수는 모두 실수로 해석되므로 정수일 필요가 없습니다.
frac	**Frac()**는 **x**의 분위수 부분을 반환합니다.
mod	**Mod()**는 정수 나누기의 음수가 아닌 나머지를 반환하는 수학적 모듈 함수입니다. 첫 번째 인수는 피제수이고 두 번째 인수는 제수이며, 두 인수는 모두 정수 값이어야 합니다.
odd	**Odd()**는 **integer_number**가 홀수 정수 또는 0일 경우 True(-1)를 반환합니다. **integer_number**가 짝수 정수일 경우는 False(0)를, **integer_number**가 정수가 아닐 경우는 NULL을 반환합니다.
round	**Round()**는 **offset** 숫자에 의해 시프트된 **step**의 가장 가까운 배수로 숫자를 올림 또는 내림한 결과를 반환합니다.
sign	**Sign()**은 **x**가 양수, 0 또는 음수인가에 따라 1, 0 또는 -1을 반환합니다.

계산 필드 식에 사용할 수 있는 조건부 함수

다음 함수를 사용하여 조건을 평가한 후 조건 값에 따라 서로 다른 대답을 반환할 수 있습니다.

조건부 함수

함수	설명
alt	**alt** 함수는 유효한 숫자 표현이 있는 첫 번째 파라메타를 반환합니다. 이와 일치하는 항목이 발견되지 않을 경우 마지막 파라메타를 반환합니다. 원하는 만큼의 파라메타를 사용할 수 있습니다.
class	**class** 함수는 첫 번째 파라메타를 클래스 간격에 할당합니다. 결과는 숫자 값으로 하위 경계, 텍스트 값으로 a<=x<b이 포함된 이중 값이며, 여기서 a와 b는 bin의 상한과 하한 값입니다.
if	**if** 함수는 함수에 제공된 조건이 True로 평가되는지, False로 평가되는지에 따른 값을 반환합니다.
match	**match** 함수는 첫 번째 파라메타를 이후의 모든 파라메타와 비교하여 일치하는 표현식의 수 위치를 반환합니다. 대/소문자가 구분됩니다.
mixmatch	**mixmatch** 함수는 첫 번째 파라메타를 이후의 모든 파라메타와 비교하여 일치하는 표현식의 수 위치를 반환합니다. 대/소문자는 구분되지 않습니다.
pick	pick 함수는 목록 내 *n*번째 표현식을 반환합니다.
wildmatch	**wildmatch** 함수는 첫 번째 파라메타를 이후의 모든 파라메타와 비교하여 일치하는 표현식의 수를 반환합니다. 비교 문자열에서 와일드카드 문자(* 및 ?)의 사용을 허용합니다. *는 문자의 순서와 일치합니다. ?는 단일 문자와 일치합니다. 대/소문자는 구분되지 않습니다.

계산 필드 식에 사용할 수 있는 NULL 함수

다음 함수를 사용하여 null 값을 반환하거나 검색할 수 있습니다.

NULL 함수

함수	설명
Null	**Null** 함수는 NULL 값을 반환합니다.
IsNull	**IsNull** 함수는 표현식의 값이 NULL인지 테스트하여, 그런 경우 -1(True)을 반환하고 그렇지 않으면 0(False)을 반환합니다.

계산 필드 식에 사용할 수 있는 수학 함수

수학 계산에 다음 함수를 사용할 수 있습니다.

수학 함수

함수	설명
e	이 함수는 자연 로그 **e**(2.71828...)의 기수를 반환합니다.
rand	이 함수는 0과 1 사이의 임의의 숫자를 반환합니다. 이는 샘플 데이터를 만드는 데 사용할 수 있습니다.

계산 필드 식에 사용할 수 있는 지수 및 로그 함수

지수 및 로그 계산에 다음 함수를 사용할 수 있습니다.

지수 및 로그 함수

함수	설명
exp	자연 로그 **e**를 밑으로 사용하는 자연 지수 함수 e^x입니다. 결과는 양수입니다.
log	**x**의 자연 로그입니다. 이 함수는 **x**> 0인 경우에만 정의됩니다. 결과는 숫자입니다.
log10	10을 밑으로 하는 **x**의 상용 로그입니다. 이 함수는 **x**> 0인 경우에만 정의됩니다. 결과는 숫자입니다.
pow	**x**의 **y** 승을 반환합니다. 결과는 숫자입니다.
sqr	**x** 제곱(**x**의 2승)입니다. 결과는 숫자입니다.
sqrt	**x**의 제곱근입니다. 이 함수는 **x** >= 0인 경우에만 정의됩니다. 결과는 양수입니다.

계산 필드 식에 사용할 수 있는 분포 함수

통계 분포 계산에 다음 함수를 사용할 수 있습니다.

분포 함수

함수	설명
CHIDIST	**CHIDIST()**는 chi^2 분포의 단측 확률을 반환합니다. chi^2 분포는 ch^{i2} 테스트와 연관이 있습니다.

함수	설명
CHIINV	CHIINV()는 chi^2 분포의 단측 확률의 역을 반환합니다.
FDIST	FDIST()는 F 확률 분포를 반환합니다.
FINV	FINV()는 F 확률 분포의 역을 반환합니다.
NORMDIST	NORMDIST()는 지정된 평균 및 표준 편차에 대한 누적 정규 분포를 반환합니다. mean = 0이고 standard_dev = 1인 경우 이 함수는 표준 정규 분포를 반환합니다.
NORMINV	NORMINV()는 지정된 평균 및 표준 편차에 대한 정규 누적 분포의 역을 반환합니다.
TDIST	TDIST()는 학생 t 분포에 대한 확률을 반환합니다. 여기서 숫자 값은 확률을 계산할 t에 대해 계산된 값입니다.
TINV	TINV()는 확률 및 자유도의 함수로서 학생 t 분포의 t 값을 반환합니다.

계산 필드 식에 사용할 수 있는 특정 지역 관련 함수

다음 함수를 사용하여 특정 지역 관련 데이터를 처리할 수 있습니다.

특정 지역 관련 함수

함수	설명
GeoMakePoint	GeoMakePoint()는 스크립트 및 차트 표현식에서 위도와 경도가 있는 포인트를 만들고 태그 지정하는 데 사용됩니다.

계산 필드 식에 사용할 수 있는 색 함수

다음 함수를 사용하여 색 속성을 설정하고 평가할 수 있습니다.

색 함수

함수	설명
ARGB	ARGB()는 표현식에서 빨강 성분 **r**, 녹색 성분 **g**, 파랑 성분 **b** 및 **alpha**의 알파 요소(불투명도)로 색이 정의된 차트 개체의 색 속성을 설정하거나 평가하는 데 사용됩니다.
HSL	HSL()은 표현식에서 0 ~ 1 사이의 **hue**, **saturation** 및 **luminosity** 값으로 색이 정의된 차트 개체의 색 속성을 설정하거나 평가하는 데 사용됩니다.
RGB	RGB()는 빨간색 구성 요소 r, 녹색 구성 요소 g 및 파란색 구성 요소 b의 세 가지 매개 변수로 정의된 색상의 색상 코드에 해당하는 정수를 반환합니다. 이러한 구성 요소는 0에서 255 사이의 정수 값을 가져야 합니다. 이 함수는 차트 개체의 색상 속성을 설정하거나 평가하기 위해 표현식에서 사용할 수 있습니다.

계산 필드 식에 사용할 수 있는 논리 함수

다음 함수를 사용하여 논리 연산을 처리할 수 있습니다.

논리 함수

함수	설명
IsNum	표현식을 숫자로 해석할 수 있으면 -1(True)을, 그렇지 않으면 0(False)을 반환합니다.
IsText	표현식에 텍스트 표현이 있으면 -1(True)을, 그렇지 않으면 0(False)을 반환합니다.

계산 필드 식에 사용할 수 있는 시스템 함수

다음 함수를 사용하여 시스템, 장치 및 Qlik Sense 앱 속성에 액세스할 수 있습니다.

시스템 함수

함수	설명
OSuser	이 함수는 현재 연결된 사용자의 이름이 포함된 문자열을 반환합니다. 데이터 로드 스크립트와 차트 표현식 모두에서 사용할 수 있습니다.
ReloadTime	이 함수는 마지막으로 데이터 로드를 완료했을 때의 타임스탬프를 반환합니다. 데이터 로드 스크립트와 차트 표현식 모두에서 사용할 수 있습니다.

필드 유형 변경

데이터가 추가되면 Qlik Sense에서 각 필드의 필드 유형을 해석합니다. 현재 다음과 같은 필드 유형이 지원됩니다.

- **일반**
- **날짜**
- **타임스탬프**
- **지리 데이터**

데이터가 올바르게 해석되지 않으면 필드 유형을 조정할 수 있습니다. 입력 및 데이터 또는 타임스탬프 필드의 표시 형식을 변경할 수도 있습니다.

테이블 편집기를 열려면 편집하려는 데이터 테이블에서 을 클릭합니다.

일부 경우에는 필드 유형이나 필드의 표시 형식을 변경할 수 없습니다.

- 테이블은 둘 이상의 테이블을 연결한 결과입니다.
- 필드는 이미 날짜 또는 타임스탬프로 인식되었습니다.

날짜 또는 타임스탬프 필드가 올바르게 인식되었는지 확인

날짜 또는 타임스탬프 필드가 날짜 또는 타임스탬프로 인식되지 않으면(즉 **일반**으로 표시되어 있는 경우), 올바르게 해석되었는지 확인할 수 있습니다.

다음과 같이 하십시오.

1. 필드 머리글 위에 있는 을 클릭합니다.
   데이터 형식 대화 상자가 열립니다.
2. **필드 유형**을 **날짜** 또는 **타임스탬프**로 변경합니다.
3. **입력 형식**의 서식 문자열을 날짜가 올바르게 해석되도록 변경합니다. 드롭다운 목록에서 사전 준비된 서식을 사용하거나 독자적인 서식 문자열을 작성할 수 있습니다.

> *서식 문자열에는 작은따옴표를 사용할 수 없습니다.*

4. 앱의 기본 형식 이외의 표시 형식을 사용하려면 **표시 형식**에서 서식 문자열을 작성하거나 선택합니다.
   비워 두면 앱의 기본 표시 형식이 사용됩니다.

날짜 또는 타임스탬프 필드의 표시 형식 변경

각 앱에는 날짜 및 타임스탬프 필드에 대한 기본 표시 형식이 있습니다. 개별 날짜 또는 타임스탬프 필드에 대한 표시 형식을 변경할 수 있습니다.

다음과 같이 하십시오.

1. 필드 머리글 위에 있는 또는 을 클릭합니다.
   데이터 형식 대화 상자가 열립니다.
2. **표시 형식**에서 서식 문자열을 변경합니다. 드롭다운 목록에서 사전 준비된 서식을 사용하거나 독자적인 서식 문자열을 입력합니다.

필드 유형을 지리 데이터로 변경

도시와 국가 이름 또는 ISO 기호와 같은 값이 포함된 필드가 지리 데이터로 인식되지 않는 경우 필드 유형을 **지리 데이터**로 변경할 수 있습니다.

다음과 같이 하십시오.

1. 필드 머리글 위에 있는 을 클릭합니다.
   데이터 형식 대화 상자가 열립니다.
2. **필드 유형** 드롭다운 메뉴에서 **지리 데이터**를 선택합니다.
3. **지리 데이터** 드롭다운 메뉴에서 지리 데이터의 유형을 선택합니다.
   옵션에는 **도시**, **국가**, **국가 코드 ISO2** 및 **국가 코드 ISO3**이 있습니다. ISO2 및 ISO3은 국가에 대한 International Standards Organization 코드인 ISO 3166에서 온 것입니다. ISO2에는 두 문자 코드가 포함되고 ISO3에는 세 문자 코드가 포함됩니다. 예를 들어 스웨덴의 코드는 SE 및 SWE입니다.
   ISO 코드를 할당할 때 필드의 값은 할당하는 코드와 일치해야 합니다. ISO3을 두 문자 코드 값을 가진 필드에 할당하면 위치 좌표가 올바르게 할당되지 않습니다.
4. **도시** 데이터의 경우 국가에 대한 지리 데이터가 포함된 테이블의 관련 필드를 선택합니다.
   관련 국가 필드는 한 개뿐일 수 있지만 국가에 대한 지리 데이터가 포함된 필드는 여러 개를 가질 수 있습니다. 예를 들어 한 필드에는 France와 같이 길이가 긴 이름이 있을 수 있고 다른 필드는 ISO2와 같은 국가 코드로 국가를 지정할 수 있습니다. 하지만 필드는 **지리 데이터**로 지정된 경우에만 **연결된 국가** 목록에 표시됩니다.
   지리 데이터 국가 필드로 지정된 필드가 없는 경우 필드를 **도시**로 지정할 때 **연결된 국가** 목록이 표시되지 않습니다.

필드에 **지리 데이터** 필드 유형이 할당되면 지리 좌표(포인트 또는 다각형 데이터)를 포함하는 필드는 사용자에 의해 또는 Qlik Sense에 의해 자동으로 이와 연결됩니다. 좌표가 포함된 연결된 필드는 **데이터 모델 뷰어**에서 볼 수 있습니다. 이러한 좌표는 **맵** 개체를 사용하는 앱에 필요합니다.

우편 지역과 같이 이름 또는 코드 형식의 지리 정보가 포함된 필드는 **지리 데이터** 필드로 지정되지 않는 경우 매핑에 사용할 수 없습니다.

지리 데이터 유형이 할당된 필드는 Mexico 및 MX 등 문자열 값을 계속 유지하지만 **맵** 개체에서 사용되는 경우에는 매핑 좌표가 포인트 또는 다각형 데이터를 포함한 필드에서 비롯됩니다.

분석에서 필드 숨기기

시트 보기 또는 Insight Advisor에서 시각화를 작성할 때 사용하지 않으려는 데이터 필드를 숨길 수 있습니다.

예를 들어 다른 필드를 계산하는 데만 사용되는 필드가 있을 수 있습니다. **데이터 관리자**에서 이러한 필드를 숨기면 시트의 자산 패널이나 Insight Advisor에서는 사용할 수 없지만 **데이터 관리자** 및 **데이터 로드 편집기**에서는 계속 사용할 수 있습니다. 이는 관련 정보만 포함하고 읽고 분석하기가 더 쉽도록 현재 분석 또는 통찰력에 중복되거나 불필요할 수 있는 정보를 제거하는 데 사용할 수 있습니다.

필드를 숨길 때 필드가 가진 모든 기존 관계(예: 연결 또는 계산에서의 사용)가 유지됩니다. 필드가 마스터 항목 또는 기존 차트 등에서 현재 사용 중인 경우 해당 위치에서 계속 사용할 수 있지만 다시 표시될 때까지 새 마스터 항목이나 시각화에서는 사용할 수 없습니다.

> ***데이터 로드 편집기**로 이동하여 자동 생성 섹션을 열면 **데이터 관리자**의 숨겨진 모든 필드를 볼 수 있습니다. 모든 숨겨진 필드는 `TAG FIELD <필드 이름>`으로 나열됩니다. `WITH '$hidden';`*

분석에서 필드 숨기기

다음과 같이 하십시오.

1. 필드 머리글 위의☰를 클릭합니다.
2. **분석에서 숨기기**를 클릭합니다.

이제 필드가 시트 보기 및 Insight Advisor에서 숨겨집니다. 숨겨진 필드가 필드 머리글 위에 👁 추가되었습니다.

숨겨진 필드 보기

다음과 같이 하십시오.

1. 필드 머리글 위의☰ 또는 👁을 클릭합니다.
2. **분석에서 표시**를 클릭합니다.

이제 필드가 시트 보기 및 Insight Advisor에서 사용 가능합니다. 필드 머리글 위의 👁은 제거됩니다.

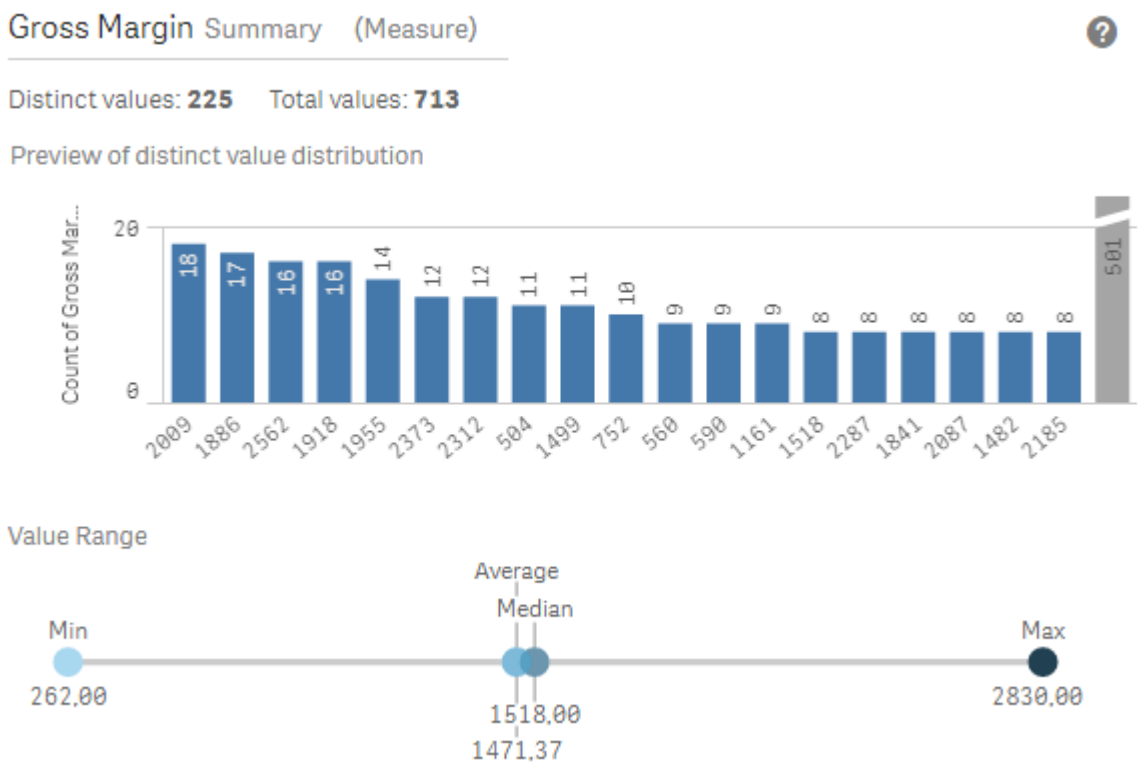
데이터를 로드하기 전에 테이블 필드 데이터 평가

Qlik Sense에 데이터를 로드하기 전에 null 값이나 이상값 등 데이터의 잠재적인 품질 문제를 검사하기 위해 **요약** 데이터 프로파일링 카드를 사용하여 데이터의 요약을 봅니다.

또한 **요약** 카드를 사용하면 필드 데이터를 차원 또는 측정값으로 보는 등의 다양한 데이터 해석을 볼 수 있습니다.

데이터 관리자에서 테이블을 편집하고 테이블 필드를 선택하여 **요약** 카드에 액세스합니다. 테이블 편집기에서 필드가 선택되면 Qlik Sense에서 데이터 유형, 메타데이터 및 현재 값을 검사합니다. 그런 다음 필드가 차원, 측정값 또는 임시 필드로 분류되고 **요약** 카드에 분석이 표시됩니다. 데이터를 차원이나 측정값으로 분류할 수 있는 필드는 미리 보기를 전환하여 데이터를 차원이나 측정값으로 표시할 수 있습니다. **요약** 카드에 데이터 필드가 분류되는 방법에 따라 Qlik Sense 시각화에서 사용할 수 있는 방법이 달라지지는 않지만 다른 데이터 프로파일링 카드의 데이터 필드에서 사용할 수 있는 변환 옵션에는 영향을 미칩니다.

요약 데이터 프로파일링 카드를 사용한 데이터 요약



요약 카드는 다음 정보를 제공합니다.

- **고유 값**: 필드의 고유 값 수
- **전체 값**: 필드의 값 수
- **고유 값 분포 미리 보기**: 20개가 넘는 고유 값이 있는 필드에서 상위 20개만 표시합니다. 다른 모든 값은 차트에서 단일 값으로 그룹화됩니다. 모든 값이 고유한 경우 막대형 차트는 표시되지 않습니다.
- **값 범위**: (측정값 및 임시 필드에만 해당) 측정값 필드의 경우 **값 범위**는 필드의 최소값, 중앙값, 평균 및 최대값을 표시하는 차트입니다. 임시 필드의 경우 **값 범위**는 필드의 데이터가 적용되는 기간입니다.
- **Null 값**: 데이터에서 null 값의 수입니다. 필드에 null 값이 있는 경우에만 이 시각화가 표시됩니다.
- **혼합 값**: 텍스트 및 숫자 값이 모두 포함된 필드에서 텍스트 값의 수입니다. 필드에 혼합 값이 있는 경우에만 이 시각화가 표시됩니다.

필드가 **요약** 카드에 분류되는 방식에 따라 다른 데이터 프로파일링 카드에서 수정될 수 있습니다. 측정값으로 설정된 필드는 **버킷** 카드를 사용하여 필드에서 만든 그룹화된 값을 가질 수 있습니다. 자세한 내용은 *측정값 데이터를 범위로 그룹화 (page 63)*를 참조하십시오.

차원으로 설정된 필드는 다음 값을 가질 수 있습니다.

- **바꾸기** 카드를 사용하여 다른 값으로 대체된 고유 값.
  자세한 내용은 *테이블의 필드 값 바꾸기 (page 57)*를 참조하십시오.
- **Null 설정** 카드를 사용하여 null 값으로 설정된 고유 값.
  자세한 내용은 *테이블에서 필드 값을 Null로 설정 (page 59)*를 참조하십시오.
- **순서** 카드를 사용하여 값에 적용된 사용자 지정 순서.
  자세한 내용은 *차원 값 순서 사용자 지정 (page 60)*를 참조하십시오.
- **분할** 카드를 사용하여 새 테이블 필드로 분할된 필드 데이터.
  자세한 내용은 *테이블에서 필드 분할 (page 61)*를 참조하십시오.

요약 카드에 액세스

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 테이블 필드를 선택합니다.

요약 카드가 나타납니다.

필드의 데이터 범주 변경

Qlik Sense에서는 필드의 데이터가 여러 범주로 분류될 수 있는 경우 필드의 데이터 범주 변경을 허용합니다.

다음과 같이 하십시오.

- **요약** 카드에서 ▼ 을 클릭하고 다른 데이터 범주를 선택합니다.

테이블의 필드 값 바꾸기

바꾸기 카드를 사용하면 테이블의 다른 필드 값을 다른 값으로 바꿀 수 있습니다.

데이터 셋에서 서로 다른 용어가 동일한 개체 또는 개념을 위해 사용될 수 있습니다. 예를 들어 국가의 전체 이름과 약어를 동일한 데이터 셋에서 찾을 수 있습니다. **바꾸기** 카드를 사용하면 이러한 인스턴스가 서로 다른 고유 값이 아닌 단일 고유 값으로 처리되도록 할 수 있습니다. 예를 들어 국가 데이터를 포함하는 필드에서 *U.S*, *US* 및 *U.S.A*를 *USA*로 바꿀 수 있습니다. **바꾸기** 카드를 사용하여 데이터 셋의 이름을 변경해야 하는 경우와 같이 개별 값을 변경할 수도 있습니다.

필드의 대체 값을 최대 5,000개의 고유 값을 포함하도록 설정할 수 있습니다.

또한 계산 필드는 대체 값을 무시하고 원래 값을 대신 사용합니다.

바꾸기 카드는 두 개의 섹션, **고유 값** 및 **대체 값**으로 구성됩니다. **고유 값**은 모든 고유 값과 대체 값을 나열합니다. **대체 값**에는 대체 용어를 입력할 필드와 대체될 값 목록이 포함됩니다. 고유 값을 선택하고, 대체 값을 입력하고, 대체를 적용하여 필드 값을 바꿉니다. **고유 값**에서 대체 값을 선택하고 이를 편집하여 대체되는 값 또는 대체에 사용되는 값을 변경할 수 있습니다. 대체를 적용하기 전에 여러 대체 값을 추가하거나 편집할 수 있습니다.

값 바꾸기

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 차원 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **바꾸기** 탭을 클릭합니다.
4. **고유 값**에서 바꿀 고유 값을 선택합니다.
5. **대체 값**에서 이러한 고유 값의 새 값을 입력합니다.
6. **바꾸기**를 클릭합니다.

대체 값 편집

대체 값에 고유 값을 더 추가하거나, 대체 값에서 고유 값을 제거하거나, 대체 값을 변경할 수 있습니다.

대체 값에 고유 값 추가

다음과 같이 하십시오.

1. **바꾸기** 카드의 **고유** 값에서 대체 값을 선택합니다.
2. **고유 값**에서 추가하려는 고유 값을 선택합니다.
3. **바꾸기**를 클릭합니다.

대체 값에서 고유 값 제거

다음과 같이 하십시오.

1. **바꾸기** 카드의 **고유** 값에서 대체 값을 선택합니다.
2. 제거하려는 각 고유 값 뒤의 을 클릭합니다.
3. **바꾸기**를 클릭합니다.

대체 값 변경

다음과 같이 하십시오.

1. **바꾸기** 카드의 **고유** 값에서 대체 값을 선택합니다.
2. **대체**에서 새 값을 입력합니다.
3. **바꾸기**를 클릭합니다.

대체 값 삭제

연결된 고유 값을 모두 제거하여 대체 값을 삭제할 수 있습니다.

다음과 같이 하십시오.

1. **바꾸기** 카드의 **고유** 값에서 대체 값을 선택합니다.
2. **모두 제거**를 클릭합니다.
3. **바꾸기**를 클릭합니다.

테이블에서 필드 값을 Null로 설정

null 설정 카드에서 차원 필드의 고유 값을 Qlik Sense에서 null 값으로 처리되도록 선택합니다.

예를 들어 소스 데이터에 null에 대해 *X*와 같은 표현이 포함되어 있는 경우 **null 설정** 카드를 사용하여 Qlik Sense에서 해당 값을 null 값으로 처리되도록 설정할 수 있습니다. 테이블에 빈 공간이 있는 필드가 포함되어 있을 경우 **null 설정** 카드를 사용하여 이 필드를 null 값으로 설정할 수 있습니다. 이러한 값을 null로 설정하여 **null 설정** 카드를 사용하여 원하지 않는 값의 데이터를 정리할 수도 있습니다.

특정 값을 null 값으로 사용하려면 **바꾸기** 카드를 사용하여 기본 null 값 **- (Null)**을 대체하면 됩니다. 자세한 내용은 *테이블의 필드 값 바꾸기 (page 57)*를 참조하십시오.

최대 5,000개의 고유 값을 포함하는 필드에서 필드 값을 null로 설정할 수 있습니다.

Null 설정 카드는 **고유 값**과 **수동 Null 값**의 두 섹션으로 구성됩니다. **고유 값**에서 값을 선택하면 **수동 Null 값**에 추가됩니다. Null 값을 적용하면 선택한 값의 모든 인스턴스가 필드 데이터에서 Null로 설정됩니다. Null로 설정된 개별 값 또는 모든 값을 복원할 수 있습니다.

필드 값을 Null로 설정

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **Null 설정** 카드를 클릭합니다.
4. **고유 값**에서 Null로 설정할 값을 선택합니다.
5. **Null 값 설정**을 선택합니다.

수동으로 Null로 설정한 값 복원

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **Null 설정** 카드를 클릭합니다.
4. **Null 설정** 카드의 **수동 Null 값**에서 다음 중 하나를 수행합니다.
    - 더 이상 Null로 설정하지 않으려는 값 뒤의 을 클릭합니다.

- Null로 설정된 모든 값을 복원하려면 **모두 제거**를 클릭합니다.

5. **Null 값 설정**을 선택합니다.

차원 값 순서 사용자 지정

사용자 지정 순서를 사용하면 시각화에서 차원 값의 순서를 설정할 수 있습니다.

많은 시각화에서 사전순 또는 숫자순이 적용될 수 있지만 일부 인스턴스에서는 다른 순서가 유용합니다. 예를 들어 도시를 포함하는 필드에서 도시가 동쪽에서 서쪽 지역의 순서로 차트에 나열되도록 사용자 지정 순서를 설정할 수 있습니다. 다음 요구 사항을 충족하는 필드에 대해 사용자 지정 순서를 정의할 수 있습니다.

- 필드가 **요약** 카드에서 차원으로 설정되어야 합니다.
- 필드가 최대 5,000개의 고유 값을 포함해야 합니다.
  필드의 값은 최대 25개가 좋지만 최대 5,000개의 고유 값을 포함하는 필드에서 **순서** 카드를 사용하도록 선택할 수 있습니다.

순서 카드는 두 개의 섹션, **현재 순서** 및 **순서 미리 보기**로 구성됩니다. **현재 순서**는 차원의 고유 값을 모두 표시합니다. 기본적으로 고유 값은 로드 순서로 구성됩니다. **현재 순서**에서 값을 원하는 순서로 끌어서 사용자 지정 순서를 설정합니다. **순서 미리 보기**는 각 고유 값의 값 수를 표시하는 막대형 차트이며 현재 순서로 구성됩니다.

사용자 지정 순서는 로드 순서로 정렬을 제외하고 Qlik Sense 시각화에서 사용할 수 있는 모든 기타 정렬 옵션을 재정의합니다. 이 필드에 대해 사전순 또는 숫자순이 필요한 경우에는 순서를 다시 설정하여 사용자 지정 순서를 제거해야 합니다.

차원 필드에서 값 순서 변경

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 차원 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **순서** 탭을 클릭합니다.
4. 값을 클릭한 채 새 순서로 끕니다.
5. 현재 순서를 취소하려면 **취소**를 클릭합니다.

   ***취소**는 새 사용자 지정 순서에만 사용할 수 있습니다. 기존 사용자 지정 순서의 값 순서를 변경할 경우 변경 사항을 취소하려면 테이블에서 다른 필드를 선택한 다음 다시 이 필드를 선택합니다.*

6. **순서 변경**을 클릭합니다.

차원 필드에서 값 순서 다시 설정

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 다시 정렬된 차원 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **순서** 탭을 클릭합니다.

4. **다시 설정**을 클릭합니다.
5. **확인**을 클릭합니다.

테이블에서 필드 분할

분할 카드를 사용하면 기존 필드의 데이터를 사용하여 새 필드를 만들 수 있습니다.

예를 들어 주소가 포함된 필드를 분할하여 우편 번호만 포함된 새 필드를 만들 수 있습니다. 이를 통해 기존 데이터의 세그먼트가 포함된 새 필드를 빠르게 만들 수 있습니다. **분할** 카드는 다음 요구 사항을 충족하는 테이블 필드로부터 새 필드를 만들 수 있습니다.

- 필드가 **요약** 카드에서 차원으로 설정되어야 합니다.

*날짜 및 시간 정보가 포함된 테이블 필드는 **데이터 관리자**에서 해당 테이블이 준비되면 자동으로 날짜 필드로 분할되고 **분할** 카드를 필요로 하지 않습니다.*

분할 카드는 템플릿 값을 포함하는 입력 필드와 값이 있는 새 필드의 미리 보기로 구성됩니다. 기본적으로 템플릿 값은 필드에서 숫자 순서로 첫 번째 값이지만 소스 필드에서 다른 값을 선택하여 템플릿으로 사용할 수 있습니다. 테이블의 모든 값을 표현하는 값을 선택해야 합니다. 이상값을 템플릿으로 사용하여 분할하면 새 필드의 품질에 영향을 미칠 수 있습니다.

필드는 필드를 분할하려는 템플릿 값에 분할 표식을 삽입하여 분할됩니다. 분할 표식은 샘플 필드에서 분할 표식을 추가할 포인트를 선택하고 선택 내용을 조정한 다음 인스턴스 또는 위치별로 분할하도록 선택하여 추가됩니다. **분할** 카드는 권장 분할 표식을 템플릿 값에 자동으로 추가할 수 있습니다.

인스턴스는 단어 사이의 문자 @ 또는 공백과 같이 선택된 구분 기호가 있는 항목입니다. 인스턴스 분할 표식의 위치는 다음 중 하나를 기준으로 합니다.

- 값의 시작(예: 값에서 @의 첫 번째 인스턴스)
- 다른 인스턴스 또는 위치 분할 표식의 오른쪽에 있는 해당 위치 (예: @ 후에 .의 첫 번째 인스턴스)

다른 인스턴스에 기준이 되는 인스턴스를 제거하면 다른 인스턴스의 위치는 분할 표식으로 설정된 다른 구분 기호의 다음 인스턴스 또는 값의 시작을 기준으로 같은 위치로 조정됩니다. 필드를 최대 9개의 구분 기호로 분할할 수 있습니다.

***분할** 카드는 분할 표식으로 지정된 문자를 사용하여 값을 분할합니다. 악센트 부호가 있는 문자와 같이 데이터가 이러한 문자의 구성 방식에 변동이 있는 경우 이러한 변동은 분할에 포함되지 않습니다.*

위치는 처음 4자 이후와 같이 필드 값의 위치입니다. 위치는 다음 중 하나를 기준으로 합니다.

- 값의 시작입니다.
- 인스턴스 분할 표식 뒤의 두 번째 문자와 같이 인스턴스 분할 표식의 오른쪽에 있는 해당 위치입니다.

오른쪽에 위치가 있는 인스턴스를 제거하면 해당 위치가 값의 시작을 기준으로 같은 위치로 이동하거나 첫 번째 다음 인스턴스 분할 표식이 왼쪽으로 이동하거나 값의 시작으로 이동합니다.

분할 표식을 추가하면 필드 미리 보기가 업데이트되어 새 필드와 해당 데이터가 표시됩니다. 필드 미리 보기에서 새 필드의 이름을 바꿀 수 있습니다. 분할을 적용하기 전에 테이블에서 분할 필드를 포함 또는 제외하도록 선택할 수도 있습니다. 분할을 적용하면 필드 미리 보기에서 선택한 필드가 테이블에 추가됩니다.

필드는 여러 번 분할될 수 있습니다.

필드를 새 필드로 분할하려면 다음을 수행합니다.

1. **분할** 카드에 액세스합니다.
2. 입력 필드에 분할 표식을 설정합니다.
3. 다음 선택적 작업을 수행할 수도 있습니다.
    - 분할 표식을 제거합니다.
    - 테이블에 추가할 새 필드를 선택합니다.
    - 새 필드의 이름을 바꿉니다.
4. 새 필드를 만듭니다.

분할 카드에 액세스

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 차원 필드를 선택합니다.
3. 데이터 프로파일링 카드에서 **분할** 탭을 클릭합니다.

분할 표식 삽입

샘플 값 내의 위치를 클릭하고 적용할 분할 유형을 선택하여 분할 표식을 삽입합니다.

▼*을 클릭하고 다른 값을 선택하여 입력 필드에 표시된 샘플 값을 변경할 수 있습니다.*

다음과 같이 하십시오.

1. **분할** 카드에서 분할 표식을 추가할 샘플 값 내의 위치를 클릭합니다.
   클릭하면 다른 모든 분할 표식까지 모든 템플릿 값이 선택됩니다.
   두 번 클릭하면 템플릿 값에서 커서의 삽입 포인트가 선택됩니다.
2. 선택 탭을 클릭한 채 끌거나 선택하려는 섹션을 강조 표시하여 선택 내용을 조정합니다.
3. 적용할 분할 유형에 해당하는 버튼을 클릭합니다.
    - **이 인스턴스**: 필드가 구분 기호의 선택된 인스턴스로 분할됩니다.
    - **모든 인스턴스**: 필드가 구분 기호의 모든 인스턴스로 분할됩니다.
    - **이 위치들**: 필드가 선택 내용의 양쪽에서 분할됩니다.
    - **이 위치**: 필드가 이 위치에서 분할됩니다.

분할 표식은 템플릿 값에 삽입됩니다.

분할 표식 제거

다음과 같이 하십시오.

- **분할** 카드에서 다음 중 하나를 수행합니다.
    - 단일 분할 표식을 제거하려면 제거할 분할 표식 위에 있는 ⊗을 클릭합니다.
    - 모든 분할 표식을 제거하려면 **다시 설정**을 클릭합니다.
      권장 분할 표식은 제거되지 않으며 제거된 권장 분할 표식은 다시 템플릿에 추가됩니다. 개별적으로 제거해야 합니다.

테이블에 추가할 필드 선택

분할 카드에서 만든 필드를 선택하여 테이블에서 포함하거나 제외시킬 수 있습니다. 기본적으로 모든 분할 필드가 포함됩니다.

다음과 같이 하십시오.

- **분할** 카드의 필드 미리 보기에서 다음 중 하나를 수행합니다.
    - 필드를 포함시키려면 필드 열 확인란을 선택합니다.
    - 필드를 제외하려면 필드 열 확인란 선택을 해제합니다.

새 필드 이름 바꾸기

다음과 같이 하십시오.

- **분할** 카드의 필드 미리 보기에 있는 필드 헤더에 새 필드 이름을 입력합니다.

새 필드 만들기

다음과 같이 하십시오.

- **분할** 카드에서 **분할**을 클릭합니다.

측정값 데이터를 범위로 그룹화

측정값은 범위로 그룹화되고 시각화에서 차원으로 사용되는 경우 분석에 대한 통찰력을 추가로 제공할 수 있습니다.

예를 들어 필드에 고객 연령이 있는 경우 값을 연령 범위로 그룹화할 수 있습니다. **버킷** 데이터 프로파일링 카드를 사용하면 필드 데이터를 범위로 그룹화하여 지정된 그룹으로 새 필드를 만들 수 있습니다. **데이터 관리자**에서 테이블을 편집하고 **요약** 카드에서 측정값으로 설정된 테이블 필드를 선택하여 **버킷** 카드에 액세스합니다.

요구 사항

다음 요구 사항을 충족하는 필드에서 측정값을 범위로 그룹화할 수 있습니다.

- 필드가 **요약** 카드에서 측정값으로 분류되어야 합니다.
- 필드가 최소 10개의 고유 값을 포함해야 합니다.
  버킷 카드는 값 범위가 낮은 필드의 경우 권장되지 않지만 비권장 필드의 경우 **버킷** 카드를 사용하도록 선택할 수 있습니다.
- 필드의 값은 20%가 넘는 혼합 형식을 포함할 수 없습니다.
  버킷 카드는 혼합 형식의 값을 포함하는 필드의 경우 권장되지 않지만 비권장 필드의 경우 **버킷** 카드를 사용하도록 선택할 수 있습니다.

__버킷__ 카드를 사용하여 만든 버킷 필드는 __요약__ 카드에서 차원으로 분류됩니다. 하지만 버킷 필드에는 __순서__ 카드를 사용하여 필드에 적용되는 사용자 지정 순서가 있으면 안 됩니다. 버킷 필드는 계산되는 필드에서 사용될 수 없습니다.

개요

버킷 카드는 제안된 그룹화 수, 그룹화 미리 보기, 각 버킷의 이름 및 값 범위를 수정할 수 있는 그룹화를 포함하는 슬라이더 막대를 제공합니다. **버킷** 필드에 새 숫자를 입력하여 제안된 그룹화 수를 수정할 수 있습니다. Qlik Sense는 최대 20개, 최소 2개의 버킷을 지원합니다. 버킷은 오른쪽에서 왼쪽으로 제거되는 반면 새 버킷은 버킷 범위의 오른쪽에 추가됩니다. 임의의 개별 버킷을 수정하지 않은 경우 각 버킷의 범위는 필드의 값 범위 내에서 고르게 맞도록 조정됩니다. 개별 버킷을 수정한 후 버킷 수를 변경하면 새 버킷이 버킷 범위의 오른쪽에 추가되고 맨 오른쪽에서 두 번째 버킷과 같은 크기의 값 범위가 제공됩니다.

데이터 버킷 미리 보기 막대형 차트는 각 버킷의 고유 값 수와 함께 버킷의 데이터 개요를 제공합니다. 버킷이 변경되면 차트도 업데이트됩니다. 버킷에 값이 없는 경우에는 차트에 막대가 없습니다.

버킷 슬라이더 막대를 사용하여 버킷을 편집할 수 있습니다. 버킷 세그먼트를 클릭하여 해당 버킷 범위를 설정하거나 버킷 이름을 변경하거나 버킷을 완전히 제거할 수 있습니다. 버킷 위로 마우스를 이동하면 버킷 이름 및 값 범위가 표시됩니다. 기본적으로 버킷 이름은 구간 표기법으로 표현된 버킷의 값 범위로 지정됩니다. 버킷 범위에는 시작 값과 최대 값 사이의 값이 포함되지만 끝 값은 제외됩니다.

단일 버킷의 값 범위를 조정할 때 Qlik Sense는 모든 버킷의 값을 시프트하여 가능하면 다른 버킷의 기존 정략적 범위를 반영하되 간격이나 중첩되는 부분이 없도록 합니다. 항상 맨 왼쪽 버킷은 하한이 없고 맨 오른쪽 버킷은 상한이 없습니다. 이렇게 하면 모든 버킷의 설정 범위를 벗어날 수 있는 어떤 값도 캡처할 수 있습니다. 버킷의 하한 범위를 수정하면 버킷 범위가 오른쪽으로 변경되고 버킷의 상한 범위를 수정하면 버킷 범위가 왼쪽으로 수정됩니다.

필드에서 버킷을 만드는 경우에는 소스 필드에서 해당 측정값으로 행에 할당된 모든 버킷을 포함하는 새 필드가 생성됩니다. 기본적으로 이름은 <필드>(버킷팅됨)으로 지정됩니다. 이 필드는 다른 테이블 필드와 마찬가지로 이름 바꾸기, 연결, 정렬 및 삭제가 가능합니다. 테이블에서 필드를 선택하고 버킷팅 옵션을 수정하여 생성된 버킷 필드에서 버킷팅을 편집할 수 있습니다. 동일한 소스 측정값 필드에서 여러 버킷 필드를 만들 수 있습니다.

측정값 그룹화

측정값을 범위로 그룹화하려면 다음을 수행합니다.

1. **버킷** 카드에 액세스합니다.
2. 필요한 경우 버킷 수를 변경합니다.

3. 필요한 경우 개별 버킷을 수정합니다.
4. 필요한 경우 버킷에 대한 수동 조정을 기본 설정 값으로 다시 설정합니다.
5. 버킷팅된 필드를 만듭니다.

버킷 카드에 액세스

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 새 값 범위 그룹을 만드는 경우 필드를 선택합니다.
3. 기존 버킷 필드를 편집하는 경우 버킷 필드를 선택합니다.

버킷 수 변경

버킷 수를 변경하면 범위 끝에 버킷을 추가하거나 범위 끝에서 버킷을 제거합니다.

다음과 같이 하십시오.

- **버킷** 카드에서 **버킷** 앞에 새 숫자를 입력합니다.

버킷 수정

버킷의 이름을 바꾸거나 버킷의 값 범위를 조정하거나 버킷을 삭제할 수 있습니다.

버킷을 수정하는 경우 맨 왼쪽의 버킷 세그먼트부터 맨 오른쪽의 버킷 세그먼트 방향으로 순서대로 버킷을 수정하는 것이 좋습니다.

버킷 이름 바꾸기

다음과 같이 하십시오.

1. **버킷** 카드에서 버킷 세그먼트를 클릭합니다.
2. 이름 필드에 새 이름을 입력합니다.
3. 변경 사항을 적용하려면 버킷 세그먼트 바깥쪽의 아무곳이나 클릭합니다.

버킷의 값 범위 조정

*10진수 값을 사용하려면 **시작** 및 **최대** 필드에 수동으로 해당 값을 입력해야 합니다.*

다음과 같이 하십시오.

1. **버킷** 카드에서 버킷 세그먼트를 클릭합니다.
2. 다음 중 하나를 수행합니다.
    - **시작** 및 **최대** 뒤에 버킷의 범위를 설정할 새 값을 입력합니다.
      버킷은 **시작** 및 최대 사이의 값은 포함하지만 **최대** 값은 제외됩니다.
    - 세그먼트의 슬라이더를 조정하여 버킷의 범위를 설정합니다.
3. 변경 사항을 적용하려면 버킷 세그먼트 바깥쪽의 아무곳이나 클릭합니다.

버킷 삭제

다음과 같이 하십시오.

1. **버킷** 카드에서 버킷 세그먼트를 클릭합니다.
2. 를 클릭합니다.

필드의 버킷 다시 설정

버킷을 다시 설정하면 버킷 카드 설정을 기본 상태로 복원할 수 있습니다. 측정값 필드의 경우 기본 상태는 Qlik Sense 권장 버킷팅입니다. 버킷 필드의 경우 기본 상태는 마지막으로 **버킷 만들기**를 클릭한 후의 필드 상태입니다.

다음과 같이 하십시오.

- **버킷** 카드에서 **기본값으로 다시 설정**을 클릭합니다.

버킷 필드 만들기

다음과 같이 하십시오.

1. **버킷** 카드에서 **버킷 만들기**를 클릭합니다.
2. **확인**을 클릭합니다.

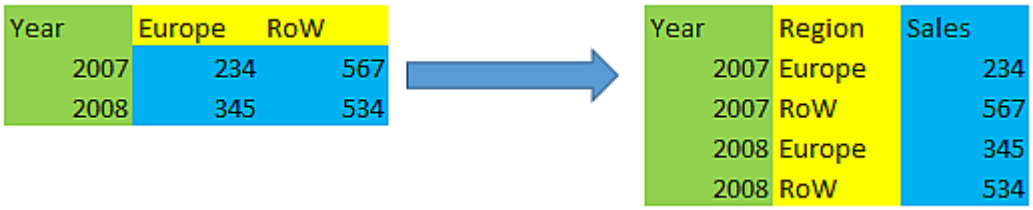
데이터 그룹화를 포함하는 새 필드가 테이블에 추가됩니다.

데이터 관리자에서 크로스탭 데이터 피벗 해제

크로스탭은 두 개의 머리글 데이터 직교 목록 사이의 값을 테이블로 나타내는 일반적인 유형의 테이블입니다.

이는 데이터를 다른 데이터 테이블에 연결하려는 경우 선택적 데이터 형식이 아닙니다. 이 항목에서는 크로스탭 형식으로 로드된 데이터의 피벗을 해제하고, 데이터 관리자를 사용하여 크로스탭의 일부를 행으로 바꾸는 방법을 설명합니다.

크로스탭 형식으로 로드된 데이터의 피벗을 해제하고 일부를 행으로 바꿈



Year	Europe	RoW
2007	234	567
2008	345	534

Year	Region	Sales
2007	Europe	234
2007	RoW	567
2008	Europe	345
2008	RoW	534

크로스탭이란?

크로스탭에는 직관적으로 읽어야 하는 값 행렬인 여러 개의 한정하는 열이 포함되어 있습니다. 이 경우 하나의 한정하는 열인 Year와 월별 판매 데이터 행렬이 있습니다.

크로스탭

Year	Jan	Feb	Mar	Apr	May	Jun
2008	45	65	78	12	78	22
2009	11	23	22	22	45	85
2010	65	56	22	79	12	56
2011	45	24	32	78	55	15
2012	45	56	35	78	68	82

이 테이블을 그대로 Qlik Sense로 로드하면 *Year*에 해당하는 필드 하나와 각 월에 해당하는 필드가 생성됩니다. 그러나 이러한 결과를 원하는 것은 아닐 것입니다. 3개의 필드를 생성하는 것을 선호할 수도 있습니다.

- 한정하는 필드(이 경우 *Year*)는 위의 테이블에서 녹색으로 표시되었습니다.
- 이 경우 월 이름 Jan - Jun으로 표시되는 특성 필드는 노란색으로 표시되었습니다. 이필드는 적절하게 *Month*로 이름을 지정할 수 있습니다.
- 데이터 필드로, 파란색으로 표시되어 있습니다. 이 경우, 판매 데이터를 나타내므로 적절히 *Sales*로 이름을 지정할 수 있습니다.

데이터 관리자 테이블 편집기에서 피벗 해제 옵션을 사용하고 Jan - Jun 필드를 선택하면 됩니다. 그러면 다음과 같은 테이블이 만들어집니다.

피벗 해제된 테이블

Year	Month	Sales
2008	1월	45
2008	2월	65
2008	3월	78
2008	4월	12
2008	5월	78
2008	6월	22
2009	1월	11
2009	2월	23
...	...	...

크로스탭을 일반 테이블로 피벗 해제

다음과 같이 하십시오.

1. 크로스탭 형식의 데이터 파일을 앱에 추가합니다.
2. 데이터 관리자의 테이블에서 기호를 클릭하여 테이블 편집기를 엽니다.
3. **피벗 해제**를 클릭합니다.

4. 행으로 바꿀 필드를 선택합니다. 피벗이 해제되지 않은 한정하는 필드가 하나 이상 있어야 합니다. 선택하는 방법에는 두 가지 방법이 있습니다.
    - 바꾸려는 필드의 필드 헤더를 클릭합니다. 한정하는 필드로 유지할 필드는 선택하지 마십시오.
    - 한정하는 필드로 유지할 필드의 필드 헤더를 클릭한 다음 필드 메뉴에서 **선택 반전**을 선택합니다. 이는 바꾸려는 필드 수가 많은 경우에 가장 간편한 방법입니다.
5. **피벗 해제 적용**을 클릭합니다.
   이제 선택한 데이터가 다음 두 개의 필드가 있는 행으로 바뀝니다. Tablename.**Attribute 필드** 및 Tablename.**Data 필드**
6. **특성 필드**의 이름을 의미 있는 이름(위 예의 경우 *Month*)으로 바꿉니다.
7. **데이터 필드**의 이름을 의미 있는 이름(위 예의 경우 *Sales*)으로 바꿉니다.

이제 교차 표가 일반 형식으로 피벗이 해제되었으므로 앱의 다른 데이터에 보다 쉽게 연결할 수 있습니다.

원래 크로스탭 테이블로 되돌리기

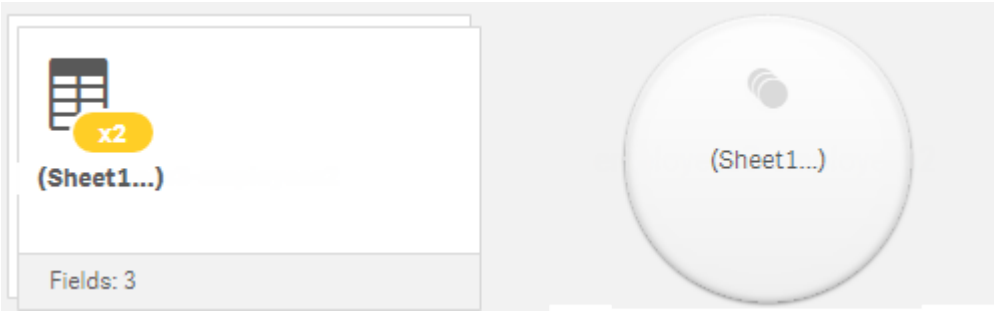
테이블 편집기에서 **피벗 해제 실행 취소**를 클릭하여 데이터 소스의 원래 크로스탭 형식으로 되돌릴 수 있습니다. 앱의 다른 데이터에 연결된 경우 연결이 삭제됩니다.

3.13 **데이터 관리자**에서 테이블 연결

연결은 두 개의 테이블을 결합된 필드가 있는 단일 테이블로 결합합니다. 콘텐츠를 통합하여 콘텐츠를 공유하는 별도의 테이블 및 필드 수를 줄입니다. **데이터 관리자**에서 테이블을 자동 또는 강제로 연결할 수 있습니다.

결합된 테이블에 포함된 데이터에 대한 보다 세분화된 제어가 필요한 경우 **조인** 작업에 대한 자세한 내용은 *데이터 관리자에서 테이블 조인 (page 72)*을 참조하십시오.

테이블 보기 및 연결 보기의 연결된 테이블



테이블 자동연결

Qlik Sense에서 하나 이상의 추가된 테이블에 동일한 수의 필드와 동일한 필드 이름이 다른 테이블에 있음을 감지하면 테이블이 **데이터 관리자**에서 자동으로 연결됩니다. 이러한 경우 Qlik Sense에서 테이블을 단일 테이블로 자동으로 연결합니다. 자동으로 연결된 테이블이 실수로 연결되었거나 연결하지 않으려는 경우 분할할 수 있습니다. 자동으로 연결된 테이블을 다른 테이블에 강제로 연결할 수 있습니다.

자동으로 연결된 테이블에는 다음과 같은 제한 사항이 있습니다.

- 필드 범주를 변경할 수 없습니다.
- 자동으로 연결된 테이블의 피벗을 해제할 수 없습니다.
- **소스에서 데이터 선택**을 사용하여 데이터를 추가하거나 제거할 수 없습니다.

테이블 간 강제 연결

동일한 필드나 데이터를 완전히 공유하지 않더라도 **데이터 관리자**에서 **연결 또는 조인** 창을 사용하여 테이블 간을 강제로 연결할 수 있습니다. 강제 연결을 사용하면 최종 연결된 테이블에서 매핑을 제어하고 필드를 제외할 수 있습니다. 강제 연결은 추가 변환을 지원합니다. 강제 연결을 사용하면 다음을 수행할 수 있습니다.

- 연결된 테이블을 다른 테이블과 연결합니다.
- 피벗 해제된 테이블을 다른 테이블과 연결합니다. 강제 연결된 테이블의 피벗을 해제할 수 있습니다.
- 계산 필드와 테이블을 연결합니다. 계산 필드는 강제 연결된 다른 필드에 연결할 수 있습니다. 강제 연결된 테이블에 계산 필드를 추가할 수 있습니다.

강제 연결된 테이블에는 다음과 같은 제한 사항이 있습니다.

- 강제 연결을 사용하려면 함께 매핑할 필요는 없지만 연결된 테이블에 포함되는 각 테이블의 필드가 하나 이상 필요합니다.
- 연결 후에는 날짜 필드의 서식을 지정할 수 없습니다. 연결하기 전에 날짜 필드에 적용할 형식이 동일해야 합니다. 연결된 날짜 필드는 **데이터 로드 편집기**에서 **DateFormat**으로 설정된 기본 시간 서식을 사용합니다.
- 연결 후에는 필드 범주를 변경할 수 없습니다.
- 연결된 테이블의 다른 필드에 매핑된 필드를 참조하는 계산 필드에는 연결된 필드의 결합된 데이터가 아니라 원래 필드의 데이터만 포함됩니다. 두 테이블이 연결된 후에 만들어지고 연결된 테이블의 필드를 참조하는 계산 필드는 해당 필드의 모든 데이터를 사용합니다.
- **소스에서 데이터 선택**을 사용하여 연결된 테이블에서 데이터를 추가 또는 제거할 수 없습니다. 그러나 **데이터 추가**를 클릭하고 원본 테이블을 선택한 다음 필드를 제외하여 필드를 제거할 수 있습니다. 제거된 필드의 데이터에 대해 Null 값이 추가됩니다.

연결 또는 조인 창은 **데이터 관리자**에서 •••을 클릭하고 **연결 또는 조인**을 클릭하고 두 테이블을 선택하여 액세스할 수 있습니다. 테이블이 **연결 또는 조인**에서 선택되면 Qlik Sense에서 필드를 분석하여 일치하는 모든 필드를 함께 자동으로 매핑합니다. 정확히 일치하지 않으면 필드는 매핑되지 않은 상태로 유지됩니다. 연결이 적용되면 매핑된 필드가 연결된 테이블에 결합되지만 매핑되지 않은 필드는 해당 값이 없는 행에 대해 null 값이 있는 개별 필드로 포함됩니다.

연결 또는 조인에서 선택된 첫 번째 테이블이 기본 테이블로 설정되며 다른 테이블이 연결되는 테이블입니다. 연결된 테이블은 수동으로 이름을 바꾸지 않는 한 기본 테이블의 테이블 및 필드 이름을 사용합니다. ⮌⮎ 버튼을 사용하여 기본 테이블을 변경할 수 있습니다. **연결 또는 조인**은 두 행으로 필드를 정렬합니다. 상단 행에는 기본 테이블 필드가 있고 하단 행에는 보조 테이블 필드가 있습니다. ⮌⮎ 버튼을 사용하여 기본 및 보조 테이블을 바꿀 수 있습니다.

매핑 편집을 사용하여 기본 매핑을 변경하고 연결된 테이블에서 매핑할 필드, 매핑 해제된 상태로 남겨둘 필드 또는 제외할 필드를 선택할 수 있습니다. **매핑 편집**에는 매핑 편집을 위한 끌어서 놓기 인터페이스와 모든 테이블 필드를 나열하는 **필드** 창이 있습니다. 필드는 기본 테이블 필드 아래로 끌어 매핑할 수 있습니다.

필드는 **필드** 창의 필드 옆에 있는 또는 필드의 맨 위 행으로 끌어 매핑되지 않은 새 필드로 추가할 수 있습니다. 매핑되지 않은 필드는 **필드** 창에서 로 표시됩니다. 연결된 테이블에서 제거된 필드는 테이블에 포함되지 않고 테이블에 연결된 후 Qlik Sense에서 사용할 수 없습니다.

매핑이 적용되고 테이블이 연결되면 필드를 편집할 수 없지만 연결된 테이블을 분할(테이블을 원래 상태로 복원)하면 테이블에서 필드를 제거할 수 있습니다.

데이터 관리자에서 테이블을 강제로 연결하려면 다음을 수행합니다.

1. 연결할 테이블을 선택합니다.
2. 원하는 경우 연결된 테이블과 필드 이름을 변경합니다.
3. 원하는 경우 연결 매핑을 편집합니다.
4. 연결 연산자를 선택합니다.
5. 테이블을 연결합니다.

연결할 테이블 선택

다음과 같이 하십시오.

1. **데이터 관리자**에서 맨 아래 행의 •••를 클릭합니다.
2. **연결 또는 조인**을 클릭합니다.
   연결 또는 조인 창이 열립니다.
3. 두 개의 테이블을 선택합니다.
   두 테이블의 필드는 **연결 또는 조인** 창에서 매핑되거나 매핑되지 않은 상태로 유지됩니다.
4. 각 필드에서 고유 값 샘플을 미리 보려면 을 클릭합니다.
5. 기본 테이블과 보조 테이블을 전환하려면 을 클릭합니다.

테이블 및 필드 이름 바꾸기

다음과 같이 하십시오.

1. **연결 또는 조인 창**에서 테이블 이름 필드에 새로운 테이블 이름을 입력합니다.
2. 필드 이름 필드에 새 필드 이름을 입력합니다.

연결 매핑 편집

다음과 같이 하십시오.

1. **연결 또는 조인** 창에서 **매핑 편집**을 클릭합니다.
2. 두 개의 필드를 매핑하려면 기본 테이블 필드 아래에서 테이블 필드를 클릭하고 끕니다.
3. 매핑되지 않은 새 필드를 추가하려면 테이블 필드를 클릭하여 필드의 위쪽 행으로 끕니다.
4. 연결된 테이블에서 필드를 제거하려면 필드에서 을 클릭합니다.
5. 제거된 필드를 다시 테이블로 되돌리려면 **필드** 창을 클릭하고 필드 옆에 있는 을 클릭합니다.
6. **매핑 편집**을 클릭하여 **매핑 편집**을 닫습니다.

연결 연산자 선택

다음과 같이 하십시오.

1. **연결 또는 조인** 창에서 **작업 선택**을 클릭합니다.
2. 목록에서 **연결**을 선택합니다.

테이블 연결

다음과 같이 하십시오.

- **연결 또는 조인** 창에서 **적용**을 클릭합니다.

이제 테이블이 연결되었습니다.

연결된 테이블 분할

Qlik Sense에서 원하지 않는 자동 연결을 수행한 경우와 같이 연결이 더 이상 필요하지 않은 경우 연결된 테이블을 해당 소스 테이블로 분할할 수 있습니다.

> *연결된 테이블을 분할하면 연결된 테이블에 있던 연결뿐만 아니라 기본 테이블과 보조 테이블을 서로 연결한 연결도 제거됩니다. 연결된 테이블을 분할하는 동안 연결을 유지하려면*↩*을 클릭하여 테이블을 분할하는 대신 연결을 실행 취소합니다.* ***데이터 관리자****에서 데이터를 로드한 후에는*↩*을 사용하여 연결을 실행 취소할 수 없습니다.*

자동으로 연결된 테이블 분할

다음과 같이 하십시오.

1. 연결된 테이블을 선택합니다.
2. ⇹를 클릭합니다.
3. 연결된 테이블에서 분할할 테이블을 선택합니다.
4. **분할**을 클릭합니다.

이제 테이블이 해당 소스 테이블로 분할되고 소스 테이블의 모든 필드가 한정됩니다. 한정된 필드의 이름은 테이블 이름과 그 뒤에 필드 이름으로 바뀌고 마침표 구두점 표시(“.” 문자)로 구분됩니다.

Table1과 Table2에는 모두 Field1 및 Field2 필드가 포함되어 있습니다. **데이터 관리자**에서 이들 테이블을 추가하면 동일한 필드인 Field1 및 Field2를 사용하여 Table1-Table2라는 테이블로 컨케트네이트됩니다.

Table1-Table2를 분할하면 그 결과로 두 개의 테이블이 생깁니다.

- Table1.Field1 및 Table1.Field2 필드가 있는 Table1
- Table2.Field1 및 Table2.Field2 필드가 있는 Table2

강제 연결된 테이블 분할

다음과 같이 하십시오.

1. 연결된 테이블을 선택합니다.
2. 를 클릭합니다.

이제 테이블이 해당 소스 테이블로 분할되었습니다. 소스 테이블의 모든 필드는 사전 연결 이름을 갖습니다. 연결된 테이블을 분할하면 한 수준의 연결만 분할하므로 분할 연결된 테이블에 속한 연결된 테이블에는 자체 연결이 유지됩니다

3.14 **데이터 관리자**에서 테이블 조인

조인은 두 테이블의 데이터를 수동으로 결합하는 데 사용할 수 있는 작업으로, 사용자가 선택하는 구성에 따라 결과가 달라질 수 있습니다.

조인은 연결에 비해 결합된 테이블에 대한 세부적인 제어가 가능합니다.

조인 작업은 두 개의 테이블을 가져와 하나로 결합하므로 하나 또는 여러 개의 공통 필드에 대한 공통 값의 중첩을 기반으로 두 원본 테이블에 있는 필드의 조합이 됩니다. **조인** 작업에 적용될 수 있는 여러 연산자는 다음과 같습니다. **외부**, **내부**, **왼쪽** 및 **오른쪽**.

조인된 테이블은 결합된 테이블의 메모리 양을 그대로 차지합니다. 조인된 테이블을 과도하게 사용하면 Qlik Sense의 속도가 저하될 수 있습니다. 조인 작업에서 제외된 정보는 테이블이 분리될 때까지 Qlik Sense에서 액세스할 수 없습니다.

조인 연산자

4개의 조인 연산자는 다음과 같습니다. **외부 조인**, **내부 조인**, **왼쪽 조인** 및 **오른쪽 조인**. 선택한 연산자에 따라 중첩되는 필드 또는 값이 포함되거나 제외됩니다.

*조인 연산자가 **왼쪽** 및 **오른쪽** 테이블을 참조하는 경우 선택 순서대로 첫 번째와 두 번째 테이블을 각각 참조합니다.*

외부 조인

외부 조인 연산자는 겹치는 필드 값이 한 테이블에 표시되거나 두 테이블에 모두 표시되는 경우 두 테이블 값의 가능한 모든 조합을 포함합니다.

First table

A	B
1	aa

A	B
2	cc
3	ee

Second table

A	C
1	xx
4	yy

조인 테이블

A	B	C
1	aa	xx
2	cc	-
3	ee	-
4	-	yy

내부 조인

내부 조인 연산자는 중첩되는 필드 값이 두 테이블에 모두 표시되는 경우 두 테이블 값의 조합만 포함합니다.

First table

A	B
1	aa
2	cc
3	ee

Second table

A	C
1	xx
4	yy

Joined table

A	B	C
1	aa	xx

왼쪽 조인

왼쪽 조인 연산자는 중첩되는 필드 값이 첫 번째 테이블에 표시되는 경우 두 테이블 값의 조합을 포함합니다.

First table

A	B
1	aa
2	cc
3	ee

Second table

A	C
1	xx
4	yy

Joined table

A	B	C
1	aa	xx
2	cc	-
3	ee	-

오른쪽 조인

오른쪽 조인 연산자는 중첩되는 필드 값이 두 번째 테이블에 표시되는 경우 두 테이블 값의 조합을 포함합니다.

First table

A	B
1	aa
2	cc
3	ee

Second table

A	C
1	xx
4	yy

Joined table

A	B	C
1	aa	xx
4	-	yy

테이블 조인

연결 또는 조인 창은 **데이터 관리자**에서 •••을 클릭하고 **연결 또는 조인**을 클릭하고 두 테이블을 선택하여 액세스할 수 있습니다. 테이블이 **테이블 연결 또는 조인**에서 선택되면 Qlik Sense에서 필드를 분석하여 일치하는 모든 필드를 함께 자동으로 매핑합니다. 정확히 일치하지 않으면 필드는 매핑되지 않은 상태로 유지됩니다. 조인이 적용되면 매핑된 필드가 조인 테이블에서 결합됩니다. 매핑되지 않은 필드는 해당 값이 없는 행의 경우 값이 null인 개별 필드로 포함되거나 중첩되는 값 인스턴스가 없는 경우 완전히 제외됩니다.

테이블 연결 또는 조인에서 선택된 첫 번째 테이블이 기본 테이블로 설정되며 다른 테이블이 조인되는 테이블입니다. 조인된 테이블은 수동으로 이름을 바꾸지 않는 한 기본 테이블의 테이블 및 필드 이름을 사용합니다. ⇆ 버튼을 사용하여 기본 테이블을 변경할 수 있습니다. **테이블 연결 또는 조인**은 두 행으로 필드를 정렬합니다. 상단 행에는 기본 테이블 필드가 있고 하단 행에는 보조 테이블 필드가 있습니다. ⇆ 버튼을 사용하여 기본 및 보조 테이블을 바꿀 수 있습니다.

매핑 편집을 사용하여 기본 매핑을 변경하고 조인된 테이블에서 매핑할 필드, 매핑 해제된 상태로 남겨둘 필드 또는 제외할 필드를 선택할 수 있습니다. **매핑 편집**에는 매핑 편집을 위한 끌어서 놓기 인터페이스와 모든 테이블 필드를 나열하는 **필드** 창이 있습니다. 필드는 기본 테이블 필드 아래로 끌어 매핑할 수 있습니다. 필드는 **필드** 창의 필드 옆에 있는 ↘ 또는 필드의 맨 위 행으로 끌어 매핑되지 않은 새 필드로 추가할 수 있습니다. 매핑되지 않은 필드는 **필드** 창에서 ⓘ로 표시됩니다. 조인된 테이블에서 제거된 필드는 테이블에 포함되지 않고 테이블에 조인된 후 Qlik Sense에서 사용할 수 없습니다.

매핑이 적용되고 테이블이 조인되면 매핑된 필드를 편집할 수 없지만 조인된 테이블을 분할(테이블을 원래 상태로 복원)하면 테이블에서 매핑된 필드를 제거할 수 있습니다.

데이터 관리자에서 테이블을 조인하려면 다음을 수행합니다.

1. 조인할 테이블을 선택합니다.
2. 원하는 경우 조인된 테이블과 필드 이름을 변경합니다.
3. 원하는 경우 매핑을 편집합니다.
4. 조인 연산자를 선택합니다.
5. 테이블을 조인합니다.

조인할 테이블 선택

다음과 같이 하십시오.

1. **데이터 관리자**에서 맨 아래 행의 •••를 클릭합니다.
2. **연결 또는 조인**을 클릭합니다.
   연결 또는 조인 창이 열립니다.
3. 두 개의 테이블을 선택합니다.
   두 테이블의 필드는 **테이블 연결 또는 조인** 창에서 매핑되거나 매핑되지 않은 상태로 유지됩니다.
4. 각 필드에서 고유 값 샘플을 미리 보려면 을 클릭합니다.
5. 기본 테이블과 보조 테이블을 전환하려면 을 클릭합니다.

테이블 및 필드 이름 바꾸기

다음과 같이 하십시오.

1. **연결 또는 조인 창**에서 테이블 이름 필드에 새로운 테이블 이름을 입력합니다.
2. 필드 이름 필드에 새 필드 이름을 입력합니다.

매핑 편집

다음과 같이 하십시오.

1. **연결 또는 조인** 창에서 **매핑 편집**을 클릭합니다.
2. 두 개의 필드를 매핑하려면 기본 테이블 필드 아래에서 테이블 필드를 클릭하고 끕니다.
3. 매핑되지 않은 새 필드를 추가하려면 테이블 필드를 클릭하여 필드의 위쪽 행으로 끕니다.
4. 조인된 테이블에서 필드를 제거하려면 필드에서 을 클릭합니다.
5. 제거된 필드를 다시 테이블로 되돌리려면 **필드** 창을 클릭하고 필드 옆에 있는 을 클릭합니다.
6. **매핑 편집**을 클릭하여 **매핑 편집**을 닫습니다.

조인 연산자 선택

다음과 같이 하십시오.

1. **연결 또는 조인** 창에서 **작업 선택**을 클릭합니다.
2. 목록에서 다음과 같은 연산자를 선택합니다. **외부 조인**, **내부 조인**, **왼쪽 조인** 또는 **오른쪽 조인**.

테이블 조인

다음과 같이 하십시오.

- **연결 또는 조인** 창에서 **적용**을 클릭합니다.

이제 테이블이 조인됩니다.

조인된 테이블 분할

더 이상 조인이 필요하지 않은 경우 조인된 테이블을 소스 테이블로 분할할 수 있습니다.

> *조인된 테이블을 분할하면 조인된 테이블에 있던 연결뿐만 아니라 기본 테이블과 보조 테이블 간의 연결도 제거됩니다. 조인된 테이블을 분할하는 동안 연결을 유지하려면*↩*을 클릭하여 테이블을 분할하는 대신 조인을 실행 취소합니다.* ***데이터 관리자****에서 데이터를 로드한 후에는*↪*을 사용하여 조인을 실행 취소할 수 없습니다.*

조인된 테이블 분할

다음과 같이 하십시오.

1. 조인된 테이블을 선택합니다.
2. ⇹를 클릭합니다.

이제 테이블이 해당 소스 테이블로 분할되었습니다. 소스 테이블의 모든 필드는 사전 조인 이름을 갖습니다. 조인된 테이블을 분할하면 한 수준의 조인만 분할하므로 분할 조인된 테이블에 속한 조인된 테이블에는 자체 조인이 유지됩니다.

3.15 **데이터 관리자**에서 테이블 및 필드 변환 세부 정보 보기

데이터 관리자에서 **세부 정보** 대화 상자를 사용하여 테이블 및 필드에서 수행되는 작업 및 변환을 볼 수 있습니다. **세부 정보** 대화 상자는 테이블의 경우 **연결** 및 **테이블** 보기, 필드의 경우 데이터 테이블 편집기에서 제공됩니다.

세부 정보에는 선택한 테이블 또는 필드의 현재 작업 및 변환이 생성된 데이터 로드 스크립트에 적용되는 순서대로 표시됩니다. 이를 통해 테이블 또는 필드의 소스, 현재 변경된 사항 및 변경 적용 순서를 쉽게 확인할 수 있습니다. 예를 들어 **세부 정보**를 사용하여 어떤 테이블이 연결되었는지 또는 필드가 재정렬되었는지 쉽게 확인할 수 있습니다.

세부 정보에 표시되는 정보는 테이블을 보는 중인지, 필드를 보는 중인지에 따라 다릅니다. 테이블 **세부 정보**에는 다음 사항이 표시됩니다.

- 선택한 테이블의 소스 테이블
- 테이블에 사용된 변환(예: 피벗 해제 및 연결)

필드 **세부 정보**에는 다음 사항이 표시됩니다.

- 선택한 필드의 소스 테이블 및 소스 필드
- 필드 유형 변경
- 필드에 사용된 변환(예: 데이터 프로파일링 카드에서 또는 연결에서)

테이블 세부 정보 보기

다음과 같이 하십시오.

- **데이터 관리자**에서 테이블을 선택하고, •••을 클릭하고, **세부 정보 보기**를 클릭합니다.

세부 정보 대화 상자가 열립니다.

필드 세부 정보 보기

다음과 같이 하십시오.

1. **데이터 관리자**에서 테이블을 선택하고 을 클릭합니다.
2. 필드 머리글 위에 있는 ☰ 을 클릭하고 **세부 정보 보기**를 클릭합니다.

세부 정보 대화 상자가 열립니다.

3.16 단계별 연습 - 강제 연결을 사용하여 테이블 결합

이 단계별 연습에서는 강제 연결을 사용하여 두 개의 비슷한 데이터 테이블을 결합하는 방법을 보여 줍니다.

강제 연결은 시트에서 분석에 사용하기 전에 데이터를 정리하는 데 사용할 수 있습니다. 두 개의 테이블을 하나의 테이블로 연결할 수 있습니다. 나중에 다른 테이블을 추가할 수도 있습니다(예: 6월부터 테이블을 처음으로 추가한 다음 7월부터 두 번째 테이블을 추가하려는 경우).

연결 개요

- Qlik Sense에서 하나 이상의 추가된 테이블에 동일한 수의 필드와 동일한 필드 이름이 다른 테이블에 있음을 감지하면 테이블이 데이터 관리자에서 자동으로 연결됩니다. 이 경우 필요하면 테이블을 분할할 수 있습니다.
- 두 개의 테이블이 동일한 필드 또는 데이터를 완전히 공유하지 않을 경우 강제로 연결할 수 있습니다. 두 개의 테이블만 강제로 연결할 수 있습니다. 예를 들어 세 개의 테이블을 연결하려면 처음 두 개의 테이블을 하나의 테이블로 연결합니다. 만들어진 테이블에 세 번째 테이블을 연결합니다.
- 아주 유사한 테이블이 아니면 자동으로 연결되지 않습니다. 강제로 연결할 수도 없습니다. 이 경우 테이블의 필드가 데이터 관리자에서 대신 연결되어야 합니다.

연습 - 강제 연결

다음은 연습을 완료하는 데 필요한 작업입니다.

1. 데이터 테이블 준비
2. 앱에 데이터 테이블 추가
3. 데이터 테이블 연결 및 앱으로 데이터 테이블 로드
4. 추가 단계 - 새 테이블 추가 및 데이터 필드 연결

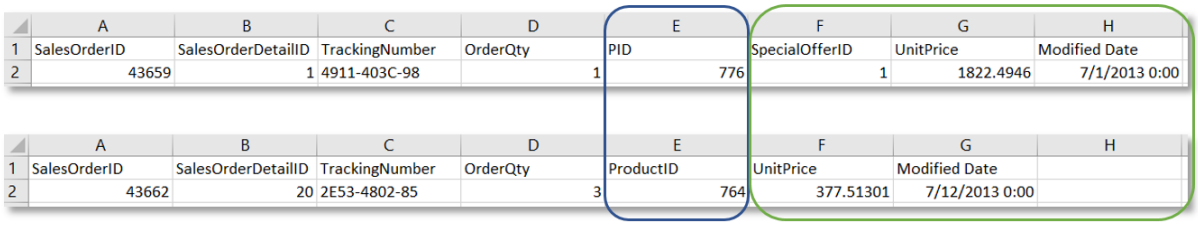
전제 조건

Qlik Sense에서 앱을 만드는 방법을 알아야 합니다.

데이터 테이블 준비

강제 연결을 설명하는 데 사용할 몇 가지 샘플 데이터가 있습니다. 두 개의 테이블에서 필드 및 데이터가 거의 동일하다면 자체 데이터를 사용할 수도 있습니다.

예를 들어 아래와 같이 데이터의 헤더 및 첫 번째 행이 있습니다. 두 개의 Excel 테이블에 붙여넣었습니다. 필드의 차이점에 주의합니다.



	A	B	C	D	E	F	G	H
1	SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	PID	SpecialOfferID	UnitPrice	Modified Date
2	43659	1	4911-403C-98	1	776	1	1822.4946	7/1/2013 0:00

	A	B	C	D	E	F	G	H
1	SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	ProductID	UnitPrice	Modified Date	
2	43662	20	2E53-4802-85	3	764	377.51301	7/12/2013 0:00	

If you want to use the sample data, copy the entire table, including the column headings, into an empty Excel file on your computer. For this walkthrough, we named the Excel tabs *Data Table 1* and *DataTable 2*. We named the Excel file *Concatenate_Data.xlsx*.

데이터 테이블 1

데이터 테이블 1

SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	PID	SpecialOfferID	UnitPrice	수정 날짜
43659	1	4911-403C-98	1	776	1	1822.4946	7/1/2013 0:00
43659	2	4911-403C-98	3	777	1	1822.4946	7/2/2013 0:00
43659	3	4911-403C-98	1	778	1	1098.9	7/2/2013 0:00
43659	4	4911-403C-98	1	771	1	1835.9946	7/2/2013 0:00
43659	5	4911-403C-98	1	772	1	1835.9946	7/3/2013 0:00
43659	6	4911-403C-98	2	773	1	1835.9946	7/3/2013 0:00
43659	7	4911-403C-98	1	774	1	1835.9946	7/4/2013 0:00

SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	PID	SpecialOfferID	UnitPrice	수정 날짜
43659	8	4911-403C-98	3	714	1	25.95636	7/4/2013 0:00
43659	9	4911-403C-98	1	716	1	25.95636	7/5/2013 0:00
43659	10	4911-403C-98	6	709	1	5.13	7/5/2013 0:00
43659	11	4911-403C-98	2	712	1	4.66785	7/6/2013 0:00
43659	12	4911-403C-98	4	711	1	18.16785	7/6/2013 0:00
43660	13	6431-4D57-83	1	762	1	377.51301	7/7/2013 0:00
43660	14	6431-4D57-83	1	758	1	787.3146	7/7/2013 0:00
43661	15	4E0A-4F89-AE	1	745	1	728.784	7/7/2013 0:00
43661	16	4E0A-4F89-AE	1	743	1	643.23387	7/8/2013 0:00
43661	17	4E0A-4F89-AE	2	747	1	643.23387	7/8/2013 0:00
43661	18	4E0A-4F89-AE	4	712	1	4.66785	7/8/2013 0:00
43661	19	4E0A-4F89-AE	4	715	1	25.95636	7/9/2013 0:00

DataTable 2

데이터 테이블 2

SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	ProductID	UnitPrice	수정 날짜
43662	20	2E53-4802-85	3	764	377.513001	7/12/2013 0:00
43662	21	2E53-4802-85	5	770	377.513001	7/12/2013 0:00
43662	22	2E53-4802-85	2	730	165.54438	7/13/2013 0:00

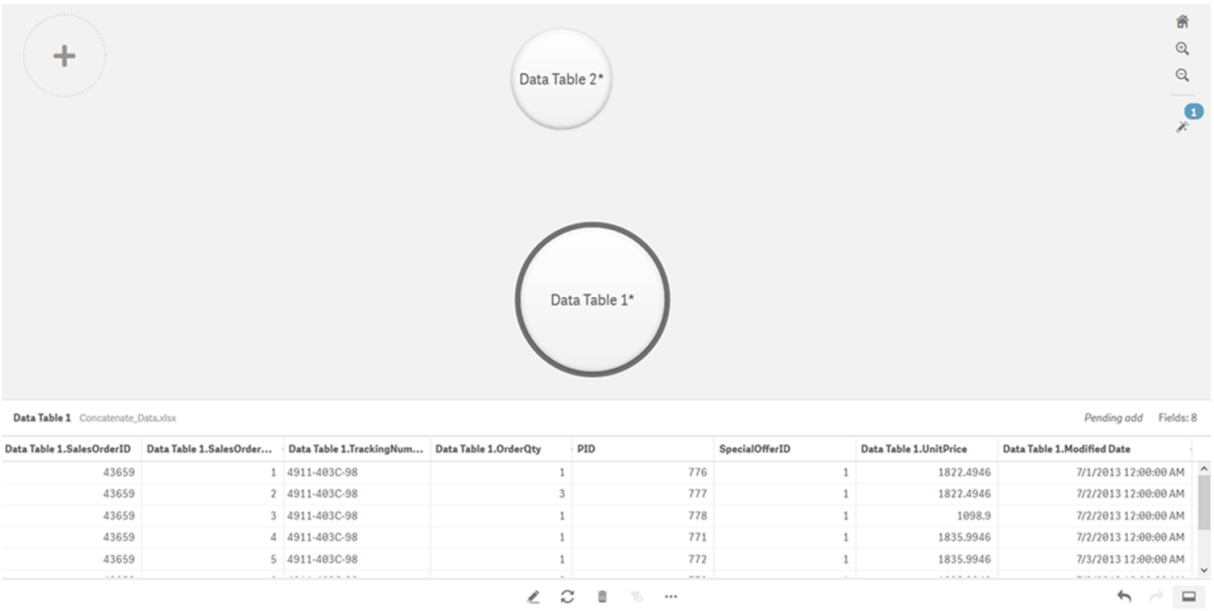
SalesOrderID	SalesOrderDetailID	TrackingNumber	OrderQty	ProductID	UnitPrice	수정 날짜
43662	23	2E53-4802-85	4	754	787.3146	7/14/2013 0:00
43662	24	2E53-4802-85	3	725	165.54438	7/14/2013 0:00
43662	25	2E53-4802-85	5	762	377.51301	7/14/2013 0:00
43662	26	2E53-4802-85	3	765	377.51301	7/14/2013 0:00
43662	27	2E53-4802-85	2	768	377.51301	7/15/2013 0:00
43662	28	2E53-4802-85	1	753	1932.2658	7/15/2013 0:00
43663	29	2E53-4802-85	1	756	787.3146	7/16/2013 0:00
43663	30	2E53-4802-85	3	763	377.51301	7/17/2013 0:00
43664	31	2E53-4802-85	1	732	321.2082	7/18/2013 0:00
43664	32	2E53-4802-85	6	758	787.3146	7/19/2013 0:00
43665	33	2E53-4802-85	1	729	165.54438	7/19/2013 0:00

앱에 데이터 테이블 추가

다음과 같이 하십시오.

1. Qlik Sense를 시작합니다.
2. 작업 영역에서 **새 앱 만들기**를 클릭합니다. **새 앱 만들기** 창이 열립니다.
3. 앱 이름을 지정한 다음 **만들기**를 클릭합니다. 앱이 생성됩니다. 앱 이름을 *ConcatenateExample*로 지정했습니다.
4. **앱 열기**를 클릭합니다. 앱이 열리고 데이터를 추가할 수 있는 대화 상자가 표시됩니다.
5. **Excel** 파일을 **파일 및 기타 소스에서 데이터 추가** 대화 상자로 끌어서 놓습니다. **데이터 관리자**의 **연결** 보기에 테이블이 표시됩니다. 해당 테이블의 데이터를 보려면 거품을 클릭합니다.

***데이터 관리자**에서 대신 데이터를 추가하는 경우 **데이터 관리자**의 **연결** 보기를 수행하기 전에 먼저 테이블 필드를 선택해야 합니다. 이 경우 두 테이블의 모든 필드를 선택합니다.*

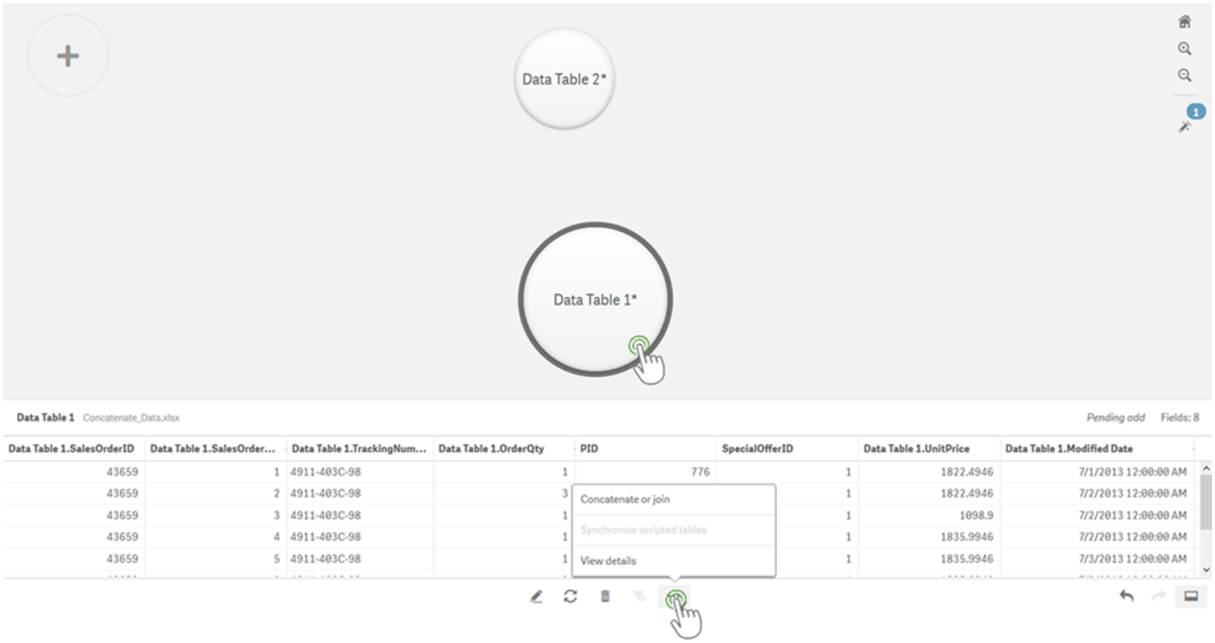


테이블 연결 및 앱으로 데이터 테이블 로드

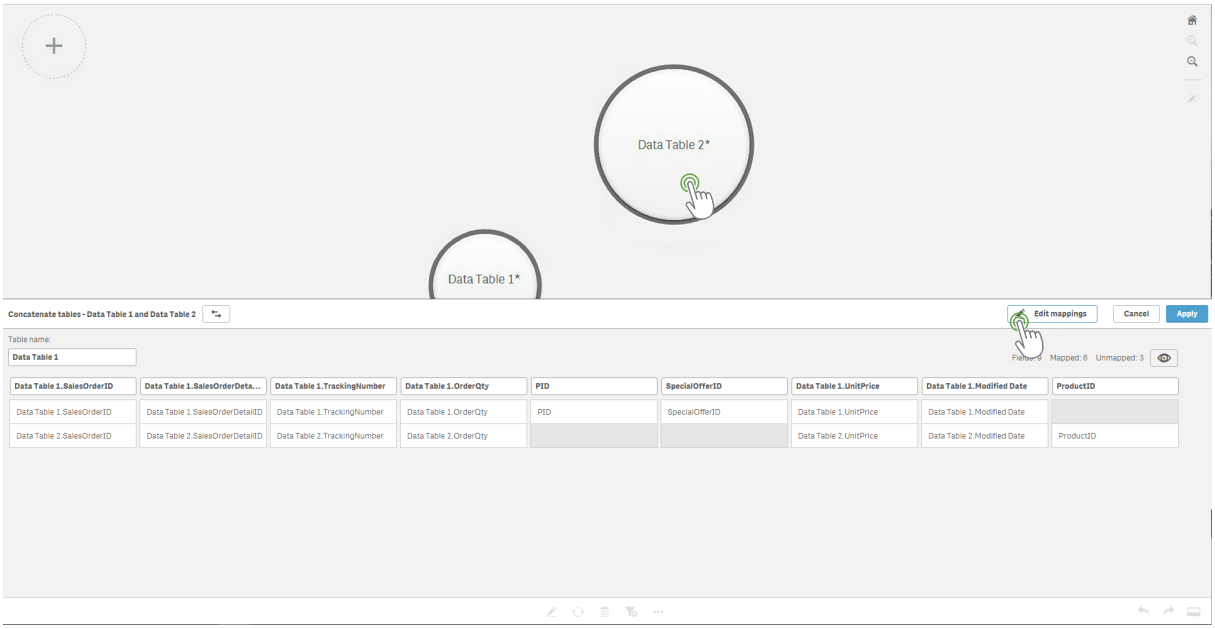
앱에 데이터 테이블이 추가된 후 테이블을 연결할 수 있습니다.

다음과 같이 하십시오.

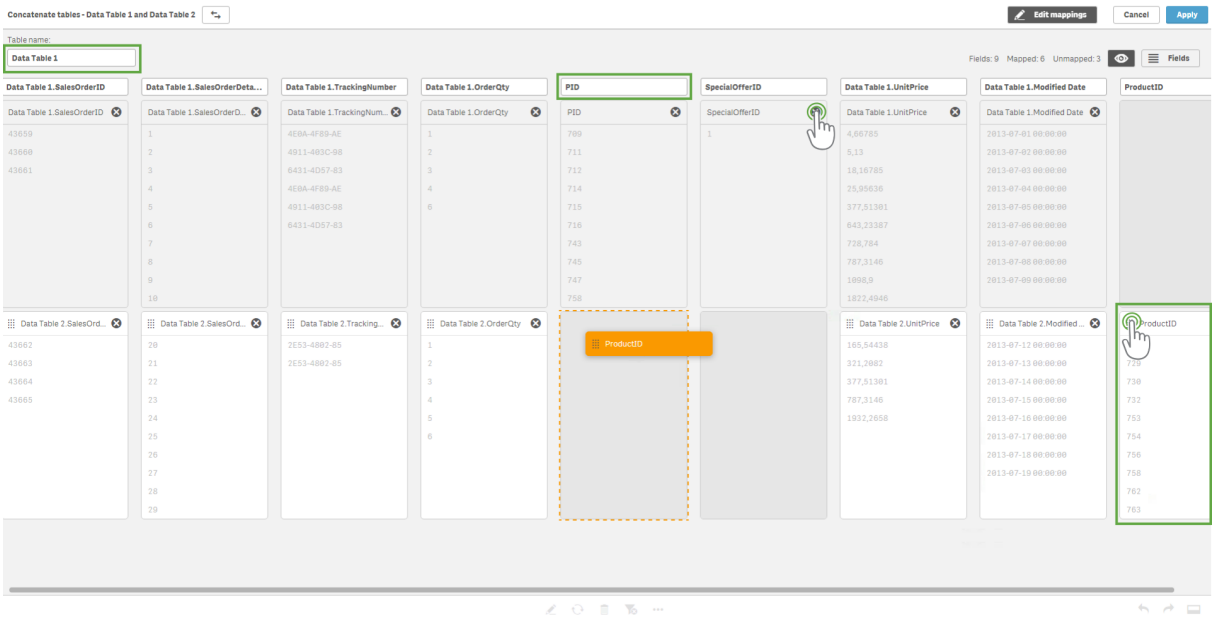
1. **데이터 관리자**의 **연결** 보기에서 거품을 클릭하여 하나의 테이블을 선택합니다. •••을 클릭한 다음 **연결 또는 조인**을 선택합니다.



2. 다른 테이블에 대한 거품을 클릭한 다음 **매핑 편집**을 클릭합니다.



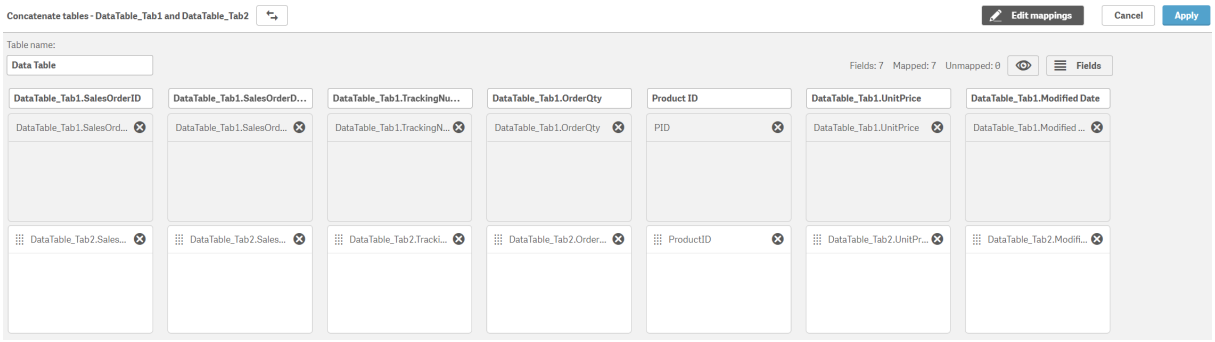
3. 이제 필요에 따라 다음을 수행할 수 있습니다.
   i. **테이블 이름**에서 테이블을 결합할 때 만들어지는 테이블의 이름을 지정합니다.
   ii. 필드 레이블을 끌어서 놓아 필드를 결합합니다.
   iii. 필드 이름을 바꿉니다.
   iv. 필드의 ⊗를 클릭하여 필드를 삭제합니다.



예제에서 수행한 작업은 다음과 같습니다.

i. 테이블 이름을 Data Table로 지정
ii. ProductID 레이블 및 필드를 PID 필드 아래로 끌어서 필드 결합
iii. PID 필드의 이름을 Product ID로 변경
iv. SpecialOfferID 필드 삭제

이제 테이블은 다음과 같이 보입니다.



4. **선택 작업** 드롭다운에서 **연결**을 클릭한 다음 **적용**을 클릭합니다. 테이블이 매핑된 필드에서 연결됩니다. *는 데이터가 아직 앱으로 로드되지 않았음을 나타냅니다.

5. **데이터 로드**를 클릭합니다. 데이터가 성공적으로 로드되었음을 나타내는 메시지가 표시됩니다. 데이터를 사용하여 시각화를 만들려면 **시트 편집**을 클릭합니다.

추가 단계 - 새 테이블 추가 및 데이터 필드 연결

위에서 제공된 샘플 데이터를 동일한 Excel 파일의 두 개의 탭에 붙여넣었습니다. 그러나 필드를 연결할 때 테이블이 같은 파일에 있을 필요는 없습니다. 테이블은 앱에 추가된 별도의 파일에 있을 수 있습니다. 나중에 다른 테이블을 추가할 수도 있습니다(예: 6월부터 테이블을 처음으로 추가한 다음 7월부터 두 번째 테이블을 추가하려는 경우).

이 예제에서는 위에서 만든 연결 테이블에 유사한 필드를 가진 다른 테이블을 추가합니다.

샘플 데이터는 다음과 같습니다. 테이블이 포함된 탭의 이름을 *DataTable_Newest*로 지정하고 데이터 파일의 이름을 *Concatenate_Data2.xlsx*로 지정합니다.

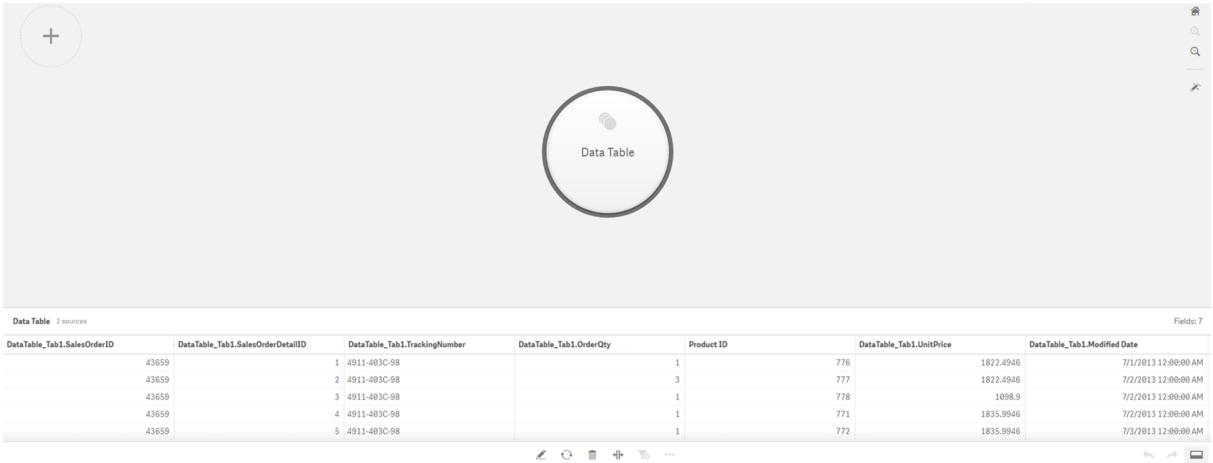
DataTable_Newest

DataTable_Newest

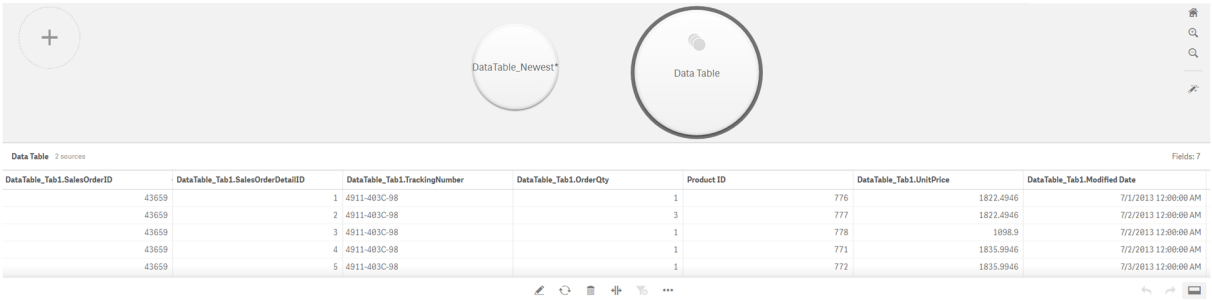
SalesOrderID	SalesOrderDetailID	TrackingNumber	ZIP	OrderQty	ID	UnitPrice	수정 날짜
43666	34	568E-472E-9C	20012	3	764	377.51301	7/12/2013 0:00
43666	34	568E-472E-9C	23220	5	770	377.51301	7/12/2013 0:00
43667	35	AB6C-4FF9-9D	30004	2	730	165.54438	7/13/2013 0:00
43668	36	AB6C-4FF9-9D	11215	4	754	787.3146	7/14/2013 0:00
43668	36	AB6C-4FF9-9D	55401	3	725	165.54438	7/14/2013 0:00
43668	36	AB6C-4FF9-9D	20003	5	762	377.51301	7/14/2013 0:00
43668	36	AB6C-4FF9-9D	15213	3	765	377.51301	7/14/2013 0:00
43669	37	C618-4998-BE	33125	2	768	377.51301	7/15/2013 0:00
43669	37	C618-4998-BE	11215	1	753	1932.2658	7/15/2013 0:00

다음과 같이 하십시오.

1. Qlik Sense 허브에서 위의 절차로 만든 앱을 클릭합니다. 앱이 열립니다.
2. 상단의 도구 모음에 있는 드롭다운 목록에서 **데이터 관리자**를 선택합니다. **데이터 관리자**가 열리고 위의 절차로 만든 테이블이 표시됩니다.



3. 데이터를 추가하려면 + 버튼을 클릭합니다.
4. 새 Excel 파일을 **이 앱에 파일 첨부** 대화 상자로 끌어서 앱에 추가합니다.
   데이터 추가 창이 열립니다.
5. **데이터 추가**를 클릭하여 앱에 데이터 테이블을 추가합니다.
   새 테이블이 앱에 추가됩니다.



6. 이제 테이블을 연결하고, 매핑을 편집하고, 데이터를 로드할 수 있습니다.

3.17 데이터 관리자에서 스크립팅된 테이블 동기화

기본적으로 데이터 로드 편집기에 추가된 스크립팅된 테이블은 **데이터 관리자**에서 사용할 수 있는 도구를 사용할 수 없습니다.

예를 들어 데이터 프로파일링 카드를 사용하여 스크립팅된 테이블의 필드를 변환하거나 **데이터 관리자**의 다른 테이블에 스크립팅된 테이블을 연결할 수 없습니다. **데이터 관리자**에서 스크립팅된 테이블을 동기화하는 경우 **데이터 관리자**의 스크립팅된 테이블을 관리되는 스크립팅된 테이블로 바꿀 수 있습니다. 이러한 테이블은 다음을 포함하여 **데이터 관리자**에 추가된 테이블과 동일한 모든 도구에 액세스할 수 있습니다.

- 테이블 편집(예: 계산 필드 추가)
- 필드 변환(예: 필드 유형 변경 또는 데이터 프로파일링 카드로 필드 변환)
- 테이블 변환(예: 테이블 피벗 해제 또는 연결)

동기화 및 관리되는 스크립팅된 테이블에는 다음과 같은 제한 사항이 있습니다.

- 스크립팅된 테이블은 관리되는 스크립팅된 테이블로 동기화되도록 데이터 로드 스크립트의 **자동 생성 섹션** 앞에 있어야 합니다. 데이터 로드 스크립트의 **자동 생성 섹션** 뒤의 테이블은 동기화되지 않습니다.
- **소스에서 데이터 선택**을 사용하여 관리되는 스크립팅된 테이블에서 필드 선택을 변경할 수 없습니다.

> *데이터 로드 스크립트에 **Exit** 문이나 동적 필드가 포함되어 있으면 스크립팅된 테이블을 동기화하지 마십시오.*

스크립팅된 테이블을 관리되는 스크립팅된 테이블로 변환하려면 **데이터 관리자**에서 스크립팅된 테이블을 동기화합니다. 동기화는 다음을 수행합니다.

- 모든 동기화된 스크립팅된 테이블을 관리되는 스크립팅된 테이블로 바꿉니다.
- 데이터 로드 스크립트에서 스크립팅된 테이블이 제거된 모든 관리되는 스크립팅된 테이블을 삭제합니다.
- 데이터 로드 스크립트에서 소스 테이블이 변경된 관리되는 스크립팅된 테이블을 업데이트합니다.

> *테이블을 동기화한 경우 다른 탭에서 **데이터 관리자**를 연 상태로 데이터 로드 편집기에서 변경해서는 안 됩니다.*

> ***데이터 관리자**에서 이미 동기화된 테이블에 대한 데이터 로드 스크립트를 변경하지 마십시오. 데이터 로드 편집기에서 필드를 제거하거나 수정하는 경우 동기화된 테이블에서 파생된 필드 또는 연결을 삭제하거나 다시 실행해야 합니다. 계산 필드 또는 **분할** 카드로 만든 필드와 같이 제거되거나 수정된 필드를 사용하는 파생된 필드는 Null 값을 표시합니다.*

동기화 후에는 **데이터 관리자**에서 다른 테이블처럼 관리되는 스크립팅된 테이블을 사용할 수 있습니다. **데이터 관리자**는 관리되는 스크립팅된 테이블과 소스 스크립팅된 테이블 간 차이점을 감지하면 다시 동기화하라는 메시지를 표시합니다.

관리되는 스크립팅된 테이블을 스크립팅된 테이블로 다시 변경하려면 **데이터 관리자**에서 삭제합니다. 다시 동기화하려면 삭제를 반복해야 합니다.

스크립팅된 테이블 동기화

다음과 같이 하십시오.

1. **데이터 관리자**에서 •••을 클릭합니다.
   또는 스크립팅된 테이블을 선택합니다.
2. **스크립팅된 테이블 동기화**를 클릭합니다.

관리되는 스크립팅된 테이블이 **데이터 관리자**의 모든 스크립팅된 테이블을 바꿉니다.

관리되는 스크립팅된 테이블 제거

다음과 같이 하십시오.

1. **데이터 관리자**에서 **테이블** 보기를 선택합니다.
2. 제거할 관리되는 스크립팅된 테이블에서 🗑을 클릭합니다.
3. **데이터 로드**를 클릭합니다.

관리되는 스크립팅된 테이블이 다시 스크립팅된 테이블로 변경됩니다.

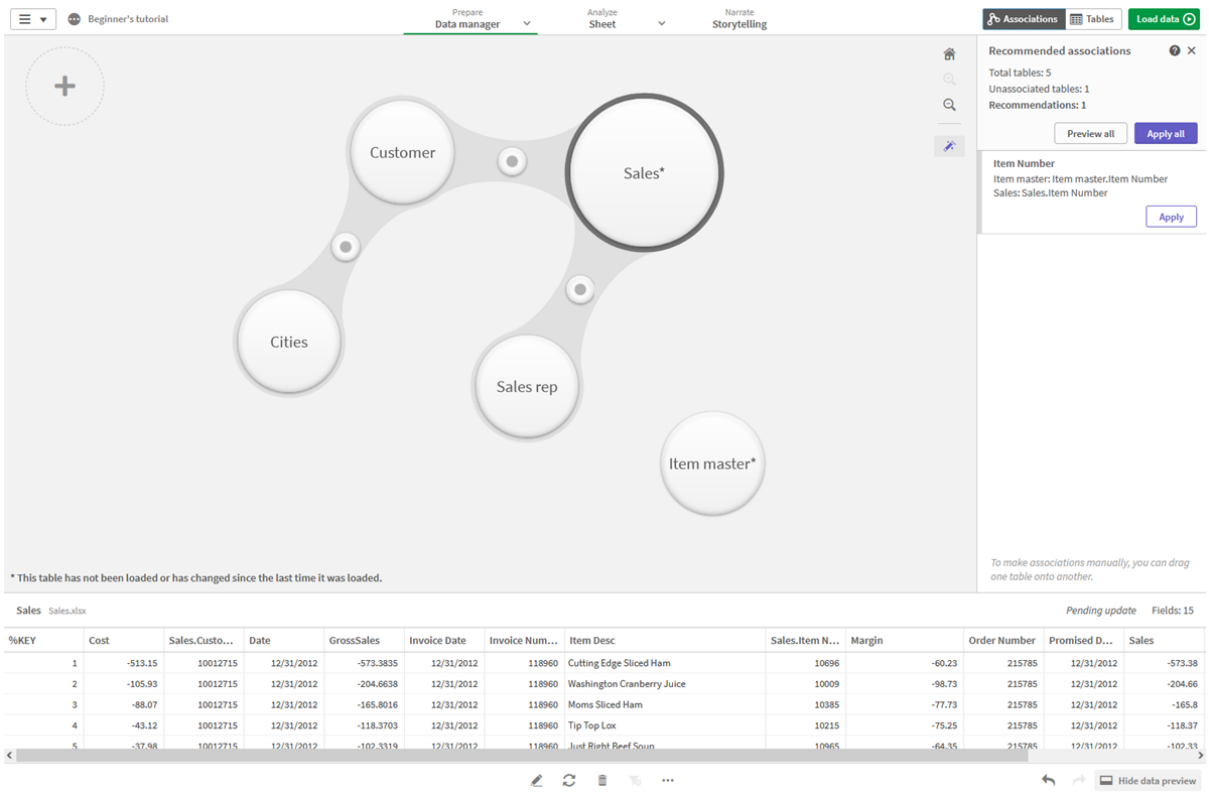
3.18 데이터 연결 관리

Qlik Sense에서 데이터를 프로파일링하여 테이블 간의 연결을 만들 수 있습니다. Qlik Sense Insight Advisor의 분석 기반 권장 사항에서 선택하여 연결을 만들거나 직접 만들 수 있습니다.

데이터를 연결하려면 데이터 프로파일링이 활성화된 상태에서 **데이터 추가** 옵션을 사용하는 것이 좋습니다. 이 옵션이 기본 옵션입니다. 앱 데이터 페이지의 오른쪽 아래 모서리에 있는 **데이터 추가** 버튼 옆에 있는 •••을 클릭하여 이 설정을 확인할 수 있습니다.

데이터 관리자의 **연결** 보기에서 데이터는 거품으로 묘사되며, 각 거품은 데이터 테이블을 나타냅니다. 거품의 크기는 테이블 내 데이터의 양을 나타냅니다. 거품 간의 링크는 테이블 간의 연결을 나타냅니다. 두 테이블 간의 연결인 경우에는 링크의 버튼을 선택하여 해당 연결을 보거나 편집할 수 있습니다.

데이터 관리자에서 연결 보기



> *대부분의 경우, 모델 보기에서 테이블 연결을 편집하는 것이 훨씬 쉽지만 테이블 편집 보기에서 **연결** 옵션을 사용하여 단일 테이블 연결을 편집할 수도 있습니다.*
>
> *자세한 내용은 테이블 편집기에서 데이터 연결 (page 39)을 참조하십시오.*

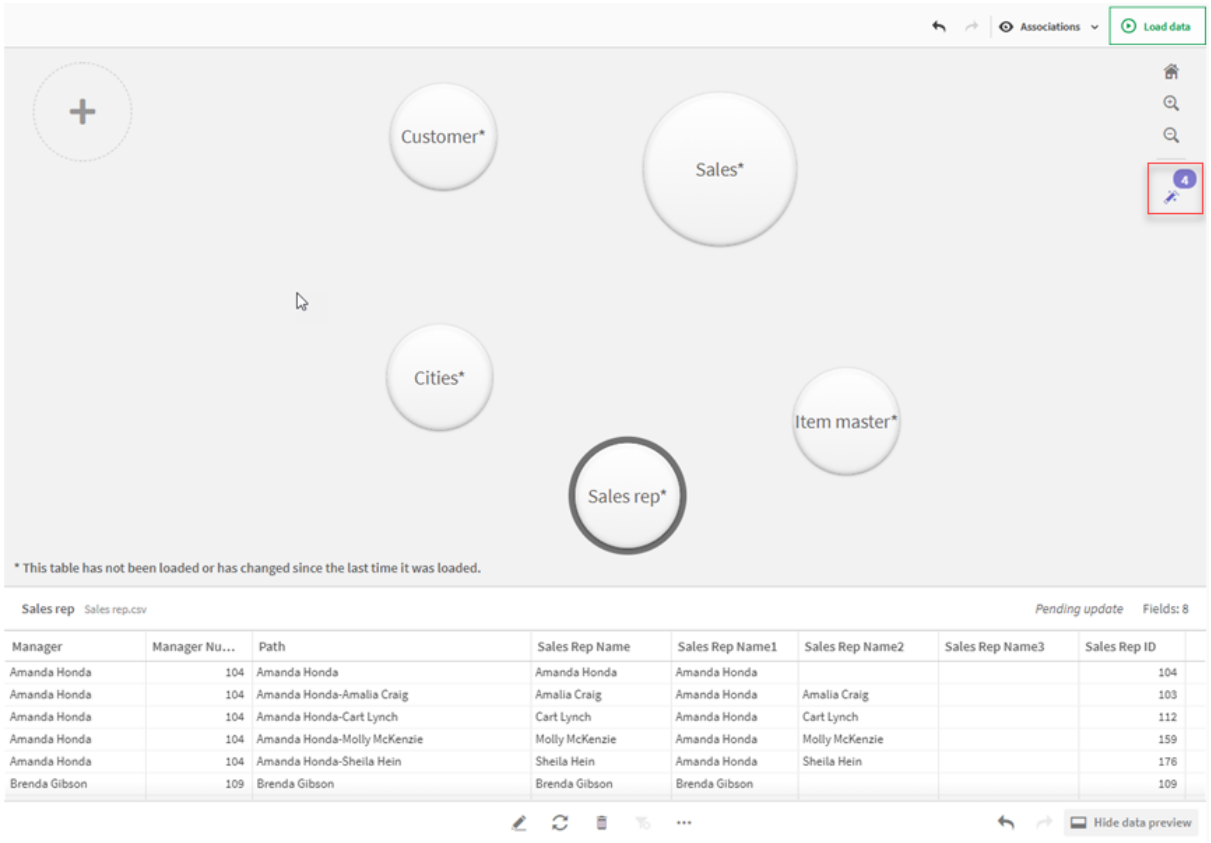
권장 연결 패널을 사용하는 테이블 연결

많은 경우 Qlik Sense Insight Advisor은 데이터 테이블 간의 연결을 권장합니다. **권장 연결** 패널을 통해 이러한 권장 사항을 보고 적용할 수 있습니다.

테이블이 있으면 기본적으로 **권장 연결** 패널이 열립니다. 닫았다가 을 클릭하여 다시 열 수 있습니다.

패널이 닫힌 상태에서 권장 사항이 있는 경우 권장 사항 수를 나타내는 위에 배지가 표시됩니다.

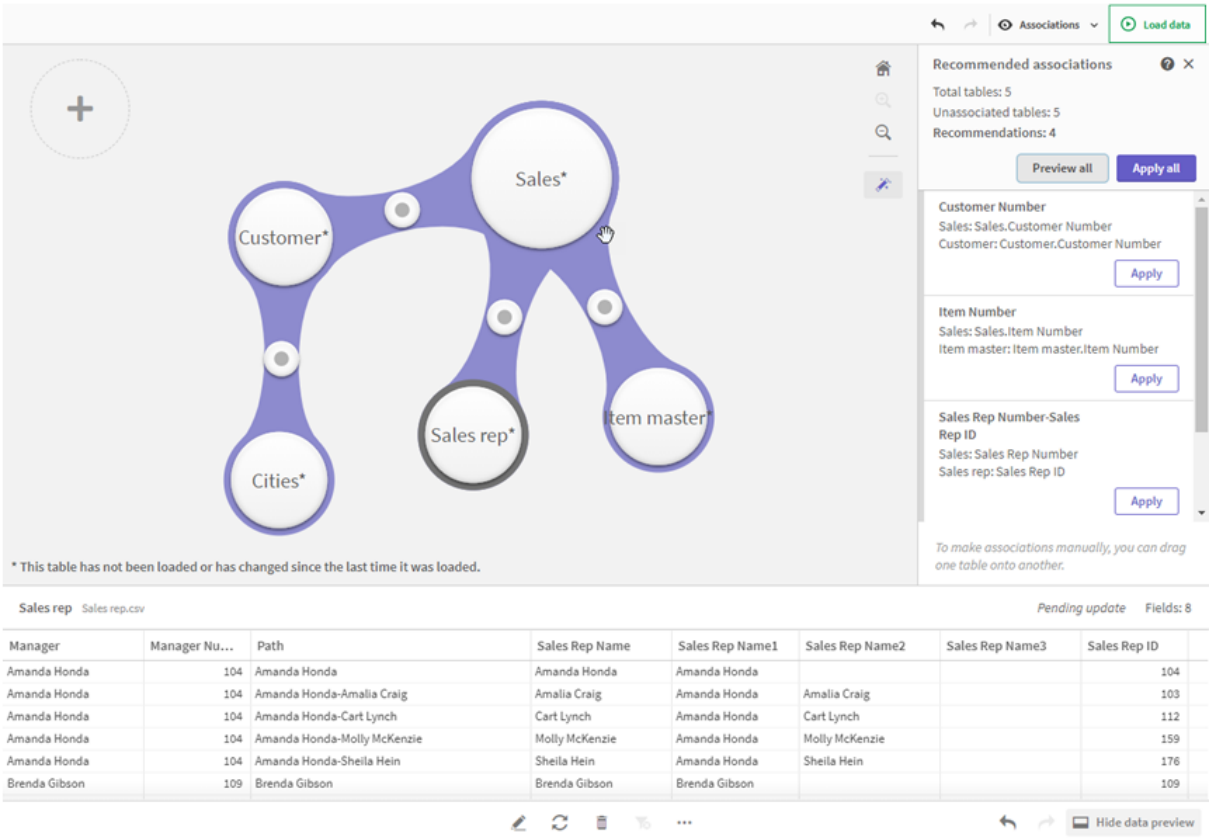
권장 연결 열기



다음과 같이 하십시오.

1. **권장 연결** 패널이 닫혀 있는 경우 연결 보기의 오른쪽 위 모서리에서 을 클릭합니다.
   패널이 오른쪽에 표시됩니다.
2. 다음 정보가 표시됩니다.
   - **총 테이블**: 테이블 총 수입니다.
   - **연결되지 않은 테이블**: 연결 관계가 없는 테이블 총 수입니다.
   - **권장 사항**: 권장 연결 총 수입니다.
   - 권장 연결 세부 정보: 권장 연결의 이름, 테이블 및 필드 이름을 콜론으로 구분해서 표시합니다.
3. 단일 권장 사항을 클릭하면 짙은 파란색으로 미리 볼 수 있습니다.

4. 권장 사항 일부만 허용하려면 필요한 특정 권장 사항의 **적용** 버튼을 클릭합니다.
5. 모든 권장 연결이 데이터 테이블에 어떻게 영향을 미치는지 보려면 **모두 미리 보기**를 클릭합니다. 미리 보는 연결은 강조 표시됩니다.
   권장 연결 패널



6. 모든 권장 연결을 적용하려면 **모두 적용**을 클릭합니다. 허용된 연결은 연한 회색으로 강조 표시됩니다.
   테이블이 어떻게 변경되는지 보려면 화면 맨 아래 있는 ▭을 클릭하면 됩니다.

이제 데이터를 사용하여 시각화 만들기를 시작할 수 있습니다.

수동으로 테이블 연결

테이블을 함께 끌면 수동으로 연결할 수 있습니다. 테이블 거품을 서로 마주 보게 끌면 녹색, 주황색 또는 빨간색 줄무늬가 표시됩니다.

- 녹색: **데이터 관리자**가 연결할 필드에 대해 매우 확신합니다. 예를 들어 두 개의 테이블에 "Sales Region"이라는 레이블이 지정된 필드가 있는 경우 **데이터 관리자**는 이 두 테이블이 연결되어야 한다고 가정합니다.
- 주황색: **데이터 관리자**가 이러한 테이블이 연결될 수 있다는 것을 꽤 확신합니다. 예를 들어 서로 다른 두 개의 필드가 레이블이 서로 다르게 지정되었지만 한자리 숫자가 포함된 경우 **데이터 관리자**는 이 둘의 데이터 유형이 비슷하기 때문에 주황색으로 플래그를 지정합니다.
- 빨간색: **데이터 관리자**가 이러한 테이블을 연결하는 방법을 모릅니다. **연결 테이블** 편집기에서 함께 연결할 테이블과 필드를 선택해야 합니다.

녹색 또는 주황색 테이블을 수동으로 연결하려고 합니다.

다음과 같이 하십시오.

1. 하나의 테이블을 녹색 또는 주황색으로 표시된 테이블 중 하나로 끕니다.
2. 연결이 자동으로 적용됩니다.

이제 테이블이 권장 필드를 사용하여 연결됩니다.

빨간색 테이블을 수동으로 연결하려고 합니다.

사용자 지정 연결을 만들면 계속 테이블을 연결할 수 있습니다.

다음과 같이 하십시오.

1. 하나의 테이블을 빨간색으로 표시된 테이블 중 하나로 끕니다.
   테이블 연결 편집기가 열립니다.
2. 왼쪽 테이블에서 연결에 사용할 필드를 선택합니다.
   단일 필드 또는 복수의 필드를 선택합니다. 또한 구분 기호를 추가하여 데이터를 해석하기 쉽게 만들거나 이미 있는 필드와 일치하도록 만들 수 있습니다. 미리 보기에서 연결 데이터가 어떻게 표시되는지 확인할 수 있습니다.
3. 오른쪽 테이블에서 왼쪽 테이블의 선택 내용과 일치시키는 데 사용할 필드를 선택합니다.
4. 만들 키 필드의 이름을 **이름**에 입력합니다.
   이 새 필드 이름은 테이블의 기존 필드 이름과 같아서는 안됩니다.
5. **연결**을 클릭합니다.

이제 테이블이 사용자 지정 연결을 사용하여 연결됩니다.

연결 해제

두 가지 방법으로 데이터 모델에 잘 맞지 않는 연결을 해제할 수 있습니다.

다음과 같이 하십시오.

- 연결 테이블 중 하나를 클릭하고 연결이 해제될 때까지 다른 테이블에서 멀리 끕니다. 또는 다음과 같이 할 수 있습니다.
- 두 거품 사이의 링크를 클릭한 다음 아래 패널에서 **삭제** 버튼을 클릭합니다.

더 이상 두 테이블이 연결되지 않습니다.

연결 편집

데이터 모델을 조정해야 하는 경우 두 테이블 간의 기존 연결을 편집할 수 있습니다.

다음과 같이 하십시오.

1. 데이터 패널을 열려면 연결 테이블 사이의 원을 클릭합니다.
   연결된 필드의 데이터에 대한 미리 보기와 함께 패널이 열립니다.

2. 을 클릭합니다.
   하나 이상의 버튼이 표시되면 각각 녹색, 주황색 또는 빨간색으로 표시됩니다. 녹색은 **데이터 관리자**가 연결을 매우 확신한다는 것이고 주황색은 어느 정도 확신하며 빨간색은 확신하지 못한다는 것입니다. 현재 연결은 회색으로 표시됩니다.
3. 연결 버튼 중 하나를 클릭합니다.
   - 권장 연결을 클릭하여 선택합니다.
   - 기존 사용자 지정 연결()을 클릭하여 연결에서 사용할 필드를 편집합니다.
   - **사용자 지정 연결**을 클릭하여 새 연결을 만듭니다.
     이 버튼은 테이블 쌍에 대한 권장 연결이 있는 경우에만 사용할 수 있습니다.
     사용자 지정 연결은 단일 필드 또는 복수의 필드를 포함할 수 있습니다.

이제 테이블 쌍 사이의 연결이 변경되었습니다.

데이터 미리 보기

연결 보기에서 테이블을 미리 보면 데이터에 대해 더욱 잘 이해할 수 있습니다.

다음과 같이 하십시오.

1. 테이블을 선택합니다.
2. 보기 하단에서를 클릭합니다.

미리 보기 창이 테이블 데이터와 함께 표시됩니다.

가상 키

두 개 이상의 데이터 테이블에 두 개 이상의 공통적인 필드가 있는 경우 이는 복합 키 관계를 나타냅니다. Qlik Sense에서는 가상 키를 만들어 자동으로 이를 처리합니다. 이 키는 복합 키에서 발생하는 모든 조합을 나타내는 익명 필드입니다.

자세한 내용은 *가상 키 (page 124)*를 참조하십시오.

다음 경우 중 하나에서 테이블 결과를 추가하는 경우 프로파일링이 활성화된 데이터만 추가할 수 있습니다.

- 6개 이상의 필드가 포함된 가상 키를 만든 경우.
- 11개 이상의 가상 키를 만든 경우.
- 중첩 가상 키(다른 가상 키가 포함된 가상 키)를 만든 경우.

이러한 경우는 문제를 해결하기 위해 데이터 테이블을 조정해야 함을 나타냅니다.

제한 사항

로드된 테이블의 구조와 테이블에 있는 데이터의 구조에 따라 연결 권장 사항이 제공되지 않는 경우도 있습니다. 이런 경우 테이블 편집기에서 연결을 조정해야 합니다.

- 다대다 관계.

- 데이터와 필드 쌍이 양방향으로 제대로 일치하지 않습니다. 다른 방향의 일치가 확연히 작으면서 큰 테이블의 필드에 100% 일치하는 적은 수의 필드 값을 포함하는 작은 테이블이 있는 경우가 이에 해당될 수 있습니다.
- 복합 키 연결.

또한 **데이터 관리자**는 **데이터 추가**를 통해 추가된 테이블만 분석합니다. 데이터 로드 스크립트를 사용하여 추가된 테이블은 **데이터 관리자**로 동기화되지 않은 경우 연결 권장 사항에 포함되지 않습니다.

자세한 내용은 *데이터 관리자에서 스크립팅된 테이블 동기화 (page 86)*를 참조하십시오.

변경 적용 및 데이터 로드

데이터 관리자에서 수행한 변경은 데이터를 로드할 때까지 앱에서 사용할 수 없습니다. 데이터를 로드하면 변경 내용이 적용되고 추가한 새 데이터는 모두 외부 데이터 소스에서 로드됩니다. 이전에 로드한 데이터는 다시 로드되지 않습니다.

데이터 관리자 바닥글의 ⟳ 버튼을 사용하여 외부 데이터 소스에서 모든 데이터를 다시 로드할 수 있습니다.

⟳ 버튼은 선택한 테이블에 대한 모든 데이터를 다시 로드합니다. 앱의 모든 테이블에 대한 모든 데이터를 로드하지는 않습니다.

데이터 관리자의 데이터가 앱 데이터와 동기화되어 있지 않으면 **데이터 로드** 버튼이 녹색으로 표시됩니다. **연결** 보기에서 모든 새 테이블 또는 업데이트된 테이블은 *로 표시되고, 삭제된 테이블은 밝은 회색으로 표시됩니다. **테이블** 보기에서 모든 새 테이블, 업데이트된 테이블 또는 삭제된 테이블은 파란색으로 강조 표시되고 테이블 상태를 보여 주는 아이콘이 표시됩니다.

- **삭제 보류** 🗑로 표시된 테이블이 삭제됩니다.
- **업데이트 보류** ⟳로 표시된 테이블이 추가, 이름 변경 또는 제거된 필드로 업데이트되거나 테이블의 이름이 변경됩니다.
- **추가 보류** ✔로 표시된 테이블이 추가됩니다.

변경 사항 적용

다음과 같이 하십시오.

- **데이터 로드**를 클릭하여 앱의 변경 내용을 로드합니다.

이제 **데이터 관리자**에서 수행한 변경 사항으로 앱 데이터가 업데이트됩니다.

4 스크립트를 사용하여 데이터 로드 및 변환

이 소개에서는 데이터 로드 스크립트를 사용하여 Qlik Sense에 데이터를 로드하는 방법에 대해 간략하게 설명합니다.

Qlik Sense는 데이터 로드 편집기에서 관리되는 데이터 로드 스크립트를 사용하여 다양한 데이터 소스에서 데이터에 연결하고 데이터를 검색합니다. 데이터 파일(예: Excel 파일 또는 .csv 파일)이 데이터 소스가 될 수 있습니다. 데이터베이스(예: Google BigQuery 또는 Salesforce 데이터베이스)도 데이터 소스가 될 수 있습니다.

데이터 관리자를 사용하여 데이터를 로드할 수도 있지만, 데이터 로드 스크립트를 만들고 편집하고 실행하려면 데이터 로드 편집기를 사용합니다.

로드할 필드와 테이블은 스크립트에서 지정합니다. 스크립팅은 데이터 소스에서 로드할 데이터를 지정하는 데 사용되는 경우가 많습니다. 스크립트 문을 사용하여 데이터 구조를 편집할 수도 있습니다.

데이터 로드 중에 Qlik Sense는 다른 테이블에서 공통 필드(키 필드)를 식별하여 데이터를 연결합니다. 앱 내 데이터의 결과 데이터 구조는 데이터 모델 뷰어에서 모니터링할 수 있습니다. 필드의 이름 변경을 통해 테이블 사이에 다른 연결을 구축하여 데이터 구조를 변경할 수 있습니다.

데이터는 Qlik Sense에 로드된 후 앱에 저장됩니다.

앱이 데이터 소스에 직접 연결되지 않은 경우 Qlik Sense에서 항상 분석이 실행됩니다. 따라서 데이터를 새로 고치려면 스크립트를 실행하여 데이터를 다시 로드해야 합니다.

4.1 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용

데이터 관리자에서 데이터 테이블을 추가하면 데이터 로드 스크립트 코드가 생성됩니다. 데이터 로드 편집기의 **자동 생성 섹션**에서 스크립트 코드를 볼 수 있습니다. 또한 생성된 스크립트 코드를 잠금 해제하여 편집할 수 있지만, 그렇게 하면 더 이상 데이터 테이블을 **데이터 관리자**에서 관리할 수 없습니다.

기본적으로 로드 스크립트에서 정의한 데이터 테이블은 **데이터 관리자**에서 관리되지 않습니다. 즉, 데이터 개요에서 테이블을 볼 수 있지만 **데이터 관리자**에서 테이블을 삭제하거나 편집할 수는 없으며, 스크립트를 사용하여 로드된 테이블에는 연결 권장 사항이 제공되지 않습니다. 그러나 스크립팅된 테이블을 **데이터 관리자**와 동기화하면 스크립팅된 테이블이 관리되는 스크립팅된 테이블로 **데이터 관리자**에 추가됩니다.

*테이블을 동기화한 경우 다른 탭에서 **데이터 관리자**를 연 상태로 데이터 로드 편집기에서 변경해서는 안 됩니다.*

스크립트 섹션을 추가하고 **데이터 관리자**에서 만든 데이터 모델을 개선하고 이 데이터 모델과 상호 작용하는 코드를 개발할 수 있지만, 일부 주의해야 할 부분이 있습니다. 사용자가 작성한 스크립트 코드는 **데이터 관리자** 데이터 모델에 방해가 될 수 있으며, 다음과 같은 경우에 문제가 발생합니다.

- 스크립트에서 **데이터 관리자**를 사용하여 추가된 테이블의 이름을 바꾸거나 삭제하는 경우.
- **데이터 관리자**를 사용하여 추가된 테이블에서 필드를 삭제하는 경우.
- **데이터 관리자**를 사용하여 추가된 테이블과 스크립트에서 로드된 테이블을 연결하는 경우.

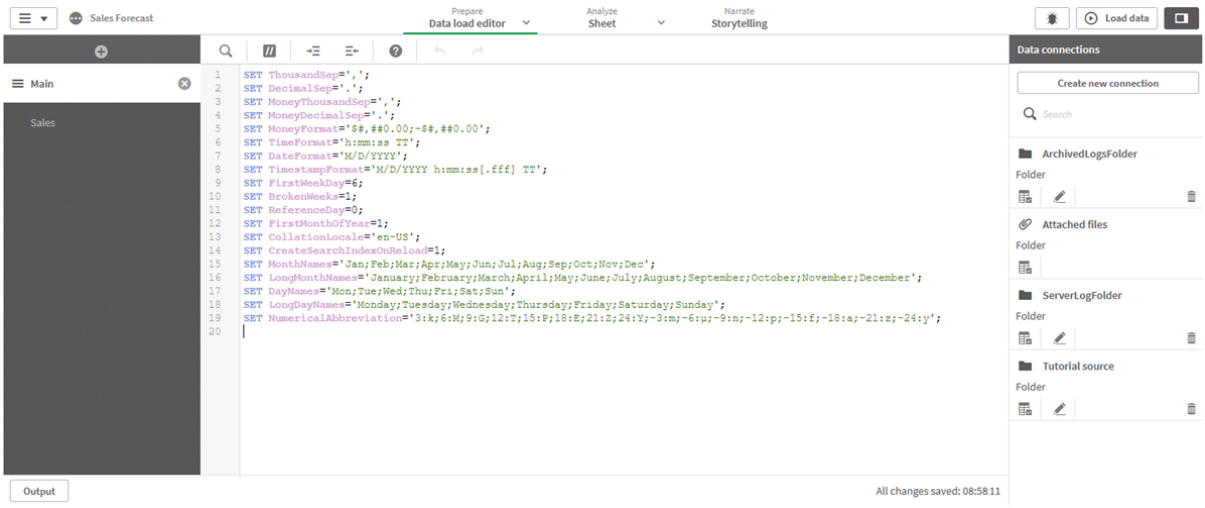
- **데이터 관리자**를 사용하여 추가된 테이블의 필드에서 **Qualify** 문을 사용하는 경우
- 스크립트에서 **Resident**를 사용하여 **데이터 관리자**를 통해 추가된 테이블을 로드하는 경우
- 생성된 코드 섹션 뒤에 스크립트 코드를 추가하는 경우. 데이터 모델에서의 결과 변경이 **데이터 관리자**에는 반영되지 않습니다.

4.2 데이터 로드 편집기 사용

데이터 로드 편집기에서 데이터 로드 스크립트를 만들어 실행하고 데이터 소스에 대한 연결을 만듭니다. 데이터를 로드했으면 분석을 위해 앱에서 사용할 수 있습니다.

탐색 막대의 **준비** 탭에서 **데이터 로드 편집기**를 클릭하여 데이터 로드 편집기를 엽니다.

데이터 로드 편집기



도구 모음

데이터 로드 편집기에서 가장 자주 사용되는 명령이 포함되어 있는 도구 모음입니다.

데이터 연결

데이터 연결에서 공통적으로 사용하는 데이터 소스(데이터베이스 또는 원격 파일)에 대한 바로 가기를 저장할 수 있습니다. 로드할 데이터에 대한 선택을 시작하는 곳이기도 합니다.

텍스트 편집기

텍스트 편집기에서 스크립트 코드를 작성하고 편집할 수 있습니다. 각 스크립트 행에는 번호가 지정되어 있으며 스크립트는 구문 구성 요소별로 다른 색으로 표시됩니다. 텍스트 편집기 도구 모음에는 **검색 및 바꾸기**, **도움말 모드**, **실행 취소** 및 **재실행** 명령이 포함되어 있습니다. 최초의 스크립트에는 필요에 따라 편집할 수 있는 일부 사전 정의된 지역 변수 설정(예: `SET ThousandSep=`)이 포함되어 있습니다.

섹션

읽고 유지 관리하기 쉽도록 스크립트를 섹션으로 구분합니다. 섹션은 맨 위에서 아래로 실행됩니다.

데이터 추가를 사용하여 데이터를 추가한 경우, 데이터를 로드하는 데 필요한 스크립트 코드가 포함되어 있는 **자동 생성 섹션**이라는 이름의 데이터 로드 스크립트 섹션이 생깁니다.

출력

출력에는 스크립트 실행 중에 생성된 모든 메시지 및 자동 저장 상태가 표시됩니다.

데이터 로드 편집기에서 데이터 소스에 연결

데이터 로드 편집기에서 데이터 연결을 통해 일반적으로 사용하는 데이터베이스, 로컬 파일 또는 원격 파일과 같은 데이터 소스에 대한 바로 가기를 저장할 수 있습니다.

데이터 연결은 영문자순으로 저장된 연결을 표시합니다. 검색 상자를 사용하면 특정 이름 또는 유형을 가진 연결로 목록을 좁힐 수 있습니다.

자신이 소유하거나 액세스 권한이 부여된 데이터 연결만 볼 수 있습니다. 필요한 경우 Qlik Sense 시스템 관리자에게 문의하여 액세스 권한을 요청하십시오.

새 데이터 연결 만들기

다음과 같이 하십시오.

1. **새 연결 만들기**를 클릭합니다.
2. 만들려는 데이터 소스 유형을 드롭다운 목록에서 선택합니다.
   선택한 데이터 소스 유형에 해당하는 설정 대화 상자가 열립니다.
3. 데이터 소스 설정을 입력하고 **만들기**를 클릭하여 데이터 연결을 만듭니다.
   연결 이름에는 사용자 이름과 도메인이 첨부되어 고유한 이름을 만듭니다.

이제 사용자가 기본 소유자인 데이터 연결이 생성되었습니다. 서버 설치에서 다른 사용자가 해당 연결을 사용할 수 있도록 하려면 Qlik Management Console에서 연결에 대한 액세스 권한을 편집해야 합니다.

생성된 연결의 설정은 데이터 소스 설정이 변경된 경우 자동으로 업데이트되지 않습니다. 따라서 사용자 이름과 비밀번호를 저장하는 경우 조심해야 하며, 특히 DSN에서 통합 Windows 보안 및 데이터베이스 로그인 간격 사이에 설정을 변경하는 경우 더욱 그렇습니다.

***새 연결 만들기**가 표시되지 않으면 데이터 연결을 추가할 수 있는 액세스 권한이 없는 것입니다. 필요한 경우 Qlik Sense 시스템 관리자에게 문의하여 액세스 권한을 요청하십시오.*

데이터 연결 삭제

다음과 같이 하십시오.

1. 삭제할 데이터 연결에서 를 클릭합니다.
2. 연결을 삭제할 것인지 확인합니다.

이제 데이터 연결이 삭제됩니다.

가 표시되지 않으면 데이터 연결을 삭제할 수 있는 액세스 권한이 없는 것입니다. 필요한 경우 Qlik Sense 시스템 관리자에게 문의하여 액세스 권한을 요청하십시오.

데이터 연결 편집

다음과 같이 하십시오.

1. **데이터 연결**에서 편집할 데이터 연결이 포함된 공간을 선택합니다.
2. 편집할 데이터 연결에서을 클릭합니다.
3. 데이터 연결 세부 정보를 편집합니다. 연결 세부 정보는 연결 유형마다 고유합니다. 연결의 자격 증명을 제공해야 할 수 있습니다.

이제 데이터 연결이 업데이트됩니다.

데이터 연결 이름을 편집하는 경우 동일한 연결을 계속 참조하려면 스크립트에서 해당 연결에 대한 기존 모든 참조(lib://)를 편집해야 합니다.

이 표시되지 않으면 데이터 연결을 업데이트할 수 있는 액세스 권한이 없는 것입니다. 필요한 경우 Qlik Sense 시스템 관리자에게 문의하십시오.

연결 문자열 삽입

대부분의 연결에서 연결 문자열이 필요합니다. 폴더 및 웹 파일 연결만 연결 문자열을 필요로 하지 않습니다.

이 기능은 Qlik Sense SaaS에서 사용할 수 없습니다.

다음과 같이 하십시오.

- 연결 문자열을 추가할 연결에서을 클릭합니다.

선택한 데이터 연결에 대한 연결 문자열이 데이터 로드 편집기의 현재 위치에 삽입됩니다.

데이터 연결에서 데이터 선택

데이터 연결에서 데이터를 선택하여 앱에서 로드하려면 다음을 수행합니다.

1. 데이터 소스에 연결되는 **새 연결을 만듭니다**(데이터 연결이 존재하지 않는 경우).
2. 연결에서 데이터를 선택합니다.

스크립트에서 데이터 연결 참조

데이터 연결을 사용하면 일반적으로 경로를 통해 파일 이름을 참조하는 스크립트의 문 및 함수에서 데이터 소스를 참조할 수 있습니다.

파일을 참조하기 위한 구문은 *'lib://(connection_name)/(file_name_including_path)'*입니다.

예: DataFiles 연결에서 파일 로드

이 예에서는 Folder 데이터 연결에 정의된 위치에서 *orders.csv* 파일을 로드합니다. 이 위치는 관리자가 Qlik Sense 서버에 만든 폴더일 수 있습니다.

```
LOAD * FROM 'lib://DataSource/orders.csv';
```

예: 하위 폴더에서 파일 로드

이 예에서는 DataSource 데이터 연결 폴더에서 *Customers/cust.txt* 파일을 로드합니다. Customers 는 데이터 연결에 정의된 위치의 하위 폴더입니다.

```
LOAD * FROM 'lib://DataSource/Customers/cust.txt';
```

예: 웹 파일에서 로드

이 예에서는 실제 URL에 대한 링크가 포함된 PublicData 웹 파일 데이터 연결에서 테이블을 로드합니다.

```
LOAD * FROM 'lib://PublicData' (html, table is @1);
```

예: 데이터베이스에서 로드

이 예에서는 DataSource 데이터베이스 연결에서 Sales_data 테이블을 로드합니다.

```
LIB CONNECT TO 'DataSource';
LOAD *;
SQL SELECT * FROM `Sales_data`;
```

데이터 연결 저장 위치

연결은 Qlik Sense Repository Service를 사용하여 저장됩니다. Qlik Sense 서버 배포에서 Qlik Management Console를 사용하여 데이터 연결을 관리할 수 있습니다. Qlik Management Console를 사용하면 데이터 연결을 삭제하고 액세스 권한을 설정하고 다른 시스템 관리 작업을 수행할 수 있습니다.

Qlik Sense Desktop에서 모든 연결은 암호화되지 않은 상태로 앱에 저장됩니다. 여기에는 연결을 만들 때 입력한 사용자 이름, 비밀번호 및 파일 경로에 대해 사용 가능한 세부 정보가 포함됩니다. 따라서 다른 사용자와 앱을 공유하는 경우, 일반 텍스트로 이러한 세부 정보를 모두 사용할 수 있습니다. 앱을 공유하기 위해 설계하는 경우 이를 고려해야 합니다.

데이터 로드 편집기에서 데이터 선택

파일 또는 데이터베이스 테이블에서 로드할 필드를 선택하거나 데이터 로드 편집기에서 **데이터 선택**을 사용하여 원하는 데이터 소스의 보기를 선택할 수 있습니다.

대화 상자에서 필드를 선택할 수 있을 뿐만 아니라 필드의 이름을 변경할 수도 있습니다. 필드 선택이 완료되면 생성된 스크립트 코드를 스크립트에 삽입할 수 있습니다.

CSV 파일 등의 일부 데이터 소스에는 단일 테이블이 포함되어 있는 반면, Microsoft Excel 스프레드시트 또는 데이터베이스 등과 같은 데이터 소스에는 여러 테이블이 포함될 수 있습니다.

> ***데이터 관리자****에 동일한 이름과 동일한 열을 가진 스크립트된 테이블로 이미 추가된 테이블은* ***데이터 로드 편집기****에서 추가하지 마십시오.*

데이터 로드 편집기의 데이터 연결에서을 클릭하여 **데이터 선택**을 엽니다.

데이터베이스에서 데이터 선택

데이터베이스에서 데이터를 선택하는 경우 데이터 소스에 여러 개의 테이블이 포함될 수 있습니다.

다음과 같이 하십시오.

1. **데이터 로드 편집기**를 엽니다.
2. 왼쪽의 **데이터 연결** 아래 데이터베이스 연결에서을 클릭합니다.
   데이터 선택 대화 상자가 표시됩니다.
3. 드롭다운 목록에서 **데이터베이스**를 선택합니다.
   연결을 구성할 때 데이터베이스 이름이 입력되므로 일부 선택 대화 상자에는 **데이터베이스** 드롭다운 목록이 없습니다.
4. 데이터베이스의 **소유자**를 선택합니다.
   테이블 목록은 선택된 데이터베이스에서 사용할 수 있는 보기와 테이블로 채워집니다.
   일부 데이터베이스는 데이터 선택 프로세스에서 소유자를 지정할 필요가 없습니다.
5. 테이블을 선택합니다.
6. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다.
   테이블 이름 옆의 상자를 선택하면 테이블 내의 모든 필드를 선택할 수 있습니다.

   > *기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드를 기준으로 연결되기 때문입니다.*

7. 추가 테이블에서 데이터를 추가하려면 이를 선택합니다.

   > *검색을 통해 필드를 필터링하면서 동시에 데이터 선택 마법사에서 필드 이름을 바꿀 수는 없습니다. 먼저 텍스트 상자에서 검색 문자열을 지워야 합니다.*

   > *동일 테이블에서 두 필드의 이름을 변경하는 것은 불가능하므로 각각은 고유한 이름을 갖습니다.*

8. 데이터 선택이 끝나면 다음을 수행합니다.
   - **스크립트 삽입**을 클릭합니다.
     데이터 선택 창이 닫히고 선택 내용에 따라 LOAD /SELECT 문이 스크립트에 삽입됩니다.

Microsoft Excel 스프레드시트에서 데이터 선택

Microsoft Excel 스프레드시트에서 데이터를 선택하면 파일에 여러 개의 시트가 포함될 수 있습니다. 각 시트는 개별 테이블로 로드됩니다. 시트에 다른 시트 또는 로드된 테이블과 동일한 필드/열 구조가 있는 경우는 예외인데, 이런 경우는 테이블이 컨케트네이트됩니다.

다음과 같이 하십시오.

1. 데이터 로드 편집기의 적절한 폴더 연결에서을 클릭합니다.
   파일 선택 대화 상자가 표시됩니다.
2. 이 폴더 연결에 액세스할 수 있는 파일 목록에서 파일을 선택합니다.
3. 데이터를 선택할 첫 번째 시트를 선택합니다. 시트 이름 옆의 상자를 선택하면 시트 내의 모든 필드를 선택할 수 있습니다.
4. 해당 시트에 대한 설정이 적절한지 확인하십시오.

테이블 데이터를 올바르게 해석하도록 도와주는 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다. Excel 스프레드시트에서는 일반적으로 첫 행에 포함된 필드 이름이 있습니다. **필드 이름 없음**을 선택하면 필드 이름은 A,B,C...로 지정됩니다.
헤더 크기	테이블 헤더(일반적으로 일반 정보를 포함하며 열 서식으로 표시되지 않는 행)로서 생략할 행 수를 설정합니다.

예
스프레드시트가 다음과 같습니다.

스프레드시트 예

컴퓨터:	AEJ12B	-	-
날짜:	2015-10-05 09	-	-
타임스탬프	**순서**	**연산자**	**산출값**
2015-10-05 09:22	00122344	A	52
2015-10-05 10:31	00153534	A	67
2015-10-05 13:46	00747899	B	86

이 경우, 처음 두 줄을 무시하고 Timestamp, Order, Operator 및 Yield 필드가 있는 테이블을 로드하고자 할 수 있습니다. 이를 위해서는 다음 설정을 사용하십시오.

처음 두 줄을 무시하고 필드를 로드하도록 설정합니다.

UI 항목	설명
헤더 크기	2 처음 두 줄을 헤더 데이터로 간주하고 파일 로드 시 무시합니다. 이 경우, Machine: 및 Date:로 시작하는 두 줄은 테이블 데이터의 일부가 아니므로 무시됩니다.
필드 이름	**포함된 필드 이름** 읽은 첫 줄을 해당 열에 대한 필드 이름으로 사용합니다. 이 경우, 처음 두 줄이 헤더 데이터이므로 읽을 첫 줄은 세 번째 줄입니다.

5. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다.

   기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드에 의해 연결되기 때문입니다.

6. 데이터 선택이 완료되면 다음을 수행합니다.
   - **스크립트 삽입**을 클릭합니다.
     데이터 선택 창이 닫히고 선택 내용에 따라 LOAD /SELECT 문이 스크립트에 삽입됩니다.

*또한 ODBC 인터페이스를 통해 Microsoft Excel 파일을 데이터 소스로 사용할 수도 있습니다. 이 경우, **모든 파일** 데이터 연결 대신 **ODBC** 데이터 연결을 사용해야 합니다.*

테이블 파일에서 데이터 선택

다양한 데이터 파일에서 데이터를 선택할 수 있습니다.

- 텍스트 파일. 이 파일에는 필드의 데이터가 쉼표, 탭, 세미콜론 등의 구분 기호로 구분되어 있습니다(쉼표로 구분된 변수(CSV) 파일).
- HTML 테이블.
- XML 파일.
- KML 파일.
- Qlik 네이티브 QVD 및 QVX 파일.
- 고정 레코드 길이 파일.
- DIF 파일(Data Interchange Format).

다음과 같이 하십시오.

1. 데이터 로드 편집기의 적절한 폴더 연결에서을 클릭합니다.
   파일 선택 대화 상자가 표시됩니다.
2. 이 폴더 연결에 액세스할 수 있는 파일 목록에서 파일을 선택합니다.
3. **파일 형식**에 적절한 파일 형식이 선택되어 있는지 확인하십시오.
4. 해당 파일에 대한 설정이 적절한지 확인하십시오. 파일 형식이 다르면 파일 설정이 달라집니다.

5. 로드할 각 필드 옆의 상자를 선택하여 로드할 필드를 선택합니다. 또한 시트 이름 옆의 상자를 선택하여 파일 내의 모든 필드를 선택할 수도 있습니다.

기존 필드 이름을 클릭하고 새 이름을 입력하면 필드 이름을 편집할 수 있습니다. 이 작업은 테이블이 다른 테이블에 연결되는 방식에 영향을 줄 수 있는데, 이들 테이블이 기본적으로 공통 필드에 의해 연결되기 때문입니다.

6. 데이터 선택을 완료했으면 **스크립트 삽입**을 클릭합니다.
7. 데이터 선택 창이 닫히고 선택 내용에 따라 LOAD /SELECT 문이 스크립트에 삽입됩니다.

공간에서 편집 권한이 있는 사용자는 해당 공간의 DataFiles를 읽거나 쓰거나 로드할 수 있습니다. 다른 사용자는 DataFiles를 볼 수 없습니다.

파일 형식에 맞는 설정 선택

구분된 테이블 파일

이러한 설정은 각 레코드가 줄 바꿈으로 구분되어 있고 각 필드는 CSV 파일처럼 구분 문자를 사용하여 구분되어 있는 단일 테이블이 포함되어 있는 구분된 테이블 파일에 대해 검증됩니다.

구분된 테이블 파일의 파일 서식 설정

UI 항목	설명
구분된 테이블 파일의 파일 서식	**구분** 또는 **고정 레코드**로 설정합니다. 선택하면, 데이터 선택 대화 상자가 선택한 파일 형식에 맞춰 조정됩니다.
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.
구분 기호	테이블 파일에서 사용되는 **구분 기호**를 설정합니다.
인용	인용 부호를 처리하는 방법을 지정하기 위해 설정합니다. **없음** = 인용 문자가 허용되지 않음 **표준** = 표준 인용 부호(필드 값의 첫 문자 및 마지막 문자로 인용 부호 사용 가능) **MSQ** = 최신 인용 부호(필드에서 복수 행의 콘텐츠 허용)
헤더 크기	테이블 헤더로 생략할 행 수를 설정합니다.
문자 집합	테이블 파일에서 사용되는 문자 집합을 설정합니다.
주석	데이터 파일은 레코드 사이에 주석이 포함될 수 있으며, 이 주석은 행 시작 부분에 하나 이상의 특수 문자(예: //)로 표시됩니다. 주석 행을 나타내는 하나 이상의 문자를 지정합니다. Qlik Sense에서는 여기에 지정된 문자로 시작하는 행을 로드하지 않습니다.
EOF 무시	데이터에 필드 값의 일부로 파일 끝 문자가 포함되는 경우, **EOF 무시**를 선택합니다.

고정 레코드 데이터 파일
고정 레코드 데이터 파일에는 각 레코드(데이터 행)에 고정된 필드 크기의 열이 여러 개 있고 일반적으로 공백 또는 탭 문자가 추가된 단일 테이블이 포함되어 있습니다.

두 가지 방법으로 필드 나누기 위치를 설정할 수 있습니다.

- **필드 나누기 위치**에 쉼표로 구분된 필드 나누기 위치를 수동으로 입력합니다. 각 위치는 필드의 시작 부분을 나타냅니다.

  1,12,24

- **필드 나누기**를 활성화하여 필드 데이터 미리 보기에서 대화형으로 필드 나누기 위치를 편집합니다. **필드 나누기 위치**가 선택한 위치로 업데이트됩니다. 다음을 수행할 수 있습니다.
  - 필드 데이터 미리 보기를 클릭하여 필드 구분선을 삽입합니다.
  - 필드 구분선을 클릭하여 삭제합니다.
  - 필드 구분선을 끌어서 이동합니다.

고정 레코드 데이터 파일의 파일 서식 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.
헤더 크기	**헤더 크기**를 테이블 헤더로서 생략할 행 수로 설정합니다.
문자 집합	테이블 파일에서 사용하는 문자 집합으로 설정됩니다.
탭 크기	테이블 파일에서 하나의 탭 문자가 나타내는 공백 수로 설정합니다.
레코드 줄 크기	테이블 파일에서 하나의 레코드 범위에 해당하는 행 수로 설정합니다. 기본값은 1입니다.

HTML 파일
HTML 파일에는 여러 개의 테이블을 포함할 수 있습니다. Qlik Sense에서는 <TABLE> 태그가 있는 모든 요소를 테이블로 해석합니다.

HTML 파일의 파일 서식 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다.
문자 집합	테이블 파일에서 사용하는 문자 집합을 설정합니다.

XML 파일
XML 형식으로 저장된 데이터를 로드할 수 있습니다.

XML 파일에 대한 특정 파일 형식 설정은 없습니다.

QVD 파일
QVD 형식으로 저장된 데이터를 로드할 수 있습니다. QVD는 네이티브 Qlik 형식이며 Qlik Sense 또는 QlikView에서만 읽고 쓸 수 있습니다. 파일 형식은 Qlik Sense 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다.

QVD 파일에 대한 특정 파일 형식 설정은 없습니다.

QVX 파일
Qlik data eXchange(QVX) 형식으로 저장된 데이터를 로드할 수 있습니다. QVX 파일은 Qlik QVX SDK를 사용하여 개발한 사용자 지정 커넥터를 통해 생성됩니다.

QVX 파일에 대한 특정 파일 형식 설정은 없습니다.

KML 파일
KML 형식으로 저장된 맵 파일을 로드하여 맵 시각화에 사용할 수 있습니다.

KML 파일에 대한 특정 파일 형식 설정은 없습니다.

스크립트 미리 보기

삽입되는 문은 스크립트 미리 보기에 표시되며 미리 보기는 **스크립트 미리 보기**를 클릭하여 숨길 수 있습니다.

LOAD 문 포함

LOAD 문 포함을 선택하면 앞에 LOAD 문이 있고 SELECT 문을 입력으로 사용하는 SELECT 문이 생성됩니다.

테이블의 필드 이름을 변경하면 이 설정에 관계없이 LOAD 문이 자동으로 삽입됩니다.

인라인 로드

인라인 로드를 사용하여 데이터 로드 편집기에서 데이터를 로드할 수 있습니다. 기본 인라인 로드는 테이블을 만들고 데이터 필드와 레코드를 삽입합니다. 예:

```
MyTable:
Load * Inline [
Country, Year, Sales
Argentina, 2014, 66295.03
Argentina, 2015, 140037.89
Austria, 2014, 54166.09
Austria, 2015, 182739.87
];
```

위의 인라인 로드에는 다음 구문이 사용됩니다.

- *MyTable:*은 데이터에 대한 테이블을 만듭니다.
- *Load * Inline*은 데이터 로드 편집기에서 인라인 데이터 로드를 나타냅니다.
- 대괄호는 데이터를 묶습니다.
- LOAD 문의 첫 번째 줄은 데이터 필드를 정의합니다.

- 쉼표는 데이터 필드와 레코드를 구분합니다.
- 세미콜론은 LOAD 문을 닫습니다.

데이터 로드 버튼을 사용하여 데이터를 로드합니다.

다음 이미지는 *Load*라는 새 스크립트 섹션의 인라인 로드를 보여 줍니다.

데이터 로드 편집기의 인라인 로드

데이터 로드 스크립트 편집

데이터 로드 편집기의 텍스트 편집기에서 스크립트를 작성할 수 있습니다. 여기서 데이터 선택 시 생성한 **LOAD** 또는 **SELECT** 문을 수동으로 변경하고 새 스크립트를 입력할 수 있습니다.

스크립트는 Qlik Sense 스크립트 구문을 사용하여 작성해야 하며 서로 다른 요소 간에 쉽게 구별할 수 있도록 다른 색으로 표시됩니다. 주석은 녹색으로 강조 표시되는 반면 Qlik Sense 구문 키워드는 파란색으로 강조 표시됩니다. 각 스크립트 행에는 번호가 표시됩니다.

편집기에는 로드 스크립트를 개발하는 데 도움을 줄 수 있는 여러 가지 기능이 포함되어 있으며 이에 대해 이 섹션에서 설명합니다.

명령 및 함수에 대한 구문 도움말 액세스

여러 가지 방법으로 Qlik Sense 구문 키워드에 대한 구문 도움말에 액세스할 수 있습니다.

도움말 포털 액세스

두 가지 방법으로 Qlik Sense 도움말 포털의 상세 도움말에 액세스할 수 있습니다.

- 도구 모음에서 을 클릭하여 구문 도움말 모드를 시작합니다. 구문 도움말 모드에서 구문 키워드(파란색과 밑줄로 표시)를 클릭하여 구문 도움말에 액세스할 수 있습니다.
- 키워드 내부 또는 끝에 커서를 놓고 Ctrl+H를 누릅니다.

> *구문 도움말 모드에서는 스크립트를 편집할 수 없습니다.*

자동 완성 기능 사용

Qlik Sense 스크립트 키워드를 입력하기 시작하면 일치하는 키워드 중에서 선택할 수 있는 자동 완성 목록이 표시됩니다. 입력을 계속하면 목록이 좁혀지고 제안 구문 및 매개 변수가 포함된 템플릿에서 선택할 수 있습니다. 도구 설명에서 문 또는 함수의 도움말 포털 설명에 대한 링크뿐만 아니라 파라미터 및 추가 문을 포함하여 함수의 구문을 표시합니다.

또한 키워드 목록을 표시하는 키보드 바로 가기(Ctrl+Space)와 도구 설명을 표시하는 키보드 바로 가기(Ctrl+Shift+Space)를 사용할 수 있습니다.

준비된 테스트 스크립트 삽입

이 기능을 사용하면 일련의 인라인 데이터 필드를 로드하는 준비된 테스트 스크립트를 삽입할 수 있습니다. 이 기능을 사용하면 테스트용 데이터 셋을 신속하게 생성할 수 있습니다.

다음과 같이 하십시오.

- Ctrl + 00을 누릅니다.

테스트 스크립트 코드가 스크립트에 삽입됩니다.

코드 들여쓰기

코드를 들여쓰면 가독성을 높일 수 있습니다.

다음과 같이 하십시오.

1. 들여쓰기를 변경할 행을 하나 이상 선택합니다.
2. 를 클릭하여 텍스트를 들여쓰거나(들여쓰기 증가), 를 클릭하여 텍스트를 내어씁니다(들여쓰기 감소).

또한 다음과 같은 키보드 바로 가기를 사용할 수도 있습니다.

Tab(들여쓰기)

Shift+Tab(내어쓰기)

텍스트 검색 및 바꾸기

스크립트 섹션에서 텍스트를 검색하고 바꿀 수 있습니다.

텍스트 검색

데이터 로드 편집기를 엽니다. 다음과 같이 하십시오.

1. 도구 모음에서 을 클릭합니다.
   검색 드롭다운 대화 상자가 표시됩니다.
2. 검색 상자에 찾으려는 텍스트를 입력합니다.

스크립트 코드의 현재 섹션에서 검색 결과가 강조 표시됩니다. 또한 찾은 텍스트 인스턴스 수가 섹션 레이블 옆에 표시됩니다.

3. ❮및❯을 클릭하여 결과를 탐색할 수 있습니다.
4. 도구 모음의🔍를 클릭하여 검색 대화 상자를 닫습니다.

> *또한 **모든 구역에서 검색**을 선택하여 모든 스크립트 섹션에서 검색할 수도 있습니다. 찾은 텍스트 인스턴스 수는 각 섹션 레이블 옆에 표시됩니다. **대/소문자 일치**를 선택하면 대문자와 소문자를 구분하여 검색할 수 있습니다.*

텍스트 바꾸기

다음과 같이 하십시오.

1. 도구 모음에서🔍을 클릭합니다.
   검색 드롭다운 대화 상자가 표시됩니다.
2. 검색 상자에 찾으려는 텍스트를 입력합니다.
3. 바꾸기 상자에 대체 텍스트를 입력하고 **바꾸기**를 클릭합니다.
4. ❯을 클릭하여 검색 텍스트의 다음 인스턴스를 찾고 다음 중 하나를 수행합니다.
   - **바꾸기**를 클릭하여 텍스트를 바꿉니다.
   - ❯을 클릭하여 다음 항목을 찾습니다.
5. 도구 모음의🔍를 클릭하여 검색 대화 상자를 닫습니다.

> *또한 **구역 내 모든 항목 바꾸기**를 클릭하여 현재 스크립트 섹션의 모든 검색 텍스트 인스턴스를 바꿀 수도 있습니다. 바꾸기 기능은 대/소문자를 구분하며 바뀐 텍스트는 바꾸기 필드에 지정된 그대로 대소문자가 표시됩니다. 바꾼 인스턴스 수에 대한 정보가 포함된 메시지가 표시됩니다.*

스크립트에 주석 추가

스크립트 코드에 주석을 삽입하거나, 주석 표시를 사용하여 스크립트 코드의 일부를 비활성화할 수 있습니다. 한 줄에서 //(슬래시 두 개)의 오른쪽에 이어지는 모든 텍스트는 주석으로 간주되며 스크립트를 실행할 때 실행되지 않습니다.

데이터 로드 편집기 도구 모음에는 코드를 주석 처리하거나 주석 처리를 제거할 수 있는 바로 가기가 포함되어 있습니다. 이 기능은 설정/해제가 교대로 작동합니다. 따라서 선택한 코드가 주석 처리되어 있으면 주석 처리되고, 주석 처리되어 있지 않으면 주석 처리가 제거됩니다.

주석 처리

다음과 같이 하십시오.

1. 주석 처리되지 않은 하나 이상의 코드 행을 선택하거나 커서를 행 시작 부분에 놓습니다.
2. //을 클릭하거나 Ctrl + K를 누릅니다.

이제 선택한 코드가 주석 처리됩니다.

주석 처리 제거

다음과 같이 하십시오.

1. 주석 처리된 하나 이상의 코드 행을 선택하거나 커서를 주석 처리된 행 시작 부분에 놓습니다.
2. //을 클릭하거나 Ctrl + K를 누릅니다.

이제 선택한 코드가 나머지 스크립트와 함께 실행됩니다.

스크립트 코드에 주석을 삽입할 수 있는 다양한 방법이 있습니다.

- ***Rem** 문을 사용합니다.*
- *코드 섹션을 /*와 */으로 감쌉니다.*

```
Rem This is a comment ;


/* This is a comment

   that spans two lines */


// This is a comment as well
```

모든 코드 선택

현재 스크립트 섹션의 모든 코드를 선택할 수 있습니다.

다음과 같이 하십시오.

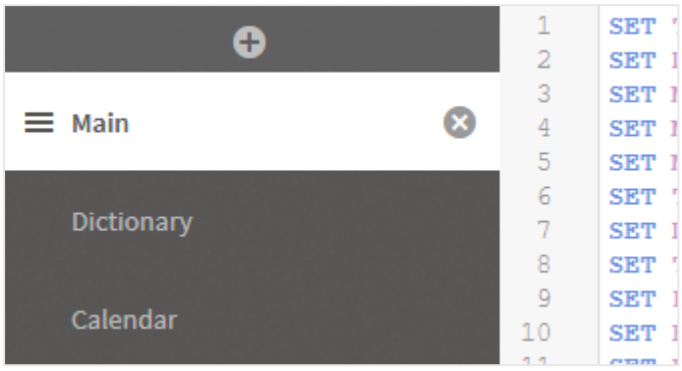
- Ctrl + A를 누릅니다.

현재 섹션의 모든 스크립트 코드가 선택됩니다.

스크립트 코드 분류

스크립트를 섹션으로 나누면 구조를 체계적으로 분류할 수 있습니다. 스크립트는 맨 위부터 섹션 순서에 따라 실행됩니다. 스크립트에는 최소 하나 이상의 섹션이 포함되어야 합니다.

데이터 로드 편집기의 기본, 사전 및 캘린더 섹션



데이터 추가를 사용하여 데이터를 추가한 경우, 데이터를 로드하는 데 필요한 스크립트 코드가 포함되어 있는 **자동 생성 섹션**이라는 이름의 데이터 로드 스크립트 섹션이 생깁니다.

스크립트 섹션 작업

스크립트 섹션을 구성하는 방법에는 여러 가지가 있습니다.

- ⊕을 클릭하여 새 스크립트 섹션을 삽입함으로써 코드를 구성합니다. 현재 선택한 섹션 뒤에 새 섹션이 삽입됩니다.
- 삭제하려면 섹션 탭 옆의⊗를 클릭합니다. 삭제 확인을 해야 합니다.

  > *스크립트 섹션 삭제는 실행 취소할 수 없습니다.*

- 섹션 이름을 클릭하고 입력하여 이름을 편집합니다. 끝나면 Enter를 누르거나 섹션 바깥쪽을 클릭합니다.
- ☰드래그 바에 커서를 놓고 섹션을 끌어서 순서를 재배열합니다.

데이터 로드 스크립트 디버그

데이터 로드 편집기에서 디버깅 유틸리티를 사용하면 중단점을 통해 스크립트의 실행을 단계별로 진행할 수 있으며, 이를 통해 스크립트 실행의 변수 값과 출력을 조사할 수 있습니다.

출력, **변수** 및 **중단점** 중 일부 또는 모두 볼 것인지 선택할 수 있습니다.

디버그 패널을 보려면 다음을 수행합니다.

- 데이터 로드 편집기 도구 모음에서 을 클릭합니다.
  디버그 패널은 데이터 로드 편집기 하단에 열립니다.

> *디버그 모드에서 실행할 때는 연결 생성, 연결 편집, 데이터 선택, 스크립트 저장 또는 데이터 로드 등의 작업을 수행할 수 없습니다. 디버그 모드는 디버그 실행과 함께 시작되며 스크립트가 실행되거나 실행이 종료될 때까지 지속됩니다.*

디버그 도구 모음

데이터 로드 편집기 디버그 패널에는 디버그 실행을 제어하는 다음과 같은 옵션이 포함된 도구 모음이 있습니다.

디버그 도구 모음 옵션

UI 항목	설명
제한된 로드	이 옵션을 사용하면 각 데이터 소스에서 로드할 데이터 행의 수를 제한할 수 있습니다. 데이터 소스가 큰 경우 실행 시간을 줄이는 데 유용합니다. 로드할 행의 수를 입력하십시오. *이 옵션은 실제 데이터 소스에만 적용됩니다. 자동으로 생성되며 인라인 로드 등은 제한되지 않습니다.*
▶	다음 중단점에 도달할 때까지 디버그 모드에서 실행을 시작하거나 계속합니다.
⏭	다음 코드 행으로 넘어갑니다.
■	여기서 실행이 종료됩니다. 모든 코드가 실행되기 전에 종료하면 결과 데이터 모델에 실행이 종료된 코드 행까지의 데이터만 포함됩니다.

출력

출력에는 디버그 실행 중에 생성된 모든 메시지가 표시됩니다. 새 메시지가 표시되는 경우 🔓을 클릭하여 출력이 스크롤되지 않도록 차단할 수 있습니다.

또한 출력 메뉴 ☰에는 다음과 같은 옵션이 포함됩니다.

출력 메뉴 옵션

UI 항목	설명
지우기	모든 출력 메시지를 삭제하려면 이 옵션을 클릭합니다.
모든 텍스트 선택	모든 출력 메시지를 선택하려면 이 옵션을 클릭합니다.
맨 아래로 스크롤	마지막 출력 메시지로 스크롤하려면 이 옵션을 클릭합니다.

변수

변수에는 모든 예약된 변수, 시스템 변수 및 스크립트에 정의된 변수가 나열되며 스크립트 실행 중에는 현재 값이 표시됩니다.

변수를 즐겨찾기로 설정

실행 중에 특정 변수를 조사하려는 경우, 해당 변수를 즐겨찾기로 설정할 수 있습니다. 즐겨찾는 변수는 변수 목록 맨 위에 표시되며 노란색 별표가 표시됩니다. 변수를 즐겨찾기로 설정하려면 다음을 수행합니다.

- 변수 옆의★을 클릭합니다.
  이제★은 노란색으로 표시되고 해당 변수는 변수 목록 맨 위로 이동되었습니다.

변수 필터링

필터를 적용하면 변수 메뉴☰에서 다음 옵션을 사용하여 선택된 유형의 변수만 표시할 수 있습니다.

변수 메뉴 옵션

UI 항목	설명
모든 변수 표시	모든 유형의 변수를 표시하려면 이 옵션을 클릭합니다.
시스템 변수 표시	시스템 변수를 표시하려면 이 옵션을 클릭합니다. 시스템 변수는 Qlik Sense에서 정의하지만 스크립트에서 해당 변수 값을 변경할 수 있습니다.
예약된 변수 표시	예약된 변수를 표시하려면 이 옵션을 클릭합니다. 예약된 변수는 Qlik Sense에서 정의하며 값은 변경할 수 없습니다.
사용자 정의 변수 표시	사용자 정의 변수를 표시하려면 이 옵션을 클릭합니다. 사용자 정의 변수는 스크립트에서 사용자가 정의한 변수입니다.

중단점

특정 코드 행에서 디버그 실행을 중단할 수 있도록 스크립트에 중단점을 추가하고 이 지점에서 변수 값과 출력 메시지를 조사할 수 있습니다. 중단점에 도달하면 실행을 중지하거나, 다음 중단점에 도달할 때까지 계속하거나, 다음 코드 행으로 건너뛰도록 선택할 수 있습니다. 스크립트의 모든 중단점이 섹션 및 행 번호에 대한 참조와 함께 나열됩니다.

중단점 추가

코드 행에 중단점을 추가하려면 다음 중 하나를 수행합니다.

- 스크립트에서 중단점을 추가할 행 번호 바로 오른쪽 영역을 클릭합니다.
  행 번호 옆의⛔은 해당 행에 중단점이 있음을 나타냅니다.

디버그 패널이 닫혔을 때도 중단점을 추가할 수 있습니다.

중단점 삭제

다음 중 하나를 수행하여 중단점을 삭제할 수 있습니다.

- 스크립트에서 행 번호 옆의⛔을 클릭합니다.
- 중단점 목록에서 중단점 옆의⊗을 클릭합니다.

또한☰을 클릭하고 **모두 삭제**를 선택하여 스크립트의 모든 중단점을 삭제할 수도 있습니다.

중단점 사용/사용 안 함

중단점을 만들면 기본적으로 사용할 수 있게 설정되며 중단점 목록에서 해당 중단점 옆에 ✔이 표시됩니다. 중단점 목록에서 중단점을 선택 및 선택 취소하면 개별 중단점을 사용하거나 사용하지 않을 수 있습니다.

또한 중단점 메뉴 ☰에서 다음 옵션도 사용할 수 있습니다.

- **모두 사용**
- **모두 사용 안 함**

로드 스크립트 저장

스크립트를 저장하면 전체 앱이 저장되지만 데이터는 자동으로 다시 로드되지 않습니다.

스크립트가 저장될 때 앱에는 이전에 다시 로드했던 데이터가 계속 포함되며, 이와 관련된 메시지가 도구 모음에 표시됩니다. 새 데이터로 앱을 업데이트하려면 데이터 로드 편집기 도구 모음에서 ⊙ 을 클릭합니다. 데이터를 로드하면 스크립트가 앱에 자동 저장됩니다.

데이터 로드 편집기는 로드 스크립트를 변경할 때 자동으로 작업을 저장합니다. CTRL+S를 누르면 강제로 저장할 수 있습니다.

> *스크립트는 Qlik Sense Desktop에서 자동으로 저장되지 않습니다. 수동으로 스크립트를 저장해야 합니다.*

스크립트를 저장하면 구문 오류를 자동으로 검사합니다. 구문 오류는 코드에서 강조 표시되며, 구문 오류가 포함된 모든 스크립트 섹션은 해당 섹션 레이블 옆에 ⓘ 이 표시됩니다.

데이터를 로드하는 스크립트 실행

도구 모음에서 ⊙ 을 클릭하여 스크립트를 실행하고 앱에서 데이터를 로드합니다. 데이터를 로드하기 전에 앱이 자동으로 저장됩니다.

데이터 로드 진행률 대화 상자가 표시되며 로드를 **중단**할 수 있습니다. 데이터 로드가 완료되면 대화 상자가 상태(**성공적으로 완료되었습니다.** 또는 **데이터 로드가 실패했습니다.**) 및 가상 키 등과 같이 발생 가능한 오류 및 경고 요약 정보로 업데이트됩니다. 또한 대화 상자가 닫힌 후에 보려는 경우 요약 정보는 **출력**에 표시됩니다.

> ***데이터 로드 진행률** 대화 상자가 실행 성공 후 항상 자동으로 닫히도록 하려면 **성공적으로 완료되면 닫기**를 선택합니다.*

데이터 로드 편집기의 키보드 바로 가기

데이터 로드 편집기 환경에서 쉽고 효과적으로 작업하는 데 사용할 수 있는 여러 가지 키보드 바로 가기가 있습니다.

키보드 바로 가기

키보드 바로 가기는 Windows에서 작업하는 경우를 기준으로 설명합니다. Mac OS의 경우 Ctrl 대신 Cmd를 사용하십시오.

데이터 로드 편집기의 키보드 바로 가기

바로 가기	동작
Ctrl+0,Ctrl+0	샘플 데이터를 생성합니다.
Alt+1	출력 패널을 표시하거나 표시된 경우 숨깁니다.
Alt+2	디버그 도구가 실행 중인 경우 변수 패널을 표시하거나 표시된 경우 숨깁니다.
Alt+3	디버그 도구가 실행 중인 경우 중단점 패널을 표시하거나 표시된 경우 숨깁니다.
Alt+F5	디버그 도구를 표시하거나 표시된 경우 숨깁니다.
Alt+F6	디버그 도구가 실행 중인 경우 디버그를 실행합니다.
Alt+F7	디버그 도구가 실행 중인 경우 디버거에서 다음 단계로 진행합니다.
Alt+F8	디버그 도구가 실행 중인 경우 디버거를 중지합니다.
F9	디버그 중단점의 삽입 및 제거를 전환합니다.
Alt+F10	오른쪽 패널을 표시하거나 표시된 경우 숨깁니다.
Alt+F11	스크립트 편집기를 전체 화면으로 확장합니다.
Ctrl+C	선택한 항목을 클립보드로 복사합니다.
Ctrl+F	검색 항목 찾기 필드를 표시하거나 열린 경우 이를 숨깁니다.
Ctrl+H	데이터 로드 편집기 또는 수식 편집기에 있을 때 현재 선택한 기능 컨텍스트에서 온라인 도움말을 엽니다.
Ctrl+K	코드 주석 행의 삽입 및 제거를 전환합니다.
Ctrl+P	현재 보기 또는 활성 시트/스토리를 인쇄합니다.
Ctrl+S	변경 사항을 저장합니다.
Ctrl+V	가장 최근에 복사한 항목을 클립보드에서 붙여넣습니다.
Ctrl+X	선택한 항목을 잘라내서 클립보드로 복사합니다. Google Chrome 브라우저를 사용하는 경우: 데이터 로드 편집기 또는 수식 편집기에서 아무 것도 선택하지 않고 커서를 행 앞에 놓으면 전체 행이 잘립니다.
Ctrl+Z	작업을 실행 취소합니다. 더 이전의 작업을 실행 취소하려면 반복합니다.
Alt+Insert	스크립트에 새 섹션을 삽입합니다.
Alt+PgUp	이전 섹션으로 이동합니다.

바로 가기	동작
Alt+PgDn	다음 섹션으로 이동합니다.
Ctrl+Shift+Enter/Return	데이터를 다시 로드합니다.
Ctrl+Shift+Space	도구 설명을 표시합니다. (Qlik Sense Desktop에서 지원되지 않음)
Ctrl+Space	자동 텍스트 문자열로 자동 완성합니다.
검색 필드에서 Enter/Return 키	검색 문자열의 다음 인스턴스를 검색합니다.
바꾸기 필드에서 Enter/Return 키	검색 문자열의 선택된 인스턴스를 바꿉니다.
검색 또는 바꾸기 필드에서 Esc 키	검색 항목 찾기 필드를 닫습니다.
Shift+Tab	스크립트에서 행을 내어씁니다.
Tab	스크립트에서 행을 들여씁니다.

4.3 스크립트 구문 및 데이터 구조에 대한 이해

추출, 변환 및 로드

일반적으로 앱에 데이터를 로드하는 방법은 추출, 변환, 로드 프로세스로 설명할 수 있습니다.

- 추출
  첫 번째 단계는 데이터 소스 시스템에서 데이터를 추출하는 것입니다. 스크립트에서 **SELECT** 또는 **LOAD** 문을 사용하여 이를 정의합니다. 이 두 문 사이의 차이점은 다음과 같습니다.
  - **SELECT** 문은 ODBC 데이터 소스 또는 OLE DB 공급자의 데이터를 선택하는 데 사용됩니다. **SELECT** SQL 문은 Qlik Sense가 아니라 데이터 공급자가 평가합니다.
  - **LOAD** 문은 파일, 스크립트에 정의된 데이터, 이전에 로드한 테이블, 웹 페이지, 이후 **SELECT** 문의 결과에서 또는 자동으로 데이터를 생성하여 데이터를 로드하는 데 사용됩니다.
- 변환
  변환 단계에는 스크립트 함수 및 규칙을 사용하여 데이터를 편집하고 원하는 데이터 모델 구조를 이끌어내는 작업이 포함됩니다. 일반적인 작업은 다음과 같습니다.
  - 새로운 값 계산
  - 코딩된 값 변환
  - 필드 이름 바꾸기
  - 테이블 조인
  - 값 집계
  - 피벗
  - 데이터 검증
- 로드
  최종 단계에서는 스크립트를 실행하여 정의한 데이터 모델을 앱에 로드합니다.

Qlik Sense에서 데이터를 효율적으로 처리할 수 있는 데이터 모델을 만들려고 합니다. 일반적으로 이런 경우, 순환 참조 없이 적절히 정규화된 스타 스키마 또는 눈송이형 스키마, 즉 각 엔터티가 별도의 테이블에 보관되는 모델을 사용해야 합니다. 다시 말해, 일반적인 데이터 모델은 다음과 같습니다.

- 측정값(단위 숫자, 판매 금액, 예산 등)을 계산하는 데 사용되는 숫자 및 차원에 대한 키가 포함된 중앙 팩트 테이블.
- 해당하는 모든 특성(제품, 고객, 범주, 캘린더, 공급자 등)과 함께 차원이 포함된 주변 테이블.

많은 경우, 로드 스크립트에서 풍부한 데이터 모델을 구축하거나 차트 표현식에서 집계를 수행하여 집계 등의 작업을 처리할 수 있습니다. 대개는 로드 스크립트에서 데이터 변환을 수행하는 경우 더 나은 성능을 얻을 수 있습니다.

종이에 데이터 모델을 그려보는 것도 좋습니다. 그렇게 하면 추출할 데이터와 수행할 변환에 대한 구조를 파악하는 데 도움이 됩니다.

데이터 로드 문

데이터는 **LOAD** 또는 **SELECT** 문으로 로드됩니다. 위 문은 각각 내부 테이블을 생성합니다. 테이블은 항상 데이터 목록으로 표시될 수 있으며, 각 레코드(행)는 개체 유형의 새로운 인스턴스가 되고 각 필드(열)는 개체의 특정 특성 또는 속성이 됩니다.

이 두 문 사이의 차이점은 다음과 같습니다.

- **SELECT** 문은 ODBC 데이터 소스 또는 OLE DB 공급자의 데이터를 선택하는 데 사용됩니다. **SELECT** SQL 문은 Qlik Sense가 아니라 데이터 공급자가 평가합니다.
- **LOAD** 문은 파일, 스크립트에 정의된 데이터, 이전에 로드한 테이블, 웹 페이지, 이후 **SELECT** 문의 결과에서 또는 자동으로 데이터를 생성하여 데이터를 로드하는 데 사용됩니다.

규칙

Qlik Sense에 데이터를 로드할 때 다음과 같은 규칙이 적용됩니다.

- Qlik Sense는 **LOAD** 또는 **SELECT** 문으로 생성된 테이블을 구분하지 않습니다. 이는 여러 테이블이 로드될 때 테이블이 **LOAD** 또는 **SELECT** 문 또는 이 둘의 조합으로 로드되어도 상관이 없음을 의미합니다.
- 문이나 데이터베이스에 있는 원래 테이블 내의 필드 순서는 Qlik Sense 논리에서 무의미합니다.
- 필드 이름은 필드를 식별하고 연결을 만드는 추가 프로세스에서 사용됩니다. 필드 이름은 대/소문자가 구분되므로 종종 스크립트에서 필드 이름을 변경해야 하는 경우가 있습니다.

스크립트의 실행

일반적인 **LOAD** 또는 **SELECT** 문의 이벤트 순서를 대략적으로 정리하면 다음과 같습니다.

1. 표현식 평가
2. **as**로 필드의 이름 바꾸기
3. **alias**로 필드의 이름 바꾸기

4. 필드 이름의 한정
5. 필드 이름이 일치하는 경우 데이터 매핑
6. 내부 테이블에 데이터 저장

필드

필드는 Qlik Sense에서 데이터를 전달하는 기본 엔터티입니다. 필드는 일반적으로 필드 값이라고 하는 여러 값을 포함합니다. 데이터베이스 용어로 Qlik Sense에서 처리되는 데이터는 데이터 파일에서 가져옵니다. 파일은 여러 필드로 구성되며 각 데이터 항목은 레코드입니다. 파일, 필드 및 레코드는 각각 테이블, 열 및 행에 해당합니다. Qlik Sense AQL 논리는 필드와 해당 필드 값에 대해서만 작동합니다.

필드 데이터는 스크립트에서 **LOAD**, **SELECT** 또는 **Binary** 문을 통해 검색합니다. 필드의 데이터를 변경하는 유일한 방법은 스크립트를 다시 실행하는 것입니다. 실제 필드 값은 레이아웃에서 사용자가 편집하거나 자동화를 통해 변경할 수 없습니다. Qlik Sense로 읽은 후에는 논리 선택 및 계산을 수행할 때만 보고 사용할 수 있습니다.

필드 값은 숫자 또는 영숫자(텍스트) 데이터로 구성됩니다. 숫자 값의 경우 실제로는 숫자 값 및 현재 서식 있는 텍스트 표시의 이중 값을 가집니다. 시트 개체 등에는 두 번째 값만 표시됩니다.

필드의 내용은 필터 창에 표시할 수 있습니다.

파생된 필드

관련성이 높은 필드 그룹이 있거나, 차원 또는 측정값을 만들 때 관련성이 있는 작은 부분으로 나눌 수 있는 정보가 필드에 포함되어 있는 경우, 파생된 필드를 생성하는 데 사용할 수 있는 필드 정의를 만들 수 있습니다. 예를 하나 들자면, 년도, 월, 주 수 또는 날짜 이름과 같은 여러 특성을 파생시킬 수 있는 데이터 필드가 있습니다. 이러한 특성은 모두 Qlik Sense 날짜 함수를 사용하여 차원 표현식에서 계산할 수 있지만, 또 다른 방법은 데이터 유형의 모든 필드에서 공통으로 사용되는 캘린더 정의를 만드는 것입니다. 필드 정의는 데이터 로드 스크립트에 저장됩니다.

*Qlik Sense에 대한 기본 캘린더 필드 정의는 **데이터 관리자**를 사용하여 로드된 날짜 필드의 autoCalendar에 포함됩니다. 자세한 내용은 앱에 데이터 추가 (page 16)를 참조하십시오.*

캘린더 필드 정의 선언

Declare 문을 사용하여 파생된 필드의 정의를 만들 수 있습니다. 필드의 다양한 특성을 정의할 수 있는 위치로, 이 사례에서는 날짜 관련 특성을 정의할 수 있습니다. 각 필드는 `<expression> **As** field_name **tagged** tag`로 기술됩니다. 하나 이상의 태그를 설정하는 것은 선택 사항이지만 파생된 필드의 정렬 순서에 영향을 미칠 수 있습니다. `$1`을 사용하여, 파생 필드를 생성해야 하는 원본 데이터 필드를 참조합니다.

캘린더 필드 정의에 대한 이름으로 autoCalendar를 사용하지 마십시오. 이 이름은 자동 생성 캘린더 템플릿용으로 예약되어 있습니다.

```
Calendar:
DECLARE FIELD DEFINITION TAGGED '$date'
   Parameters
      first_month_of_year = 1
   Fields
      Year($1) As Year Tagged ('$numeric'),
```

```
Month($1) as Month Tagged ('$numeric'),
Date($1) as Date Tagged ('$date'),
Week($1) as Week Tagged ('$numeric'),
Weekday($1) as Weekday Tagged ('$numeric'),
DayNumberOfYear($1, first_month_of_year) as DayNumberOfYear Tagged ('$numeric');
```

Derive를 사용하여 캘린더에 데이터 필드 매핑

다음 단계는 **Derive** 문을 사용하여 기존 데이터 필드를 캘린더에 매핑하는 것입니다. 이렇게 하면 파생 필드가 생성됩니다. 데이터 로드 스크립트에서 세 가지 대체 가능한 방법으로 이 작업을 수행할 수 있습니다.

- 필드 이름을 사용하여 특정 필드 매핑.
  `DERIVE FIELDS FROM FIELDS OrderDate,ShippingDate USING Calendar;`
- 하나 이상의 특정 필드 태그를 사용하여 모든 필드 매핑.
  `DERIVE FIELDS FROM EXPLICIT TAGS ('$date') USING Calendar;`
- 필드 정의(위의 예에서는 $date)의 하나 이상의 태그를 사용하여 태그가 지정된 모든 필드 매핑.
  `DERIVE FIELDS FROM IMPLICIT TAG USING Calendar;`

이 사례에서는 이 세 가지 예 중에서 사용할 수 있습니다.

시각화에서 파생 날짜 필드 사용

Qlik Sense에서는 이 예에서처럼 캘린더 정의를 만들고 필드를 매핑한 경우 파생 데이터 필드를 즉시 식별할 수 있습니다. 이들 필드는 **필드** 자산 패널의 **날짜 및 시간 필드** 섹션에서 사용할 수 있습니다. 또한 식 편집기에서, 그리고 차원을 만들거나 편집할 때에도 모든 파생 필드를 찾을 수 있습니다.

필드 태그

필드 태그를 사용하면 데이터 모델의 필드에 메타데이터를 추가할 수 있습니다. 필드 태그에는 두 가지 유형이 있습니다.

- 시스템 필드 태그
  시스템 필드 태그는 스크립트를 실행하고 데이터를 로드할 때 자동으로 생성됩니다. 일부 태그는 스크립트에서 편집할 수 있습니다. 시스템 태그 앞에는 항상 $ 기호가 붙습니다.
- 사용자 지정 필드 태그
  Tag 문을 사용하여 데이터 로드 스크립트의 필드에 사용자 지정 태그를 추가할 수 있습니다. 사용자 지정 태그는 시스템 태그와 같은 이름을 사용할 수 없습니다.

시스템 필드 태그

다음 시스템 필드 태그는 데이터 로드 시 자동으로 생성됩니다.

시스템 필드 태그

Tag	설명	스크립트에서 편집 가능 여부
$system	스크립트 실행 중에 Qlik Sense에서 생성된 시스템 필드입니다.	아니요
$key	둘 이상의 테이블 간에 연결을 제공하는 키 필드입니다.	아니요
$keypart	필드가 하나 이상의 가상 키에 속합니다.	아니요

Tag	설명	스크립트에서 편집 가능 여부
$syn	가상 키	아니요
$hidden	시각화, 차원 또는 측정값을 만들 때 필드 선택 목록에 표시되지 않는 숨겨진 필드입니다. 표현식에서 숨겨진 필드를 계속 사용할 수 있지만 필드 이름을 입력해야 합니다. **HidePrefix** 및 **HideSuffix** 시스템 변수를 사용하여 숨길 필드를 설정할 수 있습니다.	예
$numeric	필드의 모든(NULL이 아닌) 값이 숫자입니다.	예
$integer	필드의 모든(NULL이 아닌) 값이 정수입니다.	예
$text	필드 값에 숫자가 없습니다.	예
$ascii	필드 값에 표준 ASCII 문자만 포함됩니다.	예
$date	필드의 모든(NULL이 아닌) 값을 날짜(정수)로 해석할 수 있습니다.	예
$timestamp	필드의 모든(NULL이 아닌) 값을 타임스탬프로 해석할 수 있습니다.	예
$geoname	필드 값에 포인트 필드($geopoint) 및/또는 영역 필드 ($geomultipolygon)에 관한 지리적 위치의 이름이 포함됩니다.	예
$geopoint	필드 값에 [경도, 위도] 형식으로 맵의 포인트를 나타내는 도형 포인트 데이터가 포함됩니다.	예
$geomultipolygon	필드 값에 맵상의 영역을 나타내는 다각형 데이터가 포함됩니다.	예

파생된 필드 태그

다음 태그는 파생된 필드를 선언할 때 꺾은선형 차트의 연속 축에서 필드를 사용하는 방법과 표시하는 방법을 지정하는 데 사용할 수 있습니다. 데이터 로드 스크립트에서 태그를 편집할 수 있습니다.

파생된 필드 태그

Tag	설명
$axis	$axis 태그는 필드가 차트의 연속 축에 눈금을 생성하도록 지정하는 데 사용됩니다.

Tag	설명
$qualified $simplified	두 개의 서로 다른 필드를 파생시켜 축 레이블의 한정되고 간소화된 버전을 지정할 수 있습니다. 한정된 필드는 축이 더 깊은 수준으로 확대되면 레이블로 표시되어 전체 컨텍스트를 표시합니다. 예를 들어, 분기별 데이터를 표시하는 경우 다음 두 개의 필드를 생성할 수 있습니다. 간소화된 필드($simplified 태그 사용, 'Q1'과 같이 분기 표시) 및 한정된 필드($qualified 태그 사용, '2016-Q1'과 같이 연도 및 분기 표시) 시간 축이 축소되면 축은 간소화된 필드를 사용하여 레이블을 연도(2016) 및 분기(Q1)의 두 개 수준으로 표시합니다. 확대하면 해당 축은 분기 및 월에 대한 레이블을 표시하고 한정된 필드(2016-Q1)는 분기에 대한 전체 연도 컨텍스트를 제공하는 데 사용됩니다.
$cyclic	$cyclic 태그는 이중 데이터 표현을 가진 순환 필드(예: 분기 또는 월)에 사용됩니다.

시스템 필드

데이터 소스에서 추출된 필드 이외에도 Qlik Sense에서 시스템 필드도 생성됩니다. 이 모든 필드는 "$"로 시작되며 시각화에 필터 창이나 테이블 등과 같은 일반 필드처럼 표시할 수 있습니다. 시스템 필드는 데이터 로드 시 자동으로 생성되며 주로 앱 디자인 보조용으로 사용됩니다.

사용 가능한 시스템 필드

다음과 같은 시스템 필드를 사용할 수 있습니다.

사용 가능한 시스템 필드

시스템 필드	설명
$Table	로드된 모든 테이블을 포함합니다.
$Field	로드된 테이블의 모든 필드를 포함합니다.
$Fields	각 테이블의 필드 수를 포함합니다.
$FieldNo	테이블에 있는 필드의 위치를 포함합니다.
$Rows	테이블에 있는 행의 수를 포함합니다.

스크립트에서 시스템 필드를 조작할 수 없습니다.

시각화에서 시스템 필드 사용

시스템 필드 데이터는 연결되어 있습니다. 예를 들어 하나에는 $Table, 다른 하나에는 $Field가 있는 두 개의 필터 창을 추가하는 경우, 테이블 하나를 선택하면 $Field 필터 창에 선택한 테이블의 필드가 가능한 값으로 표시됩니다.

시스템 필드는 자산 패널의 필드 목록에 포함되지 않습니다. 식 편집기에 포함됩니다. 자산 패널에서 시스템 필드를 사용하려면 수동으로 입력하여 참조해야 합니다.

자산 패널의 차원에서

=$Field

필드 이름 바꾸기

원하는 연결을 얻기 위해 필드 이름을 변경해야 하는 경우가 있습니다. 필드 이름을 변경하는 3 가지 주요 이유는 다음과 같습니다.

- 이름은 다르지만 동일한 내용을 나타내는 두 개의 필드:
  - *Customers* 테이블의 *ID* 필드
  - *Orders* 테이블의 *CustomerID* 필드

  두 필드는 특정 고객 ID 코드를 나타내며 둘 다 같은 이름으로 지정되어야 합니다(예: *CustomerID*).
- 이름은 같지만 실제로는 다른 내용을 나타내는 두 개의 필드:
  - *Invoices* 테이블의 *Date* 필드
  - *Orders* 테이블의 *Date* 필드

  두 필드는 *InvoiceDate* 및 *OrderDate* 등으로 이름을 바꾸는 것이 좋습니다.
- 데이터베이스 철자 오기 또는 대소문자 규칙 차이와 같은 오류가 있을 수 있습니다.

스크립트에서 필드 이름을 바꿀 수 있기 때문에 원본 데이터를 변경할 필요가 없습니다. 아래 예에 나온 대로 필드의 이름을 변경하는 방법은 두 가지가 있습니다.

Example 1: alias 문 사용

LOAD 또는 **SELECT** 문 앞에 **alias** 문이 올 수 있습니다.

```
Alias ID as CustomerID;
LOAD * from Customer.csv;
```

Example 2: as 지정자 사용

LOAD 또는 **SELECT** 문에는 **as** 지정자를 포함할 수 있습니다.

```
LOAD ID as CustomerID, Name, Address, Zip, City, State from Customer.csv;
```

논리 테이블

각 **LOAD** 또는 **SELECT** 문은 테이블을 생성합니다. 일반적으로 Qlik Sense는 각 문의 결과를 하나의 논리 테이블로 취급합니다. 그러나 이 규칙에는 몇 가지 예외가 있습니다.

- 두 개 이상의 문이 동일한 필드 이름을 가진 테이블을 생성할 경우 해당 테이블이 연결되어 하나의 논리 테이블로 취급됩니다.
- **LOAD** 또는 **SELECT** 문 앞에 다음과 같은 한정자가 올 경우 데이터가 변경되거나 다르게 취급됩니다.

논리 테이블

테이블	설명
concatenate	이 테이블이 다른 이름을 가진 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 연결됩니다.
crosstable	이 테이블은 피벗 해제됩니다. 즉, 교차 표 형식에서 열 형식으로 변환됩니다.

테이블	설명
generic	이 테이블이 다른 여러 논리 테이블로 분할됩니다.
info	이 테이블은 논리 테이블로 로드되지 않고 파일, 사운드, URL 등과 같은 외부 정보에 대한 링크를 포함하는 정보 테이블로 로드됩니다.
intervalmatch	테이블(정확히 2개의 열을 포함해야 함)이 지정된 필드 내 불연속 숫자와 연결된 숫자 간격으로 해석됩니다.
join	Qlik Sense에서는 다른 이름의 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 공통적인 필드를 통해 이 테이블을 조인합니다.
keep	이 테이블이 다른 이름의 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 공통적인 필드로 축소됩니다.
mapping	이 테이블(2개의 열을 포함해야 함)이 다른 테이블과 연결되지 않는 매핑 테이블로 로드됩니다.
semantic	이 테이블은 논리 테이블이 아니라 조인되지 않아야 하는 관계를 포함하는 의미 체계 테이블로 로드됩니다(예: 선행자, 후속자 및 같은 유형의 다른 개체에 대한 참조).

데이터가 로드되면 논리 테이블이 연결됩니다.

테이블 이름

Qlik Sense 테이블은 Qlik Sense 데이터베이스에 저장될 때 이름이 지정됩니다. 테이블 이름은 **resident** 절이 있는 **LOAD** 문이나 **peek** 함수를 포함하는 표현식 등에서 사용할 수 있으며 레이아웃의 *$Table* 시스템 필드에서 볼 수 있습니다.

테이블의 이름은 다음 규칙에 따라 지정됩니다.

1. 레이블이 **LOAD** 또는 **SELECT** 문 바로 앞에 있는 경우 해당 레이블이 테이블 이름으로 사용됩니다. 레이블 이름 다음에는 콜론이 와야 합니다.

   ```
   Table1:
   LOAD a,b from c.csv;
   ```

2. 레이블이 지정되지 않은 경우 **LOAD** 또는 **SELECT** 문에서 키워드 **FROM** 다음에 바로 나오는 파일 이름이나 테이블 이름이 사용됩니다. 최대 32자가 사용됩니다. 파일 이름이 사용된 경우 확장명은 생략됩니다.
3. 인라인으로 로드된 테이블은 INLINExx 형식으로 이름이 지정됩니다. 여기서 xx는 숫자입니다. 첫 번째 인라인 테이블의 이름은 *INLINE01*이 됩니다.
4. 자동으로 생성된 테이블은 AUTOGENERATExx 형식으로 이름이 지정됩니다. 여기서 xx는 숫자입니다. 자동 생성된 첫 번째 테이블의 이름은 *AUTOGENERATE01*이 됩니다.

5. 위의 규칙에 따라 생성된 테이블 이름이 이전 테이블 이름과 충돌하는 경우는 이름이 -x로 확장됩니다. 여기서 x는 숫자입니다. 숫자는 더 이상 충돌이 없을 때까지 증가합니다. 예를 들어 세 개의 테이블 이름이 *Budget*, *Budget-1* 및 *Budget-2*로 지정될 수 있습니다.

테이블 이름에는 **section access**, **section application** 및 매핑 테이블의 세 가지 개별 도메인이 있습니다. **section access** 및 **section application**에서 생성된 테이블 이름은 개별적으로 처리됩니다. 참조된 테이블 이름이 섹션에서 발견되지 않는 경우 Qlik Sense는 다른 섹션도 검색합니다. 매핑 테이블은 별도로 처리되며 테이블 이름의 다른 두 도메인과는 아무런 연관이 없습니다.

테이블 레이블

테이블은 **resident** 절이 있는 **LOAD** 문이나 **peek** 함수를 포함하는 표현식으로 레이블을 지정하여 나중에 참조할 수 있습니다. 레이블은 숫자 또는 문자로 된 임의의 문자열이며 테이블을 만드는 첫 번째 **LOAD** 또는 **SELECT** 문 앞에 와야 합니다. 레이블 다음에는 콜론(":")이 와야 합니다.

공백이 포함된 레이블은 작은따옴표나 큰따옴표 또는 대괄호를 사용하여 인용해야 합니다.

Example 1:

```
Table1:
LOAD a,b from c.csv;
LOAD x,y from d.csv where x=peek('a',y,'Table1');
```

Example 2: 공백이 포함된 테이블 레이블

```
[All Transactions]:
SELECT * from Transtable;
LOAD Month, sum(Sales) resident [All Transactions] group by Month;
```

논리 테이블 간의 연결

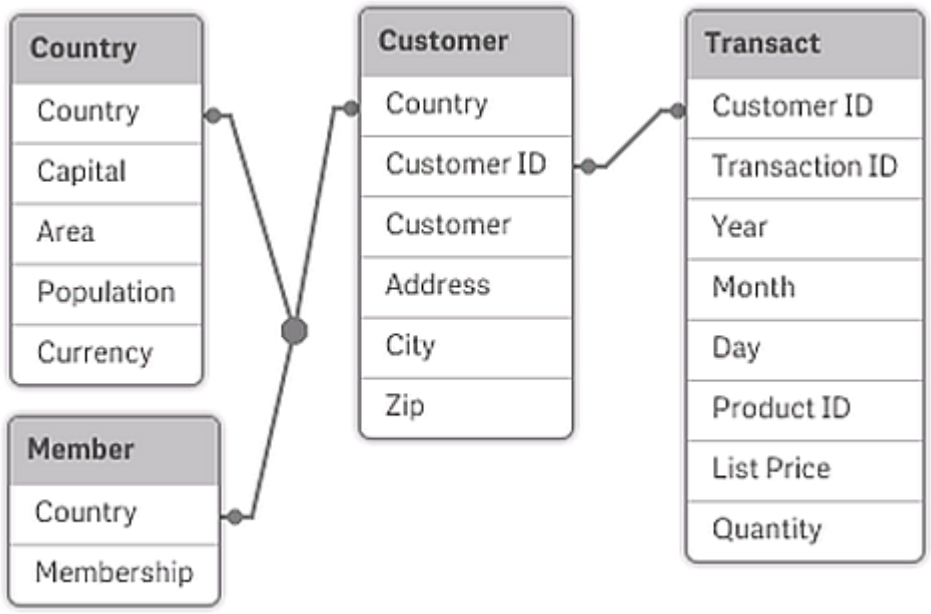
데이터베이스는 다수의 테이블을 포함할 수 있습니다. 각 테이블은 무언가에 대한 목록으로 간주할 수 있으며, 목록의 각 레코드는 어떤 유형의 개체에 대한 인스턴스를 나타냅니다.

두 테이블이 다른 것에 대한 목록(예: 한 테이블은 고객 목록이고 다른 테이블은 송장 목록인 경우)이고 고객 수와 같은 필드를 공통적으로 가지고 있는 경우 이는 일반적으로 두 테이블 간에 관계가 있다는 의미입니다. 표준 SQL 쿼리 도구에서 두 테이블은 거의 항상 조인되어야 합니다.

Qlik Sense 스크립트에서 정의된 테이블을 논리 테이블이라고 합니다. Qlik Sense는 필드 이름에 따라 테이블 간에 연결하며 선택을 수행할 때(예: 필터 창의 필드 값 선택) 조인을 수행합니다.

이는 연결이 조인과 거의 동일함을 의미합니다. 유일한 차이점은 조인은 스크립트가 실행될 때 수행된다는 점입니다. 논리 테이블은 일반적으로 조인의 결과입니다. 논리 테이블이 생성되면 연결이 완료됩니다. 연결은 항상 논리 테이블 간에 이루어집니다.

4개 테이블: Country 및 CustomerID 필드를 통해 서로 연결된 국가 목록, 고객 목록, 트랜잭션 목록 및 멤버십 목록



Qlik Sense 연결과 SQL 자연 외부 조인 비교

Qlik Sense 연결은 SQL 자연 외부 조인과 비슷합니다. 하지만 연결이 좀 더 일반적입니다. SQL의 외부 조인은 보통 한 테이블을 다른 테이블에 단방향으로 투영합니다. 연결은 항상 완전(양방향) 자연 외부 조인이 됩니다.

연결 필드 내 빈도 정보

대부분의 연결 필드, 즉 두 개 이상의 테이블 간에 공통적인 필드는 사용 시 제한 사항이 있습니다. 둘 이상의 테이블에서 필드가 발생하면 Qlik Sense가 데이터 빈도를 계산할 때 어느 테이블을 사용해야 할지 인식하는 데 문제가 생깁니다.

Qlik Sense는 데이터를 분석하여 계수할 기본 테이블을 모호하지 않게 식별하는 방법이 있는지 확인하지만 (있는 경우도 있음) 대부분의 경우 프로그램은 추측만 할 수 있습니다. 잘못된 추측은 치명적인 결과를 야기할 수 있으므로(Qlik Sense에 계산 오류가 나타날 수 있음) 이 프로그램은 연결 필드에 대한 데이터 해석이 모호한 경우 특정 작업을 허용하지 않도록 설계되었습니다.

연결 필드에 대한 제한

1. 필드를 표시하는 필터 창에 빈도 정보를 표시할 수 없습니다.
2. 필드의 통계 상자에는 대부분의 통계 엔터티가 n/a로 표시됩니다.
3. **Distinct** 수정자가 활성화되지 않은 경우 차트에서 필드의 빈도 정보에 의존하는 함수(Sum, Count 함수, Average 등)를 포함하는 표현식을 만들 수 없습니다. Qlik Sense는 재로드 후에 항상 모든 차트 표현식을 검사하여 데이터 구조 내 변화로 인해 모호성이 발생했는지 확인합니다. 모호한 표현식이 발견되면 경고 대화 상자가 표시되고 표현식이 비활성화됩니다. 문제가 해결될 때까지 표현식을 활성화할 수 없습니다. 로그 파일이 활성화되면 모든 모호한 표현식은 로그에 나열됩니다.

해결책

이러한 제한에 대처할 간단한 방법이 있습니다. 빈도를 세야 하는 테이블에서 필드를 새 이름으로 로드합니다. 그리고 빈도를 포함한 필터 창에 대한 새 필드를 차트 내 통계 상자 또는 계산에 사용합니다.

가상 키

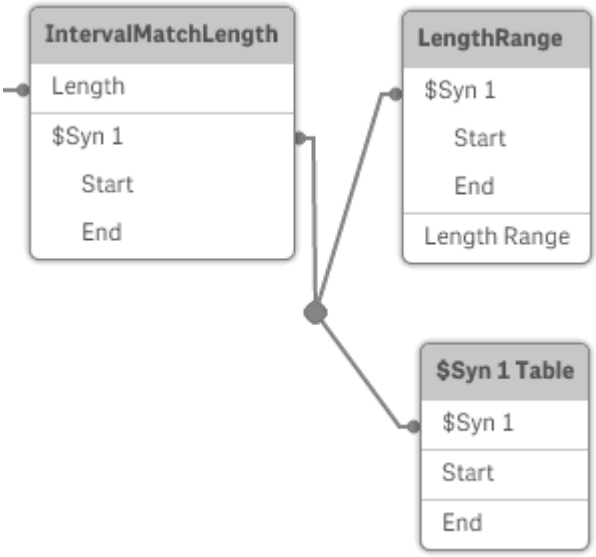
Qlik Sense는 둘 이상의 데이터 테이블에 둘 이상의 공통 필드가 있는 경우 가상 키를 만듭니다. 이 키는 복합 키에서 발생하는 모든 조합을 나타내는 익명 필드입니다.

데이터를 로드할 때 가상 키에 대한 경고를 받는 경우, 데이터 모델 뷰어에서 데이터 구조를 검토해야 합니다. 데이터 모델이 정확한지 여부를 확인해 보아야 합니다. 때때로 스크립트에서 오류로 가상 키가 생기는 경우가 많습니다.

복수의 가상 키는 종종 잘못된 데이터 모델의 증상을 보입니다. 그러나 잘못된 데이터 모델의 확실한 증거는 다른 가상 키에 기반한 가상 키를 갖는 경우입니다.

> *가상 키 개수가 증가하면 데이터 양, 테이블 구조 및 기타 요소에 따라 Qlik Sense는 이들을 원활하게 처리할 수도, 하지 않을 수도 있으며, 결국 시간 및/또는 메모리를 초과하여 사용하게 될 수 있습니다. 그런 경우 모든 가상 키를 제거하여 스크립트를 다시 작성해야 합니다.*

가상 키 $Syn 1과 관련된 세 개의 테이블



가상 키 처리

가상 키를 피해야 하는 경우 데이터 로드 스크립트에서 이를 해결하는 여러 가지 방법이 있습니다.

- 두 테이블을 논리적으로 연결하는 필드만 키로 사용하고 있는지 확인하십시오.
  - "Comment", "Remark" 및 "Description" 등의 필드는 관련성 없이 여러 테이블에 존재할 수 있으므로 키로 사용하지 않도록 해야 합니다.
  - "Date", "Company" 및 "Name" 등의 필드는 여러 테이블에 존재하며 동일한 값을 가질 수 있지만 여전히 여러 가지 역할을 가질 수 있습니다(주문 날짜/배송 날짜, 고객 회사/공급업체). 이런 경우에도 키로 사용하지 않아야 합니다.

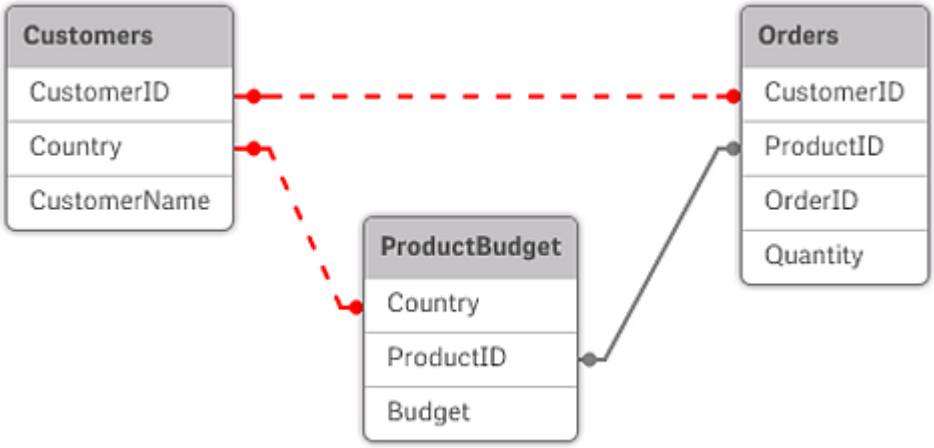
- 중복 필드를 사용하지 않도록 해야 합니다. 즉, 필요한 필드만 연결합니다. 예를 들어 날짜가 키로 사용된 경우 둘 이상의 내부 테이블에서 동일한 날짜의 year, month 또는 day_of_month를 로드하지 않도록 하십시오.
- 필요한 경우 일반적으로 AutoNumber 스크립트 함수 내에서 문자열 연결을 사용하여 사용자 지정 비복합 키를 형성합니다.

순환 참조에 대한 이해 및 해결

데이터 구조에 순환 참조("루프")가 있는 경우 두 필드 간 연결 경로가 둘 이상인 것처럼 테이블이 연결됩니다.

이러한 유형의 데이터 구조를 사용하면 데이터 해석이 모호해질 수 있으므로 최대한 사용하지 않는 것이 좋습니다.

두 개의 필드 사이에 둘 이상의 연결 경로가 있으므로 순환 참조가 있는 세 개의 테이블



Qlik Sense는 느슨하게 결합된 테이블을 통해 루프를 해체하여 순환 참조 문제를 해결합니다. Qlik Sense가 로드 스크립트를 실행하는 동안 순환 데이터 구조를 찾으면 경고 대화 상자가 표시되며 하나 이상의 테이블이 느슨한 결합으로 설정됩니다. Qlik Sense는 일반적으로 루프에서 가장 긴 테이블을 느슨하게 하려고 시도하는데, 주로 트랜잭션 테이블을 느슨하게 하는 경우가 많습니다. 데이터 모델 뷰어에서 느슨하게 연결된 테이블은 다른 테이블에 연결된 빨간 점선으로 표시됩니다.

세 테이블에서 다음을 포함하는 데이터를 로드합니다.

- 일부 국가대표 축구 팀의 이름
- 일부 도시의 축구 클럽
- 일부 유럽 국가의 도시

Excel에서 소스 데이터 테이블 보기

NationalTeams

Country	Team
Germany	Die Mannschaft
Italy	Azzurri
Spain	La Roja

Sheet1 Sheet

Clubs

City	Team
Barcelona	Barcelona
Hamburg	Altona
Madrid	Real Madrid
Milano	Milan
Munich	Bayern München
Rome	Lazio
Turin	Juventus

Sheet1 Sheet2

Cities

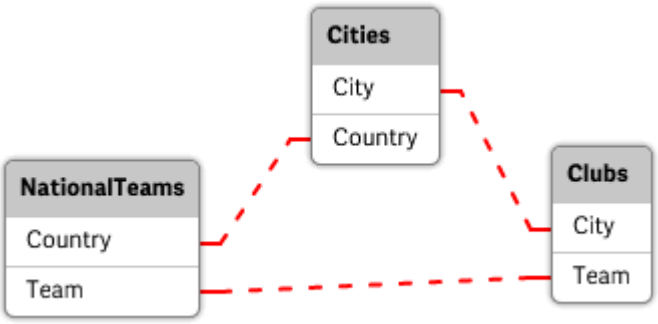
Country	City
Germany	Hamburg
Germany	Munich
Italy	Milano
Italy	Rome
Italy	Turin
Spain	Barcelona
Spain	Madrid

이 데이터 구조는 *Team*이라는 필드 이름을 국가대표 팀과 지역 클럽의 두 가지 용도로 사용하고 있으므로 별로 좋지 않습니다. 테이블의 데이터에서 해결이 불가능한 논리적 상황이 발생합니다.

Qlik Sense는 테이블을 Qlik Sense에 로드할 때 중요도가 가장 낮은 데이터 연결을 식별하고 해당 테이블의 연결을 느슨하게 만듭니다.

데이터 모델 뷰어를 열어 Qlik Sense에서 데이터 연결의 관련성을 해석하는 방법을 확인할 수 있습니다.

빨간색 점선으로 표시된 순환 참조의 보기



이제 도시와 도시가 속한 국가를 포함하는 테이블이 여러 국가의 국가대표 팀을 포함하는 테이블 및 다른 도시의 지역 클럽을 포함하는 테이블과 느슨하게 결합되었습니다.

순환 참조의 해결

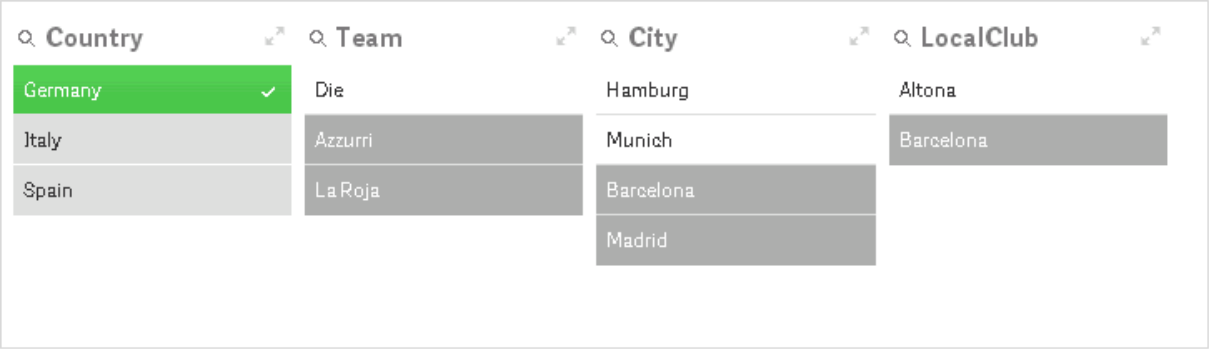
순환 참조가 발생한 경우 데이터 로드 스크립트를 편집하여 이름이 동일한 필드 중 하나에 고유한 이름을 할당해야 합니다.

다음과 같이 하십시오.

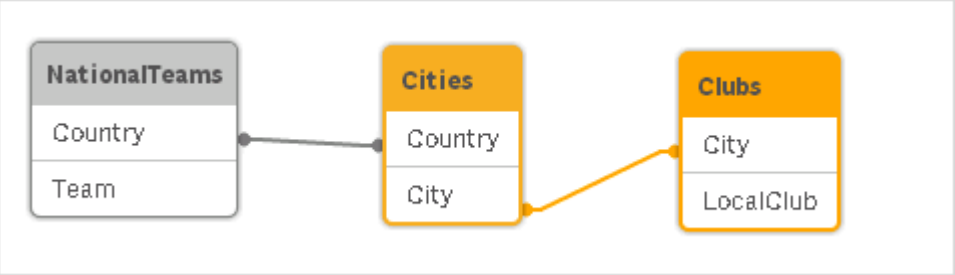
1. 데이터 로드 편집기를 엽니다.
2. 중복 필드 이름 중 하나에 대해 **LOAD** 문을 편집합니다.
   이 예에서는 지역 팀과 해당 도시를 포함하는 테이블의 **LOAD** 문에서 *Team*에 *LocalClub*이라는 새 이름을 지정합니다. 업데이트된 **LOAD** 문은 다음과 같습니다.
   ```
   LOAD City, Team as LocalClub
   ```

3. 도구 모음에서 ▶을 클릭하여 앱의 데이터를 다시 로드합니다.

이제 모든 테이블에서 처리되는 논리가 생성되었습니다. 이 예에서는 국가대표 팀으로 *Germany*를 선택하면 이탈리아의 도시와 각 도시의 지역 클럽이 연관됩니다.



데이터 모델 뷰어를 열면 느슨하게 결합된 연결이 일반 연결로 바뀐 것을 알 수 있습니다.



테이블 연결

연결은 두 테이블을 하나로 결합하는 연산입니다.

두 테이블은 단순히 서로 추가되는 것입니다. 즉, 데이터는 변경되지 않으며 결과 테이블에는 원래의 두 테이블을 합한 것과 동일한 수의 레코드가 포함됩니다. 여러 개의 연결 연산을 연속으로 수행할 수 있으며, 이런 경우 결과 테이블은 셋 이상의 테이블에서 컨케트네이트됩니다.

자동 연결

로드한 두 개 이상의 테이블에서 필드 이름과 필드의 수가 완전히 동일한 경우 Qlik Sense는 자동으로 여러 문의 내용을 한 테이블로 연결합니다.

```
LOAD a, b, c from table1.csv;
LOAD a, c, b from table2.csv;
```

결과 내부 테이블에는 a, b 및 c 필드가 생깁니다. 레코드 수는 테이블 1과 테이블 2의 레코드 수를 합한 수입니다.

> *필드의 수와 이름이 완전히 동일해야 합니다. 두 문의 순서는 관계가 없습니다.*

강제 연결

두 개 이상의 테이블이 완전하게 동일한 필드 집합을 가지지 않는 경우에도 Qlik Sense에서 두 테이블을 강제로 컨케트네이트할 수 있습니다. 이를 위해서는 스크립트에 **concatenate** 접두사를 사용하여 다른 명명된 테이블이나 최근에 생성한 테이블과 특정 테이블을 컨케트네이트하면 됩니다.

```
LOAD a, b, c from table1.csv;
concatenate LOAD a, c from table2,csv;
```

결과 내부 테이블에는 a, b 및 c 필드가 생깁니다. 결과 테이블의 레코드 수는 테이블 1과 테이블 2의 레코드 수를 합한 수입니다. 테이블 2에서 가져온 레코드의 b 필드 값은 NULL입니다.

> *이전에 로드한 테이블의 테이블 이름이 **concatenate** 문에 지정되지 않은 경우 **concatenate** 접두사는 가장 최근에 만든 테이블을 사용합니다. 따라서 두 문의 순서는 임의가 아닙니다.*

연결 방지

로드한 두 개 이상의 테이블에서 필드 이름과 필드의 수가 완전히 동일한 경우 Qlik Sense는 자동으로 여러 문의 내용을 한 테이블로 컨케트네이트합니다. **noconcatenate** 문을 사용하면 이를 방지할 수 있습니다. 그러면 연관된 **LOAD** 또는 **SELECT** 문을 사용하여 로드된 테이블이 기존 테이블과 컨케트네이트되지 않습니다.

```
LOAD a, b, c from table1.csv;
noconcatenate LOAD a, b, c from table2.csv;
```

이전에 로드한 테이블에서 데이터 로드

이미 로드된 테이블에서 데이터를 로드하고 변환하는 방법은 두 가지가 있습니다.

- **Resident LOAD** - 후속 **LOAD** 문에서 **Resident** 조건자를 사용하여 새 테이블을 로드하는 경우.
- 선행 load - 소스를 지정하지 않고 선행 **LOAD** 또는 **SELECT** 문에서 로드하는 경우.

Resident 또는 선행 LOAD?

대부분의 경우에는 어느 방법을 사용해도 동일한 결과를 얻을 수 있습니다. 일반적으로 선행 **LOAD**가 더 빠른 옵션이지만, **Resident LOAD**를 대신 사용해야 하는 경우도 있습니다.

- **LOAD** 문을 처리하기 전에 **Order_by** 절을 사용하여 레코드를 정렬하려는 경우.
- 선행 **LOAD**가 지원되지 않아서 다음 접두사 중 하나를 사용하려는 경우.
    - **Crosstable**
    - **Join**
    - **Intervalmatch**

Resident LOAD

LOAD 문에서 **Resident** 조건자를 사용하면 이전에 로드한 테이블에서 데이터를 로드할 수 있습니다. 날짜 또는 숫자 값 처리 등과 같이 Qlik Sense 함수를 사용하는 옵션이 없는 **SELECT** 문으로 로드한 데이터에서 계산을 수행하려는 경우에 유용합니다.

이 예에서는 초기 **Crosstable LOAD**에서 수행할 수 없으므로 **Resident** 로드 시 날짜 해석이 수행됩니다.

```
PreBudget:
Crosstable (Month, Amount, 1)
LOAD Account,
   Jan,
   Feb,
   Mar,
…
From Budget;

Budget:
Noconcatenate
LOAD
   Account,
   Month(Date#(Month,'MMM')) as Month,
   Amount
Resident PreBudget;

Drop Table PreBudget;
```

> ***Resident****를 사용하는 일반적인 경우는 임시 테이블을 사용하여 계산이나 필터링을 수행하려는 경우입니다. 임시 테이블을 사용하는 목적을 달성한 후에는* ***Drop table*** *문을 사용하여 제거해야 합니다.*

선행 load

선행 load 기능을 사용하면 한 단계로 테이블을 로드할 수 있지만 여러 번의 연속된 변환을 정의해야 합니다. 일반적으로 사용하는 **From** 또는 **Resident** 등의 소스 한정자를 지정하지 않고 기본적으로 아래의 **LOAD** 또는 **SELECT** 문에서 로드하는 **LOAD** 문입니다. 이런 방식으로 개수에 관계없이 **LOAD** 문을 누적할 수 있습니다. 맨 아래의 문을 처음으로 평가한 다음 그 위의 문을 평가하는 방식으로 맨 위의 문이 평가될 때까지 계속 진행됩니다.

Resident를 사용하여 동일한 결과를 얻을 수 있지만 대부분의 경우 선행 **LOAD**가 훨씬 빠릅니다.

선행 load의 또 다른 장점은 한 곳에서 계산을 유지할 수 있으며, 위에 배치된 **LOAD** 문에서 재사용할 수 있습니다.

Example 1: SELECT 문으로 로드한 데이터 변환

SELECT 문을 사용하여 데이터베이스에서 데이터를 로드하면 Qlik Sense 함수를 사용하여 **SELECT** 문에서 데이터를 해석할 수 없습니다. 해결책은 **LOAD** 문을 추가하여 **SELECT** 문 위에서 데이터 변환을 수행하는 것입니다.

이 예에서는 이전 **SELECT** 문을 소스로 사용하여, **LOAD** 문에서 Qlik Sense 함수 **Date#**을 사용하여 문자열로 저장한 날짜를 해석합니다.

```
LOAD Date#(OrderDate,'YYYYMMDD') as OrderDate;
SQL SELECT OrderDate FROM … ;
```

Example 2: 계산을 재사용하여 스크립트 간소화

이 예의 경우, 스크립트에서 계산을 두 번 이상 사용합니다.

```
LOAD Age(FromDate + IterNo() – 1, BirthDate ) as Age,
   Date(FromDate + IterNo() – 1 ) as ReferenceDate
   Resident Policies
      While IterNo() <= ToDate - FromDate + 1 ;
```

첫 번째 단계에서 계산을 도입하면 선행 **LOAD**의 Age 함수에서 재사용할 수 있습니다.

```
LOAD ReferenceDate,
   Age(ReferenceDate, BirthDate ) as Age;
LOAD *,
   Date(FromDate + IterNo() – 1 ) as ReferenceDate
   Resident Policies
      While IterNo() <= ToDate - FromDate + 1 ;
```

선행 LOAD의 제한

- **Join**, **Crosstable** 및 **Intervalmatch** 접두사는 선행 **LOAD**와 연결하여 사용할 수 없습니다.
- **distinct**를 사용하여 고유한 레코드를 로드하는 경우 **distinct**는 대상 테이블에만 영향을 주므로 첫 번째 LOAD 문에 **distinct**를 배치해야 합니다.

Qlik Sense의 데이터 유형

Qlik Sense에서는 텍스트 문자열, 숫자, 날짜, 시간, 타임스탬프 및 통화를 정확하게 처리할 수 있습니다. 이러한 데이터를 정렬하고 다양한 형식으로 표시하고 계산에 사용할 수 있습니다. 예를 들어 날짜, 시간 및 타임스탬프를 서로 더하거나 뺄 수 있음을 의미합니다.

Qlik Sense 내부의 데이터 표현

Qlik Sense의 데이터 해석 및 숫자 서식 지정 방식을 이해하려면 프로그램에서 내부적으로 데이터를 어떻게 저장하는지 알아야 합니다. Qlik Sense에 로드되는 모든 데이터는 문자열과 숫자의 두 가지 표현으로 사용할 수 있습니다.

1. 문자열 표현은 항상 사용 가능하며, 목록 상자 및 기타 시트 개체에 표시되는 표현입니다. 목록 상자 내 데이터의 서식 지정(숫자 서식)은 문자열 표현에만 영향을 줍니다.
2. 숫자 표현은 데이터를 유효한 숫자로 해석할 수 있을 때에만 사용 가능합니다. 숫자 표현은 모든 수치 계산 및 숫자 정렬에 사용됩니다.

한 필드에 읽어들인 여러 데이터 항목이 동일한 숫자 표현을 가지고 있다면 동일한 값으로 취급되며 첫 번째로 발견되는 문자열 표현을 공유하게 됩니다. 예: 숫자 1.0, 1 및 1.000을 순서대로 읽어온다면 모두 동일한 숫자 표현 1을 갖게 되며, 최초 문자열 표현은 1.0이 됩니다.

숫자 해석

숫자, 통화 또는 날짜가 포함된 데이터를 로드하면 데이터 형식을 정의했는지 여부에 따라 다르게 해석됩니다. 이 섹션에서는 두 가지 서로 다른 사례에서 데이터를 해석하는 방법을 설명합니다.

유형 정보가 포함된 데이터

ODBC를 사용하여 로드된 데이터베이스에서 데이터 유형이 정의된 숫자를 포함하는 필드는 Qlik Sense에서 각각의 형식에 따라 처리됩니다. 해당 필드의 문자열 표현은 적절한 서식이 적용된 숫자가 됩니다.

Qlik Sense는 속성 패널의 **숫자 서식**에서 측정값에 대한 숫자 서식이 변경되더라도 필드의 원래 숫자 서식을 기억합니다.

다양한 데이터 유형에 적용되는 기본 서식은 다음과 같습니다.

- 정수, 부동 소수점 숫자: 숫자의 기본 설정
- 통화: 통화의 기본 설정
- 시간, 날짜, 타임스탬프: ISO 표준 서식 지정

숫자와 통화의 기본 설정은 스크립트 숫자 해석 변수 또는 운영 체제 설정(**제어판**)을 사용하여 정의됩니다.

유형 정보가 없는 데이터

소스에서 전해진 특정 서식 정보가 없는 데이터(예: 텍스트 파일에서 나온 데이터 또는 일반 서식의 ODBC 데이터)의 경우 상황이 더 복잡해집니다. 최종 결과는 최소 6개 이상의 다양한 요소에 따라 결정됩니다.

- 소스 데이터베이스에서 데이터가 기록된 방식
- 숫자, 시간, 날짜 등에 대한 운영 체제 설정(**제어 패널**)
- 스크립트에서 선택적 숫자 해석 변수 사용
- 스크립트에서 선택적 해석 함수 사용
- 스크립트에서 선택적 서식 지정 함수 사용
- 문서 내의 숫자 서식 콘트롤

Qlik Sense는 입력 데이터를 숫자, 날짜, 시간 등으로 해석하려고 시도합니다. 데이터에 시스템 기본 설정이 사용되는 경우에 한해 Qlik Sense에서 해석과 표시 서식 지정을 자동으로 처리하며 사용자가 스크립트 또는 Qlik Sense의 설정을 변경할 필요가 없습니다.

기본적으로 완벽한 일치가 발견되기 전까지 다음과 같은 체계가 사용됩니다. (기본 서식은 운영 체제의 **제어판** 등에서 지정되거나 경우에 따라 스크립트 내의 특수 숫자 해석 변수를 통해 지정된 소수점 구분 기호, 년, 월, 일의 순서 등과 같은 서식을 의미합니다.)

Qlik Sense는 데이터를 다음과 같이 해석합니다.

1. 숫자의 기본 서식에 따른 숫자
2. 날짜의 기본 서식에 따른 날짜
3. 시간 및 날짜의 기본 서식에 따른 타임스탬프
4. 시간의 기본 서식에 따른 시간
5. 다음 서식에 따른 날짜: yyyy-MM-dd.
6. 다음 서식에 따른 타임스탬프: YYYY-MM-DD hh:mm[:ss[.fff]].
7. 다음 서식에 따른 시간: hh:mm[:ss[.fff]].
8. 통화의 기본 서식에 따른 화폐
9. 소수점 구분 기호 또는 천 단위 구분 기호 중 어느 것도 ','로 설정되지 않은 경우 소수점 구분 기호가 '.' 이고 천 단위 구분 기호가 ','인 숫자

10. 소수점 구분 기호 또는 천 단위 구분 기호 중 어느 것도 '.'로 설정되지 않은 경우 소수점 구분 기호가 ','이고 천 단위 구분 기호가 '.'인 숫자
11. 텍스트 문자열. 이 마지막 테스트는 결코 실패하지 않습니다. 데이터를 읽을 수 있다면 항상 해당 데이터를 문자열로 해석할 수 있습니다.

텍스트 파일에서 숫자를 로드할 때 잘못된 천 단위 구분 기호 또는 소수점 구분 기호와 같은 해석 문제가 발생하여 Qlik Sense에서 숫자가 잘못 해석될 수 있습니다. 가장 먼저 할 일은 스크립트 내의 숫자 해석 변수가 올바르게 정의되어 있으며 **제어판**의 시스템 설정이 올바른지 확인하는 것입니다.

Qlik Sense에서 날짜 또는 시간으로 해석한 데이터는 시각화의 속성 패널에서 다른 날짜 또는 시간 서식으로 변경할 수 있습니다.

데이터에 대해 사전 정의된 서식이 없으므로 당연히 여러 레코드의 동일한 필드에 서로 다른 서식이 지정된 데이터가 포함될 수 있습니다. 예를 들어 한 필드에 유효한 날짜, 정수 및 텍스트가 있을 수 있습니다. 따라서 데이터에 서식이 지정되지는 않지만 원래 서식대로 표시됩니다.

날짜 및 시간 해석

Qlik Sense는 데이터에서 발견된 각 날짜, 시간 및 타임스탬프를 날짜 일련 번호로 저장합니다. 날짜 일련 번호는 날짜, 시간 및 타임스탬프와 날짜 및 시간 엔터티를 기반으로 하는 산술 계산에 사용됩니다. 따라서 날짜와 시간을 더하고 빼거나 간격을 비교하는 등의 작업을 수행할 수 있습니다.

날짜 일련 번호는 1899년 12월 30일 이후로 경과된 날의 숫자(실수 값)입니다. 즉, Qlik Sense 서식에서 1900년 3월 1일부터 2100년 2월 28일까지는 Microsoft Excel 및 기타 프로그램에서 사용하는 1900 날짜 시스템과 동일합니다. 예를 들어 33857은 1992년 9월 10일에 해당합니다. 이 범위를 벗어나면 Qlik Sense에서 일반 달력과 동일하게 확장된 날짜 시스템을 사용합니다.

*필드에 1980년 1월 1일 이전의 날짜가 있는 경우 필드는 **$date** 또는 **$timestamp** 시스템 태그를 포함하지 않습니다. Qlik Sense에서 이 필드를 계속해서 날짜 필드로 인식해야 하지만 태그가 필요할 경우 **Tag** 문을 사용하여 데이터 로드 스크립트에서 수동으로 태그를 추가할 수 있습니다.*

시간에 해당하는 일련 번호는 0과 1 사이의 숫자입니다. 일련 번호 0.00000은 00:00:00에 해당하며, 0.99999는 23:59:59에 해당합니다. 혼합된 숫자는 날짜와 시간을 나타내며, 일련 번호 2.5는 1900년 1월 1일 12:00 정오를 의미합니다.

하지만 데이터는 문자열의 서식에 따라 표시됩니다. 기본적으로 **제어판**에서 지정한 설정이 사용됩니다. 또한 스크립트 내의 숫자 해석 변수 또는 서식 지정 함수를 사용하여 데이터의 서식을 설정할 수도 있습니다. 마지막으로 시트 개체의 속성 시트에서 데이터의 서식을 재지정할 수도 있습니다.

Example 1:

- 1997-08-06은 35648로 저장됨
- 09:00은 0.375로 저장됨
- 1997-08-06 09:00은 35648.375로 저장됨

다른 방법:

- 숫자 서식이 'D/M/YY'인 35648 의 경우 6/8/97과 같이 표시됨
- 숫자 서식이 'hh.mm'인 0.375의 경우 09.00과 같이 표시됨

Qlik Sense는 일련의 규칙에 따라 날짜, 시간 및 기타 데이터 유형을 해석하려 시도합니다. 하지만 최종 결과는 여기에 설명된 여러 요소의 영향을 받습니다.

Example 2:

아래 예는 다음과 같은 기본 설정을 가정한 것입니다.

- 숫자 소수점 구분 기호:
- 간단한 날짜 서식: YY-MM-DD
- 시간 서식: hh:mm

다음 테이블에는 스크립트에서 특별한 해석 함수를 사용하지 않고 Qlik Sense로 데이터를 읽을 때의 다양한 표현이 나와 있습니다.

스크립트에서 특정 해석 함수 없이 데이터를 읽을 때의 테이블

소스 데이터	Qlik Sense 기본 해석	'YYYY-MM-DD' 날짜 서식	'MM/DD/YYYY' 날짜 서식	'hh:mm' 시간 서식	'# ##0.00' 숫자 서식
0.375	0.375	1899-12-30	12/30/1899	09:00	0.38
33857	33857	1992-09-10	09/10/1992	00:00	33 857.00
97-08-06	97-08-06	1997-08-06	08/06/1997	00:00	35 648.00
970806	970806	4557-12-21	12/21/4557	00:00	970 806.00
8/6/97	8/6/97	8/6/97	8/6/97	8/6/97	8/6/97

다음 테이블에는 스크립트에서 date#(A, 'M/D/YY') 해석 함수를 사용하여 Qlik Sense로 데이터를 읽을 때의 다양한 표현이 나와 있습니다.

스크립트에서 date#(A, 'M/D/YY') 해석 함수를 사용할 때의 테이블

소스 데이터	Qlik Sense 기본 해석	'YYYY-MM-DD' 날짜 서식	'MM/DD/YYYY' 날짜 서식	'hh:mm' 시간 서식	'# ##0.00' 숫자 서식
0.375	0.375	0.375	0.375	0.375	0.375
33857	33857	33857	33857	33857	33857
97-08-06	97-08-06	97-08-06	97-08-06	97-08-06	97-08-06
970806	970806	970806	970806	970806	970806
8/6/97	8/6/97	1997-08-06	08/06/1997	00:00	35 648.00

달러 기호 확장

달러 기호 확장은 스크립트 또는 표현식에 사용되는 텍스트 대체에 대한 정의입니다. 이 프로세스는 새 텍스트가 더 짧을 경우에도 확장이라고 부릅니다. 대체는 스크립트 문 또는 표현식이 평가되기 직전에 이루어집니다. 엄밀히 말하자면 이는 매크로 확장에 속합니다.

확장은 항상 '$('로 시작하여 ')'로 끝납니다. 예: `$(variablename)`. 괄호 안의 내용은 텍스트 대체가 이루어지는 방법을 정의합니다.

달러 기호 확장은 계산할 수 있는 확장 수에 따라 제한됩니다. 중첩된 확장 수준이 1000개가 넘는 확장은 계산되지 않습니다.

변수를 사용한 달러 기호 확장

로드 스크립트 또는 차트 표현식에서 달러 기호 확장의 변수를 사용하여 다음을 수행합니다.

- 텍스트 참조
- 숫자 값 참조

텍스트 변수

스크립트 또는 표현식 내의 텍스트 대체에 변수를 사용하는 경우 다음과 같은 구문이 사용됩니다.

`$(variablename)`
`$(variablename)`은 변수의 값으로 확장됩니다. `variablename`이 존재하지 않을 경우 확장 결과는 빈 문자열이 됩니다.

예: 텍스트 변수 로드 스크립트

예 1: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
Set x = 'red';          // Assign the value "red" to variable x
Set y = 'blue';         // Assign the value "blue" to variable y
Set z = '$(x) $(y)';    // Expands x and y, returns "red blue" in variable z

// Expand x and y, return "red green blue" in variable MyString

Let MyString='$(x)'&' green '&'$(y)';

// Create table MyTable, load variable values for x, y, z into fields X, Y, Z
// Concatenate with variable MyString into field NewString

MyTable:
Load '$(x)' as X, '$(y)' as Y, '$(z)' as Z, '$(MyString)' as NewString autogenerate 1;
```

설명

이 예는 다음을 보여 줍니다.

- 변수 할당에서 변수를 확장하는 방법.
- 텍스트 작업과 결합된 변수를 확장하는 방법.

이는 가변 내용을 정적 문자열과 결합하는 동적 레이블 및 일반 텍스트 문자열을 만드는 데 유용한 설정입니다.

출력

Qlik Sense에서 다음 테이블을 만듭니다.

테이블 - 로드 스크립트의 출력

X	Y	Z	NewString
빨간색	파란색	빨간색 파란색	빨간색 파란색 녹색

예 2: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
Set vFunction = 'upper';  // Assign the string “upper” to variable vFunction
Set vField = 'String';    // Assign the string "String" to variable vField
Let vEvaluate = '$(vFunction)'&'('&'$(vField)'&')';

// The variable vEvaluate returns the value "upper(string)"

MyTable: // Create table called MyTable
Load *, $(vEvaluate) as Upper;          // vEvaluate expanded as a dynamic expression
Load *, '$(vEvaluate)' as Expression;   // vEvaluate expanded as string
Load * inline [
ID, String
1, abc
2, def
3, ghi
4, jkl ];
```

설명

Set 및 Let 문은 로드 스크립트의 변수에 값을 할당하는 데 사용됩니다. 이 둘의 차이점은 Set 문이 변수에 문자열을 할당하는 반면 Let 문은 결과 값을 변수에 할당하기 전에 문자열의 내용을 평가한다는 것입니다. 이 예의 로드 인라인 테이블은 변수 vEvaluate에 대한 서로 다른 평가를 텍스트 문자열 및 해당 표현식으로 시각화하는 데 사용되는 두 개의 선행 load 문으로 보완됩니다.

출력

Qlik Sense에서 다음 테이블을 만듭니다.

테이블 - 로드 스크립트의 출력

ID	String	Expression	Upper
1	abc	upper(String)	ABC
2	def	upper(String)	DEF
3	ghi	upper(String)	HIJ
4	jkl	upper(String)	JKL

예: 텍스트 변수 차트 표현식

예: 로드 스크립트

로드 스크립트

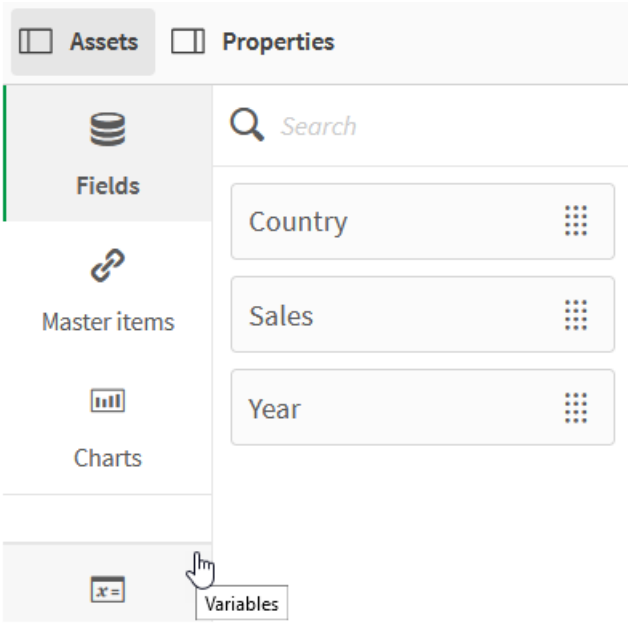
데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
// Create table SalesByCountry
SalesByCountry:
Load * Inline [
Country, Year, Sales
Argentina, 2014, 66295.03
Argentina, 2015, 140037.89
Austria, 2014, 54166.09
Austria, 2015, 182739.87
Belgium, 2014, 182766.87
Belgium, 2015, 178042.33
Brazil, 2014, 174492.67
Brazil, 2015, 2104.22
Canada, 2014, 101801.33
Canada, 2015, 40288.25
Denmark, 2014, 45273.25
Denmark, 2015, 106938.41
Finland, 2014, 107565.55
Finland, 2015, 30583.44
France, 2014, 115644.26
France, 2015, 30696.98
Germany, 2014, 8775.18
Germany, 2015, 77185.68
];
```

변수

편집 모드의 시트에서 **자산** 패널의 **변수** 대화 상자를 엽니다.

변수 대화 상자 열기

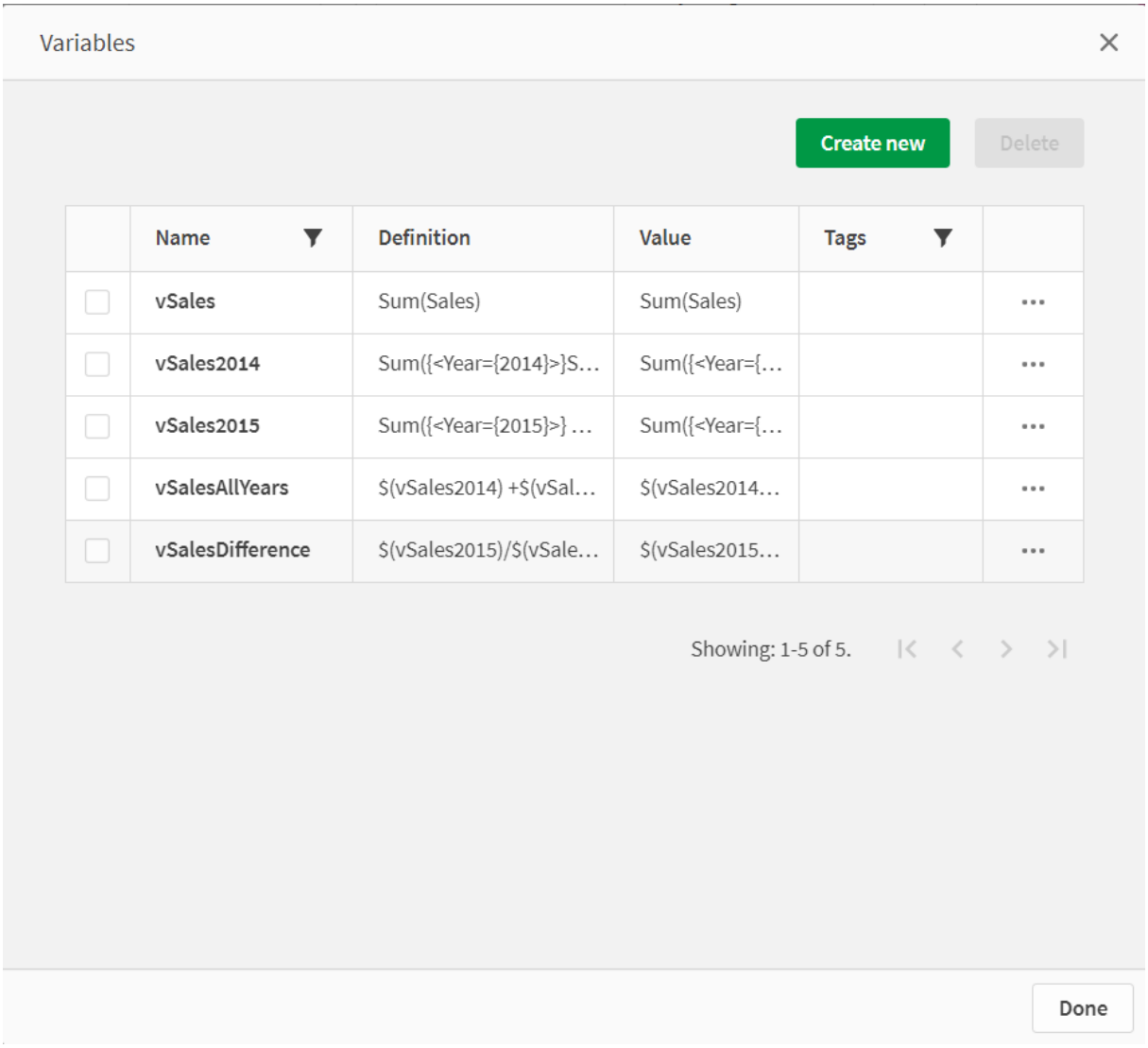


다음 변수를 만듭니다.

만들 변수

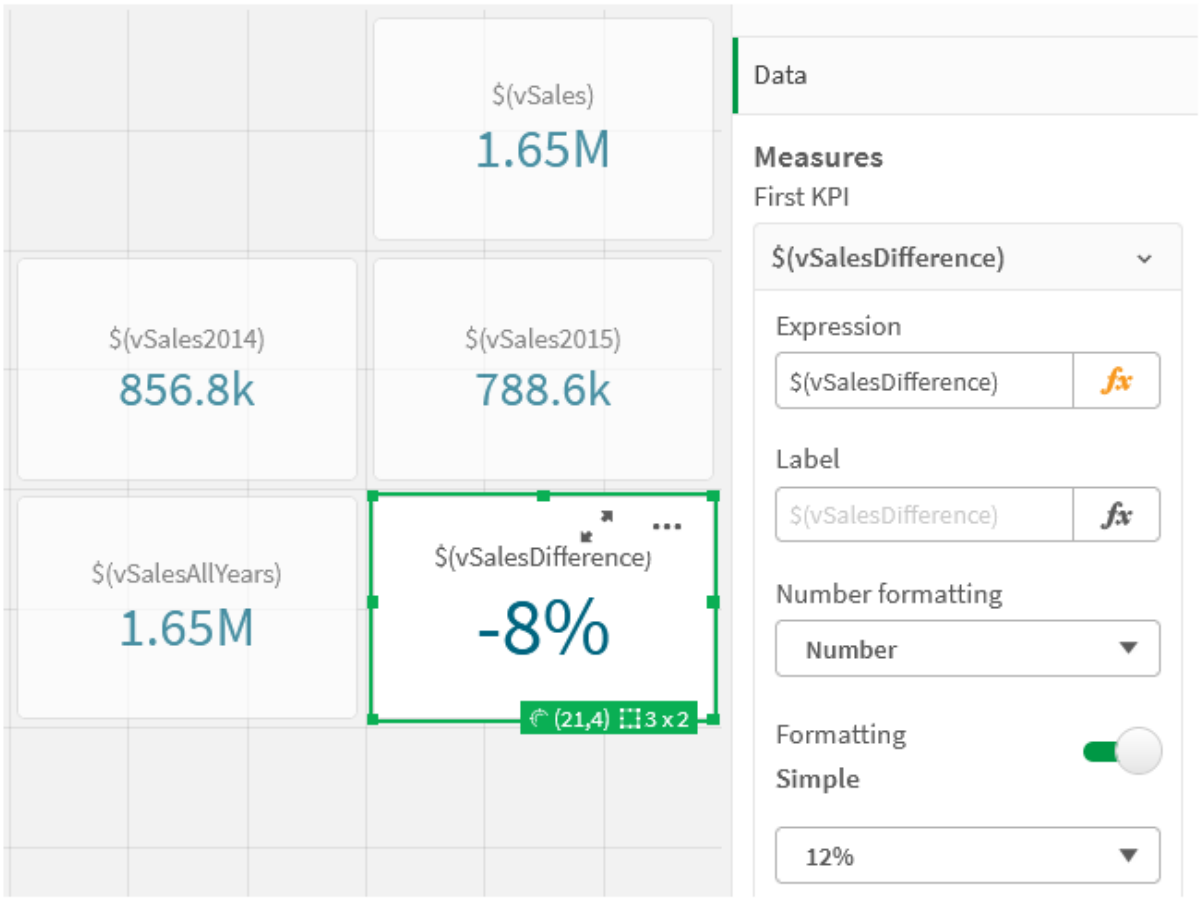
이름	정의
vSales	Sum(Sales)
vSales2014	Sum({<Year={2014}>}Sales)
vSales2015	Sum({<Year={2015}>}Sales)
vSalesAllYears	$(vSales2014) +$(vSales2015)
vSalesDifference	$(vSales2015)/$(vSales2014) - 1

변수



확장을 보려면 KPI 차트를 만듭니다.

달러 기호 확장을 사용하는 KPI



숫자 변수 확장

숫자 변수 확장의 경우 다음과 같은 구문이 사용됩니다.

`(#variablename)`

확장은 항상 변수의 숫자 값의 유효한 소수점 표현을 산출하며, 지수 표기도 가능할 수 있습니다(매우 크거나 작은 수의 경우). `variablename`이 존재하지 않거나 숫자 값을 포함하지 않는다면 NULL이 아닌 0으로 확장됩니다.

예: 숫자 변수 로드 스크립트

예 1: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
Set DecimalSep = ',';  // Set decimal comma as separator for this example.
Let X = 7/2;           // Assign the expression 7/2 to variable X.

MyTable:   // Create an inline table labeled “MyTable”
Load 1 as ID, * inline [
DecimalComma   DecimalPoint
```

```
$(X)   $(#X) ]
(delimiter is '\t');
```

설명

#vVariable 확장은 항상 변수의 숫자 값에 대한 유효한 소수점 표현을 산출합니다. 점이 아닌 쉼표가 소수점 구분 기호로 사용되고 쉼표로 구분된 목록과 충돌할 위험이 있는 경우에 유용합니다.

로드 인라인 테이블에서 이러한 변수를 확장하는 주된 이유는 $(x)의 추가 인용이 필요하지 않기 때문입니다.

출력

Qlik Sense에서 다음 테이블을 만듭니다.

테이블 - 로드 스크립트의 출력

DecimalComma	DecimalPoint
3,5	3.5

예 2: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
// The three Set statements below are required to mimic and initialize
// Format specifiers that are relevant to this particular example

Set ThousandSep=' ';           // Set example thousand separator
Set DecimalSep=',';            // Set example decimal separator
Set TimestampFormat='YYYY-MM-DD hh:mm:ss';      // Set example date format
Let vRaw = today()-1/1440;              // Timestamp minus one minute
Let vFormat = timestamp($(#vRaw));      // Formatted as timestamp

// Create MyTable as an inline table to expand variables as field values

MyTable:
Load * inline [
DecimalComma   DecimalPoint    FormattedNumber
$(vRaw)        $(#vRaw)        $(vFormat) ] (delimiter is '\t');
```

설명

#vVariable 확장은 항상 변수의 숫자 값에 대한 유효한 소수점 표현을 산출합니다. 점이 아닌 쉼표가 소수점 구분 기호로 사용되고 쉼표로 구분된 목록과 충돌할 위험이 있는 경우에 유용합니다. 또한 올바른 소수점 구분 기호 없이 변수를 확장할 때 소수 부분이 잘리기 때문에 숫자 정밀도에 영향을 미친다는 점에 유의해야 합니다.

로드 인라인 테이블에서 이러한 변수를 확장하는 주된 이유는 $(x)의 추가 인용이 필요하지 않기 때문입니다.

출력

Qlik Sense에서 다음 테이블을 만듭니다.

테이블 - 로드 스크립트의 출력

DecimalComma	DecimalPoint	FormattedNumber
44 470,00	44469.999305556	2021-09-18 23:59:00

예 3: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
// The three Set statements below are required to mimic and initialize
// format specifiers that are relevant to this particular example

Set ThousandSep=' ';           // Set example thousand separator
Set DecimalSep=',';            // Set example decimal separator
Set TimestampFormat='YYYY-MM-DD hh:mm:ss';     // Set example date format

// Assign a numerical value and a valid format specifier to vStart

Let vStart = timestamp#('2021-03-23 12:34:56','$(TimestampFormat)');

// Calculate timestamp (vStart + 3 hours) with valid decimal separator: "."

Let vStop = timestamp($(#vStart)+1/8,'YYYY-MM-DD hh:mm:ss');

// Create MyTable as an inline table to expand variables as field values

MyTable:
Load * inline [
StartTime      StopTime
$(vStart)      $(vStop) ] (delimiter is '\t');

// This is a tab delimited inline table
// Tab delimited tables are useful for avoiding conflicting list separators
```

설명

#vVariable 확장은 항상 변수의 숫자 값에 대한 유효한 소수점 표현을 산출합니다. 점이 아닌 쉼표가 소수점 구분 기호로 사용되고 쉼표로 구분된 목록과 충돌할 위험이 있는 경우에 유용합니다. 또한 올바른 소수점 구분 기호 없이 변수를 확장할 때 소수 부분이 잘리기 때문에 숫자 정밀도에 영향을 미친다는 점에 유의해야 합니다.

로드 인라인 테이블에서 이러한 변수를 확장하는 주된 이유는 $(x)의 추가 인용이 필요하지 않기 때문입니다.

출력

Qlik Sense에서 다음 테이블을 만듭니다.

테이블 - 로드 스크립트의 출력

StartTime	StopTime	
2021-03-23 12:34:56	2021-03-23 15:34:56	

대체 상태를 참조하는 변수 확장

변수는 하나의 값만 가지며 이 값은 모든 대체 상태에서 사용됩니다. 변수를 확장하면 변수가 생성된 위치 및 개체 상태에 관계없이 값도 확장됩니다.

변수가 계산된 변수인 경우, 즉 정의가 등호 기호로 시작하는 경우 변수 정의에서 대체 상태를 지정하지 않는 한 기본 상태에서 계산이 수행됩니다.

예를 들어 `MyState`라는 상태와 `vMyVar`라는 변수가 있는 경우:

```
vMyvar: =only({MyState}MyField)
```

대체 상태 이름에 대한 명시적 참조가 있는 변수 정의 내용에 따라 변수 내용이 평가되는 상태가 결정됩니다.

매개 변수를 사용한 달러 기호 확장

매개 변수를 달러 기호 확장에 사용할 수 있습니다. 이는 변수의 동적 평가를 가능하게 합니다.

확장에 사용하기 위해 만들어진 변수는 형식 매개 변수를 포함해야 합니다. 형식 매개 변수는 평가 입력을 위한 자리 표시자이며 달러 기호로 작성됩니다. `$1`, `$2`, `$3`. 숫자는 매개 변수 번호를 나타냅니다.

변수를 사용할 때 실제 매개 변수는 쉼표로 구분된 목록에 명시되어야 합니다.

예: 달러 기호 확장의 매개 변수

두 숫자 사이의 곱셈을 정의하려면 다음과 같이 작성할 수 있습니다.

```
Set MUL= $1*$2 ;
```

이는 `$1`과 `$2`를 서로 곱해야 함을 정의합니다. 사용되는 경우 이 변수는 `$1` 및 `$2`에 대한 값을 표현식에 삽입하여 달러 기호로 확장해야 합니다.

```
Set X= $( MUL(3,7) ) ;
```

값(`3` 및 `7`)은 확장에서 `$1` 및 `$2`를 대체하는 실제 매개 변수입니다. 확장은 Set 문이 구문 분석되고 실행되기 전에 이루어지며, 이는 스크립트 파서가 다음을 확인하고 있음을 의미합니다.

```
Set X= 3*7 ;
```

결과적으로 `X` 변수에는 문자열 값 `3*7`이 할당됩니다.

`Set` 대신 `Let` 문을 사용하는 경우:

```
Let X= $( MUL(3,7) ) ;
```

파서는 다음을 확인합니다.

```
Let X= 3*7 ;
```

여기에서 평가가 이루어지고 `X`는 숫자 값 `21`이 할당됩니다.

매개 변수 수

공식 매개 변수의 수가 실제 매개 변수의 수를 초과하는 경우 실제 매개 변수에 대응하는 공식 매개 변수만 확장됩니다. 실제 매개 변수의 수가 공식 매개 변수의 수를 초과하는 경우 여분의 실제 매개 변수는 무시됩니다.

예: 형식 매개 변수 및 실제 매개 변수

```
Set MUL= '$1*$2' ;

Set X= $(MUL) ;          // returns $1*$2 in X

Set X= $(MUL(10)) ;      // returns 10*$2 in X

Let X= $(MUL(5,7,8)) ;  // returns 35 in X
```

$0 매개 변수

매개 변수 $0은 호출에서 실제로 전달한 매개 변수의 수를 반환합니다.

예: 오류 처리를 알리는 방법

```
Set MUL= If($0=2, $1*$2, 'Error') ;
```

표현식을 사용한 달러 기호 확장

표현식을 달러 기호 확장에 사용할 수 있습니다.

괄호 사이의 내용은 등호로 시작해야 합니다.

$(=expression)
표현식이 평가되고 값이 확장에 사용됩니다. 예:

```
// returns a string with the current year
$(=Year(Today()));
// returns the year before the selected one
$(=Only(Year)-1);
```

예: 차트 표현식에서 측정값을 참조하여 측정값 선택

예: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
// Load string “=MinString(Measure)” into variable VSelectMeasure
Let vSelectMeasure = '=MinString(Measure)';
MyTable1: // Create table and load values for Dim and Sales
Load * inline [
Dim, Sales
A, 150
A, 200
B, 240
B, 230
```

```
C, 410
C, 330 ];


MyTable2: // Create table and load aggregations as field values for Measure
Load * Inline [
Measure
avg(Sales)
sum(Sales)
count(distinct Dim)];
```

설명

스크립트를 사용하면 차트 표현식에서 측정값을 참조하여 측정값을 선택할 수 있습니다. 차트 표현식에는 변수 확장 `$(=MinString(Measure))`가 포함됩니다. `MinString()`은 표현식에서 문자열 값을 찾고 알파벳순으로 정렬된 첫 번째 텍스트 값을 반환합니다(이 경우 `avg(Sales)`). 이를 통해 개체의 속성에 액세스하고 조작하지 않고도 시각화에 사용할 표현식을 대화식으로 결정(및 선택)할 수 있습니다.

출력

다음을 사용하여 Qlik Sense에서 다음 테이블을 만듭니다.

- 차원: `Dim`
- 측정값:
    - `='$(vSelectMeasure)'`
    - `=$(=MinString(Measure))`
    - `Avg(Sales)`

테이블 - 표현식을 사용한 달러 기호 확장

Dim	='$(vSelectMeasure)'	=$(=MinString(Measure))	Avg(Sales)
합계	**-**	**260**	**260**
A	avg(Sales)	175	175
B	avg(Sales)	235	235
C	avg(Sales)	370	370

예: 두 차원 간의 할당량 계산

예: 로드 스크립트

로드 스크립트

데이터 로드 편집기에서 다음 데이터를 인라인 로드로 로드합니다.

```
Let vDivision = '=MinString(Numerator) / MinString(Denominator)';
Load recno() as Numerator autogenerate 100;
Load recno() as Denominator autogenerate 100;
```

설명

이 스크립트는 각각 100개의 값을 포함하는 필드 쌍을 생성하며, 이 필드는 둘 사이의 할당량 계산에서 분자 및 분모로 사용됩니다. 차트 측정값에는 변수 확장 `='$( vDivision )'`이 포함되어 있어 계산할 할당량을 대화식으로 결정(및 선택)할 수 있습니다.

출력

다음을 사용하여 Qlik Sense에서 다음 테이블을 만듭니다.

- 차원:
  - `Numerator`
  - `Denominator`
- 측정값: `='$(vDivision)`

결과 할당량(합계)은 필드에서 선택한 값의 영향을 받습니다. `Numerator`(=5) 및 `Denominator`(=3)이며 두 필드 중 하나에서 새로 선택할 때마다 다시 계산됩니다.

테이블 - 표현식으로 달러 기호 확장

분자	분모	='$(vDivision)'
합계	**-**	**1,6666666666667**
5	3	1,6666666666667

파일 포함

달러 기호 확장을 사용하여 파일을 포함합니다. 구문은 다음과 같습니다.

`$(include=filename)`
위 텍스트는 등호 다음에 지정된 파일의 내용으로 대체됩니다. 이 기능은 스크립트 또는 스크립트의 일부를 텍스트 파일에 저장할 때 유용합니다.

달러 기호 확장 및 대체 상태

달러 기호 확장은 일반적으로 대체 상태에 영향을 받지 않습니다. 하지만 달러 기호 확장에 표현식이 있는 경우 달러 기호 확장이 만들어진 개체에 관련된 상태에서 평가됩니다.

예:

`$(=Sum(Amount))`
계산 결과 개체 상태의 선택 내용을 기반으로 한 합계를 반환합니다.

스크립트에서 인용 부호 사용

스크립트 문에서 다양한 방식으로 인용 부호를 사용할 수 있습니다.

LOAD 문 내부

LOAD 문 내에서는 다음 기호를 인용 부호로 사용해야 합니다.

필드 이름

설명	기호	코드 포인트	예
큰따옴표	" "	34	"string"
대괄호	[ ]	91, 93	[string]
억음 악센트 기호	` `	96	`string`

문자열 리터럴

설명	기호	코드 포인트	예
작은따옴표	' '	39	'string'

SELECT 문 내부

ODBC 드라이버에 의해 해석되는 **SELECT** 문의 경우 용도가 다를 수 있습니다. 보통 필드 이름 및 테이블 이름에 큰따옴표((Alt + 0034))를 사용하고, 리터럴에 작은따옴표((Alt + 0039))를 사용하며, 억음 악센트 기호는 사용하지 않습니다. 하지만 일부 ODBC 드라이버는 억음 악센트 기호를 인용 부호로 허용하며 이를 선호하기도 합니다. 이러한 경우 생성된 **SELECT** 문에는 억음 악센트 인용 부호가 포함됩니다.

Microsoft Access 인용 부호 예

Microsoft Access ODBC Driver 3.4(Microsoft Access 7.0에 포함)에서는 **SELECT** 문을 분석할 때 다음과 같은 인용 부호를 사용할 수 있습니다.

필드 이름 및 테이블 이름:

- [ ]
- " "
- ` `

문자열 리터럴:

- ' '

다른 데이터베이스는 다른 규칙을 사용할 수 있습니다.

LOAD 문 외부

Qlik Sense에서 표현식을 필요로 하는 **LOAD** 문 외부에서 큰따옴표는 필드 참조가 아닌 변수 참조를 의미합니다. 큰따옴표를 사용하는 경우 따옴표로 묶인 문자열은 변수로 해석되며 변수 값이 사용됩니다.

컨텍스트 외부 필드 참조 및 테이블 참조

일부 스크립트 함수는 이미 생성되었거나 **LOAD** 문의 출력에 있는 필드를 참조합니다(예: **Exists()** 및 **Peek ()**). 이러한 필드 참조는 컨텍스트 외부 필드 참조로 불리며 컨텍스트 내 즉, **LOAD** 문의 입력 테이블 내에 있는 필드를 참조하는 소스 필드 참조와 반대되는 의미입니다.

컨텍스트 외부 필드 참조 및 테이블 참조는 리터럴로 간주해야 하며 따라서 작은따옴표를 사용해야 합니다.

이름과 리터럴의 차이점

다음 예를 비교하면 이름과 리터럴의 차이점을 확실히 알 수 있습니다.

```
'Sweden' as Country
```
이 표현식이 **LOAD** 또는 **SELECT** 문에서 필드 목록의 한 부분으로 사용되면 텍스트 문자열 "*Sweden*"이 Qlik Sense의 "*Country*" 필드에 필드 값으로 로드됩니다.

```
"land" as Country
```
이 표현식이 **LOAD** 또는 **SELECT** 문에서 필드 목록의 한 부분으로 사용되면 "*land*"라는 이름의 데이터베이스 필드 또는 테이블 열의 내용이 Qlik Sense의 "*Country*" 필드에 필드 값으로 로드됩니다. 즉, *land*가 필드 참조로 처리됩니다.

숫자와 문자열 리터럴의 차이점

다음 예를 비교하면 숫자와 문자열 리터럴의 차이점을 확실히 알 수 있습니다.

```
'12/31/96'
```
이 문자열이 표현식의 한 부분으로 사용되면 첫 번째 단계로 문자열이 텍스트 문자열 "12/31/96"으로 해석되어, 결과적으로 날짜 서식이 'MM/DD/YY'인 경우에는 날짜로 해석될 수 있습니다. 이런 경우 숫자 표시와 텍스트 표시가 모두 가능한 이중 값으로 저장됩니다.

```
12/31/96
```
이 문자열이 표현식의 한 부분으로 사용되면 문자열이 숫자로 해석되어 12 나누기 31 나누기 96이 됩니다.

문자열에서 인용 부호 사용

문자열에 인용 부호로 사용할 수 있는 문자가 포함되어 있으면 문자열을 인용할 때 문자열의 시작 위치와 종료 위치를 명확하게 나타내는 것이 중요합니다. 문자열이 제대로 인용되지 않으면 스크립트가 실패하거나 데이터를 잘못 로드합니다.

인용 부호가 포함된 문자열을 인용하는 두 가지 방법이 있습니다.

특정 인용 부호를 사용하여 문자열 인용

문자열 내에서 사용되지 않는 인용 부호를 선택하고 전체 문자열을 인용할 때 사용합니다. Qlik Sense에서 이 특정 인용 부호를 사용하여 문자열의 시작 및 끝 위치를 결정합니다.

다음 인용 부호 중 하나를 사용하여 전체 문자열을 인용할 수 있습니다.

- 큰따옴표 ""
- 대괄호 []
- 억음 악센트 기호 ``
- 작은따옴표 ''

```
[테이블 '1 "2"]
```

대괄호가 문자열을 인용할 때 사용됩니다. 문자열이 다음과 같이 로드됩니다. *테이블 '1 "2"*

```
'string `Name1` "Name2'
```

작은따옴표가 문자열을 인용할 때 사용됩니다. 문자열이 다음과 같이 로드됩니다. string *`Name1` "Name2*

이스케이프 문자 사용

이스케이프 문자는 문자열을 인용하는 데 사용되는 인용 부호의 추가 인스턴스입니다. 문자열 내에 나타나는 인용 부호의 모든 인스턴스 옆에 추가해야 합니다. 모든 인용 부호가 문자열 내에서 사용되면 문자열을 인용하는 데 사용한 것과 동일한 인용 부호 유형 옆에 이스케이프 문자를 추가해야 합니다. 이스케이프 문자는 이미 문자열에서 사용 중인 인용 부호를 사용하려는 경우에도 사용할 수 있습니다.

다음 부호만 이스케이프 문자로 사용할 수 있습니다.

- 큰따옴표 ""
- 대괄호 []
- 작은따옴표 ''

```
"Michael said ""It's a beautiful day""."
```

큰따옴표 ""를 사용하여 문자열을 인용하는 경우 문자열 내에 사용된 모든 큰따옴표 옆에 큰따옴표를 추가해야 합니다.

이 문자열은 *Michael said "It's a beautiful day".*로 로드됩니다. 이스케이프 문자 ""를 사용하여 Qlik Sense 데이터 로드 편집기는 어떤 큰따옴표가 문자열의 일부인지, 어떤 인용 부호가 문자열의 끝을 나타내는지 파악합니다. 약어 *It's*에서 사용되는 작은따옴표는 문자열을 인용하는 데 사용되는 부호가 아니기 때문에 이스케이프할 필요가 없습니다.

```
'Michael said: "It''s a beautiful day".'
```

작은따옴표를 사용하여 이 문자열을 인용하는 경우 문자열 내에 사용된 작은따옴표 옆에 작은따옴표를 추가해야 합니다.

이 문자열은 *Michael said "It's a beautiful day".*로 로드됩니다. Michael이 말한 내용을 인용하기 위해 사용되는 큰따옴표 "는 문자열을 인용하는 데 사용되는 부호가 아니기 때문에 이스케이프할 필요가 없습니다.

```
[Michael said [It's a "beautiful day]].]
```

대괄호 []는 다른 두 개의 인용 부호와 다른 역할을 합니다. 대괄호를 이스케이프 문자로 사용하려면 왼쪽 대괄호 [옆이 아닌 오른쪽 대괄호] 옆에만 대괄호를 추가해야 합니다.

이 문자열은 *Michael said [It's a "beautiful day].*로 로드됩니다. 오른쪽 대괄호]만 이스케이프됩니다. 문자열에 사용된 작은따옴표 ' 및 큰따옴표 "는 이스케이프할 필요가 없습니다. 문자열을 인용할 때 사용되지 않기 때문입니다.

데이터의 와일드카드

데이터에 와일드카드를 사용할 수 있습니다. 와일드카드에는 해당 필드의 모든 값으로 해석되는 별표 기호와 해당 필드의 나머지 모든 값으로 해석되는 선택적 기호의 두 종류가 있습니다.

별표 기호

별표 기호는 이 필드의 (나열된) 모든 값, 즉 이 테이블 내의 다른 위치에 나열된 값으로 해석됩니다. 별표 기호가 스크립트의 액세스 섹션에서 로드된 테이블의 시스템 필드(*USERID, PASSWORD, NTNAME* 또는 *SERIAL*) 중 하나에서 사용된 경우 이 필드에 사용 가능한(나열되지 않은) 모든 값으로 해석됩니다.

명시적으로 지정된 경우가 아니면 별표 기호를 사용할 수 없습니다.

OtherSymbol

테이블의 다른 모든 값, 즉 로드한 데이터에서 명시적으로 발견되지 않은 모든 값을 나타내는 방법이 필요한 경우가 많습니다. 이때 사용되는 것이 **OtherSymbol**이라는 특수 변수입니다. **OtherSymbol**을 "다른 모든 값"으로 처리되도록 정의하려면 다음 구문을 사용합니다.

```
SET OTHERSYMBOL=<sym>;
```

LOAD 또는 **SELECT** 문 앞에. <sym>은 임의의 문자열이 될 수 있습니다.

그러면 정의된 기호가 내부 테이블에서 발견될 경우 Qlik Sense가 발견된 필드의 위치에서 아직 로드되지 않은 모든 값으로 기호를 정의하게 됩니다. 따라서 필드에서 **OtherSymbol** 다음에 발견된 값은 무시됩니다.

이 기능을 초기화하려면 다음 구문을 사용합니다.

```
SET OTHERSYMBOL=;
```

Table Customers

CustomerID	Name
1	ABC Inc.
2	XYZ Inc.
3	ACME INC
+	Undefined

Table Orders

CustomerID	OrderID
1	1234
3	1243
5	1248
7	1299

스크립트에서 위의 첫 번째 테이블이 로드되는 시점 앞에 다음 문을 삽입합니다.

```
SET OTHERSYMBOL=+;
```
*OrderID 1299*를 클릭할 때 1, 2 또는 3 등을 제외한 *CustomerID*에 대한 참조는 *Name* 아래에서 *Undefined*가 됩니다.

OtherSymbol*은 테이블 간의 외부 조인을 만드는 데 사용할 수 없습니다.*

NULL 값 처리

데이터베이스 쿼리 및/또는 테이블 간 조인을 통해 특정 필드의 데이터를 생성할 수 없는 경우 일반적으로 NULL 값이 생성됩니다.

개요

Qlik Sense 논리에서는 다음 값을 실제 NULL 값으로 취급합니다.

- ODBC 연결에서 반환된 NULL 값
- 데이터 로드 스크립트에서 테이블 강제 연결의 결과로 생성된 NULL 값
- 데이터 로드 스크립트에서 수행한 조인의 결과로 생성된 NULL 값
- 테이블에 표시할 필드 값 조합의 생성 결과로 생성된 NULL 값

일반적으로 이러한 NULL 값은 연결 및 선택에 사용할 수 없습니다(예외: ***NullAsValue*** *문이 사용된 경우).*

정의에 대한 텍스트 파일은 NULL 값을 포함할 수 없습니다.

ODBC에서 NULL 값 연결/선택

ODBC 데이터 소스에서 NULL 값을 연결 및/또는 선택할 수 있습니다. 이러한 목적의 스크립트 변수가 정의되어 있습니다. 다음 구문을 사용할 수 있습니다.

```
SET NULLDISPLAY=<sym>;
```
기호 <sym>은 데이터 입력의 최저 수준에서 ODBC 데이터 소스의 모든 NULL 값을 대체합니다. <sym>은 임의의 문자열이 될 수 있습니다.

이 기능을 기본 해석으로 초기화하려면 다음 구문을 사용하십시오.

```
SET NULLDISPLAY=;
```

NULLDISPLAY*를 사용하면 ODBC 데이터 소스의 데이터에만 영향을 줍니다.*

Qlik Sense 논리가 ODBC 연결에서 반환된 NULL 값을 빈 문자열로 해석하도록 하려면 모든 **SELECT** 문 앞의 스크립트에 다음을 추가하십시오.

```
SET NULLDISPLAY='';
```

여기서 ''는 실제로 중간에 아무 것도 없는 작은따옴표 2개입니다.

텍스트 파일에서 NULL 값 만들기

텍스트 파일이나 **inline** 절에서 발견될 경우 실제 NULL 값으로 해석될 기호를 정의할 수 있습니다. 다음과 같은 문을 사용합니다.

```
SET NULLINTERPRET=<sym>;
```

기호 <sym>이 NULL로 해석됩니다. <sym>은 임의의 문자열이 될 수 있습니다.

이 기능을 기본 해석으로 초기화하려면 다음 구문을 사용하십시오.

```
SET NULLINTERPRET=;
```

> ***NULLINTERPRET** 사용은 텍스트 파일과 inline 절의 데이터에만 영향을 줍니다.*

표현식에서 NULL 값 전파

NULL 값은 몇 가지 논리적이고 상당히 합리적인 규칙에 따라 표현식에 전파됩니다.

함수

파라메타가 함수에 대해 정의된 범위를 벗어나는 경우 일반적인 규칙에 따라 함수가 NULL을 반환합니다.

예

표현식	**결과**
asin(2)	반환 값: NULL
log(-5)	반환 값: NULL
round(A,0)	반환 값: NULL

위의 규칙에 이어 평가에 필요한 파라메타가 NULL인 경우 일반적으로 이 함수는 NULL을 반환합니다.

예

표현식	**결과**
sin(NULL)	반환 값: NULL
chr(NULL)	반환 값: NULL
if(NULL, A, B)	반환 값: B
if(True, NULL, A)	반환 값: NULL
if(True, A, NULL)	반환 값: A

두 번째 규칙에서 형식을 테스트하는 논리 함수의 경우 예외가 적용됩니다.

예

표현식	**결과**
isnull(NULL)	반환 값: True (-1)
isnum(NULL)	반환 값: False (0)

산술 및 문자열 연산자

이러한 연산자의 항에 NULL이 있으면 NULL이 반환됩니다.

예

표현식	결과
A + NULL	반환 값: NULL
A - NULL	반환 값: NULL
A / NULL	반환 값: NULL
A * NULL	반환 값: NULL
NULL / A	반환 값: NULL
0 / NULL	반환 값: NULL
0 * NULL	반환 값: NULL
A & NULL	반환 값: A

관계형 연산자

관계형 연산자의 항에 NULL이 있으면 특수 규칙이 적용됩니다.

예

표현식	결과
NULL (모든 관계형 연산자) NULL	반환 값: NULL
A <> NULL	반환 값: True (-1)
A < NULL	반환 값: False (0)
A <= NULL	반환 값: False (0)
A = NULL	반환 값: False (0)
A >= NULL	반환 값: False (0)
A > NULL	반환 값: False (0)

4.4 데이터 및 필드에 대한 지침

Qlik Sense로 작업할 때 알아야 할 특정 규칙 및 제한 사항이 있습니다. 예를 들면 데이터 테이블 및 필드의 상한과 Qlik Sense에 로드되는 데이터의 최대 양 등입니다.

로드되는 데이터의 양에 대한 지침

Qlik Sense에 로드할 수 있는 데이터의 양은 기본적으로 컴퓨터의 기본 메모리 용량에 따라 제한됩니다.

데이터 테이블 및 필드의 상한

매우 큰 앱을 만들 때는 Qlik Sense 앱의 경우 하나의 필드에 2,147,483,648개를 초과하는 고유 값을 부여할 수 없음에 유의하십시오.

로드할 수 있는 필드 및 데이터 테이블 수뿐 아니라 테이블 셀과 테이블 행의 수는 RAM에 의해서만 제한됩니다.

로드 스크립트 섹션의 권장 한계

로드 스크립트 섹션별로 사용할 문자의 권장 최대수는 50,000 문자입니다.

숫자 및 시간 형식에 대한 설명

많은 해석 및 서식 지정 함수에서는 서식 코드를 사용하여 숫자와 날짜의 서식을 설정할 수 있습니다. 이 항목에서는 숫자, 날짜, 시간 및 타임스탬프 함수의 형식에 대해 설명합니다. 이러한 형식은 스크립트 및 차트 함수에 모두 적용됩니다.

숫자 서식

특정 자릿수를 지정하려면 각 자리에 "0" 기호를 사용하십시오.

소수점 왼쪽에 가능한 자릿수를 지정하려면 "#" 기호를 사용합니다.

천 단위 구분 기호 또는 소수점 구분 기호의 위치를 표시하려면 해당하는 천 단위 구분 기호 및 소수점 구분 기호를 사용하십시오.

서식 코드는 구분 기호의 위치를 정의하는 데 사용됩니다. 서식 코드에서 구분 기호를 설정할 수는 없습니다. 스크립트에서는 이를 위해 **DecimalSep** 및 **ThousandSep** 변수를 사용합니다.

천 단위 구분 기호를 사용하여 자릿수를 임의의 위치로 그룹화할 수 있습니다. 예를 들어 "0000-0000-0000"(천 단위 구분 기호="-")과 같은 서식 문자열을 사용하여 "0012-4567-8912"와 같은 12자리 부품 번호를 표시할 수 있습니다.

숫자 서식의 예

숫자 서식	**설명**
# ##0	숫자를 천 단위 구분 기호가 있는 정수로 표시합니다. 이 예제에서 " "은 천 단위 구분 기호로 사용됩니다.
###0	숫자를 천 단위 구분 기호가 없는 정수로 표시합니다.
0000	숫자를 최소 네 자리 이상의 정수로 표시합니다. 예를 들어 숫자 123은 0123으로 표시됩니다.
0.000	소수 자릿수 3개를 사용하여 숫자를 표시합니다. 이 예제에서 "."은 소수점 구분 기호로 사용됩니다.

특수 숫자 서식

Qlik Sense는 이진수, 8진수, 16진수를 포함하여 2 ~ 36 사이의 모든 기수로 숫자를 해석하고 서식을 지정할 수 있습니다. 또한 로마체 서식도 처리할 수 있습니다.

특수 숫자 서식

서식	설명
이진수 서식	이진수 서식을 표시하려면 서식 코드가 (bin) 또는 (BIN)으로 시작되어야 합니다.
8진수 서식	8진수 서식을 표시하려면 서식 코드가 (oct) 또는 (OCT)로 시작되어야 합니다.
16진수 서식	16진수 서식을 표시하려면 서식 코드가 (hex) 또는 (HEX)로 시작되어야 합니다. 대문자 버전을 사용하는 경우 서식 지정에 A-F가 사용됩니다(예: 14FA). 대문자가 아닌 버전의 경우 서식 지정에 a-f가 사용됩니다(예: 14fa). 서식 코드의 대문자 표시에 관계없이 두 버전 모두 해석이 가능합니다.
10진수 서식	(dec) 또는 (DEC)를 사용하여 10진수 서식을 표시할 수 있지만 불필요합니다.
임의의 기수 서식	2 ~ 36 사이의 임의 기수로 서식을 표시하려면 서식 코드가 (rxx) 또는 (Rxx)로 시작되어야 하며, 여기서 xx는 사용할 기수를 나타내는 2자리 숫자입니다. 대문자 R을 사용하는 경우 Qlik Sense에서 서식을 지정할 때 10을 초과하는 기수는 대문자로 표시됩니다(예: 14FA). 대문자가 아닌 r을 사용하면 서식 지정에 대문자가 아닌 문자가 사용됩니다(예: 14fa). 서식 코드의 대문자 표시에 관계없이 두 버전 모두 해석이 가능합니다. 참고로 (r02)는 (bin)과 같으며, (R16)은 (HEX)와 같습니다.
로마 숫자 서식	로마 숫자를 표시하려면 서식 코드가 (rom) 또는 (ROM)으로 시작되어야 합니다. 대문자 버전을 사용하는 경우 서식 지정에 대문자가 사용됩니다(예: MMXVI). 대문자가 아닌 버전의 경우 서식 지정에 소문자가 사용됩니다(예: mmxvi). 서식 코드의 대문자 표시에 관계없이 두 버전 모두 해석이 가능합니다. 로마 숫자는 음수의 경우 음의 기호로, 영의 경우 0으로 일반화됩니다. 로마 숫자 서식에서는 소수 자릿수가 무시됩니다.

특수 숫자 서식의 예

예	결과
num(199, '(bin)')	반환 값: 11000111
num(199, '(oct)')	반환 값: 307
num(199, '(hex)')	반환 값: c7
num(199, '(HEX)')	반환 값: C7
num(199, '(r02)')	반환 값: 11000111
num(199, '(r16)')	반환 값: c7
num(199, '(R16)')	반환 값: C7
num(199, '(R36)')	반환 값: 5J
num(199, '(rom)')	반환 값: cxcix

예	결과
num(199, '(ROM)')	반환 값: CXCIX

날짜

다음 기호를 사용하여 날짜의 서식을 지정할 수 있습니다. 임의의 구분 기호를 사용할 수 있습니다.

날짜의 서식을 지정하는 기호

기호	설명
D	일을 표시하려면 각 자리에 "D" 기호를 사용하십시오.
M	월 번호를 표시하려면 "M" 기호를 사용하십시오. 한 자리 또는 두 자리의 경우, "M" 또는 "MM"을 사용하십시오. "MMM"은 운영 체제 또는 스크립트의 재정의 시스템 변수 **MonthNames**에 정의된 문자로 된 짧은 월 이름을 나타냅니다. "MMMM"은 운영 체제 또는 스크립트의 재정의 시스템 변수 **LongMonthNames**에 정의된 문자로 된 긴 월 이름을 나타냅니다.
Y	연도를 표시하려면 각 자리에 "Y" 기호를 사용하십시오.
W	요일을 표시하려면 "W" 기호를 사용하십시오. "W"는 한 자리의 요일 숫자(예: 월요일=0)를 반환합니다. "WW"는 두 자리의 숫자를 반환합니다(예: 수요일=02). "WWW"를 선택하면 운영 체제 또는 스크립트의 재정의 시스템 변수 **DayNames**에 정의된 요일 이름의 짧은 버전(예: Mon)이 표시됩니다. "WWWW"를 선택하면 운영 체제 또는 스크립트의 재정의 시스템 변수 **LongDayNames**에 정의된 요일 이름의 긴 버전(예: Monday)이 표시됩니다.

(날짜 예로 2013년 3월 31일 사용)

날짜 서식의 예

예	결과
YY-MM-DD	날짜를 13-03-31로 표시합니다.
YYYY-MM-DD	날짜를 2013-03-31로 표시합니다.
YYYY-MMM-DD	날짜를 2013-Mar-31로 표시합니다.
DD MMMM YYYY	날짜를 31 March 2013으로 표시합니다.
M/D/YY	날짜를 3/31/13으로 표시합니다.
W YY-MM-DD	날짜를 6 13-03-31로 표시합니다.

예	결과
WWW YY-MM-DD	날짜를 Sat 13-03-31로 표시합니다.
WWWW YY-MM-DD	날짜를 Saturday 13-03-31로 표시합니다.

시간

다음 기호를 사용하여 시간의 서식을 지정할 수 있습니다. 임의의 구분 기호를 사용할 수 있습니다.

시간의 서식을 지정하는 기호

기호	설명
h	시간을 표시하려면 각 자리에 "h" 기호를 사용하십시오.
m	분을 표시하려면 각 자리에 "m" 기호를 사용하십시오.
s	초를 표시하려면 각 자리에 "s" 기호를 사용하십시오.
f	초의 분위수를 표시하려면 각 자리에 "f" 기호를 사용하십시오.
tt	시간을 오전/오후 형식으로 표시하려면 시간 뒤에 "tt" 기호를 사용하십시오.

(시간 예로 18시 30분 사용)

시간 서식의 예

예	결과
hh:mm	시간을 18:30으로 표시합니다.
hh.mm.ss.ff	시간을 18.30.00.00으로 표시합니다.
hh:mm:tt	시간을 06:30:pm으로 표시합니다.

타임스탬프

위의 날짜 및 시간 표기법과 동일한 표기법이 타임스탬프에 사용됩니다.

(타임스탬프 예로 2013년 3월 31일 18시 30분 사용):

스탬프 서식의 예

예	결과
YY-MM-DD hh:mm	타임스탬프를 13-03-31 18:30으로 표시합니다.
M/D/Y hh.mm.ss.ffff	타임스탬프를 3/31/13 18.30.00.0000으로 표시합니다.

4.5 QVD 파일 작업

QVD (QlikView Data) 파일은 Qlik Sense에서 내보낸 데이터의 테이블을 포함한 파일입니다. QVD는 네이티브 Qlik 형식이며 Qlik Sense 또는 QlikView에서만 읽고 쓸 수 있습니다. 파일 형식은 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다. 파일 형식

은 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다. QVD 파일에서 데이터를 읽는 속도는 일반적으로 다른 데이터 소스에서 데이터를 읽는 것보다 10-100배 정도 빠릅니다.

QVD 파일은 표준(빠름) 모드와 최적화(매우 빠름) 모드에서 읽을 수 있습니다. 모드 선택은 스크립트 엔진에서 자동으로 결정합니다.

최적화된 로드에는 몇 가지 제한 사항이 있습니다. 필드의 이름을 변경할 수는 있지만 아래 언급된 작업 모두 최적화된 로드를 비활성화하고 표준 로드를 발생시킵니다.

- 로드된 필드의 모든 변환
- **where** 절을 사용하여 Qlik Sense가 레코드를 압축 해제하게 함
- 로드된 필드에서 **맵** 사용

QVD 파일의 용도

QVD 파일은 여러 목적으로 사용될 수 있으며 주어진 상황에서 둘 이상이 적용될 수 있습니다. 적어도 다음의 네 가지 중요 용도는 쉽게 확인할 수 있습니다.

- 로드 속도 향상

  QVD 파일에서 입력 데이터의 불변하는 또는 느리게 변하는 블록을 버퍼링하면 큰 데이터 셋에 대한 스크립트 실행 속도가 상당히 빨라집니다.

- 데이터베이스 서버의 부하 감소

  외부 데이터 소스에서 가져오는 데이터의 양이 상당히 감소할 수 있습니다. 이로 인해 외부 데이터베이스의 워크로드와 네트워크 트래픽이 감소하게 됩니다. 게다가 여러 스크립트에서 동일한 데이터를 공유할 때 해당 데이터를 소스 데이터베이스에서 QVD 파일로 한 번만 로드하면 됩니다. 다른 앱에서도 이 QVD 파일을 통해 동일한 데이터를 사용할 수 있습니다.

- 여러 앱의 데이터 통합

  binary 스크립트 문을 사용하면 단일 앱에서 다른 앱으로 데이터를 로드할 수 있습니다. 하지만 QVD 파일을 사용하면 스크립트에서 여러 앱의 데이터를 결합할 수 있습니다. 이렇게 하면 앱에서 여러 비즈니스 단위의 유사한 데이터를 통합하는 등의 작업이 가능해집니다.

- 증분

  많은 경우, QVD 기능을 사용하면 크기가 커지고 있는 데이터베이스에서 새로운 레코드만 로드하여 증분 로드를 수행할 수 있습니다.

QVD 파일 만들기

QVD 파일은 두 가지 방법으로 만들 수 있습니다.

- 스크립트에서 **store** 명령을 사용하여 명시적 생성 및 명명. 이전에 읽은 테이블 또는 그 일부를 선택한 위치의 명시적으로 명명한 파일로 내보내도록 스크립트에 명시합니다.
- 스크립트에서 자동 생성 및 유지 관리. **LOAD** 또는 **SELECT** 문 앞에 **buffer** 접두사를 추가하면 Qlik Sense가 특정 상황에서 데이터를 다시 로드할 때 원본 데이터 소스 대신 사용할 수 있는 QVD 파일이 자동으로 만들어집니다.

위의 방법으로 생성된 QVD 파일들은 읽기 속도에 차이가 없습니다.

QVD 파일에서 데이터 읽기

다음과 같은 방법을 통해 QVD 파일을 읽거나 액세스할 수 있습니다.

- 명시적 데이터 소스로 QVD 파일 로드. QVD 파일은 다른 형식의 텍스트 파일(csv, fix, dif, biff 등)과 마찬가지로 스크립트에서 **LOAD** 문을 통해 참조할 수 있습니다.
  예:
  - `LOAD * from xyz.qvd (qvd)`
  - `LOAD Name, RegNo from xyz.qvd (qvd)`
  - `LOAD Name as a, RegNo as b from xyz.qvd (qvd)`
- 버퍼링된 QVD 파일 자동 로드. **LOAD** 또는 **SELECT** 문에 **buffer** 접두사를 사용하면 읽기를 위한 명시적 문을 사용할 필요가 없습니다. 원래 **LOAD** 또는 **SELECT** 문을 사용하여 데이터를 얻는 것과는 대조적으로 Qlik Sense에서는 QVD 파일의 데이터 사용 범위를 결정합니다.
- 스크립트에서 QVD 파일 액세스. 여러 스크립트 함수(모두 **qvd**로 시작)를 QVD 파일의 XML 헤더에 있는 데이터에서 다양한 정보를 검색하는 데 사용할 수 있습니다.

QVD 형식

QVD 파일은 정확하게 하나의 데이터 테이블을 포함하며 다음 세 가지 부분으로 구성됩니다.

- 헤더.

  > *QVD 파일이 QlikView에서 생성된 경우 헤더는 테이블의 필드, 후속 정보 레이아웃 및 기타 메타데이터를 설명하는 잘 구성된 XML 헤더(UTF-8 문자 집합 사용)입니다.*

- 바이트 형식의 기호 테이블
- 비트 형식의 실제 테이블 데이터

4.6 에서 분석 연결 구성Qlik Sense Desktop

분석 연결을 사용하면 외부 분석을 Business Discovery와 통합할 수 있습니다. 분석 연결은 외부 계산 엔진을 호출하여 로드 스크립트 및 차트에서 사용할 수 있는 표현식을 확장합니다. 이렇게 하면 계산 엔진이 서버 측 확장(SSE) 역할을 합니다. 예를 들어 R에 대한 분석 연결을 만들고 데이터를 로드할 때 통계식을 사용할 수 있습니다.

Qlik Sense Desktop의 경우 *Settings.ini* 파일에서 구성해야 합니다.

다음과 같이 하십시오.

1. *Settings.ini* 파일을 엽니다.
   Qlik Sense Desktop의 경우 *C:/Users/<사용자 ID>/Documents/Qlik/Sense/* 또는 *C:/Users/AppData/Local/Programs/Qlik/Sense/Engine*에 있습니다.
   Qlik Sense의 경우 *C:/ProgramData/Qlik/Sense/Engine/*에 있습니다.
2. 다음 구성을 추가합니다(끝에 빈 줄이 있습니다).
   ```
   [Settings 7]
   SSEPlugin=<PluginConfig>[;<PluginConfig>...]
   ```

여기서 <PluginConfig>는 다음을 포함한 쉼표로 구분된 구성 요소 목록입니다.

<EngineName>,<Address>[,<PathToCertFile>,<RequestTimeout>,<ReconnectTimeout>]

새 연결을 추가하거나 기존 연결을 변경한 후 변경 사항을 적용하려면 Qlik Sense Desktop을 다시 시작해야 합니다.

Qlik Sense를 시작하기 전에 서버 측 확장(SSE) 플러그인 서버가 실행 중이어야 합니다. 그렇지 않으면 연결이 설정되지 않습니다.

Qlik 오픈 소스 SSE 리포지토리

다음 두 Qlik SSE 리포지토리는 오픈 소스입니다.

- *https://github.com/qlik-oss/server-side-extension*
  SSE 프로토콜, 일반 설명서, Python 및 C++로 작성된 예제가 포함되어 있습니다.
- *https://github.com/qlik-oss/sse-r-plugin*
  C#으로 작성된 R 플러그인이 포함되어 있지만 소스 코드만 있습니다. 사용하기 전에 플러그인을 만들어야 합니다.

요소 설명

<EngineName>: 플러그인 함수를 사용하여 앱의 표현식 내에서 사용되는 플러그인에 대한 매핑/별칭입니다. 예를 들어 Python 플러그인의 경우 *SSEPython*입니다.

<Address>: 콜론으로 구분된 두 요소의 목록입니다.

- <Host>: 플러그인의 DNS 이름(또는 IP 주소)입니다.
- <Port>: 플러그인이 수신하는 포트입니다. 일반적으로 50051입니다.

<PathToCertFile>: 플러그인과의 보안 통신에 필요한 클라이언트 인증이 포함된 폴더에 대한 파일 시스템 경로입니다. 선택 사항. 생략하면 보안되지 않은 통신이 호출됩니다. 이 경로는 단지 인증서가 있는 폴더를 가리킵니다. 인증서가 실제로 해당 폴더에 복사되어 있는지 확인해야 합니다. 세 인증서 파일의 이름은 다음과 같아야 합니다. *root_cert.pem*, *sse_client_cert.pem*, *sse_client_key.pem*. 상호 인증(서버 및 클라이언트 인증)만 허용됩니다.

<RequestTimeout>: 정수(초)입니다. 선택 사항입니다. 기본값은 0(무제한)입니다. 메시지 지속 시간에 대한 제한 시간입니다.

<ReconnectTimeout>: 정수(초)입니다. 선택 사항입니다. 기본값은 20(초)입니다. 플러그인에 대한 연결이 끊어진 후 클라이언트가 플러그인에 다시 연결을 시도하기 전의 시간입니다.

- 한 개의 SSE 플러그인 서버가 정의된 예: `SSEPlugin=SSEPython,localhost:50051`
- 두 개의 SSE 플러그인 서버가 정의된 예:
  `SSEPlugin=SSEPython,localhost:50051;R,localhost:50053`
- 한 개의 SSE 플러그인 서버가 인증서 경로 없이 정의되었지만 제한 시간이 설정된 예:
  `SSEPlugin=SSEPython,localhost:50051,,0,20`

5 Section Access를 사용하여 데이터 보안 관리

Section Access를 사용하여 응용 프로그램의 보안을 제어합니다. 기본적으로 보안 테이블을 추가하여 누가 무엇을 보게 되는지 정의하는 데이터 로드 스크립트의 일부입니다. Qlik Sense는 사용자가 응용 프로그램을 열 때 이 정보를 사용하여 데이터를 적절한 범위로 줄입니다. 즉, 앱의 일부 데이터는 ID를 기반으로 사용자에게 숨겨집니다. Section Access는 앱의 데이터와 긴밀하게 통합되며, 이에 따라 액세스를 제어합니다. 이러한 형태의 동적 데이터 축소는 테이블 행, 열 또는 둘의 조합을 대상으로 할 수 있습니다. 자세한 내용은 Qlik의 신뢰 및 보안을 참조하십시오.

5.1 로드 스크립트의 섹션

데이터 액세스 제어는 일반적으로 데이터를 로드하는 것과 동일한 방식으로 로드한 하나 이상의 보안 테이블을 통해 관리됩니다. 따라서 이러한 테이블을 표준 데이터베이스 또는 스프레드시트에 저장할 수 있습니다. 보안 테이블을 관리하는 스크립트 문은 스크립트에서 `Section Access` 문으로 시작하는 인증 섹션 내에 지정됩니다.
스크립트에 인증 섹션이 정의되어 있는 경우 앱 데이터를 로드하는 스크립트 부분은 `Section Application` 문으로 시작하는 다른 섹션에 지정해야 합니다.

```
Section Access;
Load * INLINE [
    ACCESS,  USERID,                REDUCTION
    USER,    AD_DOMAIN\ADMIN,       *
    USER,    AD_DOMAIN\A,           1
    USER,    AD_DOMAIN\B,           2
    USER,    AD_DOMAIN\C,           3
    ADMIN,   INTERNAL\SA_SCHEDULER,
];
Section Application;
T1:
Load *,
NUM AS REDUCTION;
LOAD
Chr(RecNo()+ord('A')-1) AS ALPHA,
RecNo() AS NUM
AUTOGENERATE 3;
```

로드 스크립트를 변경한 후 변경 사항을 적용하려면 항상 데이터를 다시 로드해야 합니다.

Section Access 시스템 필드

액세스 수준은 Section Access 내에서 로드되는 하나 이상의 보안 테이블에서 사용자에게 할당됩니다. 이러한 테이블에는 최소한 두 개의 시스템 필드가 포함되어야 합니다. 액세스 수준을 정의하는 필드인 ACCESS 및 USERID 또는 USER.EMAIL . 다른 선택적 시스템 필드는 사용 사례에 따라 추가될 수 있습니다. 전체 Section Access 시스템 필드 집합에 대한 설명이 아래에 나와 있습니다.

ACCESS

해당 사용자에게 있어야 하는 액세스 권한을 정의합니다.

지정된 사용자에 대해 Qlik Sense 앱에 대한 액세스 권한을 부여할 수 있습니다. 보안 테이블에서 사용자에게 ADMIN 또는 USER 액세스 수준을 할당할 수 있습니다. ADMIN 권한이 있는 사용자는 보안 테이블로 제한되지 않는 한 앱의 모든 데이터에 액세스할 수 있습니다. USER 권한이 있는 사용자는 보안 테이블에 정의된 데이터에만 액세스할 수 있습니다. 유효한 액세스 수준이 할당되지 않은 사용자는 앱을 열 수 없습니다.

Section Access가 다시 로드 시나리오에서 사용되고 스케줄러 서비스 사용자인 INTERNAL\SA_SCHEDULER에게는 다시 로드를 수행하는 ADMIN 액세스 권한이 필요합니다. 예:

```
Section Access;
LOAD * inline [
    ACCESS,  USERID
    ADMIN,   INTERNAL\SA_SCHEDULER
];
```

INTERNAL\SA_SCHEDULER 계정을 사용하지 않으려면 *가장을 사용하여 데이터 다시 로드 (page 166)*에서 대체 방법을 참조하십시오.

템플릿 앱의 On-demand 앱 생성(ODAG) 시나리오에서 Section Access가 사용되는 경우 INTERNAL\SA_API 사용자는 Section Access 테이블에서 ADMIN으로 포함되어야 합니다. 예:

```
Section Access;
LOAD * inline [
    ACCESS,  USERID
    ADMIN,   INTERNAL\SA_API
];
```

USERID

Qlik Sense 도메인 이름 및 사용자 이름에 해당하는 문자열이 포함되어 있습니다. Qlik Sense는 프록시 서비스에서 로그인 정보를 가져와서 이 필드의 값과 비교합니다.

와일드카드 문자(*)는 보안 테이블에 지정된 추가 조건에 따라 모든 사용자로 해석됩니다. 예를 들어 다음 보안 테이블에서 Qlik Sense Tenant Admins에 있는 사용자는 나열된 모든 REDUCTION 값을 볼 수 있습니다.

```
Section Access;
LOAD * INLINE [
    ACCESS,  USERID,                  GROUP,                      REDUCTION
    ADMIN,   *,                       Qlik Sense Tenant Admins,   *
    USER,    QLIK-POC\SOMEOTHERUSER1, *,                          1
    USER,    QLIK-POC\SOMEOTHERUSER2, *,                          2
    ...
];
```

USERID 및 NTNAME 둘 다 동일한 인증 정보를 사용하므로 보안 테이블의 동일한 행에서 둘 다 확인할 필요가 없습니다. 두 필드의 차이는 NTNAME이 그룹도 확인한다는 점입니다.

NTNAME

NTNAME은 레거시 QlikView 필드이며 QlikView에서 동일한 보안 테이블을 사용하지 않는 경우 USERID를 사용하는 것이 좋습니다.

Windows NT 도메인 사용자 이름 또는 그룹 이름에 해당하는 문자열을 포함해야 하는 필드입니다. 다른 인증 시스템을 사용하는 경우 인증된 사용자의 이름을 포함해야 합니다. Qlik Sense는 OS에서 로그인 정보를 가져와서 이 필드의 값과 비교합니다.

GROUP

Qlik Sense의 그룹에 해당하는 문자열이 포함되어 있습니다. Qlik Sense는 이 그룹에 대해 프록시 서비스에서 제공되는 사용자를 해결합니다.

SERIAL

SERIAL은 레거시 QlikView 필드이며 Qlik Sense만 사용하는 경우에는 사용되지 않습니다.

플랫폼에 해당하는 문자열을 포함합니다. 필드에 문자열 'QLIKSENSE' 또는 와일드카드 '*'가 포함된 경우 보안 테이블의 다른 필드에 따라 액세스가 허용될 수 있습니다.

필드 SERIAL에 라이센스 번호가 포함된 경우 Section Access 행은 문서에 대한 액세스를 거부합니다. 이 설정은 QlikView에서만 유효합니다.

OMIT

이 특정 사용자에 대해 생략할 필드의 이름이 포함되어 있습니다. 와일드카드가 사용될 수 있으며 필드는 비워둘 수 있습니다.

키 필드에 OMIT를 적용하지 않는 것이 좋습니다. 생략된 키 필드는 데이터 모델 뷰어에 표시될 수 있지만 사용자에게 혼란을 줄 수 있어 콘텐츠는 사용할 수 없습니다. 또한 시각화에서 사용되는 필드에 OMIT를 적용하면 생략된 필드에 액세스 권한이 없는 사용자에게는 시각화가 불완전할 수 있습니다.

5.2 앱에 대한 사용자 액세스 관리

가장 간단한 형태의 Section Access는 특정 사용자가 앱에 액세스하지 못하도록 제한하는 데 사용할 수 있습니다. 사용자는 제외를 통해 앱에 대한 액세스가 거부됩니다. 다시 말해, 특정 사용자 ID 가 보안 테이블에 나열되지 않으면 앱에 액세스할 수 없습니다. 이 규칙의 유일한 예외는 보안 테이블의 행 중 하나에 있는 USERID 필드에 와일드카드(*)가 지정된 경우입니다. 이 경우 와일드카드는 인증된 모든 사용자가 앱에 액세스할 수 있음을 의미합니다. 다음은 사용자 ID 목록이 있는 보안 테이블의 예입니다.

```
Section Access;
LOAD * inline [
    ACCESS,  USERID
```

```
    ADMIN,    AD_DOMAIN\ADMIN
    USER,     AD_DOMAIN\A
    USER,     AD_DOMAIN\B
];
Section Application;
```

5.3 앱의 특정 데이터에 대한 사용자 액세스 관리

동적 데이터 축소는 사용자가 앱 자체에 액세스할 수 있는 권한이 부여된 후 Qlik Sense 앱 내 데이터 테이블의 행과 열에 대한 액세스를 제한합니다.

행 수준 데이터에 대한 액세스 관리

로드 스크립트의 액세스 섹션에 있는 보안 테이블에 데이터 축소 열을 추가하여 행 수준 데이터에 대한 액세스를 제한합니다. 특정 레코드(행)는 Section Access 데이터를 실제 데이터와 연결하여 숨길 수 있습니다. 표시하거나 제외할 데이터의 선택은 스크립트의 Section Access 및 섹션 응용 프로그램 부분에 공통 이름을 가진 하나 또는 여러 개의 축소 필드를 포함하여 제어됩니다. 사용자 로그인 후 Qlik Sense는 액세스 섹션의 축소 필드 선택을 정확하게 동일한 필드 이름(필드 이름은 대문자로 작성되어야 함)으로 응용 프로그램 섹션의 필드에 일치시킵니다. 선택이 완료되면 Qlik Sense는 이러한 선택에 의해 제외된 모든 데이터를 해당 사용자에게 영구적으로 숨깁니다. 와일드카드(*)가 데이터 축소 열의 필드 값으로 사용되는 경우 사용자가 보안 테이블에서 선택한 모든 축소 필드와 연관된 레코드에 액세스할 수 있도록 허용하는 것으로 해석됩니다.

데이터 축소 열의 와일드카드 문자(*)는 보안 테이블의 모든 값만 참조합니다. 보안 테이블의 축소 열에서 사용할 수 없는 섹션 응용 프로그램에 값이 있는 경우에는 값이 줄어듭니다.

액세스 섹션에서는 모든 필드 이름 및 필드 값이 기본적으로 대문자로 변환되므로 위에서 설명한 전송에 사용된 모든 필드 이름과 이 필드의 모든 필드 값은 대문자여야 합니다.

기본적으로 Qlik Management Console 작업에서 스크립트 다시 로드를 활성화하려면 ADMIN 액세스 권한을 가진 INTERNAL\SA_SCHEDULER 계정 사용자가 필요합니다. INTERNAL\SA_SCHEDULER 계정을 사용하지 않으려면 가장을 사용하여 데이터 다시 로드 (page 166)에서 대체 방법을 참조하십시오.

사용자 ID를 기반으로 행 수준 데이터 축소

```
Section Access;
Authorization:
LOAD * inline [
    ACCESS,  USERID,                 REDUCTION
    ADMIN,   AD_DOMAIN\ADMIN,        *
    USER,    AD_DOMAIN\A,            1
    USER,    AD_DOMAIN\B,            2
    USER,    AD_DOMAIN\C,            *
    ADMIN,   INTERNAL\SA_SCHEDULER,  *
];
Section Application;
T1:
```

```
LOAD *,
NUM AS REDUCTION;
LOAD
RecNo() AS NUM
AUTOGENERATE 3;
```

이제 필드 REDUCTION(대문자)이 Section Access 및 섹션 응용 프로그램 모두에 있습니다(모든 필드 값도 대문자임). 두 필드는 일반적으로 다르고 별개이지만 Section Access를 사용할 경우 서로 연결되어 사용자에게 표시되는 레코드 수가 축소됩니다.

결과는 다음과 같습니다.

- 사용자 ADMIN은 모든 필드를 볼 수 있으며 REDUCTION = 1 또는 REDUCTION =2일 때 다른 사용자가 볼 수 있는 레코드만 볼 수 있습니다.
- 사용자 A은 모든 필드를 볼 수 있지만 REDUCTION=1에 연관된 레코드만 해당됩니다.
- 사용자 B은 모든 필드를 볼 수 있지만 REDUCTION=2에 연관된 레코드만 해당됩니다.
- 사용자 C은 모든 필드를 볼 수 있으며 REDUCTION = 1 또는 REDUCTION =2일 때 다른 사용자가 볼 수 있는 레코드만 볼 수 있습니다.

열 수준 데이터에 대한 액세스 관리

Section Access 스크립트의 보안 테이블에 OMIT 시스템 필드를 추가하여 열 수준 데이터에 대한 액세스를 제한합니다. 다음 예는 이미 행 데이터 축소가 있는 이전 예를 기반으로 합니다.

사용자 ID를 기반으로 열 데이터 축소

```
Section Access;
LOAD * inline [
    ACCESS,  USERID,                 REDUCTION,  OMIT
    ADMIN,   AD_DOMAIN\ADMIN,        *,
    USER,    AD_DOMAIN\A,            1,
    USER,    AD_DOMAIN\B,            2,          NUM
    USER,    AD_DOMAIN\C,            3,          ALPHA
    ADMIN,   INTERNAL\SA_SCHEDULER,  *,
];
Section Application;
T1:
LOAD *,
NUM AS REDUCTION;
LOAD
Chr( RecNo()+ord('A')-1) AS ALPHA,
RecNo() AS NUM
AUTOGENERATE 3;
```

Section Access에서 OMIT 필드는 사용자에게 숨겨야 하는 필드를 정의합니다.

결과는 다음과 같습니다.

- 사용자 ADMIN은 모든 필드와 더불어, REDUCTION이 1, 2 또는 3일 때 다른 사용자가 이 예에서 볼 수 있는 레코드를 볼 수 있습니다.
- 사용자 A은 모든 필드를 볼 수 있지만 REDUCTION=1에 연관된 레코드만 해당됩니다.
- 사용자 B은 NUM를 제외한 모든 필드와 REDUCTION=2과 연관된 레코드만 볼 수 있습니다.
- 사용자 C은 ALPHA를 제외한 모든 필드와 REDUCTION=3과 연관된 레코드만 볼 수 있습니다.

일부 시각화는 최소 데이터 요구 사항을 충족해야 렌더링할 수 있습니다. 결과적으로 열 수준 필드가 사용자의 데이터 보기에서 생략되면 "불완전한 시각화"가 표시될 수 있습니다.

사용자 그룹에 대한 액세스 관리

Section Access는 그룹 구성원 자격을 통해 사용자에게 표시되는 데이터 범위를 제한하는 옵션을 제공합니다. 사용자 그룹을 사용하여 데이터를 제한하려면 액세스 섹션의 보안 테이블에 GROUP 필드 이름을 추가하고 GROUP 필드에 대한 값을 정의합니다.

사용자 그룹을 기반으로 데이터 축소

```
Section Access;
LOAD * inline [
    ACCESS,  USERID,                 GROUP,   REDUCTION,  OMIT
    USER,    *,                      ADMIN,   *,
    USER,    *,                      A,       1,
    USER,    *,                      B,       2,          NUM
    USER,    *,                      C,       3,          ALPHA
    USER,    *,                      GROUP1,  3,
    ADMIN,   INTERNAL\SA_SCHEDULER,  *,       *,

];
section application;
T1:
LOAD *,
NUM AS REDUCTION;
LOAD
Chr( RecNo()+ord('A')-1) AS ALPHA,
RecNo() AS NUM
AUTOGENERATE 3;
```

결과는 다음과 같습니다.

- ADMIN 그룹에 속한 사용자는 모든 데이터 및 모든 필드를 볼 수 있습니다.
- A 그룹에 속한 사용자는 모든 필드에서 REDUCTION=1에 연결된 데이터를 볼 수 있습니다.
- B 그룹에 속한 사용자는 NUM 필드를 제외하고 REDUCTION=2에 연관된 데이터를 볼 수 있습니다.
- C 그룹에 속한 사용자는 ALPHA 필드를 제외하고 REDUCTION=3에 연관된 데이터를 볼 수 있습니다.
- GROUP1 그룹에 속한 사용자는 모든 필드에서 REDUCTION=3에 연관된 데이터를 볼 수 있습니다.

Qlik Sense는 사용자를 UserID와 비교하고 테이블의 그룹에 대해 사용자를 확인합니다. 액세스가 허용되는 그룹에 사용자가 속하거나 사용자가 일치하는 경우 앱에 액세스할 수 있습니다.

5.4 가장을 사용하여 데이터 다시 로드

기본적으로 내부 시스템 계정인 SA_SCHEDULER는 다시 로드 작업을 실행하는 데 사용됩니다. 이 계정은 권한이 상승되며 기술적으로 모든 데이터 소스를 사용할 수 있습니다. 그러나 QMC에는 가장을 사용하여 내부 시스템 계정 대신 앱 소유자의 권한으로 다시 로드 작업을 실행하는 설정이 있습니다. 이 설정을 구성하

면 SA_SCHEDULER가 아닌 앱 소유자가 다시 로드에 사용됩니다. 즉, Section Access 테이블에 SA_SCHEDULER를 추가하지 않고 앱 소유자를 추가합니다. 작업 체인 내에서 앱에는 각 소유자의 액세스 권한에 따른 소스 권한을 가진 서로 다른 소유자가 있을 수 있습니다.

5.5 Multi-Cloud 환경에서 사용자 액세스 관리

Qlik Sense Multi-Cloud 환경에는 사용자 인증 메커니즘이 혼합되어 있습니다. 일반적으로 Qlik Sense Enterprise on Windows를 사용하여 프록시 서비스가 Section Access 보안 테이블의 USERID를 확인합니다. Qlik Sense SaaS에서는 ID 공급자가 해당 인증 역할을 수행할 것입니다. 결과적으로 Qlik Sense Enterprise on Windows와 같은 온프레미스 환경에 대해 설정된 Section Access는 클라우드 환경에서 작동하지 않습니다.

Qlik Sense SaaS와 함께 OIDC ID 공급자(Qlik IdP 또는 사용자 지정 IdP)를 사용하는 경우 `subject claim`은 로그인할 때 사용자를 식별하는 데 사용됩니다. Section Access를 사용하면 보안 테이블의 USERID 필드 값이 `subject claim` 값과 비교됩니다. 테넌트를 설정하는 경우 SAM 계정 이름이 ID 공급자의 `subject claim`에 매핑되는지 확인합니다. 예를 들어 SAM 계정 이름이 AD_DOMAIN\Dev인 경우 `subject claim`을 AD_DOMAIN\Dev로 설정합니다. IdP의 `subject claim` 값을 보려면 브라우저의 테넌트 URL에 */api/v1/diagnose-claims*를 추가합니다(예: *your-tenant.us.qlikcloud.com/api/v1/diagnose-claims*). JSON 응답에서 `subject claim`은 `sub`라고 합니다.

SAM 계정 이름을 사용할 수 없는 경우 사용자를 인증하는 다른 방법이 있습니다. 이메일 주소는 다른 환경에서 동일하게 유지되는 경향이 있으므로 보안 테이블에서 USERID 대신 USER.EMAIL 필드를 사용할 수 있습니다. 다음은 보안 테이블의 예입니다.

ACCESS	USERID	USER.EMAIL	주석	COUNTRY
USER	ABC\Joe	*	Access-on-prem	United States
USER	*	joe.smith@example.com	Access-in-cloud	United States
USER	ABC\Ursula	*	Access-on-prem	Germany
USER	*	ursula.schultz@example.com	Access-in-cloud	Germany
USER	ABC\Stefan	*	Access-on-prem	Sweden
USER	*	stefan.svensson@example.com	Access-in-cloud	Sweden

인증 스크립트:

```
Section Access;
LOAD * INLINE [
    ACCESS,  USERID,      USER.EMAIL,                  COUNTRY
    USER,    ABC\Joe,     *,                           United States
    USER,    *,           joe.smith@example.com,       United States
    USER,    ABC\Ursula,  *,                           Germany
    USER,    *,           ursula.schultz@example.com,  Germany
    USER,    ABC\Stefan,  *,                           Sweden
    USER,    *,           stefan.svensson@example.com, Sweden
];
```

각 사용자에는 두 레코드가 있습니다. 하나는 온프레미스 액세스용이고 다른 하나는 클라우드 액세스용입니다. 와일드카드를 사용하면 관련 인증 필드만 사용되도록 합니다. 이 예에서 COUNTRY는 데이터 축소 필드로 사용됩니다.

5.6 Section Access 및 Insight Advisor Chat 사용

Section Access를 사용하는 앱을 Insight Advisor Chat에서 사용할 수 있도록 하려면 Section Access 스크립트에서 다음 서비스 사용자에게 관리자 액세스 권한이 있어야 합니다.

- INTERNAL/sa_repository: 이렇게 하면 사용자 액세스를 제어하기 위해 리포지토리 서비스에서 Section Access 스크립트를 사용할 수 있습니다.
- INTERNAL/sa_scheduler: 이렇게 하면 앱이 QMC 작업을 사용하여 다시 로드할 수 있습니다.

*앱 이름, 필드 이름 또는 마스터 항목 이름에 중요한 정보가 있는 경우 Section Access를 사용하는 앱을 Insight Advisor Chat에 사용할 수 있게 하여 이러한 정보가 노출될 수 있습니다. 쿼리에 대한 앱 제안에는 사용자에게 액세스 권한이 있는 스트림의 앱이 포함됩니다. 여기에는 사용자가 앱의 Section Access에서 액세스 권한이 없는 앱이 포함될 수 있습니다. 그러나 이러한 앱을 선택하면 아무 작업도 수행되지 않습니다. Section Access를 사용하여 앱에서 사용 가능한 항목을 보기 위해 **차원** 또는 **측정값**을 클릭하면 사용자가 액세스 권한이 없는 항목을 볼 수 있습니다. 그러나 이러한 항목을 클릭해도 사용자에게 데이터가 제공되지 않습니다.*

예:

```
Section Access;
LOAD * inline [
       USERID ,ACCESS
       INTERNAL\sa_repository ,ADMIN
       INTERNAL\sa_scheduler ,ADMIN
       DOMAINNAME\user1 ,ADMIN
       DOMAINNAME\user2 ,USER
       DOMAINNAME\user3 ,USER
];
```

이러한 사용자가 Section Access 스크립트에 있으면 Insight Advisor Chat에 앱을 사용할 수 있도록 할 수 있습니다. 앱을 다시 로드하면 앱을 Insight Advisor Chat에서 사용할 수 있습니다.

5.7 Section Access 사용에 대한 지침 및 팁

다음은 Section Access에 대해 알아야 할 몇 가지 중요한 사실과 유용한 힌트입니다.

- 액세스 섹션의 **LOAD** 또는 **SELECT** 문에 나열된 모든 필드는 대문자로 작성해야 합니다. 데이터베이스에서 소문자를 포함한 필드 이름은 **LOAD** 또는 **SELECT** 문을 사용하여 필드를 읽기 전에 **Upper** 함수를 사용하여 대문자로 변환하십시오.
- 데이터 모델에서 필드 이름으로 나열된 Section Access 시스템 필드 이름을 사용할 수 없습니다.
- Section Access 제어가 적용되기 전에 앱을 게시해야 합니다. 앱을 다시 로드해도 새 Section Access 스크립트 또는 변경된 Section Access 스크립트가 적용되지 않습니다.

- 스냅샷은 스냅샷을 만드는 사용자의 액세스 권한에 따라 데이터를 표시하며, 스토리에서 공유될 수 있습니다. 그러나 사용자가 앱에서 라이브 데이터를 보기 위해 스토리에서 시각화로 되돌아가면 자체 액세스 권한에 따라 제한됩니다.
- Section Access를 사용하거나 중요한 데이터로 작업할 경우에는 값이 색 구성에 의해 노출될 수 있으므로 마스터 차원 값에 색을 지정해서는 안 됩니다.
- 제한된 데이터를 노출하지 않으려면 앱을 게시하기 전에 Section Access 설정이 있는 모든 첨부 파일을 제거합니다. 앱이 게시될 때 첨부 파일이 포함됩니다. 게시된 앱을 복사하는 경우 복사본에 첨부 파일이 포함됩니다. 하지만 첨부된 데이터 파일에 Section Access 제한이 적용된 경우, 파일을 복사할 때 Section Access 설정이 유지되지 않으므로 복사된 앱 사용자는 첨부된 파일에서 모든 데이터를 볼 수 있습니다.
- 와일드카드(*)는 테이블에 있는 필드의 (나열된) 모든 값으로 해석됩니다. 스크립트의 액세스 섹션에서 로드된 테이블의 시스템 필드(USERID, GROUP) 중 하나에서 사용된 경우 이 필드에 사용 가능한 (나열되지 않은) 모든 값으로 해석됩니다.
- 보안 필드를 다른 테이블에 넣을 수 있습니다.
- QVD 파일에서 데이터를 로드할 경우 Upper 함수를 사용하면 로딩 속도가 저하됩니다.
- Section Access를 설정하여 앱에서 사용자 자신을 잠근 경우에는 데이터 없이 앱을 열고 데이터 로드 스크립트에서 액세스 섹션을 편집할 수 있습니다. 그러려면 데이터 로드 스크립트를 편집하고 로드할 수 있는 액세스 권한이 있어야 합니다.
- 이진 로드의 경우 액세스 제한이 새로운 Qlik Sense 앱에서 상속됩니다.

6 On-demand 앱으로 빅 데이터 관리

On-demand 앱을 사용하면 Qlik Sense Enterprise에서 빅 데이터 소스를 로드하고 분석할 수 있습니다. 전체 빅 데이터 저장소를 한 번에 분석하려는 것은 매우 비효율적입니다. 그렇지만 대표 시각화를 만들려면 모든 데이터를 검색할 수 있어야 합니다. Qlik Sense On-demand 앱은 사용자에게 빅 데이터 저장소 집계 보기를 제공하여 사용자가 자세한 분석을 위해 관련 데이터 하위 집합을 식별하고 로드할 수 있습니다.

On-demand 앱은 현업 사용자가 대규모 데이터 소스에서 연관 분석을 수행하도록 지원함으로써 데이터 검색의 잠재적 활용 분야를 넓혀 줍니다. 통찰력을 발견하는 데 관심 있는 데이터를 먼저 선택하고 대화형으로 On-demand 앱을 생성하여 전체 Qlik 인 메모리 기능으로 데이터를 분석할 수 있습니다.

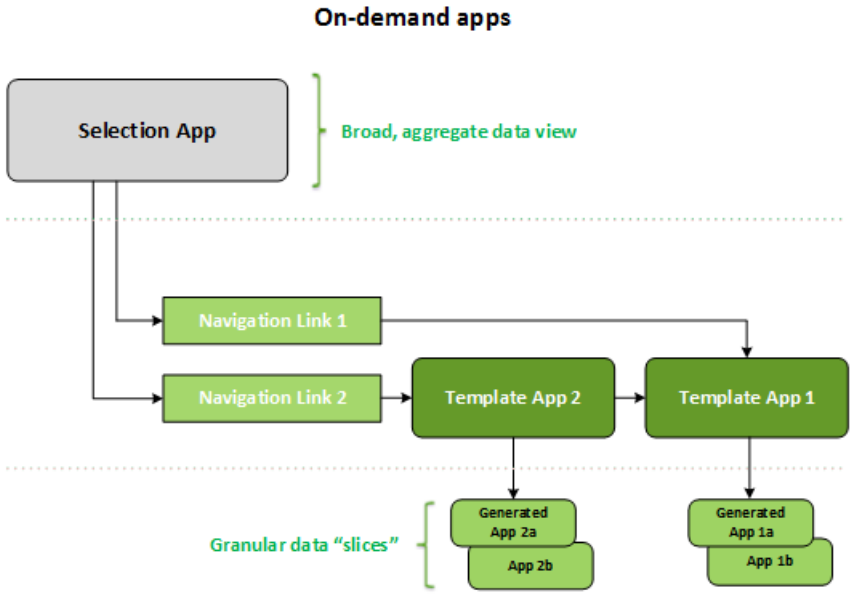
6.1 On-demand 앱 구성 요소

Qlik Sense는 빅 데이터의 집계 보기를 제공하는 선택 앱을 사용하여 빅 데이터 소스 로드 작업을 관리하며 세부적인 데이터 확대 및 분석도 가능하게 해줍니다. On-demand 앱을 만들 때 기본으로 사용되는 하나 이상의 템플릿 앱에 대한 앱 탐색 링크가 각 선택 앱에 포함되어 있습니다. 탐색 링크 및 템플릿 앱의 속성을 통해 On-demand 앱에 로드된 데이터 볼륨 및 모양을 긴밀하게 제어할 수 있습니다.

앱을 템플릿 앱에서 반복적으로 생성하여 자주 변경되는 데이터 셋을 추적할 수 있습니다. 선택 앱의 선택 내용에 따라 데이터가 필터링되는 반면 On-demand 앱 콘텐츠는 기본 데이터 소스에서 동적으로 로드됩니다. 동일한 On-demand 앱을 여러 번 생성하여 데이터가 변경될 때 새로운 분석을 수행할 수 있습니다.

On-demand 앱 생성은 On-demand 앱 서비스에 의해 제어됩니다. 이 서비스는 기본적으로 비활성화되어 있으므로 선택 및 템플릿 앱을 연결할 수 있기 전에 및 On-demand 앱을 생성하기 전에 활성화해야 합니다. On-demand 앱 서비스는 Qlik Management Console에서 관리됩니다.

On-demand 앱 구성 요소 간 관계



6.2 On-demand 앱 구성

On-demand 선택 및 템플릿 앱은 특수 로드 스크립팅이 필요하기 때문에 일반적으로 Qlik Sense 로드 스크립트 작성 경험이 있는 사용자가 만듭니다. 예를 들어 On-demand 선택 앱은 적당한 수준의 차원 세분성을 가진 데이터를 로드해야 합니다. On-demand 템플릿 앱에는 데이터 소스에서 만들어진 쿼리를 공식화하는 데 사용되는 데이터 바인딩 표현식이 있는 로드 스크립트가 포함되어 있습니다.

하나의 선택 앱은 여러 개의 템플릿 앱에 연결될 수 있으며 하나의 템플릿 앱은 여러 개의 선택 앱에 연결될 수 있습니다. 그러나 템플릿 앱의 데이터 바인딩 표현식은 연결된 선택 앱의 필드와 대응해야 합니다. 이러한 이유 때문에 선택 및 템플릿 앱은 서로 연결되어 만들어지고 종종 동일한 숙련된 스크립트 작성자에 의해 작성되는 경향이 있습니다.

> *ProgramData\Qlik\Examples\OnDemandApp\sample에 설치된 Qlik Sense Enterprise에는 샘플 On-demand 선택 및 템플릿 앱이 포함되어 있습니다.*

탐색 링크를 만들려면 템플릿 앱에 해당 바인딩이 있는 선택 앱의 필드를 이해해야 합니다. 각 탐색 링크는 총 세부 레코드 수를 계산하는 표현식을 필요로 하기 때문입니다. 이 총합은 선택 앱의 선택 상태를 통해 액세스할 수 있는 집계 레코드를 나타냅니다. 이 표현식을 만들려면 사용자가 선택 앱에서 사용할 수 있는 필드를 사용하여 템플릿 앱의 총 레코드 수를 계산하는 방법을 알아야 합니다.

선택 앱을 사용하여 On-demand 앱을 생성하는 경우 사용자가 로드 스크립트를 이해할 필요가 없습니다. On-demand 앱 탐색 링크가 만들어지면 사용자는 해당 탐색 링크를 선택 앱의 **앱 탐색** 막대 위로 끌어서 앱 탐색 포인트를 만들 수 있습니다. 그러면 On-demand 앱이 탐색 포인트에서 생성됩니다.

탐색 링크 표현식에서 계산한 최대 행이 필요한 범위 내에 있을 때 On-demand 앱 생성을 위해 탐색 포인트를 사용할 수 있습니다. 이 시점에서 사용자는 On-demand 앱을 생성할 수 있습니다. 또한 사용자는 다른 선택 집합을 만들고 이러한 다른 선택을 기반으로 추가 앱을 생성할 수 있습니다.

탐색 링크에는 링크에서 생성할 수 있는 On-demand 앱의 수에 제한이 있습니다. 최대 앱 수가 생성되면 탐색 포인트에서 앱을 생성하는 사용자가 새로운 On-demand 앱을 생성하기 전에 기존 앱 중 하나를 삭제해야 합니다. 생성된 최대 앱 수는 On-demand 앱 탐색 링크에 적용됩니다. 탐색 링크에서 하나의 On-demand 앱 탐색 포인트를 만든 경우 해당 탐색 포인트가 최대 수까지 만들 수 있습니다. 동일한 탐색 링크에서 여러 탐색 포인트를 만들면 이러한 탐색 포인트가 함께 탐색 링크에 대해 설정된 최대 수로 제한됩니다.

또한 탐색 링크는 생성된 앱의 보존 시간을 설정합니다. On-demand 앱은 보존 기간이 만료되면 자동으로 삭제됩니다.

6.3 On-demand 앱 게시

대부분의 사용자는 On-Demand 및 선택 앱이 게시된 후 사용합니다. 선택 앱이 스트림에 게시되면 해당 스트림에 대한 적절한 권한을 가진 사용자는 선택 앱을 사용하여 집계 선택을 만들고 선택 앱에 포함된 탐색 포인트에서 On-demand 앱을 생성할 수 있습니다. 게시된 모든 앱과 마찬가지로 선택 앱은 게시된 후에는 변경할 수 없습니다. 예를 들어 탐색 포인트를 추가하려면 사용자는 선택 앱의 복사본을 만들어야 합니다.

대부분의 경우 사용자는 생성된 On-demand 앱만 사용합니다. 생성된 각 앱은 별도로 게시할 수 있습니다. 실제로 앱 탐색 링크는 앱 탐색 링크에서 생성된 앱이 특정 스트림에 자동으로 게시되도록 지정할 수 있습니다. 실제로 앱 탐색 링크는 앱 탐색 링크에서 생성된 앱이 특정 스트림에 자동으로 게시되도록 지정할 수 있습니다. 그런 다음 사용자는 앱이 게시된 스트림에서 생성된 On-demand 앱으로 로드된 선택된 데이터 조각을 탐색합니다.

6.4 On-demand 앱의 장점

On-demand 앱은 비즈니스 사용자와 IT 부서가 다양한 방식으로 빅 데이터 환경에서 값을 파생할 수 있도록 도와줍니다. On-demand 앱:

- 사용자에게 대화형으로 기간, 고객 세그먼트 또는 지역과 같은 데이터의 하위 집합으로 앱을 채울 수 있는 “쇼핑 목록” 환경을 제공합니다.
- 메모리에 호스팅된 잠재 하위 집합에서 완전한 Qlik Sense 기능을 제공합니다.
  반면 Direct Discovery는 큰 데이터 소스를 관리할 수는 있지만 모든 관련 데이터를 메모리에 보관하지는 않습니다. Direct Discovery를 사용하면 측정값 데이터는 실행될 때까지 소스에 존재합니다.
- IT가 데이터 볼륨 또는 차원 선택에 따라 앱의 크기를 제어하고 앱을 호출할 수 있습니다.
- Teradata Aster, MapR, SAP BEx 및 SAP HANA의 PLACEHOLDER 함수와 같은 비 SQL 데이터 소스에 대한 액세스를 제공합니다.
  비 SQL 쿼리를 수행하는 것은 Direct Discovery가 SQL 데이터 소스에서만 사용할 수 있는 것과는 대조적입니다.
- 사용자 지정 가능한 SQL 및 로드 스크립트 생성을 허용합니다.
- 모든 경우에 섹션 액세스를 허용합니다.

6.5 제한 사항

On-demand 앱에서 Qlik NPrinting을 사용할 수 없습니다.

6.6 On-demand 선택 앱 만들기

On-demand 선택 앱은 Qlik associative engine이 효율적이고 효과적으로 연결할 수 있도록 대규모 데이터 셋의 하위 집합을 선택하는 수단을 제공합니다. 매우 큰 데이터 볼륨 환경에서는 선택 앱이 적당한 수준으로 세분화된 차원만 로드하는 것이 좋습니다. 예를 들어 해당 데이터가 분기, 지역 및 제품 범주별로 집계된 판매 데이터를 기반으로 하는 선택 앱은 다음과 같은 **SQL SELECT** 문을 사용할 수 있습니다.

```
SELECT SUM(S.UNIT_COST) AS TOTAL_UNIT_COST, SUM(S.QUANTITY) AS TOTAL_QUANTITY, SUM(S.UNIT_
PRICE * S.QUANTITY) AS TOTAL_SALE, SUM( (S.UNIT_PRICE - S.UNIT_COST) * QUANTITY) AS TOTAL_
PROFIT, SUM(1) AS TOTAL_LINE_ITEMS, S.REGION, S.YEARQUARTER, S.PRODCAT, FROM SALE_DETAIL S
GROUP BY S.REGION, S.YEARQUARTER, S.PRODCAT
```

On-demand 측정값 표현식 속성은 일반적으로 데이터를 로드하는 데 사용된 SQL GROUP BY 쿼리에서 계산된 집계 결과를 기반으로 합니다. 선택 앱은 **GROUP BY** 쿼리를 사용하여 SALE_DETAIL 레코드를 집계하므로 집계 함수(이 경우 **SUM**)는 UNIT_COST, QUANTITY의 측정값 필드와 TOTAL_SALE 및 TOTAL_PROFIT에 대해 계산된 값에 사용해야 합니다.

SUM(1) AS TOTAL_LINE_ITEMS는 지역, 분기 및 제품 범주의 모든 고유한 조합에 대한 총 판매 품목 수를 정확하게 측정할 수 있는 방법을 제공합니다. On-demand 앱 제작에 사용되는 링크를 만들 때 On-demand 앱에 로드되는 레코드 수를 제어하는 방법으로 측정값 표현식을 제공해야 합니다. SALE_DETAIL 예에서 사용자가 여러 제품 범주, 지역 및/또는 분기를 선택할 때 TOTAL_LINE_ITEMS에 대한 합계를 계산하여 선택 항목이 On-demand 앱의 레코드 한계를 초과하는지 여부를 판단할 수 있습니다.

> *ProgramData\Qlik\Examples\OnDemandApp\sample에 설치된 Qlik Sense Enterprise on Windows에 샘플 On-demand 선택 앱이 포함되어 있습니다.*

레코드 한계는 앱 탐색 링크를 만들기 위해 선택 앱이 템플릿 앱에 연결될 때 지정됩니다. 레코드 한계는 앱 탐색 링크를 만들기 위해 선택 앱이 템플릿 앱에 연결될 때 지정됩니다. 각 앱 탐색 링크에는 레코드 한계가 있습니다. 선택 앱에서 여러 탐색 링크를 만들 수 있습니다. 여러 앱 탐색 링크는 일반적으로 여러 데이터 보기를 생성하기 위해 선택 앱을 다양한 템플릿 앱에 연결합니다.

개별 On-demand 앱 탐색 링크는 게시를 위해 선택 앱에 포함될 수 있습니다. 선택 앱에 포함되면 앱 탐색 링크를 사용하여 하나 이상의 앱 탐색 포인트를 만듭니다. 이를 통해 특정 시트 사용자가 해당 링크의 템플릿 앱을 기반으로 On-demand 앱을 만들 수 있습니다.

6.7 On-demand 템플릿 앱 만들기

On-demand 템플릿 앱은 하나의 중요한 차이점이 있는 일반 Qlik Sense 앱입니다. 해당 로드 스크립트에는 데이터 소스에서 만들어진 쿼리를 공식화하는 데 사용되는 데이터 바인딩 표현식이 포함되어 있습니다. 이러한 데이터 바인딩 표현식은 On-demand 앱 생성 시간에 사용되어 선택 앱의 선택 상태에서 값을 읽어서 이 값을 템플릿 앱 스크립트에 바인딩하여 생성된 앱에 사용자가 제어하는 데이터 하위 집합이 로드됩니다.

템플릿 앱은 일반적으로 선택 앱과 동일한 데이터 소스에 연결됩니다. 선택 앱의 로드 스크립트는 중요한 차원 및 측정값의 대화형 시각화를 제공하면서 데이터 볼륨을 줄이기 위해 일반적으로 집계된 데이터를 로드합니다. 템플릿 앱의 로드 스크립트는 보다 세밀한 데이터의 제어된 하위 집합을 로드하는 쿼리를 사용합니다.

On-demand 템플릿 앱은 데이터를 직접 로드하지 않습니다. 템플릿 앱 연결은 유효해야 하지만 연결이 제대로 작동하는지 테스트하려면 On-demand 앱을 생성해야 합니다. On-demand 앱이 생성되면 On-demand 선택 앱의 선택 상태를 로드하기 위해 On-demand 앱 서비스에서 로드 스크립트를 수정합니다. On-demand 앱이 오류 없이 생성되면 선택 앱과 템플릿 앱의 연결이 올바르게 작동하는 것입니다.

템플릿 앱의 구조

템플릿 앱은 On-demand 앱 탐색 링크를 사용하여 선택 앱에 연결됩니다. 앱 탐색 링크에는 On-demand 앱에 데이터가 로드될 때 쿼리되는 레코드 수를 제어하는 속성이 포함되어 있습니다. 템플릿 앱의 로드 스크립트에는 On-demand 앱에 데이터가 로드될 때 실행되는 쿼리를 공식화하는 데 사용되는 선택 앱의 필드 데이터를 지정하는 데이터 바인딩 표현식이 포함되어 있습니다.

ProgramData\Qlik\Examples\OnDemandApp\sample에 설치된 Qlik Sense Enterprise on Windows에 샘플 On-demand 템플릿 앱이 포함되어 있습니다.

2020년 6월에 데이터 바인딩 표현식에 대한 새 구문이 도입되었습니다. 수량 제약 조건, _n 접미사 및 서식 사양을 포함하여 이전 구문 및 접두사 od_, ods_, odo_, odso_ 및 odb_는 전과 마찬가지로 동작합니다. 앱이 2020년 6월 이전의 Qlik Sense 버전에서 작동해야 하는 경우 이전 구문을 사용합니다. 이전 구문의 경우 On-demand 템플릿 앱 만들기(이전 버전)*를 참조하십시오.*

새 접두사를 사용할 때 _n 접미사는 지원되지 않습니다.

기본 데이터 바인딩 표현식의 양식은 다음과 같습니다.

$(odag_FIELDNAME)

odag_ 접두사는 선택 앱의 선택 상태를 템플릿 앱에서 정의한 On-demand 앱의 로드 스크립트에 바인딩하는 데 사용됩니다. odag 접두사 다음에 오는 데이터 바인딩 표현식 부분은 선택 앱의 필드와 일치하는 이름이어야 합니다. On-demand 앱이 생성되면 선택 앱의 현재 선택 상태는 각 필드에 대해 바인딩할 원하는 값을 가져오는 데 사용됩니다. On-demand 앱이 생성되면 선택 앱의 현재 선택 상태는 각 필드에 대해 바인딩할 원하는 값을 가져오는 데 사용됩니다. 새로 만든 On-demand 앱의 로드 스크립트에 있는 **$(odag_FIELDNAME)** 표현식은 발생할 때마다 선택 앱의 선택 상태에서 해당 필드에 대해 선택된 값 목록으로 대체됩니다.

보다 특수화된 데이터 바인딩을 위해 다른 접두사를 사용할 수 있습니다. 특수한 경우에 맞게 조정하고 LOAD 문을 최적화하는 방법을 알아보려면 *On-demand 템플릿 앱의 바인딩 표현식 (page 176)*를 참조하십시오.

On-demand 바인딩을 로드 스크립트의 **SELECT** 및 **WHERE** 문에 직접 삽입할 수 있습니다. 바인딩을 **WHERE** 문에 직접 추가하면 문에서 다른 조건과 함께 조합하기가 더 쉽습니다.

로드 스크립트를 만들 때 자리 표시자 변수 **$(odagActive)**를 추가할 수 있습니다. 이렇게 하면 모든 데이터를 로드하지 않고 동적 보기에 대한 마스터 차트를 만들 수 있도록 샘플 데이터를 템플릿 앱에 로드할 수 있습니다.

다음 예에서 샘플 On-demand 템플릿 로드 스크립트를 설명합니다.

몇 가지 샘플 데이터 추가

이 예에서는 바인딩이 완료되지 않은 경우에도 앱을 로드할 수 있도록 샘플 값을 추가합니다.

```
IF '$(odagActive)'='' THEN
trace ODAG variables not inserted! Loading sample data.;
SET 'odag_Origin Code' = '''LAX''';
SET 'odag_Destination Code' = '''JFK''';
SET odagn_Year = 2015;
SET odag_Quarter = '''1''';
SET 'odag_Ticket Carrier Code' = '''CA''';
SET 'odag_Fare Class' = '''X''';
END IF;
```

템플릿 앱에서 데이터 로드

다음은 샘플 데이터를 로드하고 생성된 **odag_FIELDNAME** 바인딩으로 필터링하기 위한 샘플 로드 스크립트입니다. **odagn_<Field Name>** 바인딩은 이중 값에서 숫자를 선택하고 기본적으로 인용 부호를 사용하지 않습니다.

```
SQL SELECT *
FROM FlightDB.Flights
WHERE "Origin Code" IN ($(odag_Origin Code))
AND "Destination Code" IN ($(odag_Destination Code))
AND "Year" IN ($(odagn_Year))
AND "Quarter" IN ($(odag_Quarter))
AND "Ticket Carrier Code" IN ($(odag_Ticket Carrier Code))
AND "Fare Class" IN ($(odag_Fare Class));
```

단일 사인온(SSO)

On-demand 앱은 SSO를 지원하는 데이터 소스와 함께 단일 사인온(SSO)을 사용할 수 있습니다. 엔진 및 데이터 소스는 SSO를 허용하도록 구성해야 합니다.

SSO에 대해 엔진 및 데이터 소스를 구성한 후 템플릿 앱은 템플릿 앱 스크립트에 다음 구문을 추가하여 SSO를 활성화해야 합니다.

```
///!ODAG_SSO
```

On-demand 앱 서비스는 On-demand 앱이 생성될 때 및 On-demand 앱이 다시 로드될 때마다 스크립트를 구문 분석합니다.

On-demand 앱이 SSO에서 로드되면 최종 사용자의 ID가 데이터 소스로 전송됩니다. 최종 사용자는 템플릿 앱의 데이터 연결에서 사용되는 소스에 대한 액세스 권한이 있어야 합니다. 대규모 데이터 집합이 선택된 경우에도 사용자에게 해당 소스에 대한 액세스 권한이 있는 데이터만 로드됩니다.

단일 사인온(SSO)을 사용하는 템플릿 앱에서 생성된 On-demand 앱은 게시할 수 없습니다.

템플릿 앱에 대한 노드 다시 로드

관리자는 템플릿 앱에 부하 분산 규칙을 설정하여 다중 노드 환경에서 On-demand 앱을 다시 로드할 위치를 제어할 수 있습니다. 부하 분산 규칙이 설정되면 템플릿 앱에 대한 링크로 생성된 모든 앱이 템플릿 앱에 적용되는 규칙에 따라 로드됩니다.

On-demand 템플릿 앱의 바인딩 표현식

템플릿 앱의 데이터 바인딩은 On-demand 앱에 데이터가 로드될 때 실행되는 쿼리를 공식화하는 데 사용되는 해당 선택 앱의 데이터를 지정합니다.

바인딩 표현식의 기본 형식(**$(odag_FIELDNAME)**)이 표현식을 바인딩하는 데 사용할 수 있는 유일한 방법은 아닙니다. 다른 접두사를 사용하여 선택 내용을 세분화하고 탬플릿 앱이 데이터를 정확하게 로드할 수 있도록 할 수 있습니다.

원래 On-demand App Generation용 Qlik Sense 확장을 사용하여 만든 템플릿 앱은 필드의 다양한 선택 내용을 바인딩하기 위해 아래 설명된 방법을 사용하여 변경해야 합니다.

사용 가능한 바인딩 접두사

일반 접두사 형식은 **odag[s|o][n][cnt]**입니다. 여기서,

- s - 선택한 값만 포함
- o - 선택적 값만 포함
- n - 숫자 버전 선택, 기본적으로 인용 부호로 묶지 않음
- cnt - 실제 값 대신 값 수 삽입

다음 표에는 사용 가능한 바인딩 접두사의 모든 버전 목록이 나와 있습니다. 이 예에서는 필드 이름 *MyField*를 1,2,3(선택한 값(녹색 값))으로 가정하고 4(선택한 선택적 값(흰색 값)인)로 가정합니다.

바인딩 접두사

접두사	설명	예	대체 값
odag_	선택한 값(녹색)과 선택적 값(흰색)으로 대체됩니다. 값의 텍스트 버전을 선택합니다. 이 접두사는 문자열 값의 표준 접두사입니다.	*$(odag_MyField)*	'1','2','3','4'
odagcnt_	해당 odag_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화에 사용됩니다.	○ *$(odagcnt_MyField)*	4

접두사	설명	예	대체 값
odagn_	선택한 값(녹색)과 선택적 값(흰색)으로 대체됩니다. 값의 숫자 버전을 선택합니다. 이 접두사는 숫자 값의 표준 접두사입니다. *데이터 모델이 선택된 값이나 선택적 필드 값이 없는 경우 표현식에서 noValue를 지정해야 합니다. 예: `$(odagn_MyField){"noValue":"-99999"}`.* *자세한 내용은 값 따옴표 및 구분 기호 변경 (page 179)을 참조하십시오.*	*$(odagn_MyField)*	1,2,3,4
odagncnt_	해당 odagn_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화용입니다.	*$(odagncnt_MyField)*	4
odago_	선택적 값(흰색)으로 대체됩니다. 값의 텍스트 버전을 선택합니다. 이 접두사는 쿼리 최적화용입니다. *대규모 데이터베이스 최적화 (page 178).*	*$(odago_MyField)*	'4'
odagocnt_	해당 odago_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화용입니다.	*$(odagocnt_MyField)*	1
odagon_	선택적 값(흰색)으로 대체됩니다. 값의 숫자 버전을 선택합니다. 이 접두사는 쿼리 최적화용입니다. *대규모 데이터베이스 최적화 (page 178).*	*$(odagon_MyField)*	4
odagoncnt_	해당 odagon_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화용입니다.	*$(odagoncnt_MyField)*	1
odags_	선택한 값(녹색)으로 대체됩니다. 값의 텍스트 버전을 선택합니다. 이 접두사는 쿼리 최적화용입니다. *대규모 데이터베이스 최적화 (page 178).*	*$(odags_MyField)*	'1','2','3'
odagscnt_	해당 odags_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화용입니다.	*$(odagscnt_MyField)*	3
odagsn_	선택한 값(녹색)으로 대체됩니다. 값의 숫자 버전을 선택합니다. 이 접두사는 쿼리 최적화용입니다. *대규모 데이터베이스 최적화 (page 178).*	*$(odagsn_MyField)*	1,2,3
odagsncnt_	해당 odagsn_ 바인딩의 값 수로 대체됩니다. 이 접두사는 쿼리 최적화용입니다.	*$(odagsncnt_MyField)*	3

빈 값이 텍스트 버전에서 필터링됩니다. 숫자가 아닌 값과 NaN 값이 숫자 버전에서 필터링됩니다.

대규모 데이터베이스 최적화

odags_ 및 odagsn_ 접두사는 쿼리를 최적화하기 위한 것입니다. 바인딩된 필드에 선택 항목이 없는 경우 odag_는 모든 값을 포함하지만 odags_는 아무 값도 포함하지 않습니다. 경우에 따라 odags_ 및 odagscnt_ 접두사를 사용하는 것이 더 효율적입니다. 이렇게 하면 값 집합이 비어 있는지 테스트할 수 있습니다. 예를 들어 다음은 *odag_MyField*의 모든 값을 테스트하는 것보다 *MyField*에 선택된 항목이 없는 경우 더 효과적입니다.

```
WHERE ($(odagscnt_MyField)=0 OR MyColumn IN ($(odags_MyField)))
```

odags_는 On-demand 필드가 아닌 선택 앱에서 선택할 대체 필드가 있는 경우에 사용할 수 없습니다. 예를 들어 사용자가 *CountryName*에서 선택해도 바인딩 표현식이 연결된 필드 *CountryCode*에 있는 경우 odags_를 사용할 수 없습니다. 이러한 경우 대신 odago_를 사용할 수 있습니다. odago_ 바인딩에 값이 없으면 모든 값을 포함해야 하거나 어떤 값도 포함할 수 없음을 의미할 수 있습니다.

바인딩 숫자 값

On-demand 앱에 바인딩할 데이터가 문자열 대신 숫자로 구성되어 있으면 숫자 필드에서 인용 부호로 묶는 동작을 비활성화하는 것이 유용합니다. 예를 들어 판매 레코드에 숫자 DAY_OF_WEEK 열이 포함되어 있고 선택 앱의 사용자가 DAY_OF_WEEK의 임의 조합을 선택하게 하려는 경우 선택 앱을 로드하는 데 사용되는 집계 쿼리를 확대하면 DAY_OF_WEEK가 **SELECT** 목록과 **GROUP BY** 목록에 모두 포함됩니다. DAY_OF_WEEK 값을 선택할 때 이 값이 따옴표로 묶여 있으면 데이터베이스가 문자열에서 숫자로의 자동 유형 변환을 지원하지 않는 경우 런타임 쿼리 오류가 발생할 수 있습니다.

이 상황을 처리하기 위해 바인딩 표현식 접미사의 숫자 버전을 사용할 수 있습니다. 그러면 문자열 값이 아닌 선택 앱에서 숫자 값을 사용하도록 필드 바인딩이 적용됩니다. 다음 숫자 버전을 사용할 수 있습니다.

- odagn_
- odagon_
- odagsn_

숫자 버전을 사용하여, 선택한 값을 저장하는 이중 값의 숫자 부분에서 값이 선택되며, 이 값은 기본적으로 인용 부호로 묶지 않습니다.

특정 수를 선택하도록 요구

경우에 따라 On-demand 앱 쿼리에 특정 필드의 특정 숫자 또는 값 범위가 포함되도록 요구할 필요가 있을 수 있습니다. 예를 들어 시작일과 종료일 사이의 모든 판매를 가져오기 위해 On-demand 앱 쿼리에 **BETWEEN** 절을 포함하는 경우 YEARQUARTER 필드의 바인드 표현식에 [2]의 접미사 구문을 사용할 수 있습니다. 그러면 YEARQUARTER에 대해 다음과 같이 정확히 2개의 값을 선택해야 합니다.

```
$(odag_YEARQUARTER)[2]
```

선택 앱의 On-demand 앱 탐색 포인트는 정확히 두 개의 YEARQUARTER 값이 선택되어 있지 않으면 비활성화된 상태로 유지됩니다. 정확히 두 개의 YEARQUARTER 값을 선택해야 한다는 메시지가 표시됩니다.

선택 수량 제약 조건은 선택 앱과 On-demand 앱 간에 필수 연결을 만듭니다. 이는 수량 제약 조건을 사용하지 않는 바인드 표현식과는 다릅니다. 예를 들어 다음과 같이 템플릿 앱의 스크립트에 수량 제약 조건이 없는 바인드 표현식이 포함된 경우입니다.

```
$(odag_MYFIELD)
```

선택 앱에 MYFIELD라는 필드가 포함되어야 한다거나 필드가 있어도 해당 필드의 선택된 값이 존재해야 한다는 요구 사항이 없습니다. 선택 앱에 MYFIELD라는 필드가 포함되어 있지 않거나 사용자가 단순히 해당 필드에서 아무런 선택을 하지 않는 경우, 레코드 제한 값 조건을 충족하기 위해 다른 선택을 하면 On-demand 앱 탐색 포인트를 계속 사용할 수 있습니다.

반면에 바인드 표현식은 다음과 같습니다.

```
$(odag_MYFIELD)[1+]
```

이제 선택 앱은 두 가지 요구 사항이 있습니다.

- 선택 앱에 MYFIELD라는 필드가 포함되어 있어야 합니다.
- 사용자가 MYFIELD에 대해 하나 이상의 값을 선택해야 합니다.

이 유형의 바인드 표현식은 템플릿 앱과 함께 사용할 수 있는 선택 앱을 제한하기 때문에 신중하게 사용해야 합니다. 선택 수량 요구 사항을 해당 템플릿 앱에 연결하는 모든 선택 앱에 부과하려는 확신이 없는 한 이 수량 제약 조건을 템플릿 앱의 바인딩에 사용해서는 안 됩니다.

데이터 바인딩 프로세스를 수행하기 위해 On-demand 앱 서비스는 스크립트의 주석에 영향을 받지 않는 문자열 대체 방법을 사용합니다. 즉, 이러한 주석에 앱 생성 후 바인딩된 값 목록이 포함되기를 원하지 않는다면 주석에 바인드 표현식을 사용하지 않아야 합니다.

다른 수량 제약 조건은 가능합니다. 다음 표에는 선택 수량 제약 조건의 다양한 조합이 요약되어 있습니다.

선택 수량 제약 조건의 다양한 조합

제약 조건 패턴	선택 요구 사항
$(odag_YEARQUARTER)[2]	정확히 2개의 YEARQUARTER 값을 선택해야 합니다.
$(odag_YEARQUARTER)[2-4]	2개에서 4개 사이의 YEARQUARTER 값을 선택해야 합니다.
$(odag_YEARQUARTER)[2+]	2개 이상의 YEARQUARTER 값을 선택해야 합니다.
$(odag_YEARQUARTER)[2-]	최대 2개까지 YEARQUARTER 값을 선택할 수 있습니다.

템플릿 앱의 모든 수량 제약 조건이 충족되었는지 결정하는 검사는 앱 생성 프로세스 중에 수행됩니다. 수량 제약 조건을 위반하면 앱 생성 요청이 거부되고 오류 메시지가 표시됩니다.

값 따옴표 및 구분 기호 변경

선택 앱에서 선택된 필드의 값 목록이 템플릿 앱의 스크립트에 삽입된 경우 값은 작은따옴표로 묶이고 쉼표로 구분됩니다. 이러한 문자가 인용 부호와 구분 기호에 대한 기본 문자입니다. 이러한 값은 각 필드의 바인딩 문에 추가된 구문에서 변경할 수 있습니다. 예:

```
$(odag_ORIGIN){"quote": "|", "delimiter": ";"}
```

이러한 새 값은 선택 앱에서 가져온 바인딩 값 목록을 작성할 때 사용됩니다. 예를 들어 선택된 값이 해당 연도의 처음 3개월인 경우 목록은 다음과 같이 구성됩니다.

```
|January|;|February|;|March|
```

인용 부호 및 구분 기호 문자의 기본값은 대부분의 표준 SQL 데이터베이스에서 작동합니다. 그러나 일부 SQL 데이터베이스에는 작동하지 않을 수 있으며 NoSQL 및 REST 등 여러 동적 데이터 소스에서는 작동하지 않습니다. 이러한 소스의 경우 인용 부호 및 구분 기호 문자를 변경하려면 이 바인딩 표현식을 추가해야 합니다.

다음 표에는 따옴표 및 구분 기호 문자를 변경하는 형식 매개 변수가 나와 있습니다.

형식 매개 변수

매개 변수	기본값	정의
quote	텍스트 접두사의 경우 '(작은따옴표) 숫자 접두사의 경우 비어 있음	각 값의 앞뒤에 추가됨
delimiter	,(쉼표)	모든 값 사이에 추가됨
quoteReplace	"(큰따옴표)	값이 비어 있지 않고 따옴표가 비어 있지 않은 경우에는 값 내에 있는 따옴표의 모든 값이 지정된 문자열로 대체됩니다. *quoteReplace는 숫자 접두사(예: odagn_)의 매개 변수로 지원되지 않습니다. 숫자 접두사는 quoteReplace를 무시합니다.*
noValue	(비어 있음)	필드에 대해 선택된 값이 없는 경우 대신 이 값이 사용됩니다. 이 매개 변수는 특정 필드에 대해 선택된 값이 없는 경우에 유용합니다. 값은 소스 데이터에 없는 값으로 설정해야 합니다. 숫자 값의 경우 예를 들어 데이터베이스의 모든 값이 양인 경우 음수 값을 사용합니다. *인용 부호로 묶지 않은 값의 경우 선택된 필드 값이 빈 집합일 수 있으면 noValue를 지정해야 합니다.*

다음 표에는 odag_ 및 odagn_에 대한 서식 사양 및 생성된 값이 나와 있습니다. 생성된 삽입 값은 기본 데이터 값 VAL1, VAL2를 기반으로 합니다.

odag_ 예제 서식 사양 및 생성된 값

서식 사양	설명	생성된 삽입 값
지정되지 않음	값 목록이 쉼표로 구분되고 '로 묶입니다.	'VAL1','VAL2'

서식 사양	설명	생성된 삽입 값
{"quote": "", "delimiter": ""}	연결된 값	VAL1VAL2
{"quote": "X", "delimiter": "Y"}	값이 X로 묶이고 Y로 구분됩니다.	XVAL1XYXVAL2X
{"quote": "XX", "delimiter": "YY"}	값이 XX로 묶이고 YY로 구분됩니다.	XXVAL1XXYYXXVAL2XX
{"quote": "X"}	값이 X로 묶이고 ,(기본값)로 구분됩니다.	XVAL1X,XVAL2X
{"delimiter": "YY"}	'(기본값)로 묶이고 YY로 구분됩니다.	'VAL1'YY'VAL2'
{"quote": ""}	값이 인용 부호로 묶이지 않고 ,로 구분됩니다.	VAL1,VAL2
{"quote": "A", "quoteReplace": "\\A"}	값이 A로 묶이고 쉼표(기본값)로 구분됩니다. 필드 내부 값이 \A로 대체됩니다. *이 예에서 \는 json 형식의 이스케이프 문자이므로 두 개의 \가 필요합니다.*	AV\AL1A,AV\AL2A

odagn_ 예제 서식 사양 및 생성된 값

서식 사양	설명	생성된 삽입 값
지정되지 않음	값 목록이 인용 부호로 묶이지 않고 쉼표로 구분됩니다. 숫자 값이 사용됩니다.	VAL1,VAL2
{"delimiter": "YY"}	값이 인용 부호로 묶이지 않고 YY로 구분됩니다.	VAL1YYVAL2
{"quote": "A", "quoteReplace": "\\A"}	odag_ 접두사에 비해 quoteReplace 매개 변수는 무시됩니다.	AVAL1A,AVAL2A

개별 값 처리

개별 필드 값 처리가 필요한 경우 inline 메서드를 사용하여 변수 Values의 값을 생성하고 **Replace** 또는 다른 함수를 사용하여 임의 처리를 수행합니다. 아래 예에서는 **Replace**가 자리 표시자 값으로 사용됩니다.

```
MyTempBindingData:
LOAD * INLINE [VAL
$(odag_MyField){"quote": "", "delimiter": "\n"}
];

_TempTable:
LOAD Concat(chr(39) & Replace(text, from_str, to_str) & chr(39), ',') as CombinedData Resident
MyTempBindingData;
LET Values = Peek('CombinedData',0,'_TempTable');
drop table _TempTable;
drop table MyTempBindingData;
```

6.8 On-demand 앱 만들기

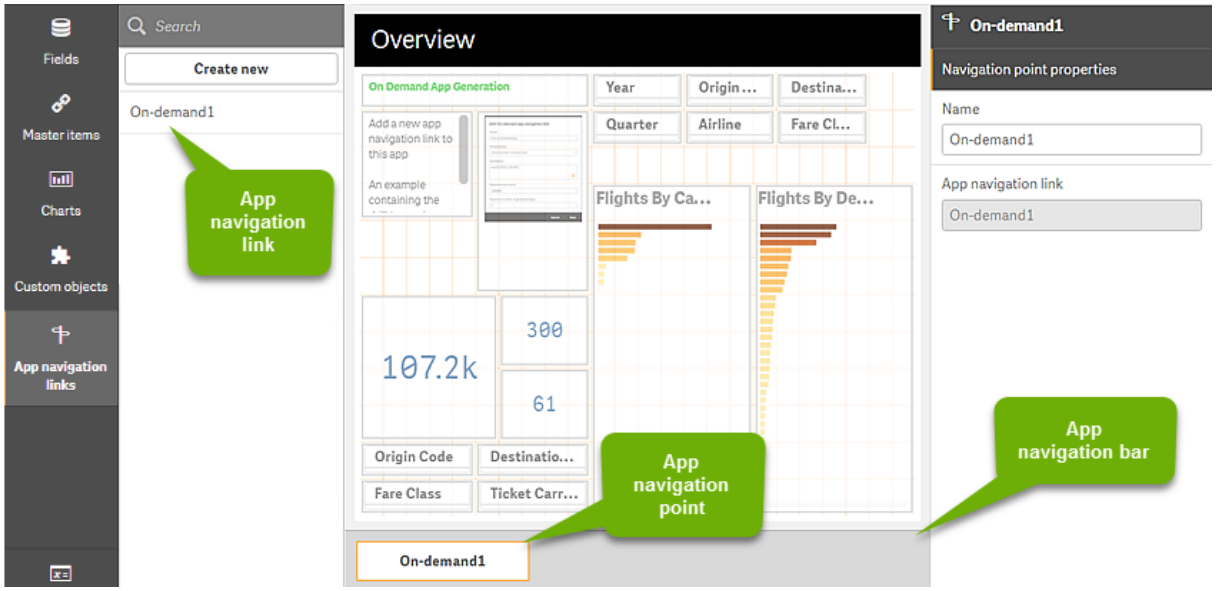
On-demand 앱은 선택 앱에서 집계 양식으로 로드된 데이터의 하위 집합을 로드합니다. On-demand 앱은 On-demand 선택 앱을 On-demand 템플릿 앱에 연결하여 구성됩니다. 선택 앱과 템플릿 앱은 On-demand 앱의 기본 핵심 구성 요소입니다.

On-demand 앱을 만들려면 서로 연결될 수 있는 선택 및 템플릿 앱을 먼저 만들어야 합니다. 연결하려면 선택 및 템플릿 앱에 공통적으로 함께 바인딩할 수 있는 데이터 필드가 있어야 합니다.

하나의 선택 앱은 여러 개의 템플릿 앱에 연결될 수 있으며 하나의 템플릿 앱은 여러 개의 선택 앱에 연결될 수 있습니다. 그러나 템플릿 앱의 데이터 바인딩 표현식은 연결된 선택 앱의 필드와 대응해야 합니다.

On-demand 앱 탐색 링크는 선택 앱을 템플릿 앱에 조인합니다. On-demand 앱 탐색 링크는 선택 앱에 만들어집니다. 탐색 링크가 정의되면 선택 앱의 **앱 탐색** 막대에 On-demand 앱 탐색 포인트로 추가될 수 있습니다. 앱의 각 시트에는 자체 **앱 탐색** 막대가 있습니다. 그런 다음 사용자는 앱 탐색 포인트에서 On-demand 앱을 생성합니다.

On-demand 앱 작성 보기의 예



선택한 각기 다른 데이터 조합이 포함된 여러 개의 On-demand 앱을 동일한 앱 탐색 포인트에서 생성할 수 있습니다.

단일 앱 탐색 링크에 대한 포인터를 동일한 선택 앱의 여러 시트에 추가할 수 있습니다. 또한 시트에는 여러 앱 탐색 링크에서 만들어진 여러 앱 탐색 포인트가 있을 수 있습니다.

탐색 링크와 탐색 포인트가 있는 선택 앱이 완성되면 On-demand를 생성할 수 있습니다.

다음과 같이 하십시오.

1. On-demand 선택 앱을 열고 **편집**을 선택합니다.
2. 왼쪽 패널에서 **앱 탐색 링크**를 선택합니다.

3. **새로 만들기** 단추를 클릭하여 **새 On-demand 앱 탐색 링크 만들기** 대화 상자를 엽니다.
4. 새 On-demand 앱 탐색 링크의 이름을 지정합니다.
5. **On-demand 템플릿 앱**을 선택합니다.
   템플릿 앱 드롭다운 목록의 모든 앱이 유효한 템플릿 앱은 아닙니다. 템플릿 앱으로 구성되고 데이터 바인딩 표현식이 작업 중인 선택 앱의 필드에 대응하는 앱을 선택해야 합니다. 그러지 않으면 앱 탐색 링크에서 생성된 On-demand 앱에 오류가 발생합니다.
6. 선택 앱에서 선택 상태를 통해 액세스할 수 있는 집계 레코드로 표시되는 총 세부 레코드 수를 계산하는 식을 작성합니다.
   이 표현식은 대개 **SUM** 함수를 사용하여 선택된 레코드의 총계를 가져옵니다. 그 결과는 로드할 데이터의 양이 On-demand 앱 생성에 대해 지정된 범위 내에 있는지 확인하는 데 사용됩니다.
7. **최대 행 개수**를 지정합니다.
   최대 행 개수 값은 On-demand 앱이 로드할 수 있는 레코드 수의 상한을 설정하며 이 값은 **표현식** 항목의 함수에 의해 계산됩니다. 선택 앱의 행 예측값 표현식으로 계산된 레코드 수가 **최대 행 개수** 값보다 큰 경우 On-demand 앱을 생성할 수 없습니다. 행 예측값 표현식으로 계산된 레코드 수가 **최대 행 개수** 값으로 설정된 상한 이하인 경우에만 앱을 생성할 수 있습니다.
   최대 행 개수에 사용되는 표현식을 만들려면 선택 앱의 사용 가능한 필드에서 전체 레코드 수가 계산되는 방식을 알아야 합니다.
8. **생성된 최대 앱 수**를 지정합니다.
   선택 앱의 **앱 탐색** 막대에 있는 동일한 On-demand 앱 탐색 포인트에서 여러 개의 On-demand 앱을 생성할 수 있습니다. 여러 개의 앱을 생성하는 이유는 각 앱마다 서로 다른 데이터 선택을 포함할 수 있기 때문입니다. 최대 앱 수가 생성되면 탐색 포인트에서 앱을 생성하는 사용자가 새로운 On-demand 앱을 생성하기 전에 기존 앱 중 하나를 삭제해야 합니다.
   생성된 최대 앱 수는 On-demand 앱 탐색 링크에 적용됩니다. 탐색 링크에서 하나의 On-demand 앱 탐색 포인트를 만든 경우 해당 탐색 포인트가 최대 수까지 만들 수 있습니다. 그러나 동일한 탐색 링크에서 여러 탐색 포인트를 만든 경우 이러한 탐색 포인트에서 생성된 On-demand 앱의 총 수가 **생성된 최대 앱 수**의 설정으로 제한됩니다.
9. **보존 기간** 필드에 탐색 링크에서 생성된 앱이 삭제되기 전에 보존되는 시간을 숫자 값으로 입력합니다.
10. **보존 기간** 필드의 오른쪽에 있는 드롭다운 메뉴에서 보존 기간의 시간 단위를 선택합니다.
    보존 시간 옵션은 시간, 일 또는 **만료되지 않음**입니다.
    탐색 링크에서 생성된 모든 On-demand 앱은 이 설정에 따라 보존됩니다. 생성된 On-demand 앱의 수명은 현재 시간과 마지막 데이터 로드 시간의 차이입니다. On-demand 앱의 수명에 대한 이 계산은 게시된 앱과 게시되지 않은 앱에서 동일합니다. On-demand 앱이 생성된 후 수동으로 게시되면 수명 계산은 그대로 유지됩니다. 생성된 앱의 마지막 데이터 로드를 기반으로 합니다.

    *또한 On-demand 앱 서비스에는 익명 사용자가 생성한 앱에 적용되는 보존 시간 설정이 있습니다. 이 설정은 고유한 ID로 로그인한 사용자의 보존 시간에는 영향을 미치지 않습니다. 익명 사용자가 생성한 앱의 경우 보존 시간은 탐색 링크 및 Qlik Management Console에서 설정되는 On-demand 앱 서비스 설정의 **보존 시간** 설정보다 짧습니다.*
11. **열 때 기본 보기** 드롭다운 메뉴에서, 탐색 링크에서 생성된 앱이 열릴 때 처음 표시할 시트를 선택합니다.
    탐색 링크가 만들어지는 선택 앱의 시트 중 하나 또는 **앱 개요**를 선택할 수 있습니다.
12. **게시** 드롭다운 메뉴에서 탐색 링크에서 생성된 앱이 게시될 스트림을 선택합니다.

선택한 스트림에 게시할 수 있는 권한이 있어야 합니다. 선택한 스트림에 대한 게시 권한이 없는 경우 탐색 링크에서 On-demand 앱을 생성하려는 시도가 실패합니다.
생성된 앱을 게시할 스트림을 선택할 때 On-demand 앱의 의도된 사용자에게 스트림에 대한 읽기 권한이 있는지 확인해야 합니다.
게시되지 않음(작업 영역에 저장됨)을 선택하여 생성된 앱을 게시하지 않고 사용자 작업 영역에 저장할 수도 있습니다.

익명 사용자가 게시된 선택 앱을 사용할 수 있는 경우 익명 사용자가 액세스할 수 있는 스트림에 게시하도록 On-demand 앱 탐색 링크를 구성해야 합니다. 탐색 링크에서 생성된 On-demand 앱이 자동으로 게시되지 않으면 익명 사용자가 해당 앱을 생성하려고 시도할 때 오류 메시지가 표시됩니다.

앱이 생성된 후에는 수동으로 게시할 수 있습니다.

13. **만들기**를 클릭하면 새 On-demand 앱 탐색 링크가 **앱 탐색 링크** 목록에 나타납니다.
14. 앱 탐색 링크를 선택 앱의 **앱 탐색** 막대로 끕니다.
    앱 탐색 링크를 선택 앱으로 끌면 On-demand 앱 탐색 포인트가 만들어집니다. 새 On-demand 앱 탐색 포인트의 속성이 오른쪽 패널에 표시됩니다. 원하는 경우 탐색 포인트의 이름을 변경할 수 있습니다.
15. 시트 편집기에서 **완료**를 클릭합니다.
    이제 On-demand 선택 앱이 사용 또는 게시할 준비가 되었습니다. 이제 On-demand 선택 앱이 사용 또는 게시할 준비가 되었습니다. 선택 앱 사용자는 선택 앱의 **앱 탐색** 막대에 있는 탐색 포인트에서 On-demand 앱을 생성할 수 있습니다.

7 동적 보기를 사용한 데이터 관리

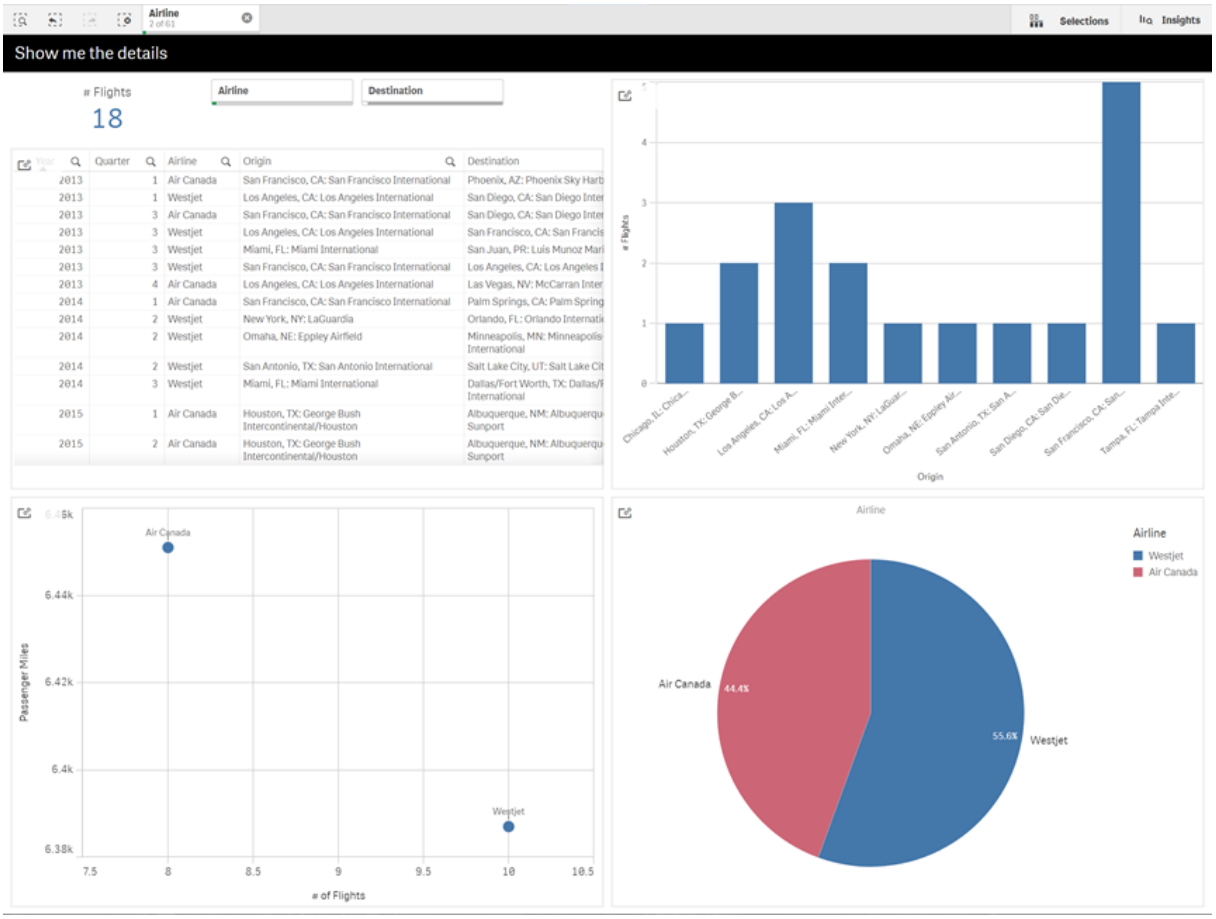
동적 보기에서 사용자는 탐색할 분석 소스 및 시각화의 데이터를 새로 고치는 시기를 직접 제어할 수 있습니다.

동적 보기를 사용하면 선택 시 동적으로 새로 고칠 수 있는 차트에서 대규모 데이터 집합의 관련 하위 집합을 쿼리하고 볼 수 있습니다. 이렇게 하면 많은 데이터 볼륨 또는 빠르게 변환하는 데이터 시나리오에 대한 시각화를 최신으로 유지할 수 있습니다.

7.1 동적 보기 개요

동적 보기를 사용하면 기본 앱을 다른 앱에 연결할 수 있습니다. 그런 다음 해당 앱의 마스터 시각화를 기본 앱에서 사용할 수 있습니다. 이렇게 하면 앱 작성자가 템플릿 앱의 마스터 시각화를 다른 앱에서 동적 차트로 사용할 수 있습니다. 기본 앱에 추가할 수 있는 동적 보기의 수에는 제한이 없습니다.

동적 보기 및 동적 차트를 사용한 시트 보기



동적 보기는 세 가지 기본 구성 요소로 이루어집니다.

- 동적 보기: 템플릿 앱에 연결되고 앱 작성자가 템플릿 앱에서 기본 앱으로 마스터 시각화를 추가할 수 있는 기본 앱에 추가된 메커니즘입니다.

- 동적 보기 템플릿 앱:클라우드 데이터베이스와 같은 데이터 소스에 대한 연결이 포함된 Qlik Sense 앱입니다.
- 동적 차트: 기본 앱에 추가할 수 있고 사용자가 수동으로 새로 고칠 수 있는 동적 보기 템플릿 앱의 마스터 시각화입니다.

템플릿 앱과 기본 앱이 동일한 데이터를 사용할 필요는 없습니다. 고객 구매와 관련된 데이터 집합이 있는 경우 날씨 데이터가 포함된 템플릿 앱에 동적 보기를 추가하여 모든 상관 관계를 확인할 수 있습니다.

템플릿 앱의 소스에서 쿼리한 데이터를 기본 앱의 값을 사용하여 필터링할 수 있는 경우 템플릿 앱의 스크립트에 바인딩 표현식을 사용할 수 있습니다. 이렇게 하면 동적 보기에서 템플릿 앱의 데이터 소스에서 기본 앱의 선택 내용과 특히 관련된 데이터의 하위 집합만 쿼리할 수 있습니다. 예를 들어 기본 앱의 SalesDate 필드를 템플릿 앱의 DailyTemperatureReadingDate 필드로 바인딩할 수 있습니다.

이 하위 집합 기능은 기본 앱에 집계 데이터가 포함되어 있고 동적 보기 데이터가 소스는 같지만 기본 앱 데이터보다 세밀한 경우에 유용합니다(예: 기본 앱에 월별 판매 및 제품 브랜드가 포함되어 있고 템플릿 앱에는 일별 판매 및 제품 이름이 포함된 경우). 템플릿 앱에 바인딩 표현식을 추가하는 방법에 대한 자세한 내용은 *On-demand 템플릿 앱의 바인딩 표현식 (page 176)*을 참조하십시오.

동적 보기는 모든 종류의 데이터에 사용할 수 있습니다. 동적 보기는 데이터베이스에서 데이터 집계를 수행하는 것이 더 좋은, 많은 데이터 볼륨 또는 빠르게 변화하는 데이터 시나리오에 특히 유용합니다. 이렇게 하면 데이터 소스에서 데이터 전송 지연을 방지할 수 있습니다.

동적 보기는 **자산** 패널에서 액세스할 수 있습니다. 동적 보기는 QMC에서 관리자가 활성화할 수 있습니다. 동적 보기 활성화에 대한 자세한 내용은 On-demand 앱 관리를 참조하십시오.

동적 보기는 On-demand 앱 생성과 유사합니다. 둘 다 템플릿 앱을 사용하여 On-demand 데이터를 제공하지만 동적 보기의 경우 전체 On-demand 앱을 생성하는 대신 시트에서 개별 차트를 사용할 수 있습니다. 또한 On-demand 앱을 사용하는 경우 On-demand 템플릿 앱을 사용하여 동적 보기를 만들 수 있습니다. On-demand 앱 생성에 대한 자세한 내용은 *On-demand 앱으로 빅 데이터 관리 (page 170)*를 참조하십시오.

동적 보기

동적 보기를 만드는 경우 템플릿 앱을 선택하고 선택적으로 행 제한 표현식을 적용하여 동적 보기에서 액세스 가능한 데이터 양을 제어합니다. 동적 보기를 만든 다음 템플릿 앱에서 마스터 시각화를 시트에 추가할 수 있습니다.

여러 동적 보기가 동일한 템플릿 앱을 사용할 수 있습니다. 각 동적 보기는 개별적으로 새로 고쳐집니다. 동적 보기의 템플릿 앱 스크립트에 바인드 표현식이 사용된 경우 기본 앱의 선택 내용으로 해당 템플릿 앱을 사용하는 각 동적 보기로 로드되는 데이터를 제어할 수 있습니다. 동일한 템플릿을 사용하는 두 개의 동적 보기를 사용하여 데이터가 세분화된 두 개별 하위 집합의 차트를 나란히 비교할 수 있습니다. 예를 들어 동일한 템플릿 앱을 사용하는 두 개의 동적 보기가 있는 경우 기본 앱 *SaledDate* 필드에서 2018년 1월 1일을 선택하고 하나의 동적 보기를 새로 고칠 수 있습니다. 그런 다음 2019년 1월 1일로 선택을 변경한 후 다른 동적 보기를 새로 고친 다음 동적 차트를 비교할 수 있습니다.

사용자가 동적 보기가 포함된 앱에 액세스하면 On-demand 앱이 해당 **작업**에 추가됩니다. 이 앱에는 현재 데이터가 포함된 동적 보기 템플릿 앱이 포함되어 있으며 기본 앱을 동적 보기로 채우는 데 이 앱이 사용됩니다. 동적 보기를 새로 고칠 때마다 새 버전으로 바뀝니다. 사용자가 동적 보기 템플릿 앱의 소유자가 아닌 경우 로드 스크립트가 제거됩니다. 이 앱은 마지막 새로 고침 후 24시간 내에 삭제됩니다.

동적 보기 만들기 및 편집에 대한 자세한 내용은 *동적 보기를 사용한 데이터 관리 (page 185)*를 참조하십시오.

동적 보기 사용에 대한 자세한 내용은 *동적 보기 및 차트 사용 (page 189)*을 참조하십시오.

동적 보기 템플릿 앱

동적 보기 템플릿 앱은 데이터 및 마스터 시각화를 사용하여 동적 보기를 제공하는 데 사용되는 Qlik Sense 앱입니다.

동적 보기 템플릿에는 기본 앱에서의 선택 내용을 기반으로 데이터 소스에서 만든 쿼리를 공식화하는 데 사용되는 데이터 바인딩 표현식을 포함한 로드 스크립트가 있을 수 있습니다. 바인딩 표현식은 일반적으로 사용자가 Qlik Sense 로드 스크립트 작성 경험이 있는 사용자가 만듭니다. 템플릿 앱에는 동적 차트를 활성화하는 동안 제공한 입력 매개 변수를 기반으로 하는 쿼리 필터 조건이 있을 수 있습니다.

동적 보기 템플릿 앱의 데이터 모델이 완료되면 마스터 시각화를 템플릿 앱에 추가할 수 있습니다. 그런 다음 동적 보기를 통해 이 마스터 시각화에 액세스하고 다른 앱에 동적 차트로 추가할 수 있습니다.

템플릿 앱 만들기에 대한 자세한 내용은 *On-demand 템플릿 앱 만들기 (page 173)*를 참조하십시오.

동적 차트

동적 차트는 동적 보기 템플릿 앱의 마스터 차트에서 파생됩니다. 동적 차트는 동적 보기를 사용하여 다른 앱의 시트에 추가할 수 있습니다. 다른 Qlik Sense 차트와 달리 사용자는 동적 보기의 소스 데이터가 차트의 새로 고침 옵션을 사용하여 새로 고쳐지는 시점을 제어할 수 있습니다. 동적 보기의 데이터가 바인드 표현식에서 제어되는 경우 Qlik Sense는 기본 앱 선택 상태를 추적합니다. 기본 앱의 선택 상태가 변경되어 새 값이 설정될 때마다 동적 보기의 각 차트에 오래된 데이터 아이콘이 나타나고 동적 보기의 바인딩된 필드는 보기의 마지막 새로 고침에 사용된 값과 더 이상 일치하지 않습니다.

동적 차트 사용에 대한 자세한 내용은 *동적 보기 및 차트 사용 (page 189)*을 참조하십시오.

7.2 동적 보기 제한

동적 보기에는 다음과 같은 제한 사항이 있습니다.

- 동적 보기는 스토리에는 지원되지 않습니다. 동적 차트의 스냅샷을 스토리에 추가할 수 있지만 동적 차트가 포함된 소스를 사용할 수 없습니다.
- 동적 보기는 Qlik NPrinting에는 지원되지 않습니다.
- 동적 보기는 대시보드 및 시각화 번들 확장을 지원합니다. 다른 확장은 지원하지 않습니다.
- 동적 보기 소유권은 앱 소유권에 따라 변경되지 않습니다.
- 동적 보기는 관리되는 공간의 앱에서 만들 수 없습니다.
- 동적 보기는 Qlik Sense Desktop에서 지원되지 않습니다.
- 동적 차트는 컨테이너에 배치할 수 없습니다.
- 격자 울타리 컨테이너 및 컨테이너는 동적 보기에서 마스터 차트로 지원되지 않습니다.
- 익명 사용자에게는 동적 보기가 지원되지 않습니다.

7.3 스트림 및 동적 보기

액세스 권한이 있는 모든 앱에 대한 동적 보기를 만들 수 있습니다. 게시된 앱에 대한 동적 보기는 스트림에서, 게시되지 않은 앱에 대한 동적 보기는 **작업**에서 만들 수 있습니다. 다른 사용자가 소유한 게시된 앱에 대한 동적 보기를 액세스 권한이 있는 스트림에서 만들 수도 있습니다.

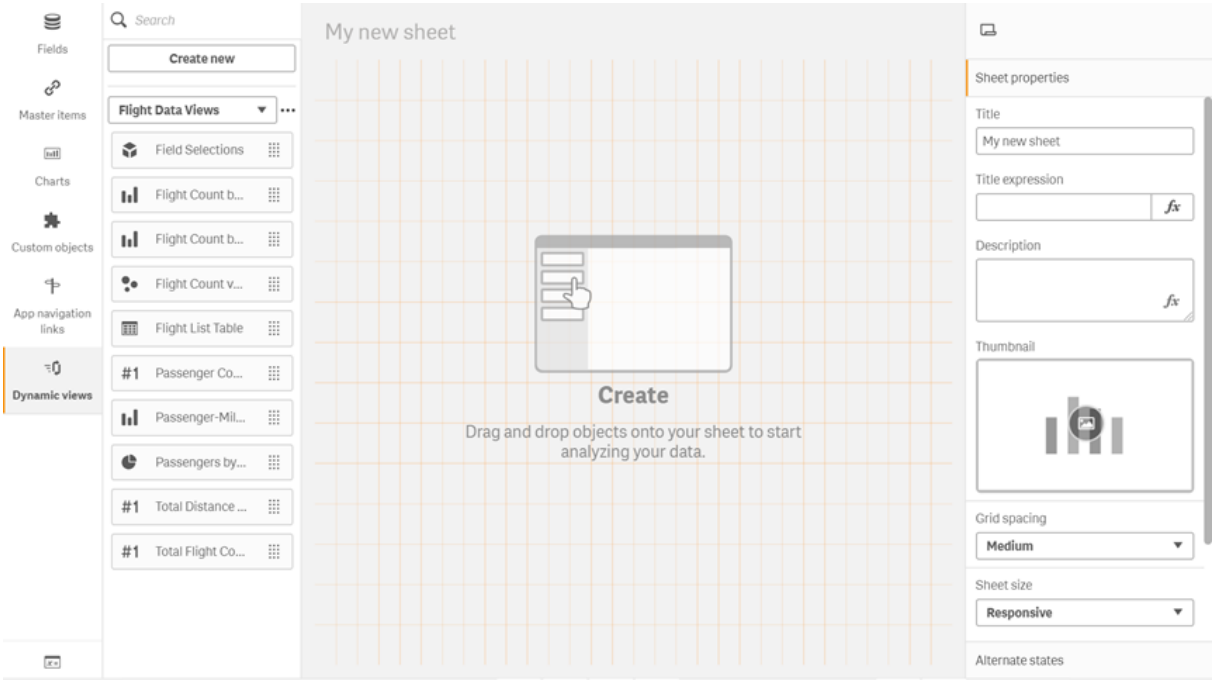
동적 보기가 있는 앱에 액세스할 수 있는 사용자는 템플릿 앱에 액세스할 수 없는 경우에도 동적 보기를 사용할 수 있습니다.

게시된 앱에는 동적 보기를 추가할 수 없습니다. 새로운 동적 보기를 추가하려면 앱을 복제하고 다시 게시하면 됩니다.

7.4 동적 보기 및 차트 만들기

동적 보기는 시트 보기의 **자산** 패널에서 추가할 수 있습니다. 동적 보기에서 동적 차트를 시트에 추가할 수 있습니다.

*새 시트의 **자산** 패널에서 동적 보기*



동적 보기 만들기

동적 보기를 만들 때 동적 보기의 일부로 가져올 행 수를 제한할 수 있습니다. 이렇게 하면 동적 보기가 아주 큰 데이터 집합에 연결될 경우 지나치게 많은 행이 반환되는 것을 막을 수 있습니다.

다음과 같이 하십시오.

1. 시트의 **편집** 모드에서 **동적 보기**를 클릭합니다.
2. **새로 만들기**를 클릭합니다.

3. **이름** 다음에 보기의 이름을 입력합니다.
4. **템플릿 앱** 다음에 템플릿 앱을 선택합니다.
5. **행 제한** 다음에 선택적으로 **행 제한 표현식**을 사용할지 선택하고 선택적으로 **최대 행 수**를 설정합니다.
6. **만들기**를 클릭합니다.

시트에 동적 차트 추가

동적 차트는 **자산** 패널에서 시트에 추가할 수 있습니다.

템플릿 앱에서 편집 권한이 있는 경우 동적 차트를 기반으로 하는 마스터 차트를 편집할 수 있습니다.

동적 차트를 편집하려면 시트가 **편집** 모드일 때 동적 차트를 선택하고 **소스 앱 편집**을 선택합니다.

동적 보기 편집

동적 보기를 편집하고 행 제한을 변경할 수 있습니다. 동적 보기에서 사용되는 템플릿 앱은 변경할 수 없습니다.

다음과 같이 하십시오.

1. 동적 보기 드롭다운에서 동적 보기를 선택합니다.
2. •••을 클릭하고 을 클릭합니다.

7.5 동적 보기 및 차트 사용

동적 차트는 다른 Qlik Sense 시각화와 상호 작용할 수 있습니다. 또한 사용자는 동적 보기에서 데이터를 새로 고칠 시점을 직접 제어할 수 있습니다. 동적 보기의 템플릿 앱이 해당 로드 스크립트에서 바인드 표현식을 사용하는 경우 동적 보기의 데이터를 새로 고치는 쿼리에는 일반적으로 기본 앱의 선택 상태에서 선택한 값을 사용하는 필터 조건이 포함됩니다. 사용자가 동적 보기를 새로 고치면 보기와 연결된 모든 동적 차트가 새 데이터로 업데이트됩니다.

스냅샷 찍기 및 탐색 메뉴 열기와 같은 차트 옵션의 컨텍스트 메뉴에 액세스하려면 차트를 마우스 오른쪽 버튼으로 클릭하고 **동적 차트**를 선택합니다.

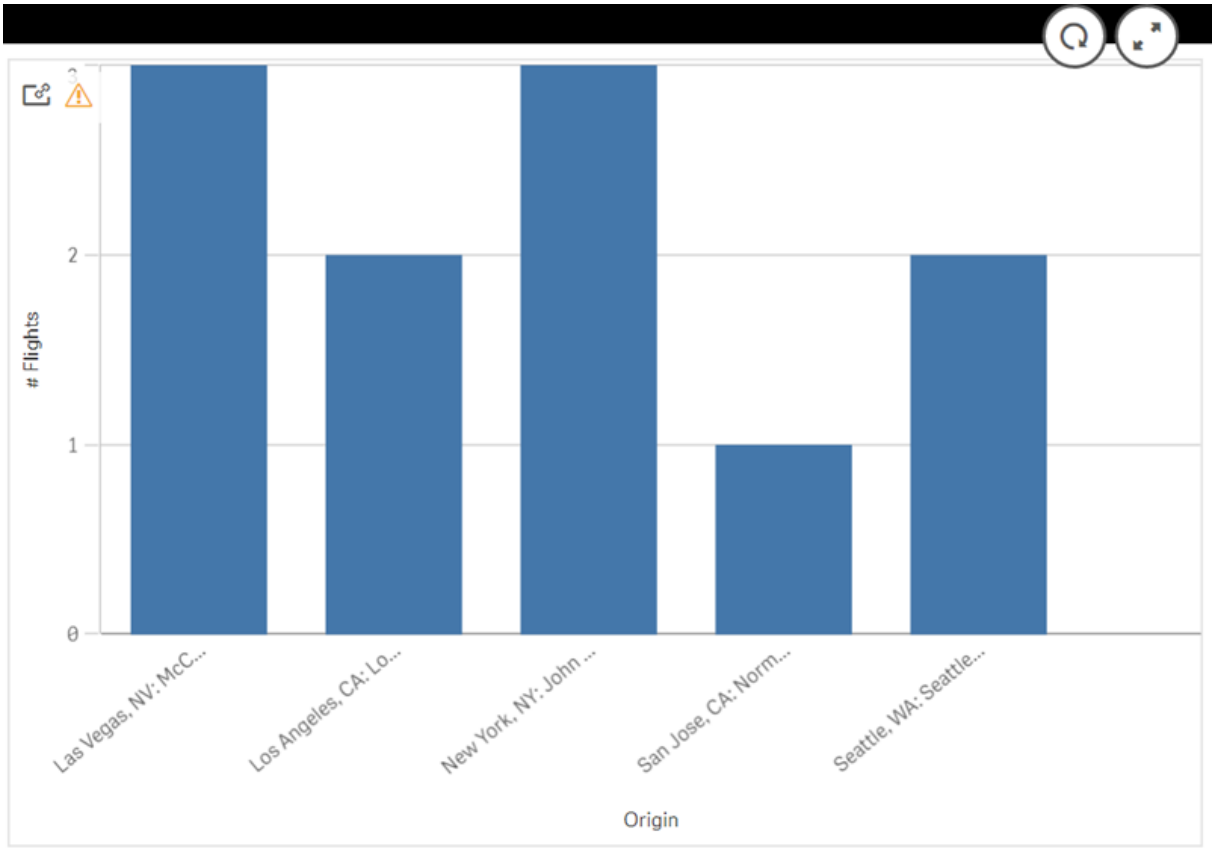
동적 차트를 통해 동적 보기와 상호 작용할 수 있습니다. 동적 차트를 마우스 오른쪽 버튼으로 클릭하고 **동적 보기**를 선택하면 차트, 보기 제약 조건을 새로 고치고 동적 보기를 마지막으로 새로 고치는 데 사용된 동적 보기의 바인딩 필드 값을 볼 수 있습니다.

동적 보기에서 선택

동적 차트는 표준 Qlik Sense 차트와 동일한 연관 선택 모델을 사용합니다. 동적 차트에서 선택하면 기본 앱에서의 선택에 영향을 미치지 않으며 기본 앱 선택 막대에 나타나지 않습니다. 동적 보기에서 동적 차트를 마우스 오른쪽으로 클릭하고 **동적 보기**를 선택한 다음 **선택 해제**를 선택하면 동적 보기 내의 선택을 지울 수 있습니다.

바인드 표현식이 동적 보기의 템플릿 앱에 있는 경우 기본 앱에서 선택한 내용은 다음 새로 고침 시 동적 보기에 로드되는 데이터에 영향을 미칩니다. Qlik Sense는 각 동적 보기의 마지막 새로 고침 시의 선택 상태를 추적합니다. 동적 보기가 마지막으로 새로 고쳐졌을 때부터 선택 작업으로 동적 보기의 바인딩된 필드에서 사용되는 값이 변경될 때마다 동적 보기의 데이터는 오래된 데이터로 간주됩니다. 데이터가 오래된 데이터로 간주될 때마다 동적 보기의 모든 차트에 오래된 데이터 아이콘이 표시됩니다. 오래된 차트를 새로 고치면 각 보기의 바인딩된 필드의 업데이트된 선택한 값 집합을 사용하여 동적 보기의 모든 차트가 새로 고쳐집니다. 동적 보기를 새로 고치는 대신 마지막으로 새로 고쳤을 때와 일치하도록 기본 앱의 선택 상태를 복원하려면 해당 동적 보기의 차트에서 컨텍스트 메뉴를 사용하여 **동적 보기**, **마지막 선택 표시**를 선택하고 **적용**을 클릭합니다.

오래된 데이터 알림이 있는 동적 차트



동적 보기의 제약 조건 중 하나라도 충족하지 못할 경우(예: 현재 선택으로 인해 행 수가 최대 행 수를 초과하는 경우) 해당 보기의 모든 차트에 데이터가 표시되지 않습니다. 기본 앱의 선택 상태가 변경되어 모든 제약 조건을 충족할 때까지 동적 보기를 새로 고칠 수 없습니다.

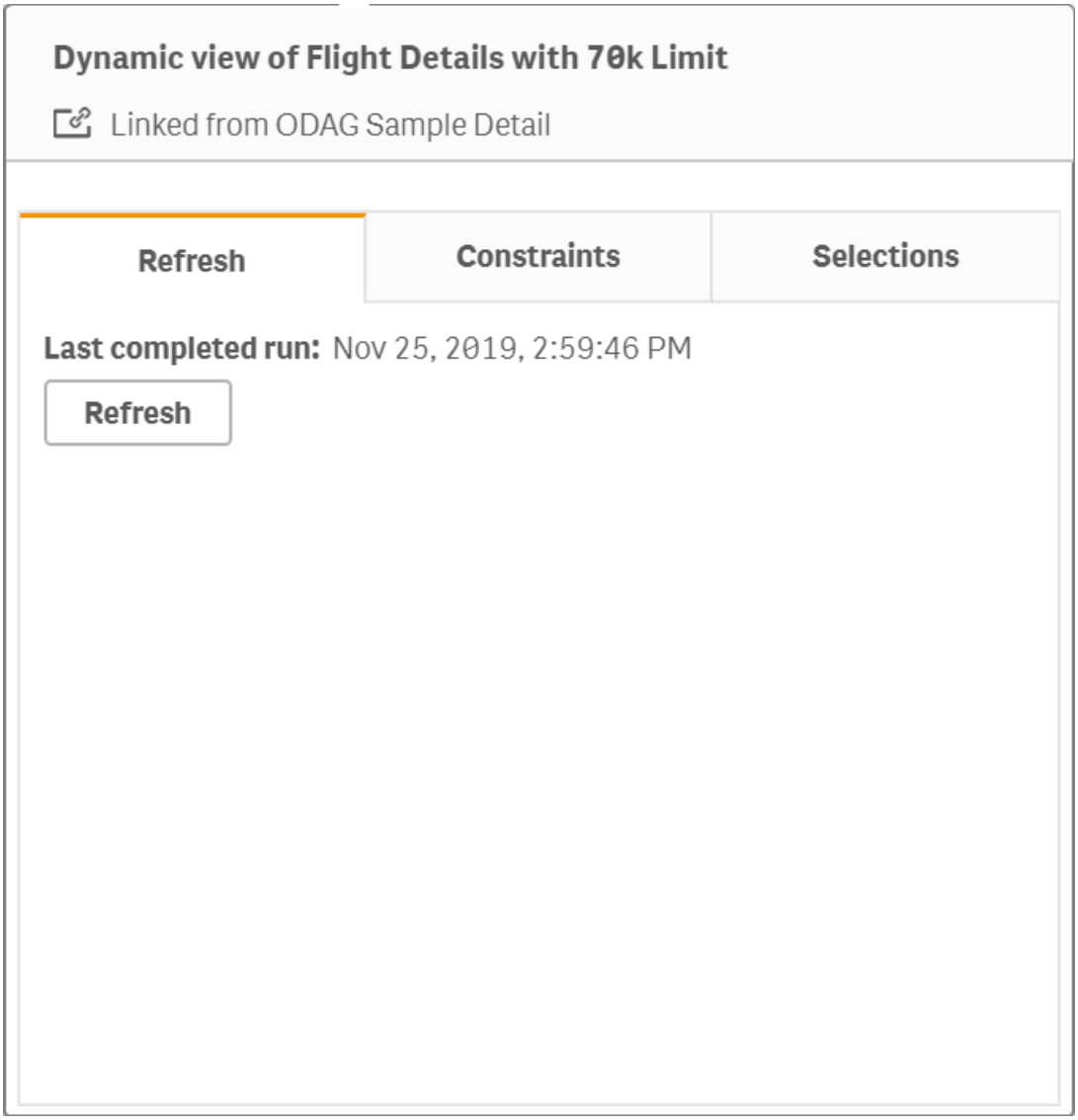
선택 내용이 제약 조건을 초과하는 동적 차트

동적 보기 세부 정보 보기

동적 보기 세부 정보에서 새로 고침 내역, 제약 조건 및 동적 보기에서 사용한 선택 항목을 볼 수 있습니다. 동적 차트의 모서리에 있는 **동적 보기 세부 정보** 아이콘을 클릭하면 세부 정보에 액세스할 수 있습니다.

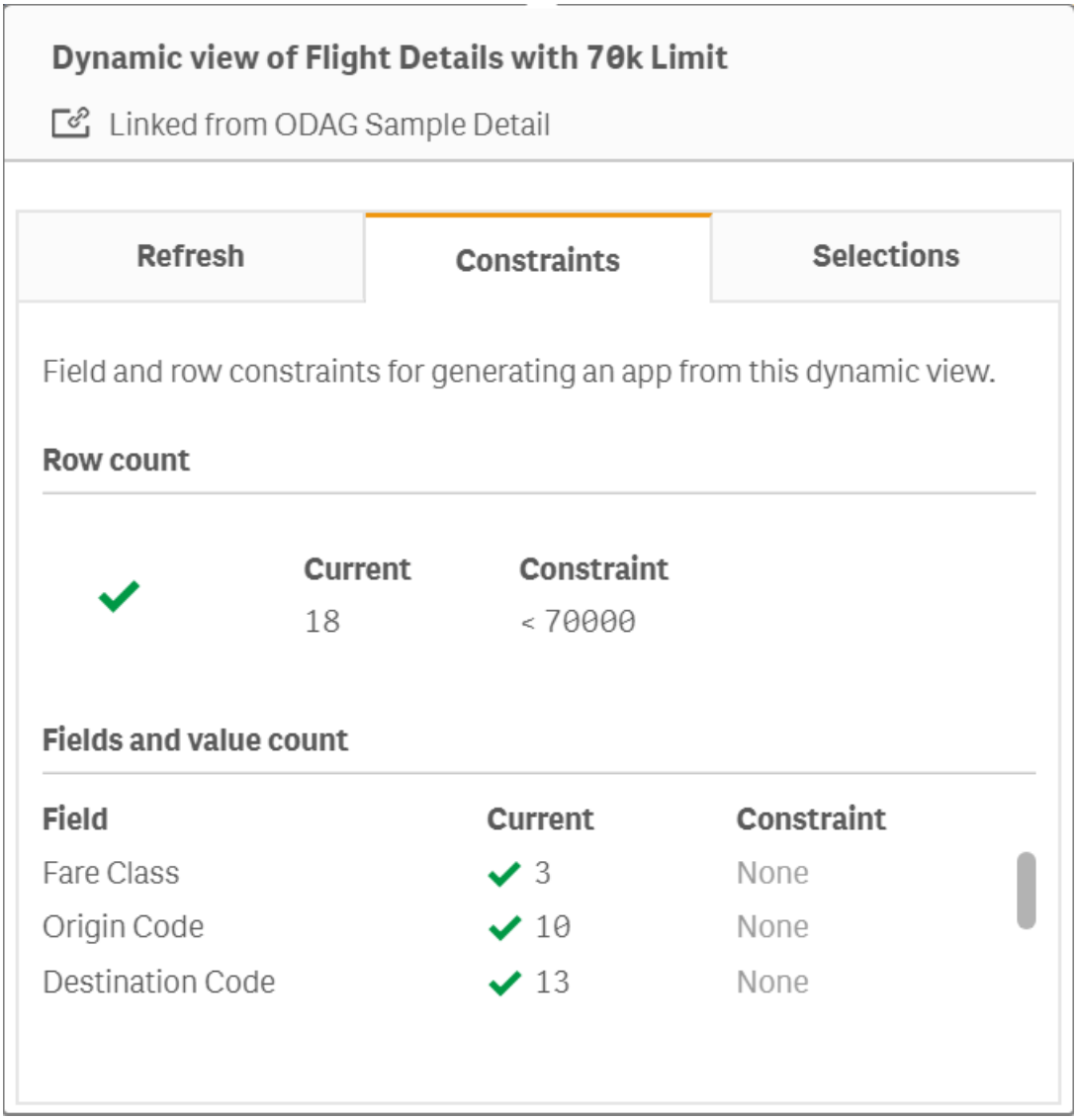
새로 고침은 동적 보기의 데이터가 마지막으로 새로 고쳐진 시간을 표시합니다.

동적 보기 세부 정보*에서 새로 고침 세부 정보*



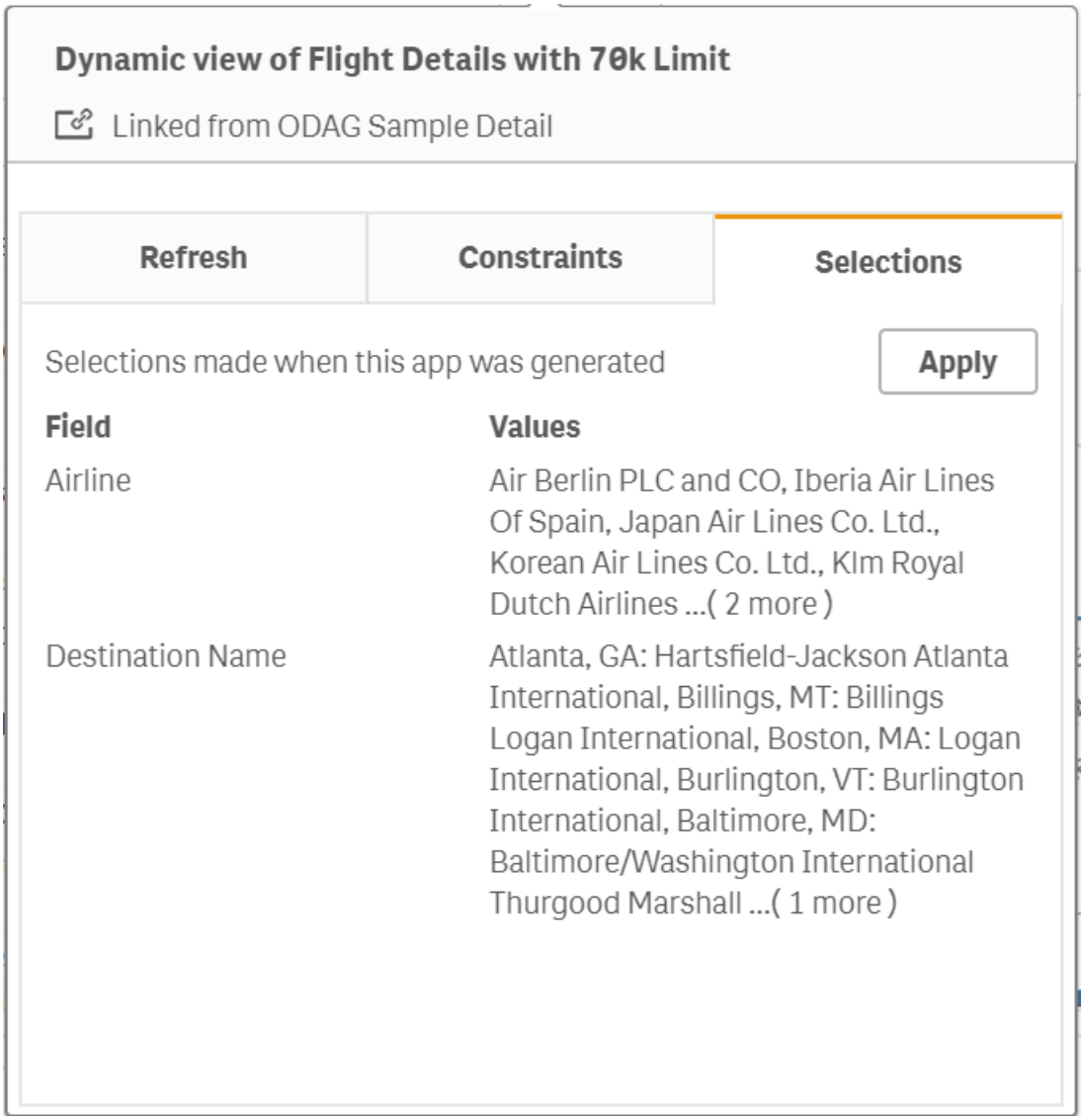
제약 조건은 동적 보기의 콘텐츠를 생성하는 데 적용된 필드 및 행 제약 조건을 표시합니다.

동적 보기 세부 정보*에서 현재 제약 조건*



선택은 이 보기의 데이터를 생성하는 데 적용된 선택을 표시합니다. 동적 보기의 데이터를 변경하도록 새로 선택하고 동적 보기를 아직 새로 고치지 않은 경우 **적용**을 클릭하여 동적 보기의 원래 선택을 복원할 수 있습니다

동적 보기 세부 정보*에서 현재 선택*



동적 보기 새로 고침

해당 보기에서 동적 차트를 선택하고 . 동적 차트에서 새로 고침 옵션을 사용하면 동일한 동적 보기의 모든 차트를 새로 고칠 수 있습니다. 새로 고치면 동적 보기의 동적 차트 내 모든 항목이 선택 해제됩니다.

8 데이터 소스에 연결

Qlik Sense을(를) 사용하면 데이터가 어디에 저장되어 있든 광범위한 Qlik 커넥터 및 기타 데이터 연결 유형을 사용하여 데이터에 연결할 수 있습니다.

데이터 연결을 만들면 Qlik Sense에 저장되므로 일반적으로 사용하는 데이터 소스에서 데이터를 신속하게 선택하고 로드할 수 있습니다. 데이터베이스, 소셜 미디어 데이터, 로컬 파일, 원격 파일 및 웹 파일에 연결합니다.

8.1 연결 만들기

데이터 소스에서 데이터를 선택하기 위해 새로운 데이터 연결을 만들거나 저장된 데이터 연결을 사용할 수 있습니다. 데이터 연결을 만들고 다음에서 저장된 연결에 액세스할 수 있습니다.

- 데이터 관리자의 **데이터 추가**.
  새로운 데이터를 앱에 신속하게 추가하고 연결을 만드는 데 도움을 받을 수 있습니다.
- 데이터 로드 편집기의 **데이터 연결**.
  새 데이터 연결 또는 기존 데이터 연결에서 데이터를 선택하거나 스크립트 편집기를 사용하여 데이터 연결에서 데이터를 로드합니다. 또한 기존 데이터 연결을 편집할 수 있습니다.

> *사용자가 소유한 데이터 연결 또는 읽거나 업데이트할 수 있는 액세스 권한이 있는 데이터 연결만 볼 수 있습니다. 필요한 경우 Qlik Sense 시스템 관리자에게 문의하여 액세스 권한을 요청하십시오.*

8.2 데이터 연결 유형

Qlik Sense를 사용하면 데이터가 어디에 있든 데이터에 액세스할 수 있습니다. 다음과 같은 데이터 연결 유형을 Qlik Sense에서 사용할 수 있습니다. Qlik다운로드 사이트에서 커넥터를 다운로드하여 Qlik Sense와 함께 사용할 수 있습니다.

이러한 데이터 소스에 액세스할 수 있는 많은 커넥터가 Qlik Sense에서 기본 제공되며, 다른 커넥터도 추가할 수 있습니다. 각 데이터 연결 유형에는 구성해야 하는 특정한 설정이 있습니다.

첨부된 파일

끌어서 놓기를 사용하여 데이터 파일을 앱에 직접 첨부합니다.

데이터베이스 커넥터

> *Qlik Sense Enterprise만 해당합니다.*

사전 구성된 ODBC 데이터베이스 커넥터를 사용하여 ODBC 데이터 소스에 연결합니다.

- Amazon Athena
- Amazon Redshift
- Apache Drill
- Apache Hive
- Apache Phoenix
- Apache Spark
- Azure SQL
- Azure Synapse
- Cloudera Impala
- Databricks
- Google BigQuery
- IBM DB2
- Microsoft SQL Server
- MongoDB
- MySQL Enterprise
- Oracle
- PostgreSQL
- Presto
- Snowflake
- Sybase ASE
- Teradata

Essbase

Essbase 데이터 셋에 연결합니다.

로컬 또는 네트워크 파일

Qlik Sense Enterprise만 해당합니다.

로컬 또는 네트워크 파일에 연결하여 데이터를 선택하고 로드합니다.

DSN을 통한 ODBC 연결

Qlik Sense Enterprise만 해당합니다.

ODBC을(를) 사용하여 데이터베이스 관리 시스템(DBMS)에 연결합니다. 해당하는 DBMS용 ODBC 드라이버를 설치하고 데이터 소스 DSN을(를) 만듭니다.

Qlik 웹 커넥터

소셜 미디어 또는 웹 기반 데이터 소스에 연결합니다. 별도의 설치가 필요합니다.

- Amazon S3
- AYLIEN News V2
- AYLIEN Text Analysis
- Azure Storage
- Bitly V2
- Box
- Dropbox
- Facebook Insights
- GitHub
- Google Ads
- Google Ad Manager
- Google AdWords
- Google Analytics
- Google Calendar
- Google Drive and Spreadsheets
- Google Search Console
- Helper Connector
- JIRA
- Mailbox IMAP
- Mailbox POP3
- MeaningCloud
- Microsoft Dynamics CRM V2
- OData
- Outlook 365
- Office 365 SharePoint
- RegEx Connector
- Repustate
- Sentiment140
- SMTP Connector
- Strava
- SugarCRM
- SurveyMonkey
- Watson Natural Language Understanding
- Twitter
- YouTube Analytics

REST

REST 데이터 소스에 연결합니다. REST 커넥터는 특정 REST 데이터 소스에 맞게 조정되지 않으며 REST API을(를) 통해 노출된 데이터 소스에 연결하는 데 사용할 수 있습니다.

Salesforce

Qlik Sense Enterprise만 해당합니다.

Salesforce.com 계정에 연결합니다.

SAP

Qlik Sense Enterprise만 해당합니다.

SAP NetWeaver에서 데이터를 로드합니다.

웹 파일

Qlik Sense Enterprise only

웹 URL에서 웹 기반 데이터 소스에 연결합니다.

웹 저장소 공급자 커넥터

웹 저장소 공급자의 파일 기반 데이터에 연결합니다.

- Dropbox
- Google Drive

타사 커넥터

타사 커넥터를 사용하면 Qlik Sense에서 직접 지원되지 않는 데이터 소스에 연결할 수 있습니다. 타사 커넥터는 QVX SDK을(를) 사용하여 개발하거나 타사 개발자에 의해 제공됩니다. 표준 Qlik Sense 설치에서는 타사 커넥터를 사용할 수 없습니다.

8.3 데이터 연결 저장 위치

연결은 Qlik Sense Repository Service에 의해 리포지토리 데이터베이스에 저장됩니다. Qlik Sense 서버 배포 시에는 Qlik Management Console를 사용하여 데이터 연결을 관리합니다. Qlik Management Console을 사용하면 데이터 연결을 삭제하고 액세스 권한을 설정하고 다른 시스템 관리 작업을 수행할 수 있습니다.

Qlik Sense Desktop에서 모든 연결은 암호화되지 않은 상태로 앱에 저장됩니다.

Qlik Sense Desktop 연결에 해당 연결을 만들 때 입력한 사용자 이름, 비밀번호 및 파일 경로에 대한 모든 세부 정보가 저장되기 때문에, 앱을 다른 사용자와 공유하는 경우 이러한 저장된 세부 정보를 일반 텍스트로 사용할 수 있게 됩니다. 앱을 공유하기 위해 설계하는 경우 이를 고려해야 합니다.

8.4 파일에서 데이터 로드

Qlik Sense에서는 다양한 형식의 파일에서 데이터를 읽을 수 있습니다.

파일 형식

지원되는 데이터 파일 형식에는 여러 가지가 있습니다.

- 텍스트 파일: 필드의 데이터는 쉼표, 탭 또는 세미콜론 등의 구분 기호로 구분되어야 합니다. 예: 쉼표로 구분된 변수(CSV) 파일.
- HTML 테이블
- Excel 파일:

  > *암호로 보호되는 Excel 파일 또는 Excel Binary Workbook 파일(.xlsb)의 데이터는 로드할 수 없습니다.*

  자세한 내용은 *Microsoft Excel 스프레드시트에서 데이터 로드 (page 201)*을 참조하십시오.
- XML 파일
- Qlik 네이티브 QVD 및 QVX 파일
- 고정 레코드 길이 파일
- Data Interchange Format(DIF) 파일: DIF 파일은 데이터 로드 편집기에서만 로드할 수 있습니다.

연결 유형

다양한 데이터 연결 유형에서 파일을 로드할 수 있습니다.

- 로컬 및 네트워크 파일 폴더: 자세한 내용은 *로컬 및 네트워크 파일 폴더에서 파일 로드 (page 200)*을 참조하십시오.
- **첨부된 파일** 폴더: 이 폴더는 삭제하거나 편집할 수 없습니다. 이 폴더에는 앱에 업로드 및 첨부된 파일이 포함되어 있습니다(Qlik Sense Desktop에서 사용할 수 없음). 자세한 내용은 *앱에 데이터 추가 (page 16)*을 참조하십시오.
- 웹상의 파일: *웹 리소스에서 파일 로드 (page 200)*

> *DataFiles 연결의 파일 확장명은 대/소문자를 구분합니다. 예를 들어 .qvd입니다.*

파일에서 데이터를 로드하는 방법

여러 가지 방법으로 파일에서 데이터를 로드할 수 있습니다.

> *공간에서 편집 권한이 있는 사용자는 해당 공간의 DataFiles를 읽거나 쓰거나 로드할 수 있습니다. 다른 사용자는 DataFiles를 볼 수 없습니다.*

데이터 로드 편집기에서 데이터 연결로부터 데이터 선택

데이터 연결로 이동하고 **데이터 선택** 대화 상자를 사용하여 로드할 데이터를 선택할 수 있습니다.

스크립트 코드를 작성하여 파일에서 데이터 로드

파일은 스크립트의 **LOAD** 문을 사용하여 로드됩니다. **LOAD** 문에는 전체 스크립트 표현식이 포함될 수 있습니다. 다른 Qlik Sense 앱에서 데이터를 읽으려면 **Binary** 문을 사용할 수 있습니다.

로컬 및 네트워크 파일 폴더에서 파일 로드

폴더 연결을 사용하여 로컬 및 네트워크 파일 폴더에서 파일을 로드할 수 있습니다.

데이터 연결 설정

UI 항목	설명
경로	데이터 파일이 포함된 폴더의 경로입니다. 다음 중 하나를 수행할 수 있습니다. 폴더를 선택하거나 유효한 로컬 경로를 입력하거나 UNC 경로를 입력합니다. 유효한 로컬 경로의 예: *C:\data\DataFiles* UNC 경로의 예: *\\myserver\filedir* *매핑된 네트워크 드라이브를 경로에 사용할 수 없습니다.*
Name	데이터 연결 이름입니다.

웹 리소스에서 파일 로드

웹 파일 데이터 연결을 사용하여 FTP, HTTP 또는 HTTPS 등의 웹 리소스에서 파일을 로드할 수 있습니다. Qlik Sense에서 지원하는 모든 유형의 파일을 사용할 수 있습니다.

웹 파일 데이터 연결 설정

UI 항목	설명
URL	프로토콜 식별자를 포함하여, 연결할 웹 파일에 대한 전체 URL입니다. **http://unstats.un.org/unsd/demographic/products/socind/Dec.%202012/1a.xls** FTP 파일에 연결하는 경우 URL의 사용자 이름 및 비밀번호 부분에 : 또는 @와 같은 특수 문자를 사용해야 할 수 있습니다. 이 경우, 특수 문자를 퍼센트 문자 및 문자의 ASCII 16진 코드로 바꿔야 합니다. 예를 들어, :을 '%3a'로, @를 '%40'으로 바꿔야 합니다.
이름	데이터 연결 이름입니다.

웹 파일 데이터 연결에 설정된 URL은 기본적으로 정적이지만 URL은 서식 사양 설정 **URL is**로 재정의할 수 있습니다. 이는 동적으로 만들어진 URL에서 데이터를 로드해야 하는 경우에 유용합니다.

동적으로 만들어진 URL에서 데이터 로드

이 예에서는 Qlik Community New to Qlik Sense 포럼의 처음 10페이지에서 포럼 게시물을 로드하려고 합니다. 포럼 페이지의 각 페이지에는 20개의 게시물이 포함되어 있으며 URL의 최종 매개 변수인 `start`는 페이지의 첫 번째 게시물로 표시할 게시물을 설정합니다. 다음 URL 예에서는 페이지에 게시물 번호 20으로 시작하는 게시물과 그 다음 20개의 게시물이 표시됩니다.

```
https://community.qlik.com/community/qlik-sense/new-to-qlik-
sense/content?filterID=contentstatus%5Bpublished%5D~objecttype~objecttype%5Bthread%5D&itemView
=detail&start=20
```

카운터 *i*를 사용하여 한 단계를 20으로 지정하여 180이 될 때까지 단계별로 페이지를 진행합니다. 즉, **For** 루프를 10번 실행합니다.

페이지를 로드하려면 시작 페이지를 **URL is** 설정에서 URL의 끝에 있는 $(i)로 바꿉니다.

```
For i = 0 to 180 step 20
LOAD
    Title1,
    "Author",
    F6 As Replies,
    Views,
    "Latest activity"
FROM [lib://x2]
(URL IS [https://community.qlik.com/community/qlik-sense/new-to-qlik-
sense/content?filterID=contentstatus%5Bpublished%5D~objecttype~objecttype%5Bthread%5D&itemView
=detail&start=$(i)], html, utf8, embedded labels, table is @1);
Next i;
```

그러면 제목, 작성자, 답글 수 및 보기 수, 최근 활동 시간과 함께 가장 최근의 포럼 게시물 200개가 테이블에 로드됩니다.

Microsoft Excel 스프레드시트에서 데이터 로드

Qlik Sense에서는 Microsoft Excel 스프레드시트에서 데이터를 읽을 수 있습니다. 지원되는 파일 형식은 *XLS*, *XLSX*, *XLW* 및 *XLSM*입니다. 데이터 관리자에서 데이터 추가를 사용하거나 데이터 로드 편집기에서 데이터를 선택할 수 있습니다. 두 경우 모두 스프레드시트 파일에서 시트의 이름 지정된 영역, 단일 시트, 선택된 시트 또는 모든 시트를 선택할 수 있습니다. 각 시트는 동일한 필드 구조를 가지고 있어서 하나의 테이블로 컨케트네이트되는 경우를 제외하고, 개별 테이블로 로드됩니다.

Microsoft Excel 스프레드시트를 로드하는 경우에는 해당 스프레드시트를 Qlik Sense 앱의 데이터 소스로 사용하고 있는 것입니다. 즉, Microsoft Excel 시트는 Qlik Sense에서 테이블이 되며, Qlik Sense 앱의 시트가 되지는 않습니다.

스프레드시트를 로드하기 전에 Microsoft Excel에서 몇몇 변경 작업을 수행하는 것이 유용할 때가 있습니다.

Microsoft Excel 시트에서 데이터 선택

Microsoft Excel 시트에서 데이터를 선택하는 경우 테이블 데이터를 올바르게 해석하도록 도와주는 설정이 있습니다.

테이블 데이터를 올바르게 해석하도록 도와주는 설정

UI 항목	설명
필드 이름	테이블에 **포함된 필드 이름**이 있는지, 또는 **필드 이름 없음**이 있는지 지정합니다. Excel 스프레드시트에서는 일반적으로 첫 행에 포함된 필드 이름이 있습니다. **필드 이름 없음**을 선택하면 필드 이름은 A,B,C...로 지정됩니다.
헤더 크기	테이블 헤더(일반적으로 일반 정보를 포함하며 열 서식으로 표시되지 않는 행)로서 생략할 행 수를 설정합니다.

예
스프레드시트가 다음과 같습니다.

스프레드시트 예

컴퓨터:	AEJ12B	-	-
날짜:	2015-10-05 09	-	-
타임스탬프	**순서**	**연산자**	**산출값**
2015-10-05 09:22	00122344	A	52
2015-10-05 10:31	00153534	A	67
2015-10-05 13:46	00747899	B	86

이 경우, 처음 두 줄을 무시하고 Timestamp, Order, Operator 및 Yield 필드가 있는 테이블을 로드하고자 할 수 있습니다. 이를 위해서는 다음 설정을 사용하십시오.

처음 두 줄을 무시하고 필드를 로드하도록 설정합니다.

UI 항목	설명
헤더 크기	2 처음 두 줄을 헤더 데이터로 간주하고 파일 로드 시 무시합니다. 이 경우, Machine: 및 Date:로 시작하는 두 줄은 테이블 데이터의 일부가 아니므로 무시됩니다.
필드 이름	**포함된 필드 이름** 읽은 첫 줄을 해당 열에 대한 필드 이름으로 사용합니다. 이 경우, 처음 두 줄이 헤더 데이터이므로 읽을 첫 줄은 세 번째 줄입니다.

Qlik Sense에서 로드하기 쉽도록 Microsoft Excel 스프레드시트 준비

Microsoft Excel 스프레드시트를 Qlik Sense에 로드하려는 경우 데이터 로드 스크립트에서 다양한 함수를 사용하여 데이터를 변환 및 정리할 수 있지만 Microsoft Excel 스프레드시트 파일에서 직접 소스 데이터를 준비하는 것이 더 편리할 수 있습니다. 이 섹션에서는 최소한의 스크립트 코딩으로 Qlik Sense에 로드할 수 있도록 스프레드시트를 준비하는 데 도움이 되는 몇 가지 팁을 제공합니다.

열 머리글 사용

Microsoft Excel에서 열 머리글을 사용하는 경우 Qlik Sense에서 데이터를 선택할 때 **포함된 필드 이름**을 선택하면 자동으로 필드 이름으로 사용됩니다. 또한 레이블에 줄 바꿈을 넣지 않고 머리글을 시트의 첫 번째 줄로 입력하는 것이 좋습니다.

데이터 서식 지정

내용이 원시 데이터 상태에서 테이블로 정렬되어 있다면 Microsoft Excel 파일을 Qlik Sense에 더 쉽게 로드할 수 있습니다. 다음과 같은 경우는 피하는 것이 좋습니다.

- 합계 및 카운트 등의 집계. 집계는 Qlik Sense에서 정의하고 계산할 수 있습니다.
- 중복 머리글.
- 주석 등과 같이 데이터의 일부가 아닌 추가 정보. 가장 좋은 방법은 파일을 Qlik Sense에 로드할 때 손쉽게 생략할 수 있는 주석 전용 열을 사용하는 것입니다.
- 교차 표 데이터 레이아웃. 예를 들어 각 달마다 하나의 열을 사용한다면, 대신 "달"이란 열을 지정하고 한 행에 한 달씩 12행에 동일한 데이터를 입력하는 것이 좋습니다. 그러면 항상 이를 Qlik Sense에서 교차 표 형식으로 볼 수 있습니다.
- 중간 머리글, 예를 들어 "A 부서"라고 지정된 줄 다음에 A 부서와 관련된 줄이 이어지는 경우. 대신 " 부서"라는 열을 만들고 여기에 해당 부서 이름을 기입하는 것이 좋습니다.
- 병합된 셀. 병합하는 대신 셀 값을 모든 셀에 나열하십시오.
- 값이 위 이전 값에서 유추되는 빈 셀. 반복되는 값이 있을 경우 빈 곳을 채워 모든 셀에 데이터 값이 포함되도록 해야 합니다.

영역에 이름 지정

시트의 일부만을 보려고 하는 경우, 열과 행의 영역을 선택하고 Microsoft Excel에서 여기에 이름을 지정할 수 있습니다. Qlik Sense는 시트에서뿐 아니라 이름이 지정된 영역에서도 데이터를 로드할 수 있습니다.

일반적으로, 원시 데이터를 이름이 지정된 영역으로 정의하고, 추가 주석과 범례는 모두 이름이 지정된 영역 밖에 유지할 수 있습니다. 그러면 더 쉽게 Qlik Sense에 데이터를 로드할 수 있습니다.

비밀번호 보호 제거

Qlik Sense에서는 비밀번호로 보호된 파일이 지원되지 않으므로 Qlik Sense에 로드하기 전에 스프레드시트에서 비밀번호 보호를 제거해야 합니다.

Excel Binary Workbook 파일(.xlsb) 로드

Excel Binary Workbook 파일(.xlsb)을 Qlik Sense에 직접 로드할 수 없습니다. 해결책은 ODBC 연결을 사용하는 것입니다.

8.5 데이터베이스에서 데이터 로드

다음 커넥터를 사용하여 상용 데이터베이스 시스템의 데이터를 Qlik Sense로 로드할 수 있습니다.

- 커넥터는 DSN 연결 없이, 사용 허가된 ODBC 드라이버를 통해 데이터베이스에서 데이터를 직접 로드할 수 있도록 특별히 개발되었습니다. 자세한 내용은 Qlik 커넥터: 데이터베이스를 참조하십시오.
- Microsoft ODBC 인터페이스 또는 OLE DB를 사용하는 커넥터입니다. Microsoft ODBC를 사용하려면, DBMS를 지원하는 드라이버를 설치해야 하며 Windows **제어판**의 **ODBC 데이터 원본 관리자**에서 데이터베이스를 ODBC 데이터 소스로 구성해야 합니다.

ODBC 데이터베이스에서 데이터 로드

두 가지 방법으로 데이터베이스에서 데이터를 로드할 수 있습니다.

Qlik 사용 허가된 ODBC 드라이버 중 하나를 통해 데이터베이스에 직접 연결하려면 Qlik 커넥터 도움말 사이트에서 Database 커넥터에 대한 지침을 참조하십시오.

자세한 내용은 ODBC 커넥터 패키지를 참조하십시오.

Qlik 사용권이 있는 ODBC 드라이버는 다음과 같은 데이터베이스를 지원합니다.

- Amazon Redshift
- Apache Drill
- Apache Hive
- Apache Phoenix
- Apache Spark
- Azure SQL
- Cloudera Impala
- Google BigQuery
- Microsoft SQL Server
- MongoDB
- MySQL Enterprise
- Oracle
- PostgreSQL
- Presto
- Sybase ASE
- Teradata

Microsoft ODBC 인터페이스를 사용하려면 다음을 수행합니다.

1. 액세스하려는 데이터베이스에 대한 ODBC 데이터 소스가 있어야 합니다. 이 데이터 소스는 Windows **제어판**의 **ODBC 데이터 원본 관리자**에서 구성됩니다. 아직 데이터 소스가 없다면 추가하고 Microsoft Access 데이터베이스 등을 가리키도록 구성해야 합니다.
2. 데이터 로드 편집기를 엽니다.

3. 1단계에서 설명한 ODBC 연결을 가리키는 **ODBC** 데이터 연결을 만듭니다.
4. 데이터 연결에서를 클릭하여 데이터 선택 대화 상자를 엽니다.

이제 데이터베이스에서 데이터를 선택하고 데이터를 로드하는 데 필요한 스크립트 코드를 삽입할 수 있습니다.

ODBC

ODBC 데이터 소스에서 데이터 로드

- 가장 일반적인 ODBC 소스를 지원하는 Qlik ODBC Connector Package의 Database 커넥터를 사용합니다. 이렇게 하면 Qlik Sense에서 Microsoft Windows **ODBC 데이터 원본 관리자**를 사용할 필요 없이 데이터 소스를 정의할 수 있습니다. ODBC Connector Package에서 Qlik 사용 허가된 ODBC 드라이버 중 하나를 통해 데이터베이스에 직접 연결하려면 Qlik Connectors 도움말 사이트에서 Database 커넥터에 대한 지침을 참조하십시오.
- 해당하는 DBMS용 ODBC 드라이버를 설치하고 데이터 소스 DSN을 만들 수 있습니다. 여기에 대해서는 이 섹션에서 설명합니다.

Qlik Sense를 사용하여 ODBC를 통해 DBMS(Database Management System)에 액세스할 수 있습니다.

> ***새 연결(ODBC) 만들기*** *대화 상자에 구성된* ***사용자 DSN*** *연결이 표시됩니다. Qlik Sense Desktop 사용 시 DSN 연결 목록에 ODBC Connector Package에 포함된 ODBC 드라이버가 표시됩니다. 이름에 첨부된 "Qlik-"으로 식별됩니다(예: Qlik-db2). 이러한 드라이버는 새 ODBC 연결을 만드는 데 사용할 수 없습니다. ODBC Connector Package에서 데이터베이스 커넥터에 의해 배타적으로 사용됩니다. ODBC Connector Package의 ODBC 드라이버는 서버 환경에서 Qlik Sense 사용 시 표시되지 않습니다.*

대체 옵션으로는 데이터베이스의 데이터를 Qlik Sense에서 읽을 수 있는 파일로 내보내는 방법이 있습니다.

일반적으로 일부 ODBC 드라이버는 Microsoft Windows와 함께 설치됩니다. 추가 드라이버는 소프트웨어 판매점에서 구입하거나, 인터넷에서 찾거나, DBMS 제조업체에서 제공받을 수 있습니다. 일부 드라이버는 자유롭게 재배포할 수 있습니다.

> *서버 환경에서 Microsoft Access Database 드라이버에 제한 사항이 있습니다. 문제를 방지하려면 SQL Server Express Edition을 사용합니다.*
> Microsoft Access Database Engine 2016 재배포 가능

여기서 설명하는 ODBC 인터페이스는 클라이언트 컴퓨터의 인터페이스입니다. ODBC를 사용하여 네트워크 서버에 있는 다중 사용자 관계형 데이터베이스에 액세스하려는 경우, 클라이언트가 서버의 데이터베이스에 액세스하는 데 사용되는 추가 DBMS 소프트웨어가 필요할 수 있습니다. 필요한 소프트웨어에 대한 자세한 내용은 DBMS 공급자에게 문의하시기 바랍니다.

ODBC 데이터 연결 설정

ODBC 데이터 연결 설정

UI 항목	설명
사용자 DSN **시스템 DSN**	연결할 DSN 유형을 선택합니다. **사용자 DSN** 소스의 경우, **32비트 연결 사용**을 통해 32비트 드라이버를 사용할지 여부를 지정해야 합니다. **시스템 DSN** 연결은 **32비트** 또는 **64비트**에 따라 필터링할 수 있습니다.
Single Sign-On	SAP HANA 데이터 소스에 연결하는 경우 Single Sign-On (SSO)을 사용할 수 있습니다. 이 옵션을 선택하지 않으면 **사용자 이름** 및 **비밀번호**로 자격 증명을 지정하지 않는 한 엔진 서비스 사용자 자격 증명이 사용됩니다. 이 옵션을 선택하면 엔진 서비스 사용자 또는 **사용자 이름** / **비밀번호** 자격 증명이 Windows 로그온에 사용되며 뒤이어 현재 사용자 자격 증명을 사용하여 SAML(SAP HANA) 에 대한 후속 로그온이 이루어집니다.
사용자 이름	데이터 소스에서 필요한 경우, 연결할 사용자 이름입니다. 엔진 서비스 사용자 자격 증명을 사용하려는 경우나 데이터 소스가 자격 증명을 필요로 하지 않는 경우에는 이 필드를 비워 둡니다.
비밀번호	데이터 소스에서 필요한 경우, 연결할 비밀번호입니다. 엔진 서비스 사용자 자격 증명을 사용하려는 경우나 데이터 소스가 자격 증명을 필요로 하지 않는 경우에는 이 필드를 비워 둡니다.
이름	데이터 연결 이름입니다.

ODBC 드라이버 추가

사용자의 데이터베이스에 액세스할 수 있도록 하려면 사용자 DBMS (DataBase Management System)에 적합한 ODBC 드라이버를 Qlik Sense에 설치해야 합니다. 사용 중인 DBMS에 대한 자세한 내용은 설명서를 참조하십시오.

사용자의 데이터베이스에 액세스할 수 있도록 하려면 사용자 DBMS에 적합한 ODBC 드라이버를 Qlik Sense에 설치해야 합니다. 이는 외부 소프트웨어입니다. 따라서 아래 지침은 모든 공급업체의 소프트웨어와 일치하지 않을 수 있습니다. 자세한 내용은 사용 중인 DBMS에 대한 설명서를 참조하십시오.

다음과 같이 하십시오.

1. **제어판**의 **관리 도구** 아이콘을 두 번 클릭합니다.
2. **데이터 소스(ODBC)** 아이콘을 두 번 클릭합니다.
   ODBC 데이터 소스 관리자 대화 상자가 나타납니다.
3. Qlik Sense에서 사용할 데이터베이스를 선택합니다.
4. **데이터 소스** 대화 상자에서 **드라이버** 탭을 선택합니다.

드라이버 탭에서 설치된 ODBC 드라이버 목록을 볼 수 있습니다. 해당 DBMS가 나열되지 않은 경우 이에 대한 드라이버를 설치해야 합니다. ODBC 드라이버와 함께 제공된 설치 프로그램을 실행합니다. 예를 들어 Microsoft ODBC 설치 프로그램입니다.

64비트 및 32비트 버전 ODBC 구성

Microsoft Windows 운영 체제의 64비트 버전에는 다음과 같은 버전의 Microsoft Open DataBase Connectivity (ODBC) Data Source Administrator 도구(*Odbcad32.exe):*가 포함되어 있습니다.

- *Odbcad32.exe* 파일의 32비트 버전은 *%systemdrive%\Windows\SysWOW64* 폴더에 있습니다.
- *Odbcad32.exe* 파일의 64비트 버전은 *%systemdrive%\Windows\System32* 폴더에 있습니다.

ODBC 데이터 소스 만들기

액세스하려는 데이터베이스에 대한 ODBC 데이터 소스를 만들어야 합니다. 이 작업은 ODBC 설치 도중 또는 이후 단계에서 완료할 수 있습니다.

> *데이터 소스 만들기를 시작하기 전에 데이터 소스를 **사용자 DSN**으로 할지, **시스템 DSN**(권장)으로 할지 결정해야 합니다. 올바른 사용자 자격 증명으로만 사용자 데이터 소스에 연결할 수 있습니다. 서버 설치에서는 일반적으로 다른 사용자와 데이터 소스를 공유할 수 있도록 시스템 데이터 소스를 만들어야 합니다.*

다음과 같이 하십시오.

1. *Odbcad32.exe*를 엽니다.
2. **시스템 DSN** 탭으로 이동하여 시스템 데이터 소스를 만듭니다.
3. **추가**를 클릭합니다.

   설치된 ODBC 드라이버 목록을 보여주는 **새 데이터 원본 만들기** 대화 상자가 나타납니다.
4. 올바른 ODBC 드라이버가 나열되는 경우 이를 선택하고 **마침**을 클릭합니다.

   선택한 데이터베이스 드라이버에 대한 대화 상자가 나타납니다.
5. **Microsoft Access Driver (*.mdb, *.accdb)**를 선택하고 **마침**을 클릭합니다.

   > *목록에서 이 드라이버를 찾을 수 없으면 Microsoft 다운로드 웹 사이트에서 다운로드하여 설치할 수 있습니다.*
6. 데이터 소스 이름을 지정하고 필요한 파라메타를 설정합니다.
7. 데이터 소스의 이름을 *스크립팅 자습서 ODBC*로 지정합니다.
8. **데이터베이스**: 아래에서 **선택...**을 클릭합니다.
9. **디렉터리** 아래에서 *Sales.accdb* 파일(자습서 예제 파일)의 위치로 이동합니다.
10. *Sales.accdb* 파일이 왼쪽의 텍스트 상자에 표시되어 있으면 클릭하여 데이터베이스 이름으로 만듭니다.
11. **확인**을 차례로 3번 클릭하여 모든 대화 상자를 닫습니다.
12. **확인**을 클릭합니다.

ODBC 데이터 연결 사용 시 모범 사례

ODBC 데이터 연결이 포함된 앱 이동

Qlik Sense 사이트/Qlik Sense Desktop 설치 간에 앱을 이동하는 경우, 데이터 연결이 포함됩니다. 앱에 ODBC 데이터 연결이 포함된 경우 새 배포에 관련된 ODBC 데이터 소스가 존재하는지 확인해야 합니다. ODBC 데이터 소스는 동일하게 이름을 지정하고 구성되어야 하며 동일한 데이터베이스 또는 파일을 가리켜야 합니다.

파일 기반 ODBC 데이터 연결에 연결하는 경우의 보안 측면

파일 기반 드라이버를 사용하는 ODBC 데이터 연결은 연결 문자열에 연결된 데이터 파일의 경로를 노출합니다. 연결을 편집할 때 데이터 선택 대화 상자 또는 특정 SQL 쿼리에서 경로가 노출될 수 있습니다.

이 문제가 염려되는 경우 가능하면 폴더 데이터 연결을 사용하여 데이터 파일에 연결하는 것이 좋습니다.

테이블에서 대규모 데이터 집합의 미리 보기 중지

대규모 데이터 집합이 있고 ODBC 데이터 소스를 **데이터 관리자** 또는 **데이터 로드 편집기**에 추가하는 동안 데이터를 표시하지 않으려면 Shift 키를 누른 채 ODBC 데이터 연결을 선택합니다.

OLE DB

Qlik Sense는 외부 데이터 소스 연결에 OLE DB (Object Linking and Embedding, Database) 인터페이스를 지원합니다. 대다수의 외부 데이터베이스가 OLE DB를 통해 액세스됩니다.

OLE DB 데이터 연결 설정

OLE DB 데이터 연결 설정

UI 항목	설명
공급자	사용 가능한 공급자 목록에서 **공급자**를 선택합니다. 새 연결을 만들 때만 사용할 수 있습니다.
데이터 소스	연결할 **데이터 소스**의 이름을 입력합니다. 이 이름은 서버 이름이거나, 경우에 따라 데이터베이스 파일의 경로일 수 있습니다. 이 이름은 사용 중인 **OLE DB** 공급자에 따라 달라집니다. 새 연결을 만들 때만 사용할 수 있습니다. Microsoft Office 12.0 Access Database Engine OLE DB Provider를 선택한 경우, 전체 파일 경로를 포함하여 Access 데이터베이스 파일의 이름을 입력합니다. *C:\Users\{user}\Documents\Qlik\Sense\Apps\Tutorial source files\Sales.accdb* *데이터 소스에 대한 연결에 실패하면 경고 메시지가 표시됩니다.*
연결 문자열	데이터 소스에 연결할 때 사용할 연결 문자열입니다. 이 연결 문자열에는 **공급자** 및 **데이터 소스**에 대한 참조가 포함되어 있습니다. 연결을 편집할 때만 사용할 수 있습니다.

UI 항목	설명
Windows 통합 보안	이 옵션을 사용하면 Qlik Sense 서비스를 실행하는 사용자의 기존 Windows 자격 증명을 사용할 수 있습니다.
특정 사용자 이름 및 비밀번호	이 옵션을 사용하면 데이터 소스 로그인 자격 증명에 대해 **사용자 이름** 및 **비밀번호**를 입력해야 합니다.
사용자 이름	데이터 소스에서 필요한 경우, 연결할 사용자 이름입니다. **Windows 통합 보안**을 사용하는 경우 또는 데이터 소스에서 자격 증명을 필요로 하지 않는 경우는 이 필드를 비워둡니다.
비밀번호	데이터 소스에서 필요한 경우, 연결할 비밀번호입니다. **Windows 통합 보안**을 사용하는 경우 또는 데이터 소스에서 자격 증명을 필요로 하지 않는 경우는 이 필드를 비워둡니다.
Load 데이터베이스 선택...	연결을 테스트하려면 **로드**를 클릭한 다음 **데이터베이스 선택...**을 클릭하여 데이터 연결에 사용할 데이터베이스를 선택합니다. *데이터 연결에서 데이터를 선택할 때 다른 사용 가능한 데이터 소스의 데이터베이스를 모두 사용할 수 있습니다.*
이름	데이터 연결 이름입니다.

파일 기반 OLE DB 데이터 연결에 연결하는 경우의 보안 측면

파일 기반 드라이버를 사용하는 OLE DB 데이터 연결은 연결 문자열에 연결된 데이터 파일의 경로를 노출합니다. 연결을 편집할 때 데이터 선택 대화 상자 또는 특정 SQL 쿼리에서 경로가 노출될 수 있습니다.

이 문제가 염려되는 경우 가능하면 폴더 데이터 연결을 사용하여 데이터 파일에 연결하는 것이 좋습니다.

테이블에서 대규모 데이터 집합의 미리 보기 중지

대규모 데이터 집합이 있고 OLE DB 데이터 소스를 **데이터 관리자** 또는 **데이터 로드 편집기**에 추가하는 동안 데이터를 표시하지 않으려면 Shift 키를 누른 채 OLE DB 데이터 연결을 선택합니다.

데이터베이스 내의 논리

데이터베이스 응용 프로그램의 여러 테이블을 Qlik Sense 논리에 동시에 포함할 수 있습니다. 필드가 둘 이상의 테이블에 존재할 경우 해당 테이블은 이 키 필드를 통해 논리적으로 연결됩니다.

값을 선택하면 해당 선택 내용과 호환되는 모든 값이 선택 사항으로 표시됩니다. 그 외 모든 값은 제외됨으로 표시됩니다.

여러 필드의 값을 선택할 경우 논리적 AND가 가정됩니다.

동일한 필드에서 여러 값을 선택할 경우 논리적 OR이 가정됩니다.

어떤 경우에는 필드 내에서 선택한 값을 논리적 AND로 설정할 수 있습니다.

8.6 를 사용하여 대규모 데이터 셋에 액세스Direct Discovery

Direct Discovery를 사용하면 간단한 스타 스키마 구조의 특정 SQL 소스에서 빅 데이터 셋을 로드하여 인 메모리 데이터와 결합할 수 있습니다.

인 메모리 및 Direct Discovery 데이터를 선택하여 Qlik Sense의 연결 색(녹색, 흰색, 회색)으로 데이터 셋 간의 연결을 확인할 수 있습니다. 이 접근 방법에는 많은 제한이 있지만 특정 시각화에서 두 데이터 셋 모두의 데이터를 함께 분석할 수 있습니다. 실시간 솔루션이 되도록 설계되지 않았습니다.

제한 사항을 극복할 수 있는 새로운 Direct Discovery 개발은 아직 이루어지지 않았습니다.

Qlik Sense On-demand 앱은 빅 데이터 소스를 로드하고 분석할 수 있는 보다 유연한 접근 방식을 제공합니다.

는 인 메모리 데이터와 더 큰 데이터 셋을 매끄럽게 연결하는 집계 쿼리를 통해 추가 소스 데이터에 대한 액세스를 제공함으로써 Qlik Sense 인 메모리 데이터 모델의 연관 기능을 확장합니다. Direct Discovery는 현업 사용자가 제한 없이 빅 데이터 소스 연관 분석을 수행할 수 있는 역량을 강화합니다. 인 메모리 및 Direct Discovery 데이터를 선택하여 Qlik Sense의 연결 색과 같은 녹색, 흰색, 회색으로 데이터 셋 간의 연결을 확인할 수 있습니다. 시각화는 두 데이터 셋의 데이터를 함께 분석할 수 있습니다.

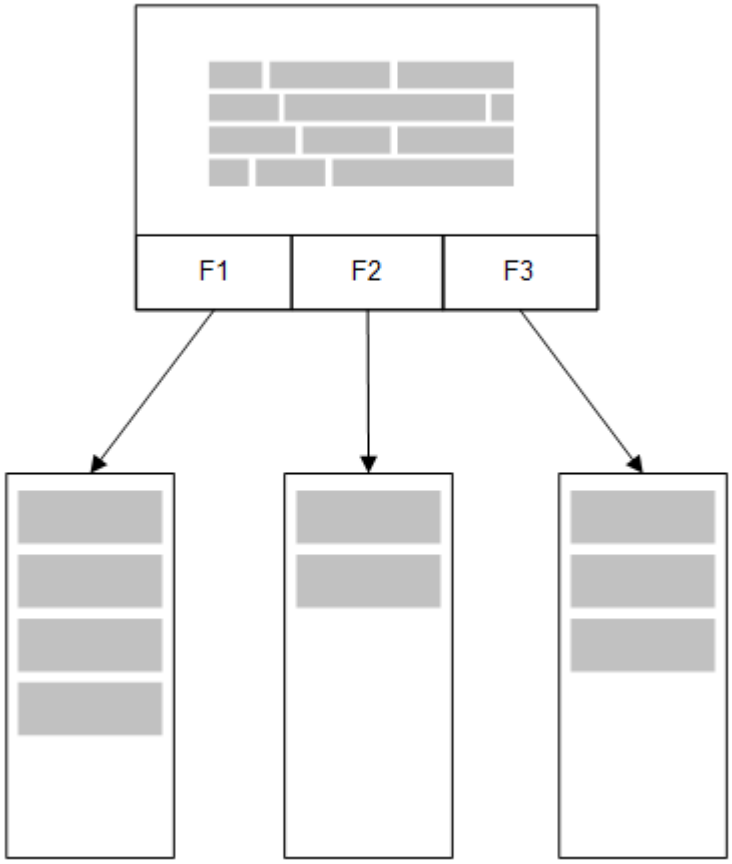
Direct Discovery에서 데이터는 특수 스크립트 구문인 **DIRECT QUERY**를 사용하여 선택됩니다. Direct Discovery 구조가 설정되면 인 메모리 데이터와 함께 Direct Discovery 필드를 사용하여 Qlik Sense 개체를 만들 수 있습니다. Qlik Sense 개체의 Direct Discovery 필드를 사용하면 외부 데이터 소스에 대해 SQL 쿼리가 자동으로 실행됩니다.

On-demand 앱은 대규모 데이터 셋에 액세스하기 위한 또 다른 방법을 제공합니다. Direct Discovery와 달리 On-demand 앱은 메모리에 호스팅된 잠재 하위 집합에서 완전한 Qlik Sense 기능을 제공합니다.

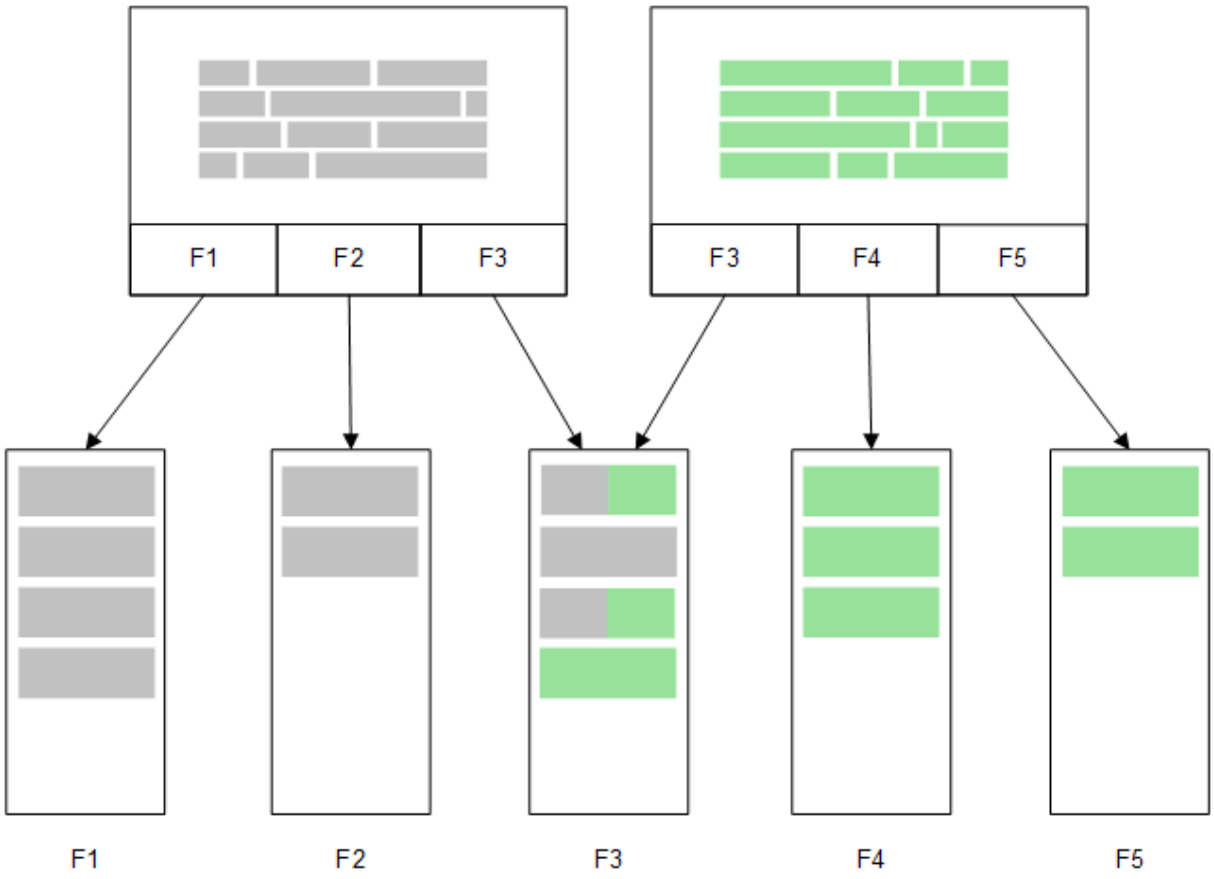
Direct Discovery와 인 메모리 데이터의 차이점

인 메모리 모델

Qlik Sense의 인 메모리 모델에서는 로드 스크립트의 테이블에서 선택된 필드의 모든 고유 값이 필드 구조로 로드되고 연관 데이터가 동시에 테이블로 로드됩니다. 필드 데이터 및 연관 데이터는 모두 메모리에 보관됩니다.

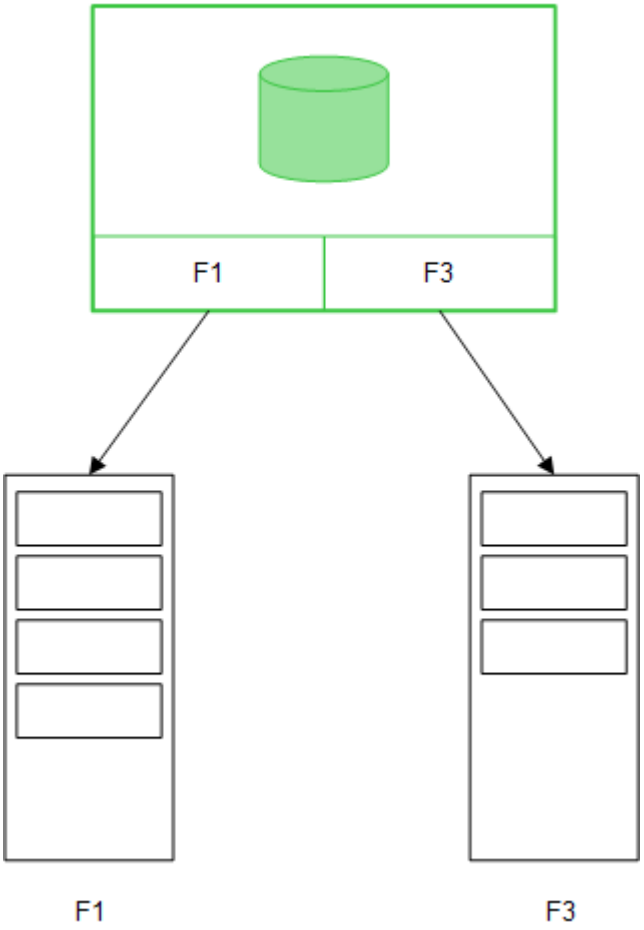


두 번째 관련 테이블이 메모리에 로드되면 공통 필드를 공유하게 되고, 해당 테이블이 새 고유 값을 공통 필드에 추가하거나 기존 값을 공유할 수 있습니다.

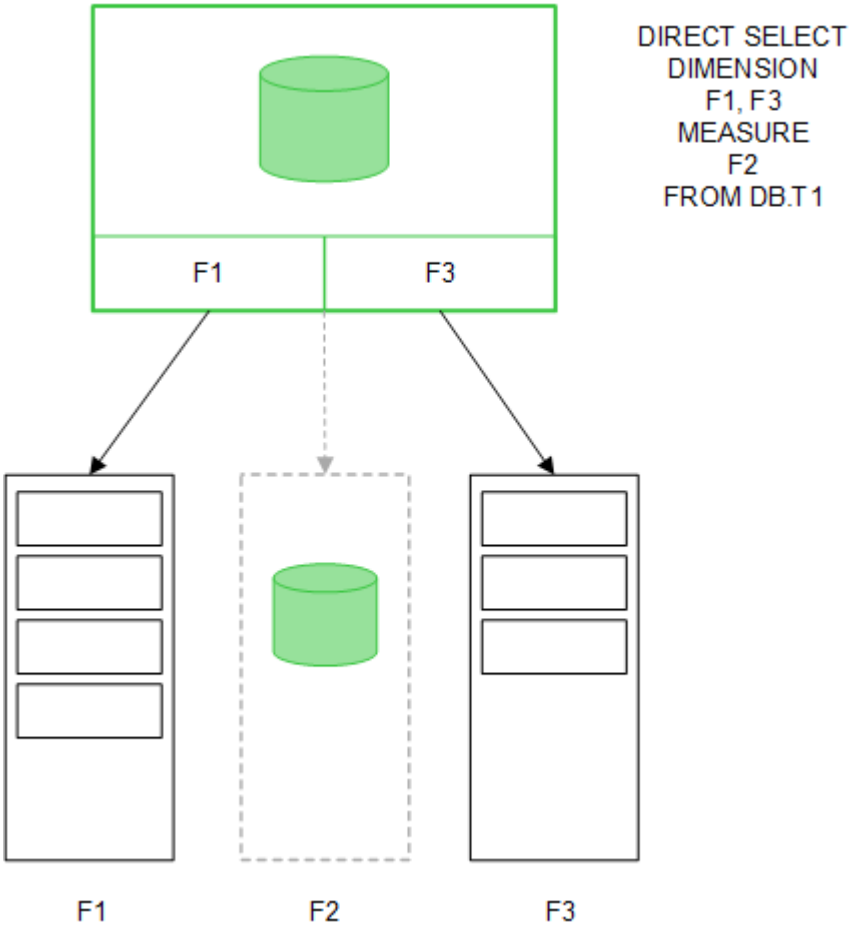


Direct Discovery

Direct Discovery **LOAD** 문(**Direct Query**)을 사용하여 테이블 필드를 로드할 경우 **DIMENSION** 필드만 있는 유사한 테이블이 생성됩니다. 인 메모리 필드와 마찬가지로 **DIMENSION** 필드의 고유 값이 메모리에 로드됩니다. 하지만 필드 간 연결은 데이터베이스에 남습니다.



MEASURE 필드 값도 데이터베이스에 남습니다.



Direct Discovery 구조가 설정되면 Direct Discovery 필드를 특정 시각화 개체와 함께 사용할 수 있고, 이를 인 메모리 필드와 연결하는 데 사용할 수도 있습니다. Direct Discovery 필드를 사용하면 Qlik Sense에서 외부 데이터에 대해 실행할 적절한 SQL 쿼리를 자동으로 만듭니다. 선택이 수행되면 Direct Discovery 필드의 연관 데이터 값이 데이터베이스 쿼리의 WHERE 조건에 사용됩니다.

선택을 수행할 때마다 Qlik Sense가 만든 SQL 쿼리를 실행하는 방식으로 소스 데이터베이스 테이블에서 계산이 수행되면서 Direct Discovery 필드가 있는 시각화가 다시 계산됩니다. 시각화를 다시 계산해야 하는 시기는 계산 조건 기능을 사용하여 지정할 수 있습니다. 지정된 조건이 충족될 때까지 Qlik Sense가 시각화를 다시 계산하기 위해 쿼리를 보내지 않습니다.

인 메모리 필드와 Direct Discovery 필드 간 성능 차이

인 메모리 처리 속도가 소스 데이터베이스의 처리 속도보다 항상 빠릅니다. Direct Discovery의 성능은 Direct Discovery 쿼리를 처리하는 데이터베이스를 실행 중인 시스템의 성능에 따라 달라집니다.

Direct Discovery의 경우 표준 데이터베이스 및 쿼리 조정 모범 사례를 사용할 수 있습니다. 모든 성능 조정은 소스 데이터베이스에서 수행되어야 합니다. Direct Discovery는 Qlik Sense 앱에서 쿼리 성능 조정을 지원하는 기능을 제공하지 않습니다. 하지만 연결 풀링 기능을 사용하여 데이터베이스에 대한 비동기, 병렬 호출을 수행할 수 있습니다. 풀링 기능을 설정하기 위한 로드 스크립트 구문은 다음과 같습니다.

```
SET DirectConnectionMax=10;
```

Qlik Sense의 캐싱도 전반적인 사용자 환경을 개선합니다. 아래 *캐싱 및 Direct Discovery (page 216)*을 참조하십시오.

Direct Discovery **DIMENSION** 필드의 경우 일부 필드의 연결을 분리하여 성능을 개선할 수도 있습니다. **DIRECT QUERY**에 **DETACH** 키워드를 사용하면 됩니다. 분리된 필드는 연결 쿼리 대상이 아니지만 여전히 필터의 일부이므로 선택 시간이 단축됩니다.

Qlik Sense 인 메모리 필드 및 Direct Discovery **DIMENSION** 필드는 둘 다 모든 데이터를 메모리에 보관하지만 이러한 필드가 로드되는 방식에 따라 메모리의 로드 속도가 달라집니다. Qlik Sense 인 메모리 필드는 동일한 값을 가진 필드가 여러 개 있을 경우 한 필드 값의 사본만 보관합니다. 하지만 모든 필드 데이터가 로드된 후 중복 데이터가 선별됩니다.

또한 **DIMENSION** 필드는 한 필드 값의 사본만 저장하지만 중복 값이 데이터베이스에서 선별된 후 메모리에 로드됩니다. 일반적으로 Direct Discovery를 사용할 때처럼 대량의 데이터를 처리할 때는 인 메모리 필드에 사용되는 **SQL SELECT** 로드를 사용하는 것보다 **DIRECT QUERY** 로드로 로드하는 것이 데이터 로드 속도가 훨씬 더 빠릅니다.

인 메모리 데이터와 데이터베이스 데이터의 차이점

DIRECT QUERY는 인 메모리 데이터와 연결할 때 대/소문자를 구분합니다. Direct Discovery는 쿼리된 데이터베이스 필드의 대/소문자 구분 여부에 따라 소스 데이터베이스에서 데이터를 선택합니다. 데이터베이스 필드가 대/소문자를 구분하지 않을 경우 Direct Discovery 쿼리에서 인 메모리 쿼리가 반환하지 못하는 데이터가 반환될 수 있습니다. 예를 들어 대/소문자를 구분하지 않는 데이터베이스에 다음 데이터가 있을 경우 Direct Discovery의 "Red" 값 쿼리는 4개 행을 모두 반환합니다.

예 테이블

ColumnA	ColumnB
red	one
Red	two
rED	three
RED	four

반면에 인 메모리에서 "Red,"를 선택하면 다음 값만 반환됩니다.

```
Red two
```

Qlik Sense는 데이터베이스와 일치하지 않지만 선택한 데이터와는 일치하는 항목이 반환되는 수준으로 데이터를 정규화합니다. 따라서 인 메모리 쿼리는 Direct Discovery 쿼리보다 일치 값을 더 많이 생성할 수 있습니다. 예를 들어 다음 테이블에서 숫자 "1"에 대한 값은 숫자 주변의 공백 위치에 따라 다릅니다.

주변 공백의 위치가 다르기 때문에 숫자 "1"에 대해 다른 값을 가진 테이블

ColumnA	ColumnB
' 1'	space_before
'1'	no_space

ColumnA	ColumnB
'1 '	space_after
'2'	two

표준 Qlik Sense 인 메모리에 데이터가 있는 **필터 창**에서 ColumnA에 대해 "1"을 선택하면 처음 3개 행이 연결됩니다.

Associated rows

ColumnA	ColumnB
' 1'	space_before
'1'	no_space
'1 '	space_after

필터 창에 Direct Discovery 데이터가 포함된 경우 "1"을 선택하면 "no_space"만 연결될 수 있습니다. Direct Discovery 데이터에 대해 반환되는 일치 항목은 데이터베이스에 따라 다릅니다. 일부는 "no_space"만 반환하며, SQL Server 등 일부는 "no_space" 및 "space_after"를 반환합니다.

캐싱 및 Direct Discovery

Qlik Sense의 캐싱은 쿼리의 선택 상태 및 연관된 쿼리 결과를 메모리에 저장합니다. 동일한 유형의 선택이 수행되면 Qlik Sense는 소스 데이터를 쿼리하는 대신 캐시에 저장된 쿼리를 활용합니다. 다른 선택이 수행되면 데이터 소스에 대해 SQL 쿼리가 수행됩니다. 캐싱된 결과는 사용자 간에 공유됩니다.

1. 사용자가 초기 선택을 적용합니다.
   SQL이 기본 데이터 소스로 전달됩니다.
2. 사용자가 선택을 해제하고 초기 선택과 동일한 선택을 적용합니다.
   캐시 결과가 반환되며, SQL은 기본 데이터 소스로 전달되지 않습니다.
3. 사용자가 다른 선택을 적용합니다.
   SQL이 기본 데이터 소스로 전달됩니다.

제한 시간은 **DirectCacheSeconds** 시스템 변수로 캐싱에 설정할 수 있습니다. 제한 시간에 도달하면 Qlik Sense가 이전 선택 내용에 대해 생성된 Direct Discovery 쿼리 결과에 대한 캐시를 지웁니다. 그런 다음 Qlik Sense는 선택 내용에 대해 소스 데이터를 쿼리하고 지정된 제한 시간 동안 캐시를 다시 만듭니다.

DirectCacheSeconds 시스템 변수를 사용하지 않을 경우 Direct Discovery 쿼리 결과의 기본 캐시 시간은 30분입니다.

Direct Discovery 필드 유형

Direct Discovery에는 DIMENSION, MEASURE 및 DETAIL의 세 가지 데이터 필드 유형이 있습니다. 로드 스크립트에서 **Direct Query** 문을 사용하여 Direct Discovery 선택을 수행할 때 데이터 필드에 유형이 설정됩니다.

모든 Direct Discovery 필드는 인 메모리 필드와 조합하여 사용할 수 있습니다. 일반적으로 차원으로 사용될 불연속 값이 있는 필드는 DIMENSION 키워드로 로드해야 하며, 집계에만 사용될 숫자 데이터는 MEASURE 필드로 표시해야 합니다.

아래 테이블에는 Direct Discovery 필드 유형의 특성과 사용법이 요약되어 있습니다.

Direct Discovery 필드 유형

필드 유형	인 메모리?	연결 형성?	차트 표현식에 사용 가능?
DIMENSION	예	예	예
MEASURE	아니요	아니요	예
DETAIL	아니요	아니요	아니요

DIMENSION 필드

DIMENSION 필드는 메모리에 로드되며, 인 메모리 데이터와 Direct Discovery 필드 내의 데이터 사이에 연결을 만드는 데 사용할 수 있습니다. Direct DiscoveryDIMENSION 필드는 차트 내에서 차원 값을 정의하는 데에도 사용됩니다.

MEASURE 필드

반면 MEASURE 필드는 "메타 수준"에서 인식됩니다. MEASURE 필드는 메모리에 로드되지 않습니다(데이터 모델 뷰어에 표시되지 않음). 인 메모리가 아니라 데이터베이스 내에서 MEASURE 필드에 있는 데이터의 집계가 가능하도록 하는 것이 목적입니다. 그렇지만 표현식에서 표현식 구문을 변경하지 않고 MEASURE 필드를 사용할 수 있습니다. 따라서 최종 사용자가 데이터베이스의 Direct Discovery 필드 사용을 쉽게 알 수 있습니다.

MEASURE 필드와 함께 사용할 수 있는 집계 함수는 다음과 같습니다.

- **Sum**
- **Avg**
- **Count**
- **Min**
- **Max**

DETAIL 필드

DETAIL 필드는 사용자가 차트 표현식에 사용하지는 않지만 표시하려고 하는 정보 또는 세부 사항을 제공합니다. DETAIL로 지정된 필드에는 일반적으로 주석과 같이 의미 있는 방식으로 집계할 수 없는 데이터가 포함됩니다.

어느 필드나 DETAIL 필드로 지정할 수 있습니다.

에서 지원하는 데이터 소스Direct Discovery

Qlik SenseDirect Discovery는 32비트 및 64비트 연결 모두에서 다음 데이터 소스에 대해 사용할 수 있습니다.

- ODBC/OLEDB 데이터 소스 - SQL Server, Teradata 및 Oracle을 포함한 모든 ODBC/OLEDB 소스 지원
- SQL을 지원하는 커넥터 - SAP SQL 커넥터, SQL 호환 데이터 저장소용 사용자 지정 QVX 커넥터

32비트 및 64비트 연결이 모두 지원됩니다.

SAP

SAP의 경우 Direct Discovery는 Qlik SAP SQL Connector와 함께만 사용할 수 있으며 **SET** 변수에 다음 파라메타가 필요합니다.

```
SET DirectFieldColumnDelimiter=' ';
SET DirectIdentifierQuoteChar=' ';
```

SAP는 쉼표가 아닌 공백으로 열을 구분하는 OpenSQL을 사용하므로 위 set 문은 ANSI SQL과 OpenSQL의 차이를 조정하기 위해 대체를 수행합니다.

Google Big Query

Direct Discovery는 Google Big Query와 함께 사용할 수 있으며, set 변수에 다음 파라메타가 필요합니다.

```
SET DirectDistinctSupport=false;
SET DirectIdentifierQuoteChar='[]';
SET DirectIdentifierQuoteStyle='big query'
```

Google Big Query는 **SELECT DISTINCT** 또는 인용된 열/테이블 이름을 지원하지 않으며, '[]'를 사용하는 ANSI 이외 인용 구성을 가집니다.

MySQL 그리고 Microsoft Access

Direct Discovery는 MySQL 및 Microsoft Access와 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 set 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='``';
```

DB2, Oracle 및 PostgreSQL

Direct Discovery는 DB2, Oracle 및 PostgreSQL와 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 set 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='"';
```

Sybase 그리고 Microsoft SQL Server

Direct Discovery는 Sybase와 Sybase 및 Microsoft SQL Server와 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 set 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='[]';
```

Apache Hive

Direct Discovery는 Apache Hive와 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 set 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='';
```

Cloudera Impala

Direct Discovery는 Cloudera Impala와 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 set 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='[]';
```

이 파라메타는 Qlik ODBC Connector Package에서 Cloudera Impala Connector를 사용할 때 필요합니다. DSN을 통해 ODBC를 사용할 때는 필요하지 않을 수도 있습니다.

를 사용할 때의 제한 사항Direct Discovery

제한 사항을 극복할 수 있는 새로운 Direct Discovery 개발은 아직 이루어지지 않았습니다.

지원되는 데이터 유형

일부 데이터 유형은 Direct Discovery에서 지원되지 않습니다. Qlik Sense에서 특정 소스 데이터 형식을 정의해야 하는 경우가 있습니다. 로드 스크립트에서 "SET Direct...Format" 구문을 사용하여 데이터 서식을 정의합니다. 다음 예는 Direct Discovery의 소스로 사용된 소스 데이터베이스의 날짜 서식을 정의하는 방법을 보여줍니다.

```
SET DirectDateFormat='YYYY-MM-DD';
```

또한 Direct Discovery에서 생성된 SQL 문의 현재 값의 서식을 지정하는 방법을 제어하는 데 사용되는 두 가지 스크립트 변수가 있습니다.

```
SET DirectMoneyFormat (default '#.0000')
SET DirectMoneyDecimalSep (default '.')
```

이 두 변수의 구문은 **MoneyFormat** 및 **MoneyDecimalSep**과 동일하지만 사용법에서 두 가지 중요한 차이점이 있습니다.

- 이는 표시 형식이 아니므로 통화 기호 또는 천 단위 구분 기호가 포함되지 않아야 합니다.
- 기본값은 로캘에 따라 결정되지 않지만 값에 묶여 있습니다. (로캘별 형식에는 통화 기호가 포함됩니다.)

Direct Discovery는 특히 SQL Server와 같은 일부 데이터베이스에서 요구하는 대로 확장 문자 문자열 리터럴(N'<확장 문자열>')에 대한 SQL 표준 형식을 사용하여 확장 유니코드 데이터의 선택을 지원할 수 있습니다. 이 구문은 스크립트 변수 **DirectUnicodeStrings**를 통해 Direct Discovery에서 사용할 수 있습니다. 이 변수를 "true"로 설정하면 문자열 리터럴 앞에 "N"을 사용할 수 있습니다.

보안

Direct Discovery를 사용할 때는 보안 모범 사례에 영향을 줄 수 있는 다음과 같은 사항을 고려해야 합니다.

- Direct Discovery 기능과 함께 동일한 앱을 사용하는 모든 사용자가 동일한 연결을 사용합니다. 인증 통과 및 사용자별 자격 증명이 지원되지 않습니다.
- 섹션 액세스는 서버 모드에서만 지원됩니다.
- 섹션 액세스는 집합 크기가 큰 조인에는 지원되지 않습니다.

- NATIVE 키워드 표현식으로 데이터베이스 내에서 사용자 지정 SQL 문을 실행할 수 있으므로 로드 스크립트 내의 데이터베이스 연결 설정에서 데이터베이스에 대해 읽기 전용 액세스 권한이 있는 계정을 사용해야 합니다.
- Direct Discovery에는 로깅 기능이 없지만 ODBC 추적 기능을 사용할 수 있습니다.
- 클라이언트의 요청으로 데이터베이스가 포화될 수 있습니다.
- 서버 로그 파일에서 자세한 오류 메시지를 확인할 수 있습니다.

지원되지 않은 Qlik Sense 기능

Direct Discovery의 대화형, SQL 구문 관련 특성으로 인해 다음과 같은 일부 기능이 지원되지 않습니다.

- 고급 계산(집합 분석, 복합 표현식)
- 계산 차원
- Direct Discovery 필드를 사용하는 개체에 대한 비교 분석(대체 상태)
- 스마트 검색에서는 Direct Discovery **MEASURE** 및 **DETAIL** 필드가 지원되지 않습니다.
- Direct Discovery **DETAIL** 필드의 검색
- Direct Discovery 테이블에 액세스하는 응용 프로그램에서 이진 로드
- Direct Discovery 테이블의 가상 키
- 스크립트 내의 테이블 이름 규칙은 Direct Discovery 테이블에 적용되지 않습니다.
- 로드 스크립트에서 **DIRECT QUERY** 키워드 뒤에 와일드카드 문자 * 사용

  ```
  (DIRECT QUERY *)
  ```

- LONG 데이터 유형 열이 있는 Oracle 데이터베이스 테이블
- 과학적 표기법에서 [-9007199254740990, 9007199254740991]의 범위를 벗어나는 정수(Big). 반올림 오차 및 불확실한 동작을 일으킬 수 있습니다.
- Snowflake 데이터베이스 스키마
- 데이터 관리자에서 데이터 준비
- Qlik Sense Enterprise SaaS
- Microsoft Excel로 다운로드
- 오프라인 모바일 iOS 앱
- 고급 분석 통합
- 확장
- Qlik GeoAnalytics
- 마스터 차원 및 측정값에 색상 할당
- Qlik Sense Enterprise on Windows 2017년 6월 및 이후 버전에 포함된 새 시각화
- 비SQL 소스 및 비SQL 문(예: SAP HANA에서 PLACEHOLDER 함수)
- 다음 커넥터는 지원되지 않습니다.
  - Qlik Salesforce 커넥터
  - Qlik REST 커넥터
  - Qlik 웹 커넥터
  - SAP NetWeaver에서 사용할 Qlik 커넥터
- Direct Discovery 쿼리에서 생성한 SQL 최적화

- 인 메모리 테이블과의 조합에서 집합 크기가 큰 조인은 데이터 소스의 SQL 버퍼 한도를 초과하는 큰 IN 절을 생성할 수 있습니다.
- Qlik Visualization 및 Dashboard 번들 개체
- 통찰력
- 알림
- 동적 보기
- 사용자 지정 도구 설명

에서 다중 테이블 지원Direct Discovery

Direct Discovery에서 ANSI SQL 조인 기능을 사용하여 둘 이상의 테이블 또는 보기를 로드할 수 있습니다.

단일 차트에서 모든 계수는 Qlik Sense의 동일한 논리 테이블에서 파생되어야 하지만 조인 문을 통해 연결된 소스에서 가져온 여러 테이블의 조합일 수 있습니다. 하지만 다른 테이블에서 가져온 차원을 동일한 차트에 사용할 수 있습니다.

예를 들어 Direct Discovery에 로드한 테이블을 **Where** 절 또는 **Join** 절을 사용하여 연결할 수 있습니다.

- Direct Discovery는 대규모 데이터 셋이 포함된 메모리 시나리오에서 단일 팩트/다중 차원으로 배포될 수 있습니다.
- Direct Discovery는 다음 중 하나의 기준에 부합하는 둘 이상의 테이블과 함께 사용할 수 있습니다.
    - 조인 내 키 필드의 집합 크기가 작습니다.
    - 조인 내 키 필드의 집합 크기가 크며, **DirectEnableSubquery**가 true로 설정되었고 모든 테이블이 Direct Discovery에 조인되었습니다.
- Direct Discovery는 모든 테이블이 Direct Discovery 형식이므로 3NF(제3정규형) 시나리오에 배포하기에는 적합하지 않습니다.

Where 절을 사용하여 Direct Discovery 테이블 연결

이 예제 스크립트에서는 AW2012 데이터베이스에서 데이터를 로드합니다. **Where** 절에서 공통 ProductSubCategoryID 필드를 사용하여 Product 및 ProductSubcategory 테이블을 연결합니다.

```
      Product_Join:
DIRECT QUERY
DIMENSION
   [ProductID],
   [AW2012].[Production].[Product].[Name] as [Product Name],
   [AW2012].[Production].[ProductSubcategory].[Name] as [Sub Category Name],
   Color,
   [AW2012].[Production].[Product].ProductSubcategoryID as [SubcategoryID]
MEASURE
   [ListPrice]
FROM [AW2012].[Production].[Product],
     [AW2012].[Production].[ProductSubcategory]
WHERE [AW2012].[Production].[Product].ProductSubcategoryID =
      [AW2012].[Production].[ProductSubcategory].ProductSubcategoryID   ;
```

Join On 절을 사용하여 Direct Discovery 테이블 연결

Join On 절을 사용하여 Direct Discovery 테이블을 연결할 수도 있습니다. 이 예제 문에서는 SalesOrderID 필드를 통해 SalesOrderDetail 테이블에 SalesOrderHeader 테이블을 조인하고, Customer ID 필드를 통해 SalesOrderHeader 테이블에 Customer 테이블을 조인합니다.

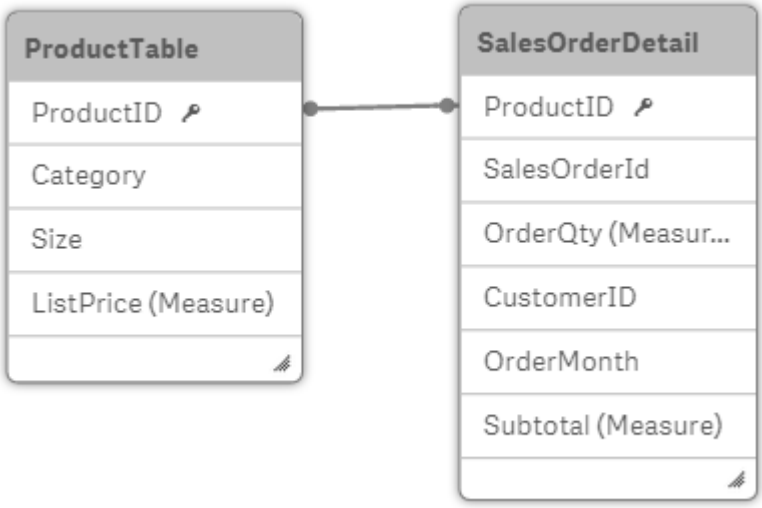
이 예에서는 동일한 논리 테이블의 계수를 만들기 때문에 동일한 차트에서 사용할 수 있습니다. 예를 들어 SubTotal 및 OrderQty를 계수로 사용하여 차트를 만들 수 있습니다.

```
Sales_Order_Header_Join:
DIRECT QUERY
DIMENSION
   AW2012.Sales.Customer.CustomerID as CustomerID,
   AW2012.Sales.SalesOrderHeader.SalesPersonID as SalesPersonID,
   AW2012.Sales.SalesOrderHeader.SalesOrderID as SalesOrderID,
   ProductID,
   AW2012.Sales.Customer.TerritoryID as TerritoryID,
   OrderDate,
   NATIVE('month([OrderDate])') as OrderMonth,
   NATIVE('year([OrderDate])') as OrderYear
MEASURE
   SubTotal,
   TaxAmt,
   TotalDue,
   OrderQty
DETAIL
   DueDate,
   ShipDate,
   CreditCardApprovalCode,
   PersonID,
   StoreID,
   AccountNumber,
   rowguid,
   ModifiedDate
FROM AW2012.Sales.SalesOrderDetail
   JOIN AW2012.Sales.SalesOrderHeader
   ON (AW2012.Sales.SalesOrderDetail.SalesOrderID =
       AW2012.Sales.SalesOrderHeader.SalesOrderID)
   JOIN AW2012.Sales.Customer
   ON(AW2012.Sales.Customer.CustomerID =
      AW2012.Sales.SalesOrderHeader.CustomerID);
```

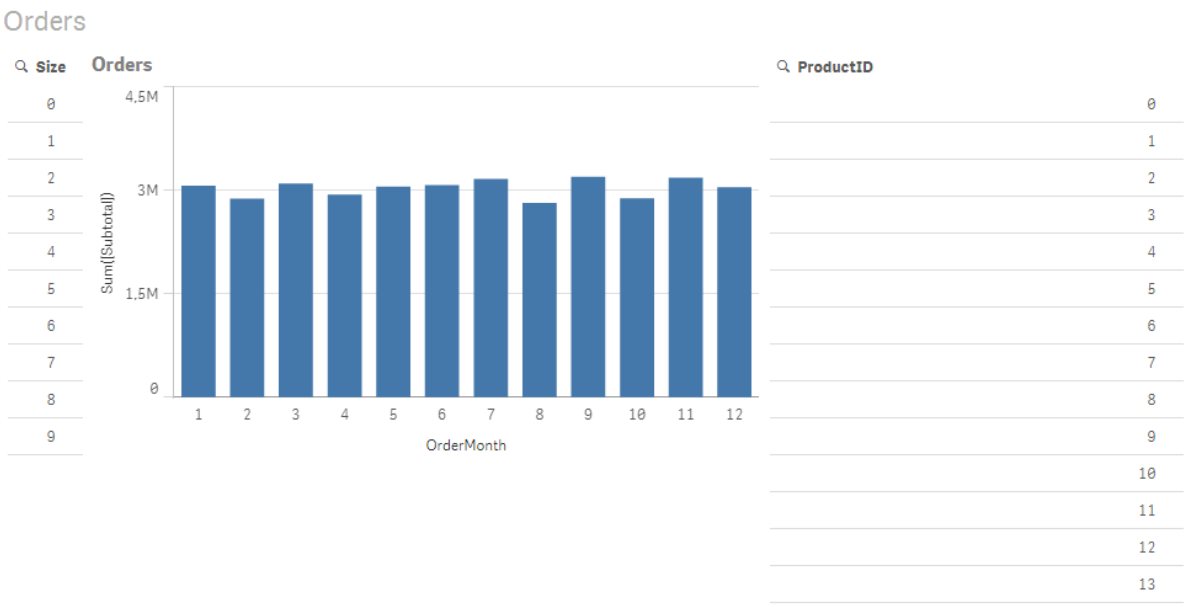
에서 하위 쿼리 사용하기Direct Discovery

테이블에 조인되는 키 필드의 집합 크기가 크다면, 즉 다수의 고유값이 있다면 **WHERE key_field IN** 절에 다수의 값이 포함될 수 있으므로 Qlik Sense에서 선택하는 내용에 따라 매우 큰 SQL 문이 생성될 수 있습니다. 이 경우 가능한 해결책은 Qlik Sense에서 대신 하위 쿼리를 만드는 것입니다.

이를 설명하기 위해 제품 ID(ProductID)를 사용하여 제품 테이블(ProductTable)과 판매 주문 테이블(SalesOrderDetail)을 두 테이블이 Direct Discovery 모드를 사용하는 상태로 연결하는 예를 사용합니다.

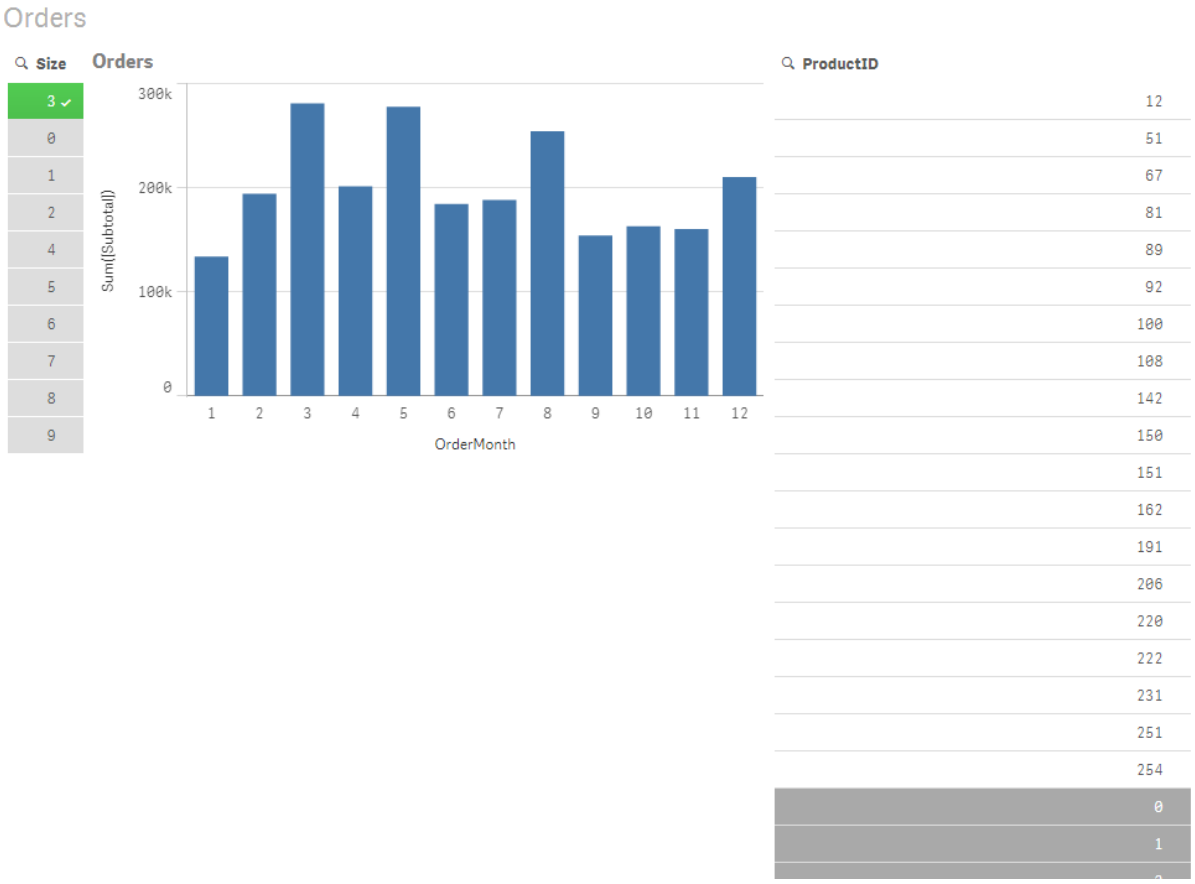


OrderMonth를 차원으로, Sum(Subtotal)을 측정값으로 사용하는 차트 및 Size를 선택하기 위한 필터 상자를 만듭니다.



시나리오 1: 집합 크기 작음

이 시나리오의 경우 제품 테이블에 포함된 고유 제품의 수가 266개로, 적습니다. Size에서 선택을 하면 Direct Discovery는 선택한 크기(이 경우 19개 제품)와 일치하는 제품 ID가 포함된 **WHERE ProductID IN** 절을 사용하여 데이터를 반환하기 위한 SQL 문을 생성합니다.



생성된 SQL 문은 다음과 같습니다.

```
SELECT ProductID, month([OrderDate]), SUM(OrderQty), SUM(SubTotal)
FROM SalesTable
WHERE ProductID IN ( 12, 51, 67, 81, 89, 92, 100, 108, 142, 150, 151, 162, 191, 206, 220, 222,
251, 254)
GROUP BY ProductID, month([OrderDate])
```

시나리오 2: 하위 쿼리 사용

같은 예에서, 예를 들어 20.000개의 많은 고유 제품이 포함되어 있다면 차원 필터로 Size를 선택할 경우 수천 가지 제품 ID가 포함된 **WHERE ProductID IN** 절이 포함된 SQL 문이 생성됩니다. 이 경우 메모리 또는 성능과 관련된 제한이나 문제로 인해 결과 문이 데이터 소스에서 처리하기에 너무 커질 수 있습니다.

해결책은 **DirectEnableSubquery**를 true로 설정하여 Qlik Sense에서 대신 하위 쿼리를 만들도록 하는 것입니다. 생성된 SQL 문은 다음과 같습니다.

```
SELECT ProductID, month([OrderDate]), SUM(OrderQty), SUM(SubTotal)
FROM SalesTable
WHERE ProductID IN
( SELECT DISTINCT "AW2012"."dbo"."PRODUCT"."PRODUCTID" WHERE "AW2012"."dbo"."PRODUCT"."SIZE"
IN (3))
GROUP BY ProductID, month([OrderDate])
```

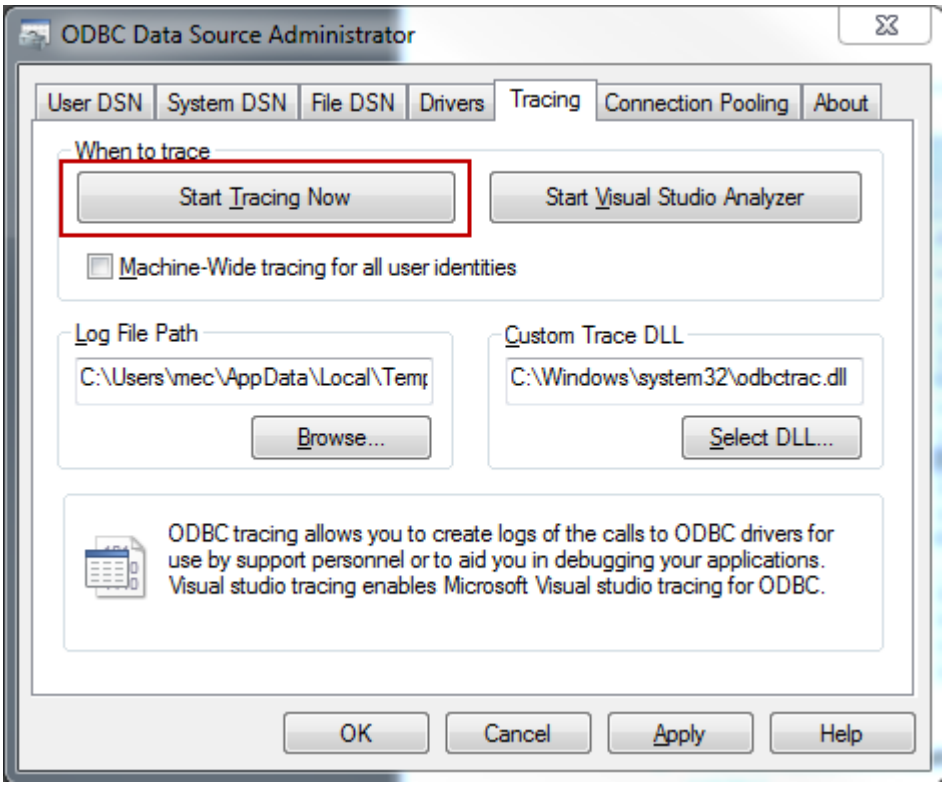
WHERE ProductID IN 절의 크기는 더 이상 선택 내용에 기인한 키 수에 따라 결정되지 않습니다.

하위 쿼리를 사용할 때는 다음과 같은 제한이 적용됩니다.

- 하위 쿼리 구문은 다른 테이블의 데이터를 사용하여 차트를 필터링하는 것과 관련된 데이터를 선택할 때만 호출됩니다.
- 키 수가 아니라 키에 포함된 데이터의 양이 결정 요인입니다.
- 하위 쿼리는 관련된 모든 테이블이 Direct Discovery 모드일 때만 호출됩니다. 메모리 모드에서 포함된 테이블의 데이터를 사용하여 차트를 필터링하면 **IN** 절이 생성됩니다.

Direct Discovery 액세스 로깅

데이터 소스에 전달되는 Direct Discovery SQL 문은 데이터베이스 연결의 추적 파일에 기록될 수 있습니다. 표준 ODBC 연결의 경우 **ODBC 데이터 원본 관리자**를 통해 추적이 시작됩니다.



결과 추적 파일에는 사용자 선택 및 상호 작용을 통해 생성된 SQL 문이 자세히 표시됩니다.

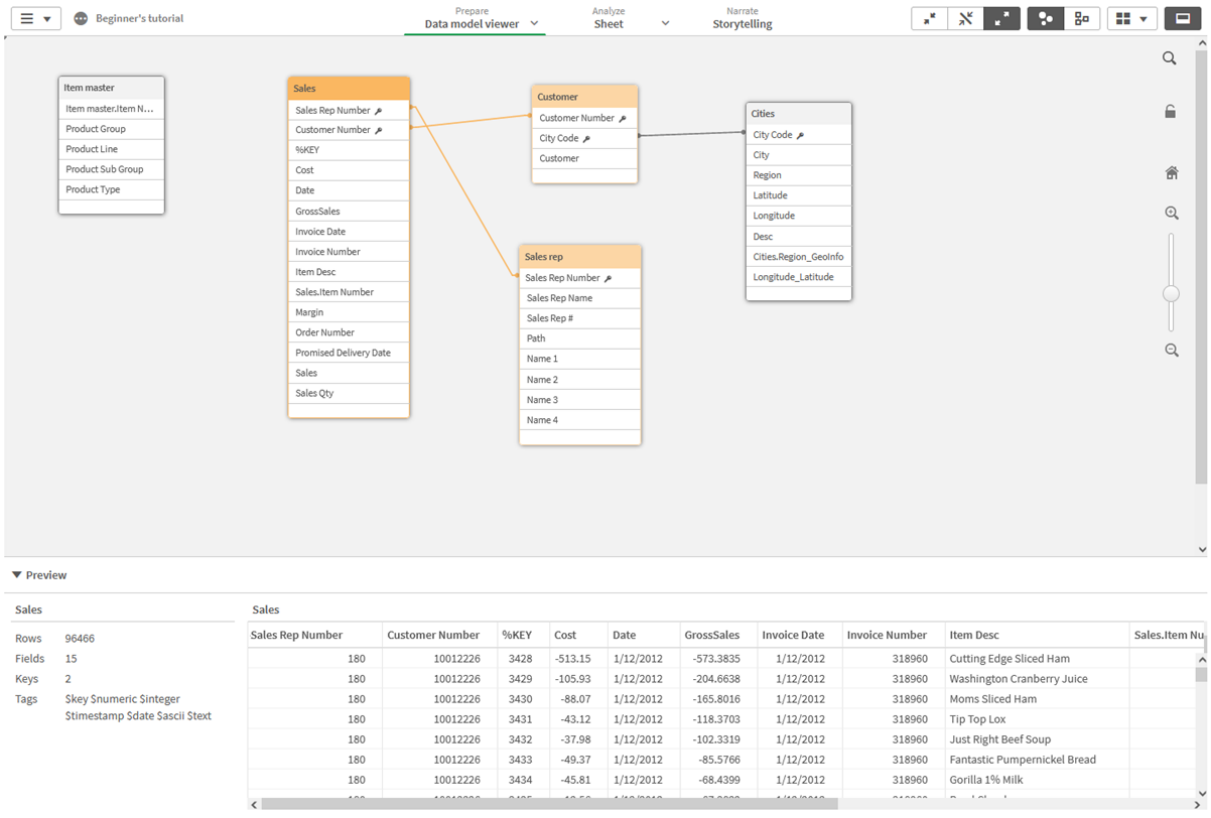
9 데이터 모델 보기 및 변환

데이터 모델 뷰어는 앱의 데이터 구조에 대한 개요이며 테이블 및 필드에 대한 상세한 메타데이터도 볼 수 있습니다. 또한 데이터 필드에서 차원 및 측정값을 만들 수 있습니다.

상단 도구 모음의 **준비** 탭에서 **데이터 모델 뷰어**를 클릭하여 데이터 모델 뷰어를 엽니다.

각 데이터 테이블은 테이블 이름이 제목이고 테이블의 모든 필드가 나열되어 있는 상자로 표현됩니다. 테이블 연결은 순환 참조를 나타내는 점선이 포함된 선으로 표시됩니다. 테이블 또는 필드를 선택하면 연결이 강조 표시되어 필드 및 테이블이 어떻게 연결되어 있는지 즉시 보여줍니다. 을 클릭하여 특정 테이블 및 필드를 검색할 수 있습니다.

데이터 모드는 앱의 데이터 구조를 표시합니다.



확대/축소 수준을 변경하려면 을 클릭하거나 슬라이더를 사용합니다. 아이콘 을 클릭하여 확대/축소 수준을 1:1로 복원합니다.

미리 보기 창에서 테이블 또는 필드의 내용을 조사할 수 있습니다. 또한 필드를 선택하는 경우 차원 및 측정값을 앱에 추가할 수 있습니다. 자세한 내용은 *데이터 모델 뷰어에서 테이블 및 필드 미리 보기 (page 230)*을 참조하십시오.

9.1 테이블 이동

테이블을 캔버스로 끌어서 테이블을 이동할 수 있습니다. 테이블 위치는 앱을 저장할 때 저장됩니다.

캔버스 오른쪽 부분에서 을 클릭하여 테이블 레이아웃(위치 및 크기)을 잠글 수 있습니다. 테이블 레이아웃의 잠금을 해제하려면 을 클릭합니다.

또한 도구 모음의 아래에서 옵션을 사용하여 레이아웃을 자동으로 배열할 수 있습니다.

테이블 이동 옵션

UI 항목	Name	설명
	그리드 레이아웃	그리드에 테이블을 배치합니다.
	자동 레이아웃	창에 맞도록 테이블을 배치합니다.
	레이아웃 복원	데이터 모델 뷰어를 마지막으로 열었을 때의 레이아웃 상태로 복원합니다.

9.2 테이블 크기 조정

테이블 오른쪽 아래 모서리의 화살표를 사용하여 테이블의 표시 크기를 조정할 수 있습니다. 표시 크기는 앱을 저장할 때 저장되지 않습니다.

또한 도구 모음의 자동 표시 크기 옵션을 사용할 수도 있습니다.

테이블 크기 조정 옵션

UI 항목	Name	설명
	모두 축소	테이블 이름만 표시되도록 모든 테이블을 최소화합니다.
	연결된 필드 표시	테이블 이름 및 다른 테이블에 대한 연결이 있는 모든 필드를 표시하도록 모든 테이블의 크기를 줄입니다.
	모두 확장	테이블의 모든 필드가 표시되도록 모든 테이블을 최대화합니다.

9.3 데이터 모델 성능

다음은 데이터 모델 성능에 영향을 미칠 수 있는 표시기입니다. 각각은 앱 유용성을 개선하는 각각의 모범 사례입니다.

데이터 모델 성능 모범 사례

동작	설명
가상 키 제거	Qlik Sense는 둘 이상의 데이터 테이블에 둘 이상의 공통 필드가 있는 경우 가상 키를 만듭니다. 이는 스크립트 또는 데이터 모델에 오류가 있음을 의미할 수 있습니다.

동작	설명
데이터 모델에서 순환 참조 제거	두 개의 필드에 둘 이상의 연결이 있는 경우 순환 참조가 발생합니다. Qlik Sense는 테이블 중 하나에 대한 연결을 변경하여 이 문제를 해결하려고 합니다. 하지만 모든 순환 참조 경고를 해결해야 합니다..
적절한 데이터 세분성	필요한 데이터만 로드해야 합니다. 예를 들어 사용자 그룹에서 주, 월 및 연도로 나뉜 데이터만 필요합니다. 집계된 데이터로 로드하거나 로드 스크립트 내에서 데이터를 집계하여 메모리를 절약할 수 있습니다. 사용자가 낮은 수준의 세분성으로 데이터를 시각화해야 하는 경우 ODAG 또는 문서 연결을 사용할 수 있습니다.
가능한 경우 **QVDs** 사용	QVD는 Qlik Sense에서 내보낸 데이터의 테이블을 포함한 파일입니다. 이 파일 형식은 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다. 파일 형식은 스크립트에서 데이터를 읽는 속도에 최적화되어 있지만 크기는 매우 작습니다. QVD 파일에서 데이터를 읽는 속도는 일반적으로 다른 데이터 소스에서 데이터를 읽는 것보다 10-100배 정도 빠릅니다.
로드 시 **QVD** 파일 최적화	QVD 파일은 표준(빠름) 모드와 최적화(매우 빠름) 모드에서 읽을 수 있습니다. 모드 선택은 스크립트 엔진에서 자동으로 결정합니다. 최적화된 로드에는 몇 가지 제한 사항이 있습니다. 필드의 이름을 변경할 수는 있지만 다음 작업 모두 표준 로드를 발생시킵니다. • 로드된 필드의 모든 변환 • **where** 절을 사용하여 Qlik Sense가 레코드를 압축 해제하게 함 • 로드된 필드에서 **Map** 사용
증분 로드 활용	앱이 지속적으로 업데이트되는 데이터베이스의 대용량 데이터에 연결되는 경우 전체 데이터 집합을 다시 로드하려면 시간이 오래 걸릴 수 있습니다. 대신 증분 로드를 사용하여 데이터베이스의 새 레코드나 변경된 레코드를 검색해야 합니다.

동작	설명
Snowflake 모델 통합	눈송이형 데이터 모델이 있는 경우 Join 접두사 또는 기타 매핑을 사용하여 데이터 테이블 중 일부를 조인함으로써 데이터 테이블 수를 줄일 수 있습니다. 이것은 큰 팩트 테이블에 특히 중요합니다. 실제 좋은 방법은 하나의 큰 테이블만 갖는 것입니다.
필드 수가 적은 테이블 비정규화	필드 수가 적은 두 개의 테이블이 있는 경우 두 테이블을 조인하여 성능을 개선할 수 있습니다. .
매핑 로드를 사용하여 조회(리프) 테이블 비정규화	한 테이블의 필드를 다른 테이블에 하나만 추가해야 하는 경우 Join 접두사를 사용하면 안 됩니다. ApplyMap 조회 함수를 사용해야 합니다. .
날짜 필드에서 타임스탬프 제거 또는 분리	문자열 표현이 더 크고 고유 값의 수가 더 많아 타임스탬프가 있는 경우 날짜 필드가 공백을 채울 수 있습니다. 분석에 정밀도가 필요하지 않은 경우 *Timestamp(Floor (YourStimestamp, 1/24))*를 사용하여 타임스탬프를 가장 가까운 시간 등으로 반올림하거나 *Date(Floor(YourTimestamp))*를 사용하여 시간 구성 요소를 완전히 제거할 수 있습니다. 타임스탬프가 필요한 경우 날짜 자체에서 분리할 수 있습니다. 동일한 Floor() 함수를 사용하고 다음 줄과 함께 사용하여 추출된 시간으로 새 필드를 만들 수 있습니다. *Time(Frac(YourTimestamp))*.
데이터 모델에서 불필요한 필드 제거	데이터 모델에서 필요한 필드만 로드해야 합니다. Load *와 SELECT를 사용하지 마십시오. 다음 필드는 유지합니다. • 분석에 필요한 필드 • 앱에서 실제로 사용되는 필드
볼륨이 큰 데이터를 처리할 때 링크 테이블 방지	가능한 경우 링크 테이블을 사용해야 합니다. 그러나 큰 데이터 볼륨을 처리하는 경우 연결된 테이블이 링크 테이블보다 성능이 뛰어납니다.

동작	설명
연결된 차원을 새 필드로 분리	연결된 차원을 별도의 필드로 분리해야 합니다. 이렇게 하면 필드에서 고유한 값의 발생 수가 줄어듭니다. 이 방법은 타임스탬프를 최적화하는 방법과 유사합니다.
가능한 경우 **AutoNumber** 사용	먼저 **QVD** 파일에서 데이터를 로드하여 최적화된 로드를 만든 다음 **AutoNumber** 문을 사용하여 값을 기호 키로 변환할 수 있습니다.
데이터 섬 방지	데이터 섬은 유용하기도 하지만 대부분 성능에 영향을 미칩니다. 선택 값에 대한 섬을 만드는 경우 변수를 사용합니다.
증분 타임프레임을 기반으로 **QVD** 저장	**QVD**를 분할하여(예: 월별) 저장해야 합니다. 그러면 이 소규모 월별 **QVD**는 모든 데이터가 필요하지 않을 수 있는 다양한 앱을 지원할 수 있습니다.

추가 모범 사례는 *데이터 모델링의 모범 사례 (page 233)*를 참조하십시오.

9.4 데이터 모델 뷰어에서 테이블 및 필드 미리 보기

데이터 모델 뷰어에서 화면 하단의 패널에서 모든 데이터 테이블을 미리 볼 수 있습니다. 미리 보기를 사용하면 테이블 또는 필드의 내용을 빠르게 조사할 수 있습니다. 또한 필드를 선택하는 경우 차원 및 측정값을 앱에 추가할 수 있습니다.

또한 선택한 테이블 또는 필드의 메타데이터가 미리 보기 패널에 표시됩니다.

두 가지 방법으로 미리 보기 패널을 표시하고 숨길 수 있습니다.

- 도구 모음에서 ▭을 클릭합니다.
- **미리 보기** 헤더를 클릭합니다.

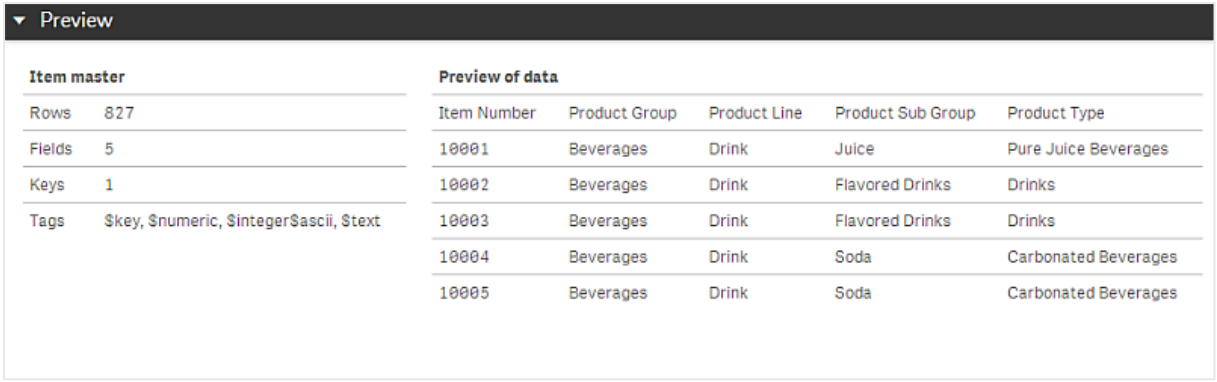
Direct Discovery 데이터는 미리 보기 에 표시되지 않습니다.

테이블의 미리 보기 표시

다음과 같이 하십시오.

- 데이터 모델 뷰어에서 테이블 헤더를 클릭합니다.

선택한 테이블의 필드 및 값과 함께 미리 보기 패널이 표시됩니다.

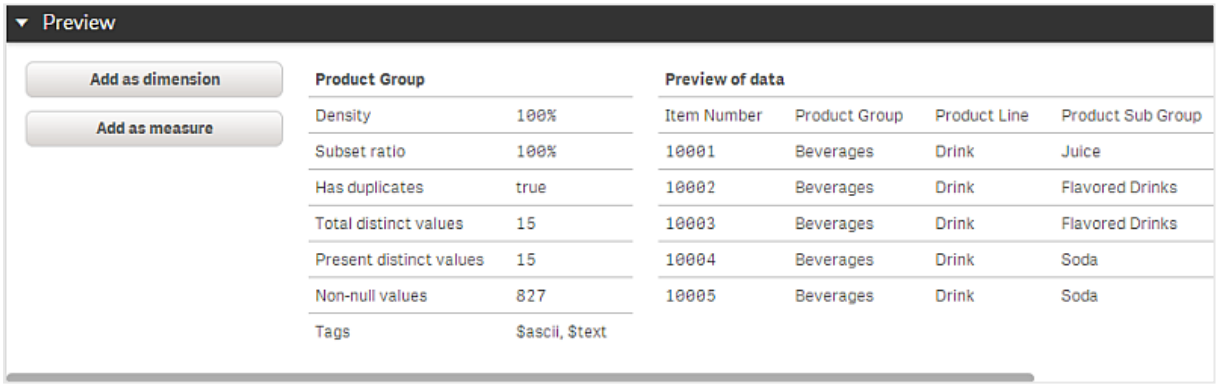


필드의 미리 보기 표시

다음과 같이 하십시오.

- 데이터 모델 뷰어에서 테이블 필드를 클릭합니다.

선택한 필드 및 값, 해당 필드의 메타데이터와 함께 미리 보기 패널이 표시됩니다. 또한 필드를 마스터 차원 또는 측정값으로 추가할 수도 있습니다.



- **밀도**는 이 필드에서 NULL이 아닌 값을 가진 레코드 수와 테이블의 전체 레코드 수를 비교한 값입니다.
- **하위 집합 비율**은 이 테이블에서 발견된 필드의 고유 값의 수와 데이터 모델의 다른 테이블에 포함된 이 필드의 전체 고유 값의 수를 비교한 값입니다. 이 값은 키 필드에만 관련됩니다.
- 필드가 **[완전 키]**로 표시된 경우 모든 행에 고유한 키 값이 포함됩니다.

9.5 데이터 모델 뷰어에서 마스터 차원 만들기

게시되지 않은 앱에서 작업하는 경우, 재사용 가능하도록 마스터 차원을 만들 수 있습니다. 게시된 앱의 사용자는 마스터 차원에 액세스할 수 있지만 수정할 수는 없습니다. 게시된 앱에서는 데이터 모델 뷰어를 사용할 수 없습니다.

다음과 같이 하십시오.

1. 데이터 모델 뷰어에서 필드를 선택하고 **미리 보기** 패널을 엽니다.
2. **차원으로 추가**를 클릭합니다.
   선택한 필드와 함께 **새 차원 만들기** 대화 상자가 열립니다. 선택한 필드 이름은 차원의 기본 이름으로도 사용됩니다.
3. 원하는 경우 이름을 변경하고 선택적으로 설명, 색 및 태그를 추가합니다.
4. **만들기**를 클릭합니다.
5. **완료**를 클릭하여 대화 상자를 닫습니다.

이제 차원이 자산 패널의 마스터 항목 탭에 저장됩니다.

*각 차원을 추가한 후 **차원 추가**를 클릭하여 여러 차원을 마스터 항목으로 신속하게 추가할 수 있습니다. 완료되면 **완료**를 클릭합니다.*

Direct Discovery 테이블은 데이터 모델 뷰어에서 에 의해 표시됩니다.

9.6 데이터 모델 뷰어에서 마스터 측정값 만들기

게시되지 않은 앱에서 작업하는 경우, 재사용 가능하도록 마스터 측정값을 만들 수 있습니다. 게시된 앱의 사용자는 마스터 측정값에 액세스할 수 있지만 수정할 수는 없습니다.

다음과 같이 하십시오.

1. 데이터 모델 뷰어에서 필드를 선택하고 **미리 보기** 패널을 엽니다.
2. **측정값으로 추가**를 클릭합니다.
   선택한 필드와 함께 **새 측정값 만들기** 대화 상자가 열립니다. 선택한 필드 이름은 측정값의 기본 이름으로도 사용됩니다.
3. 측정값에 대한 표현식을 입력합니다.
4. 원하는 경우 이름을 변경하고 선택적으로 설명, 색상 및 태그를 추가합니다.
5. **만들기**를 클릭합니다.

이제 측정값이 자산 패널의 마스터 항목 탭에 저장됩니다.

10 데이터 모델링의 모범 사례

이 섹션에서는 데이터의 구조와 연결하려는 데이터 모델에 따라 Qlik Sense 앱에 데이터를 로드할 수 있는 다양한 방법에 대해 설명합니다.

10.1 데이터 열을 행으로 변환

다음과 같은 데이터가 있는 경우 별도의 필드에 판매 수치를 표시하고 싶습니다.

원래 데이터 테이블

연도	Q1	Q2	Q3	Q4
2013	34	54	53	52
2014	47	56	65	67
2015	57	56	63	71

제안 조치

테이블을 로드할 때 **Crosstable** 접두사를 사용하십시오.

결과는 다음과 같습니다.

Crosstable 접두사 적용 후 테이블

Year	Quarter	Sales
2013	Q1	34
2013	Q2	54
2013	Q3	53
2013	Q4	52
2014	Q1	47
...	...	...

10.2 데이터 행을 필드로 변환

아래와 비슷한 세 개의 필드가 포함된 일반적인 테이블이 있는 경우 각 특성을 개별 테이블로 표시하고 싶습니다.

세 개의 필드가 포함된 일반 테이블

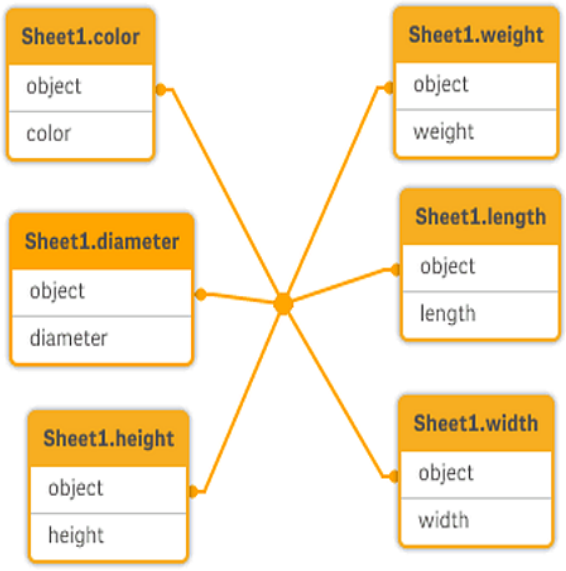
Object	Attribute	Value
ball	color	red
ball	diameter	25

Object	Attribute	Value
ball	weight	3
box	color	56
box	height	30
box	length	20
box	width	25

제안 조치

Generic 로드 접두사를 사용하여 일반 데이터 모델을 만듭니다.

다음과 같은 데이터 모델을 얻게 됩니다.



10.3 계층 수준으로 구성된 데이터(예: 조직 체계) 로드

다음과 같이 인접한 노드 테이블에 데이터가 저장되어 있습니다.

인접 노드 테이블

NodeID	ParentNodeID	Title
1	-	General manager
2	1	Country manager
3	2	Region manager

제안 조치

Hierarchy 접두사를 통해 데이터를 로드하여 확장 노드 테이블을 만듭니다.

확장된 노드 테이블

NodeID	ParentNodeID	Title	Level1	Level2	Level3
1	-	General manager	General manager	-	-
2	1	Country manager	General manager	Country manager	-
3	2	Region manager	General manager	Country manager	Region manager

10.4 대규모 데이터베이스에서 신규 또는 업데이트된 레코드만 로드

대량의 레코드가 포함된 데이터베이스가 있는 경우 앱의 데이터를 갱신하기 위해 전체 데이터베이스를 다시 로드하고 싶지 않습니다. 새 레코드 또는 업데이트된 레코드만 로드하고 데이터베이스에서 삭제된 레코드는 제거하고 싶습니다.

제안 조치

QVD 파일을 사용하여 증분 로드 솔루션을 구현하십시오.

10.5 공통 필드를 사용하여 두 테이블의 데이터 결합

Qlik Sense에서는 공통 필드를 사용하여 자동으로 테이블을 연결하지만 해당 테이블 결합 방법을 제어하고 싶습니다.

제안 조치 : Join / Keep

Join 또는 **Keep** 접두사를 사용하여 두 테이블을 단일 내부 테이블로 결합할 수 있습니다.

제안 조치 : 매핑

테이블을 조인하는 다른 방법은 매핑 테이블에서 연관 값의 조회를 자동화하는 매핑을 사용하는 것입니다. 이 방법으로 로드할 데이터 양을 줄일 수 있습니다.

10.6 간격에 불연속 값 일치

불연속 숫자 값 테이블(Event)이 있고 이를 하나 이상의 간격(Start 및 End)에 일치시키고 싶습니다.

불연속 숫자 값 테이블(Event)

Time	이벤트	주석
00:00	0	이동 1 시작
01:18	1	라인 중단
02:23	2	라인 다시 시작 50%
04:15	3	라인 속도 100%

Time	이벤트	주석
08:00	4	이동 2 시작
11:43	5	종료 또는 생산

간격이 있는 테이블(Start 및 End)

시작	끝	순서
01:00	03:35	A
02:30	07:58	B
03:04	10:27	C
07:23	11:43	D

제안 조치

IntervalMatch 접두사를 사용하여 Time 필드를 Start 및 End에 의해 정의된 간격과 연결하십시오.

시작 및 종료로 간격이 명확하게 정의되어 있지 않고 아래 테이블에서처럼 변경 타임스탬프만 있는 경우, 간격 테이블을 만들어야 합니다.

변경 타임스탬프가 있는 테이블

통화	데이터 변경	환율
EUR	-	8.59
EUR	28/01/2013	8.69
EUR	15/02/2013	8.45
USD	-	6.50
USD	10/01/2013	6.56
USD	03/02/2013	6.30

10.7 일관성 없는 필드 값 처리

여러 테이블에 일관성 없이 이름이 지정된 필드 값이 포함된 데이터가 있습니다. 예를 들어 한 테이블에는 Country에 US 값이 있고 다른 테이블에는 United States이 포함되어 있습니다. 이런 경우 연결이 수행되지 않습니다.

테이블 1

국가	지역
미국	메릴랜드
미국	아이다호
미국	뉴욕
미국	캘리포니아

Table 2

Country	Population
United States	304
Japan	128
Brazil	192
China	1333

제안 조치

매핑 테이블을 사용하여 데이터 정리 작업을 수행하십시오. 그러면 필드 값을 비교하여 올바른 연결을 수행할 수 있습니다.

10.8 일관성 없는 필드 값 대소문자 처리

여러 테이블에 일관성 없이 서식 지정된 필드 값이 포함된 데이터가 있습니다. 예를 들어 한 테이블에는 Type에 single 값이 있고 다른 테이블의 동일한 필드에는 Single이 포함되어 있습니다. 이런 경우 Type 필드에 single과 Single 값을 모두 포함하게 되면 연결이 수행되지 않으므로 대소문자 처리가 중요합니다.

테이블 1

유형	가격
단일	23
더블	39

테이블 2

유형	색
단일	빨강
단일	파랑
더블	흰색
더블	Black

제안 조치

데이터 추가를 사용하여 데이터를 로드한 경우에는 데이터 관리자에서 이를 해결할 수 있습니다.

다음과 같이 하십시오.

1. 데이터 관리자의 테이블 편집기에서 Table2를 엽니다.
2. Type 필드의 이름을 Table2.Type으로 변경합니다.
   데이터 프로파일링이 활성화된 상태에서 **데이터 추가**를 사용하여 테이블을 추가한 경우에는 해당 필드가 자동 연결을 방지하기 위해 이미 Table2.Type으로 이름이 지정되어 있을 수 있습니다. 이런 경우는 이 절치에서 두 테이블을 연결합니다.

3. 표현식 `Lower(Table2.Type)`을 사용하여 계산 필드를 만들고 Type으로 이름을 지정합니다.
4. **데이터 로드**를 클릭합니다.

이제 Table1 및 Table2는 single과 double처럼 소문자 값만 포함하는 Type 필드에 의해 연결되어야 합니다.

대소문자를 다르게 사용하고 싶은 경우에도 비슷한 절차를 통해 작업을 수행할 수 있지만 테이블은 동일한 이름의 필드를 사용하여 연결된다는 점을 기억하십시오.

- Single처럼 모든 값의 첫 문자를 대문자로 표시하려면 대신 Table1의 계산된 Type 필드를 만들고 표현식 `Capitalize(Table1.Type)`을 사용합니다.
- SINGLE처럼 모든 값을 대문자로 표시하려면 두 테이블 모두에서 계산된 Type 필드를 만들고 표현식 `Upper(Table1.Type)` 및 `Upper(Table2.Type)`을 각각 사용합니다.

10.9 특정 지역 관련 데이터를 로드하여 맵으로 데이터 시각화

국가별 또는 매장별 판매량 데이터 등과 같이 맵을 사용하여 시각화하고 싶은 데이터가 있습니다. 맵 시각화를 사용하려면 영역 또는 포인트 데이터를 로드해야 합니다.

제안 조치

KML 파일 또는 Excel 파일에서 데이터 값 위치를 일치시킬 영역 또는 포인트 데이터를 로드할 수 있습니다. 또한 실제 맵 배경도 로드해야 합니다.

10.10 증분 로드를 통해 새 레코드 및 업데이트된 레코드 로드

앱에 지속적으로 업데이트되는 데이터베이스 소스의 데이터가 대량으로 포함된 경우 전체 데이터 셋을 다시 로드하려면 시간이 오래 걸릴 수 있습니다. 이 경우 데이터베이스에서 새 레코드 또는 변경된 레코드를 로드할 수 있습니다. 다른 데이터는 모두 이미 앱에서 사용할 수 있습니다. QVD 파일을 사용하는 증분 로드를 통해 이 작업을 수행할 수 있습니다.

기본 프로세스는 아래의 설명과 같습니다.

1. 데이터베이스 소스 테이블에서 새 데이터나 업데이트된 데이터를 로드합니다.
   느린 프로세스이지만 제한된 수의 레코드만 로드됩니다.
2. 앱에서 이미 사용 가능한 데이터를 QVD 파일에서 로드합니다.
   많은 수의 레코드가 로드되지만 훨씬 빠른 프로세스입니다.
3. 새로운 QVD 파일을 만듭니다.
   다음 번에 증분 로드를 수행할 때 이 파일을 사용합니다.
4. 로드된 모든 테이블에 대해 이 절차를 반복합니다.

다음 예에는 증분 로드가 사용되는 사례가 나와 있습니다. 하지만 소스 데이터베이스 구조와 작업 모드에 따라 좀더 복잡한 솔루션이 필요할 수 있습니다.

- 추가만(일반적으로 로그 파일에 사용)
- 삽입만(업데이트 또는 삭제는 제외)

- 삽입 및 업데이트(삭제는 제외)
- 삽입, 업데이트 및 삭제

최적화 모드 또는 표준 모드에서 QVD 파일을 읽을 수 있습니다. (사용되는 방법은 작업의 복잡성에 따라 Qlik Sense 엔진에 의해 자동으로 선택됩니다.) 최적화 모드는 표준 모드보다 약 10배 빠르거나 데이터베이스를 일반적인 방식으로 로드하는 것보다 약 100배 빠릅니다.

추가만

가장 간단한 사례로는 레코드가 추가되기만 하고 삭제되지는 않는 로그 파일이 있습니다. 다음 조건이 적용됩니다.

- 데이터베이스는 텍스트 파일(ODBC, OLE DB 또는 기타 데이터베이스는 지원되지 않음)에 포함된 로그 파일(또는 레코드가 추가되기만 하고 삽입 또는 삭제되지는 않는 다른 파일)이어야 합니다.
- Qlik Sense에서는 이전에 읽은 레코드의 수를 기록하고 파일의 끝에 추가된 레코드만 로드합니다.

```
Buffer (Incremental) Load * From LogFile.txt (ansi, txt, delimiter is '\t', embedded labels);
```

삽입만(업데이트 또는 삭제는 제외)

데이터가 단순한 로그 파일이 아닌 데이터베이스에 존재하는 경우 추가 방식은 효과가 없습니다. 하지만 최소한의 추가 작업으로 문제를 해결할 수 있습니다. 다음 조건이 적용됩니다.

- 데이터 소스는 모든 데이터베이스일 수 있습니다.
- Qlik Sense에서 마지막 스크립트 실행 후 데이터베이스에 삽입된 레코드를 로드합니다.
- Qlik Sense에서 새 레코드 여부를 인식하려면 ModificationTime 필드(또는 유사한 필드)가 필요합니다.

```
QV_Table:

SQL SELECT PrimaryKey, X, Y FROM DB_TABLE

WHERE ModificationTime >= #$(LastExecTime)#

AND ModificationTime < #$(BeginningThisExecTime)#;



Concatenate LOAD PrimaryKey, X, Y FROM File.QVD;

STORE QV_Table INTO File.QVD;
```

SQL WHERE 절에서 해시 기호는 날짜의 시작과 끝을 정의합니다. 사용 중인 데이터베이스의 올바른 날짜 구문은 데이터베이스 설명서에서 확인하십시오.

삽입 및 업데이트(삭제는 제외)

다음 사례는 이전에 로드된 레코드의 데이터가 스크립트 실행 도중 변경된 경우 적용할 수 있습니다. 다음 조건이 적용됩니다.

- 데이터 소스는 모든 데이터베이스일 수 있습니다.
- Qlik Sense에서 마지막 스크립트 실행 후 데이터베이스에 삽입되거나 데이터베이스에서 업데이트된 레코드를 로드합니다.
- Qlik Sense에서 새 레코드 여부를 인식하려면 ModificationTime 필드(또는 유사한 필드)가 필요합니다.
- Qlik Sense에서 QVD 파일의 업데이트된 레코드를 정렬하려면 기본 키 필드가 필요합니다.
- 이 솔루션은 QVD 파일을 최적화 모드보다는 표준 모드에서 읽습니다. 그래도 전체 데이터베이스를 로드하는 것보다 속도가 상당히 빠릅니다.

```
QV_Table:

SQL SELECT PrimaryKey, X, Y FROM DB_TABLE

WHERE ModificationTime >= #$(LastExecTime)#;



Concatenate LOAD PrimaryKey, X, Y FROM File.QVD

WHERE NOT Exists(PrimaryKey);



STORE QV_Table INTO File.QVD;
```

삽입, 업데이트 및 삭제

처리가 가장 어려운 사례는 스크립트 실행 도중 레코드가 실제로 소스 데이터베이스에서 삭제되는 경우입니다. 다음 조건이 적용됩니다.

- 데이터 소스는 모든 데이터베이스일 수 있습니다.
- Qlik Sense에서 마지막 스크립트 실행 후 데이터베이스에 삽입되거나 데이터베이스에서 업데이트된 레코드를 로드합니다.
- Qlik Sense에서 마지막 스크립트 실행 후 데이터베이스에서 삭제된 레코드를 제거합니다.
- Qlik Sense에서 새 레코드 여부를 인식하려면 ModificationTime 필드(또는 유사한 필드)가 필요합니다.
- Qlik Sense에서 QVD 파일의 업데이트된 레코드를 정렬하려면 기본 키 필드가 필요합니다.
- 이 솔루션은 QVD 파일을 최적화 모드보다는 표준 모드에서 읽습니다. 그래도 전체 데이터베이스를 로드하는 것보다 속도가 상당히 빠릅니다.

```
Let ThisExecTime = Now( );


QV_Table:

SQL SELECT PrimaryKey, X, Y FROM DB_TABLE

WHERE ModificationTime >= #$(LastExecTime)#

AND ModificationTime < #$(ThisExecTime)#;


Concatenate LOAD PrimaryKey, X, Y FROM File.QVD

WHERE NOT EXISTS(PrimaryKey);


Inner Join SQL SELECT PrimaryKey FROM DB_TABLE;


If ScriptErrorCount = 0 then

STORE QV_Table INTO File.QVD;

Let LastExecTime = ThisExecTime;

End If
```

10.11 Join 및 Keep을 사용한 테이블 결합

조인은 두 테이블을 하나로 결합하는 연산입니다. 결과 테이블의 레코드는 원래 테이블에 있는 레코드의 조합이며, 결과 테이블 내의 일정한 조합에 기여하는 2개의 레코드가 하나 또는 여러 공통 필드에 대해 공통의 값을 가지는 이른바 자연 조인이라 불리는 조인이 일반적으로 수행됩니다. Qlik Sense의 경우 스크립트에서 조인을 수행하여 논리 테이블을 생성할 수 있습니다.

이미 스크립트에 있는 테이블을 조인할 수 있습니다. 그러면 Qlik Sense 논리에서는 별도의 테이블이 아니라 조인의 결과인 단일 내부 테이블을 보게 됩니다. 경우에 따라서는 이러한 작업 방식이 필요할 수 있지만 단점이 되는 경우도 있습니다.

- 로드한 테이블이 커져서 Qlik Sense가 느리게 작동할 수 있습니다.
- 일부 정보가 손실될 수 있습니다. 원래 테이블의 빈도(레코드의 수)를 더 이상 사용할 수 없게 될 수 있습니다.

테이블을 Qlik Sense에 저장하기 전에 두 테이블 중 하나 또는 둘 다 테이블 데이터의 교집합으로 축소하는 효과를 제공하는 Keep 기능은 명시적 조인을 사용해야 하는 경우의 수를 줄이기 위해 설계되었습니다.

이 설명서에서는 일반적으로 내부 테이블이 생성되기 전에 수행되는 조인에 대해 조인이라는 용어를 사용합니다. 내부 테이블이 생성된 후 수행되는 연결도 기본적으로 조인입니다.

SQL SELECT 문 내의 조인

일부 ODBC 드라이버의 경우 **SELECT** 문에서 조인을 수행할 수 있습니다. 이 경우 **Join** 접두사를 사용하여 조인할 때와 거의 동일합니다.

그러나 대부분의 ODBC 드라이버는 완전(양방향) 외부 조인을 수행할 수 없습니다. 이 경우 왼쪽 또는 오른쪽 외부 조인만 수행할 수 있습니다. 왼쪽(오른쪽) 외부 조인은 조인 키가 왼쪽(오른쪽) 테이블에 존재하는 조합만 포함합니다. 완전 외부 조인은 모든 조합을 포함합니다. Qlik Sense는 자동으로 완전 외부 조인을 수행합니다.

또한 **SELECT** 문에서 조인을 수행하는 것이 Qlik Sense에서 조인을 수행하는 것보다 훨씬 복잡합니다.

```
SELECT DISTINCTROW
[Order Details].ProductID, [Order Details].
UnitPrice, Orders.OrderID, Orders.OrderDate, Orders.CustomerID
FROM Orders
RIGHT JOIN [Order Details] ON Orders.OrderID = [Order Details].OrderID;
```

이 **SELECT** 문은 가상의 회사로 보내는 주문을 포함하는 테이블과 주문 세부 정보를 포함하는 테이블을 조인합니다. 이 작업은 오른쪽 외부 조인이므로 *OrderDetails*의 모든 레코드가 포함되며 *Orders* 테이블에 존재하지 않는 *OrderID*를 가지는 레코드도 포함됩니다. 반면에 *Orders*에는 존재하지만 *OrderDetails*에는 존재하지 않는 주문은 포함되지 않습니다.

Join

Join을 수행하는 가장 간단한 방법은 스크립트에서 join 접두사를 사용하여 내부 테이블을 다른 명명된 테이블이나 최근에 생성한 테이블과 조인하는 것입니다. 조인은 외부 조인이 되어 두 테이블에서 가능한 모든 값의 조합을 만듭니다.

```
LOAD a, b, c from table1.csv;
join LOAD a, d from table2.csv;
```

결과 내부 테이블에는 a, b, c, d 필드가 생깁니다. 레코드 수는 두 테이블의 필드 값에 따라 달라집니다.

*조인하려는 필드의 이름이 완전히 동일해야 합니다. 조인하려는 필드의 수는 임의입니다. 일반적으로 테이블은 하나 이상의 공통 필드를 가집니다. 공통 필드가 없을 경우 테이블의 카티션 곱이 생성됩니다. 모든 필드가 공통인 경우도 가능하지만 일반적으로 의미가 없습니다. 이전에 로드한 테이블의 이름을 **Join** 문에 지정하지 않은 경우 **Join** 접두사는 최근에 만든 테이블을 사용합니다. 따라서 두 문의 순서는 임의가 아닙니다.*

Keep

데이터 로드 스크립트에서 명시적인 **Join** 접두사는 두 테이블의 완전 조인을 수행합니다. 결과는 한 테이블입니다. 이러한 조인의 결과로 매우 큰 테이블이 만들어지는 경우가 많습니다. Qlik Sense의 주요 기능 중에는 테이블을 조인하는 대신 테이블을 연결하여 메모리 사용량 축소, 속도 향상, 그리고 엄청난 유연성 향상을 제공하는 기능이 있습니다. 유지 기능은 명시적 조인을 사용해야 하는 경우의 수를 줄이기 위해 설계되었습니다.

두 개의 **LOAD** 또는 **SELECT** 문 사이에 **Keep** 접두사를 사용하면 Qlik Sense에 저장하기 전에 하나 또는 두 테이블을 테이블 데이터의 교집합으로 축소하는 효과를 제공할 수 있습니다. **Keep** 접두사 앞에는 항상 **Inner**, **Left** 또는 **Right** 키워드 중 하나가 와야 합니다. 테이블에서 레코드의 선택은 해당하는 조인과 동일한 방법으로 수행됩니다. 그러나 두 테이블은 조인되지 않으며 Qlik Sense에 별도의 명명된 두 테이블로 저장됩니다.

Inner

데이터 로드 스크립트에서 **Join** 및 **Keep** 접두사 앞에 **Inner** 접두사가 올 수 있습니다.

이 접두사가 **Join** 앞에 사용될 경우 두 테이블 간에 내부 조인을 수행하도록 지정됩니다. 결과 테이블은 양쪽의 전체 데이터 셋이 포함된 두 테이블 간의 조합만 포함합니다.

Keep 앞에 사용할 경우 Qlik Sense에 저장되기 전에 두 테이블을 공통 교집합으로 축소하도록 지정됩니다.

이 예에서는 소스 테이블 Table1 및 Table2:

Table 1

A	B
1	aa
2	cc
3	ee

Table2

A	C
1	xx
4	yy

Inner Join

먼저 테이블에 대한 **Inner Join**을 수행하여 두 테이블에 존재하는 유일한 레코드인 하나의 행만 포함된 VTable을 만들고 두 테이블에서 데이터가 결합되도록 합니다.

```
VTable:
SELECT * from Table1;
inner join SELECT * from Table2;
```

VTable

A	B	C
1	aa	xx

Inner Keep

대신 **Inner Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 공통 필드 A를 통해 연결됩니다.

```
VTab1:
SELECT * from Table1;
VTab2:
inner keep SELECT * from Table2;
```

VTab1

A	B
1	aa

VTab2

A	C
1	xx

Left

데이터 로드 스크립트에서 **Join** 및 **Keep** 접두사 앞에 **left** 접두사가 올 수 있습니다.

이 접두사가 **Join** 앞에 사용될 경우 두 테이블 간에 왼쪽 조인을 수행하도록 지정됩니다. 결과 테이블은 첫 번째 테이블의 전체 데이터 셋이 포함된 두 테이블 간의 조합만 포함합니다.

Keep 앞에 사용할 경우 Qlik Sense에 저장되기 전에 두 번째 테이블을 첫 번째 테이블과의 공통 교집합으로 축소하도록 지정됩니다.

이 예에서는 소스 테이블 Table1 및 Table2:

Table1

A	B
1	aa
2	cc
3	ee

Table2

A	C
1	xx
4	yy

먼저 테이블에 대한 **Left Join**을 수행하여 Table1의 모든 행이 포함되고 Table2의 해당 행에 있는 필드와 결합된 VTable을 만듭니다.

```
VTable:
SELECT * from Table1;
left join SELECT * from Table2;
```

VTable

A	B	C
1	aa	xx
2	cc	-
3	ee	-

대신 **Left Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 공통 필드 A를 통해 연결됩니다.

```
VTab1:
SELECT * from Table1;
VTab2:
left keep SELECT * from Table2;
```

VTab1

A	B
1	aa
2	cc
3	ee

VTab2

A	C
1	xx

Right

데이터 로드 스크립트에서 **Join** 및 **Keep** 접두사 앞에 **right** 접두사가 올 수 있습니다.

이 접두사가 **Join** 앞에 사용될 경우 두 테이블 간에 오른쪽 조인을 수행하도록 지정됩니다. 결과 테이블은 두 번째 테이블의 전체 데이터 셋이 포함된 두 테이블 간의 조합만 포함합니다.

Keep 앞에 사용할 경우 Qlik Sense에 저장되기 전에 첫 번째 테이블을 두 번째 테이블과의 공통 교집합으로 축소하도록 지정됩니다.

이 예에서는 소스 테이블 Table1 및 Table2:

Table1

A	B
1	aa
2	cc
3	ee

Table2

A	C
1	xx
4	yy

먼저 테이블에 대한 **Right Join**을 수행하여 Table2의 모든 행이 포함되고 Table1의 해당 행에 있는 필드와 결합된 VTable을 만듭니다.

```
VTable:
SELECT * from Table1;
right join SELECT * from Table2;
```

VTable

A	B	C
1	aa	xx
4	-	yy

대신 **Right Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 공통 필드 A를 통해 연결됩니다.

```
VTab1:
SELECT * from Table1;
VTab2:
right keep SELECT * from Table2;
```

VTab1

A	B
1	aa

VTab2

A	C
1	xx
4	yy

10.12 조인에 대한 대안으로 매핑 사용

Qlik Sense에서 **Join** 접두사는 데이터 모델에서 여러 데이터 테이블을 결합하는 강력한 방법입니다.

한 가지 단점은 결합된 테이블이 커지고 성능 문제를 일으킬 수 있다는 점입니다. 다른 테이블의 단일 값을 조회해야 하는 경우 **Join**에 대한 대안은 매핑을 사용하는 것입니다. 이렇게 하면 조인에 의해 테이블의 레코드 수가 변경되어 불필요한 데이터가 로드되고 계산 속도가 느려지고 계산 오류가 발생할 가능성을 방지할 수 있습니다.

매핑 테이블은 두 개의 열로 구성되는데, 비교 필드(입력)와 매핑 값 필드(출력)입니다.

이 예에서는 주문 테이블(Orders) 하나가 있으며 고객의 국가를 알아야 하는데, 이 정보는 고객 테이블(Customers)에 저장되어 있습니다.

Orders 데이터 테이블

OrderID	OrderDate	ShipperID	Freight	CustomerID
12987	2007-12-01	1	27	3
12988	2007-12-01	1	65	4
12989	2007-12-02	2	32	2
12990	2007-12-03	1	76	3

Customers 데이터 테이블

CustomerID	Name	Country	...
1	DataSales	Spain	...
2	BusinessCorp	Italy	...
3	TechCo	Germany	...
4	Mobecho	France	...

고객의 국가(Country)를 조회하기 위해서는 다음과 같은 매핑 테이블이 필요합니다.

매핑 테이블

CustomerID	Country
1	Spain
2	Italy
3	Germany
4	France

MapCustomerIDtoCountry라는 이름의 매핑 테이블이 다음과 같이 스크립트에 정의되어 있습니다.

```
MapCustomerIDtoCountry:
Mapping LOAD CustomerID, Country From Customers ;
```

다음 단계는 주문 테이블을 로드할 때 **ApplyMap** 함수를 사용하여 매핑을 적용하는 것입니다.

```
Orders:
    LOAD *,
        ApplyMap('MapCustomerIDtoCountry', CustomerID, null()) as Country
```

```
From Orders ;
```

ApplyMap 함수의 세 번째 파라메타는 매핑 테이블에서 값을 발견하지 못했을 때 반환할 값(이 경우는 **Null ()**)을 정의하는 데 사용됩니다.

결과 테이블은 다음과 같이 표시됩니다.

결과 테이블

OrderID	OrderDate	ShipperID	Freight	CustomerID	Country
12987	2007-12-01	1	27	3	Germany
12988	2007-12-01	1	65	4	France
12989	2007-12-02	2	32	2	Italy
12990	2007-12-03	1	76	3	Germany

10.13 데이터 로드 스크립트에서 교차 표 작업

크로스탭은 두 개의 머리글 데이터 직교 목록 사이의 값을 테이블로 나타내는 일반적인 유형의 테이블입니다. 이는 데이터를 다른 데이터 테이블에 연결하려는 경우 선택적 데이터 형식이 아닙니다.

이 항목에서는 크로스탭의 피벗을 해제하는 방법, 즉 데이터 로드 스크립트에서 **LOAD** 구문의 **crosstable** 접두사를 사용하여 크로스탭의 일부를 행으로 바꾸는 방법을 설명합니다.

하나의 한정하는 열이 있는 크로스탭 피벗 해제

크로스탭 앞에는 한정하는 여러 개의 열이 오는 경우가 많은데, 이러한 열은 읽기 쉽게 되어 있어야 합니다. 이 경우 하나의 한정하는 열인 Year와 월별 판매 데이터 행렬이 있습니다.

하나의 한정하는 열이 있는 크로스탭

Year	Jan	Feb	Mar	Apr	May	Jun
2008	45	65	78	12	78	22
2009	11	23	22	22	45	85
2010	65	56	22	79	12	56
2011	45	24	32	78	55	15
2012	45	56	35	78	68	82

이 테이블을 그대로 Qlik Sense로 로드하면 *Year*에 해당하는 필드 하나와 각 월에 해당하는 필드가 생성됩니다. 그러나 이러한 결과를 원하는 것은 아닐 것입니다. 3개의 필드를 생성하는 것을 선호할 수도 있습니다.

- 한정하는 열(이 경우 *Year*)은 위의 테이블에서 녹색으로 표시되었습니다.
- 이 경우 월 이름 Jan - Jun으로 표시되는 특성 필드는 노란색으로 표시되었습니다. 이필드는 적절하게 *Month*로 이름을 지정할 수 있습니다.
- 데이터 행렬 값으로, 파란색으로 표시되어 있습니다. 이 경우, 판매 데이터를 나타내므로 적절히 *Sales*로 이름을 지정할 수 있습니다.

LOAD 또는 **SELECT** 문에 **crosstable** 접두사를 추가하면 원하는 결과를 얻을 수 있습니다.

```
crosstable (Month, Sales) LOAD * from ex1.xlsx;
```

이는 Qlik Sense에서 다음과 같은 테이블을 만듭니다.

crosstable 접두사가 LOAD 또는 SELECT 문에 추가된 테이블

Year	Month	Sales
2008	1월	45
2008	2월	65
2008	3월	78
2008	4월	12
2008	5월	78
2008	6월	22
2009	1월	11
2009	2월	23
...	...	...

두 개의 한정하는 열이 있는 크로스탭 피벗 해제

이 경우에는 왼쪽에 한정하는 열이 두 개 있고 뒤이어 행렬 열이 나옵니다.

두 개의 한정하는 열이 있는 크로스탭

Salesman	Year	Jan	Feb	Mar	Apr	May	Jun
A	2008	45	65	78	12	78	22
A	2009	11	23	22	22	45	85
A	2010	65	56	22	79	12	56
A	2011	45	24	32	78	55	15
A	2012	45	56	35	78	68	82
B	2008	57	77	90	24	90	34
B	2009	23	35	34	34	57	97
B	2010	77	68	34	91	24	68
B	2011	57	36	44	90	67	27
B	2012	57	68	47	90	80	94

한정하는 열의 수는 다음과 같이 **crosstable** 접두사의 세 번째 파라메타로 지정할 수 있습니다.

```
crosstable (Month, Sales, 2) LOAD * from ex2.xlsx;
```

이는 Qlik Sense에서 다음과 같은 결과를 만듭니다.

crosstable 접두사에 대한 세 번째 파라메타로 지정된 한정하는 열이 있는 테이블

Salesman	Year	Month	Sales
A	2008	1월	45
A	2008	2월	65
A	2008	3월	78
A	2008	4월	12
A	2008	5월	78
A	2008	6월	22
A	2009	1월	11
A	2009	2월	23
...	...	...	...

10.14 일반 데이터베이스

일반 데이터베이스는 필드 이름이 한 열의 필드 값으로 저장되고 필드 값은 다른 열에 저장되는 테이블입니다. 일반 데이터베이스는 일반적으로 다양한 개체의 특성에 사용됩니다.

아래 GenericTable 예제를 보십시오. 다음은 ball과 box라는 두 개체를 포함하는 일반 데이터베이스입니다. color 및 weight와 같은 일부 특성은 두 개체에 공통적이지만 diameter, height length 및 width와 같은 다른 특성은 공통적인 특성이 아니라는 것을 확실히 알 수 있습니다.

GenericTable

개체	특성	값
ball	color	red
ball	diameter	10 cm
ball	weight	100 g
box	color	black
box	height	16 cm
box	length	20 cm
box	weight	500 g
box	width	10 cm

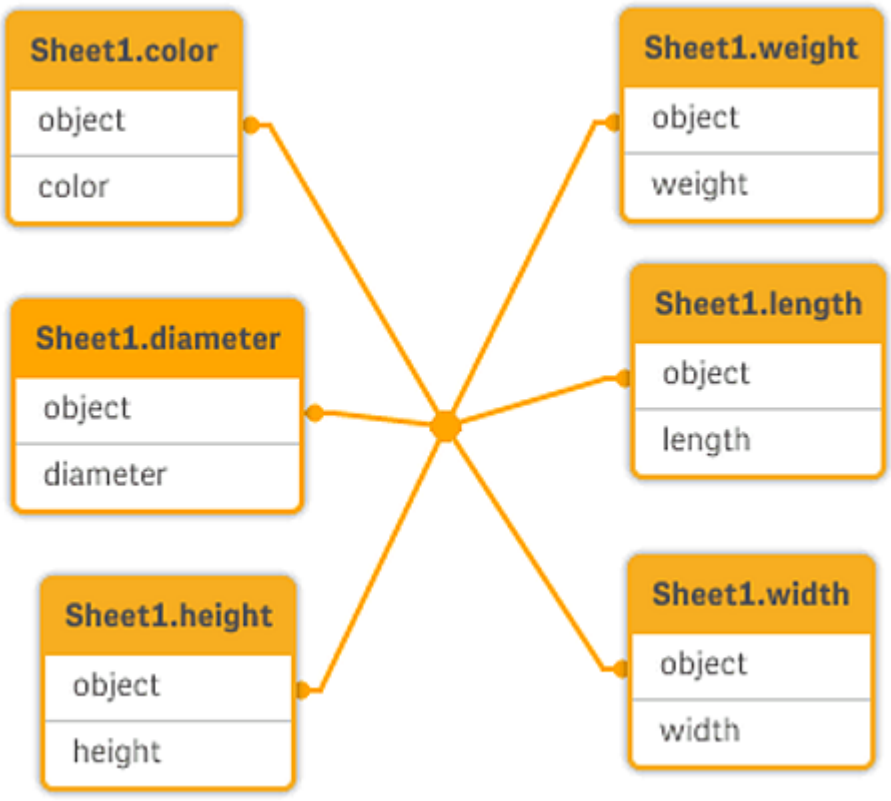
한편, 이러한 특성 중 여러 가지가 특정 개체에는 해당되지 않기 때문에 각 특성에 고유한 열을 지정하는 방식으로 데이터를 저장하는 것은 적합하지 않을 것입니다.

그러나 length, color 및 weight를 하나로 섞어서 표시하면 알아보기 어려울 것입니다.

이 데이터베이스를 표준 방법으로 Qlik Sense에 로드하고 테이블에 데이터가 표시되면 다음과 같이 나타납니다.

object ▲	attribute	value
ball	color	red
ball	diameter	10 cm
ball	weight	100 g
box	color	black
box	height	16 cm
box	length	20 cm
box	weight	500 g
box	width	10 cm

하지만 테이블을 일반 데이터베이스로 로드하면 2열과 3열이 다른 테이블로 분할되며 각각 고유한 값의 추가 열이 됩니다.



이를 수행하는 구문은 다음 예와 같이 간단합니다.

예

로드 스크립트

예제 스크립트를 앱에 추가하고 실행합니다. 결과를 보기 위해 결과 열에 나열된 필드를 앱의 시트에 추가합니다.

```
Sheet1: Generic Load * inline [ object, attribute, value ball, color, red ball, diameter, 10 cm ball, weight, 100 g box, color, black box, height, 16 cm box, length, 20 cm box, weight, 500 g box, width, 10 cm ];
```

결과

결과 테이블

개체	color	diameter	length	height	width	weight
ball	red	10 cm	-	-	-	100 g
box	black	-	20 cm	16 cm	10 cm	500 g

10.15 간격과 불연속 데이터의 연결

LOAD 또는 **SELECT** 문에 **intervalmatch** 접두사를 사용하면 불연속 숫자 값을 하나 이상의 숫자 간격과 연결할 수 있습니다. 이는 프로덕션 환경에서 사용할 수 있는 매우 강력한 기능입니다.

Intervalmatch 예

아래 두 테이블을 보십시오. 첫 번째 테이블에는 여러 주문의 생산 시작 날짜와 종료 날짜가 나와 있습니다. 두 번째 테이블에는 몇 가지 불연속 이벤트가 나와 있습니다. 생산 중단으로 영향을 받은 주문이나 특정 주문 제품을 생산한 근무조 등을 확인하기 위해 이러한 불연속 이벤트를 주문과 연결하려면 어떻게 해야 할까요?

OrderLog 테이블

시작	끝	순서
01:00	03:35	A
02:30	07:58	B
03:04	10:27	C
07:23	11:43	D

EventLog 테이블

Time	이벤트	주석
00:00	0	이동 1 시작
01:18	1	라인 중단

Time	이벤트	주석
02:23	2	라인 다시 시작 50%
04:15	3	라인 속도 100%
08:00	4	이동 2 시작
11:43	5	생산 종료

먼저 두 테이블을 일반적인 방법으로 로드하고 *Time* 필드를 *Start* 및 *End* 필드로 정의된 간격과 연결합니다.

```
SELECT * from OrderLog;
SELECT * from EventLog;
Intervalmatch (Time) SELECT Start,End from OrderLog;
```

이제 아래와 같이 Qlik Sense에서 테이블을 만들 수 있습니다.

시작 및 끝으로 정의된 간격에 연결된 시간 필드가 있는 테이블

Time	이벤트	주석	순서	시작	끝
0:00	0	이동 1 시작	-	-	-
1:18	1	라인 중단	A	1:00	3:35
2:23	2	라인 다시 시작 50%	A	1:00	3:35
4:15	3	라인 속도 100%	B	2:30	7:58
4:15	3	라인 속도 100%	C	3:04	10:....
8:00	4	이동 2 시작	C	3:04	10:....
8:00	4	이동 2 시작	D	7:23	11:....
11:43	5	생산 종료	E	7:23	11:.....

생산 라인 중단에 가장 큰 영향을 받은 것은 주문 *A*이지만 주문 *B*와 *C*의 경우에도 라인 속도 저하에 영향을 받았음을 손쉽게 확인할 수 있습니다. 주문 *C*와 *D*만 *Shift 2*에 의해 부분적으로 처리되었습니다.

intervalmatch를 사용할 때는 다음 사항에 주의하십시오.

- **intervalmatch** 문을 실행하기 전에 불연속 데이터 포인트(이 예에서는 *Time*)가 포함된 필드를 Qlik Sense로 읽어와야 합니다. **intervalmatch** 문은 이 필드를 데이터베이스 테이블에서 읽지 않습니다.
- **intervalmatch LOAD** 또는 **SELECT** 문으로 읽은 테이블에는 항상 정확히 두 개의 필드(위 예에서는 *Start* 및 *End*)가 포함되어 있어야 합니다. 다른 필드에 대한 링크를 설정하려면 별도의 **LOAD** 또는 **SELECT** 문(위의 예에서 첫 번째 **SELECT** 문)의 추가 필드와 함께 간격 필드를 읽어야 합니다.
- 간격은 항상 완료된 상태입니다. 즉, 간격에는 종료 지점이 포함되어야 합니다. 제한이 숫자가 아닐 경우 간격이 무시되며(정의되지 않음), 제한이 NULL일 경우 간격이 무기한으로(무제한) 확장됩니다.
- 간격은 겹칠 수 있으며 불연속 값은 일치하는 모든 간격에 연결됩니다.

확장된 intervalmatch 구문을 사용하여 느리게 변화하는 차원 문제 해결

확장된 **intervalmatch** 구문을 사용하여 잘 알려진 소스 데이터의 느리게 변화하는 차원 문제를 처리할 수 있습니다.

예제 스크립트:

```
SET NullInterpret='';

IntervalTable:
LOAD Key, ValidFrom, Team
FROM 'lib://dataqv/intervalmatch.xlsx' (ooxml, embedded labels, table is IntervalTable);

Key:
LOAD
Key,
ValidFrom as FirstDate,
date(if(Key=previous(Key),
previous(ValidFrom) - 1)) as LastDate,
Team
RESIDENT IntervalTable order by Key, ValidFrom desc;

drop table IntervalTable;

Transact:
LOAD Key, Name, Date, Sales
FROM 'lib://dataqv/intervalmatch.xlsx' (ooxml, embedded labels, table is Transact);

INNER JOIN intervalmatch (Date,Key) LOAD FirstDate, LastDate, Key RESIDENT Key;
```

누락된 값은 NULL 값이 아니라 빈 문자열로 정의되므로, **nullinterpret** 문은 테이블 파일에서 데이터를 읽어올 때만 필요합니다.

*IntervalTable*에서 데이터를 로드하면 다음과 같은 테이블이 만들어집니다.

Table with data loaded from IntervalTable

Key	FirstDate	Team
000110	2011-01-21	Southwest
000110	-	Northwest
000120	-	Northwest
000120	2013-03-05	Southwest
000120	2013-03-05	Northwest
000120	2013-03-05	Southwest

nullasvalue 문을 사용하여 NULL 값을 나열된 필드에 매핑할 수 있습니다.

previous 및 **order by**를 사용하여 *Key*, *FirstDate*, *LastDate*(특성 필드)를 만들면 나중에 *IntervalTable*이 키 테이블로 대체된 후 삭제됩니다.

*Transact*에서 데이터를 로드하면 다음과 같은 테이블이 만들어집니다.

Transact에서 로드된 데이터가 있는 테이블

키	이름	날짜	판매
000110	Spengler Aaron	2009-08-18	100

키	이름	날짜	판매
000110	Spengler Aaron	2009-12-25	200
000110	Spengler Aaron	2011-02-03	300
000110	Spengler Aaron	2011-05-05	400
000120	Ballard John	2011-06-04	500
000120	Ballard John	2013-01-20	600
000120	Ballard John	2013-03-10	700
000120	Ballard John	2013-03-13	800
000120	Ballard John	2013-09-21	900

inner join 다음에 나오는 **intervalmatch** 문은 *Transact* 테이블을 연결하는 가상 키로 위 키를 대체하여 다음과 같은 테이블을 만듭니다.

inner join 다음에 **intervalmatch** 문이 있는 테이블

키	팀	이름	처음 날짜	마지막 날짜	날짜	판매
000110	Northwest	Spengler Aaron	-	2011-01-20	2009-08-18	100
000110	Northwest	Spengler Aaron	-	2011-01-20	2009-12-25	200
000110	Southwest	Spengler Aaron	2011-01-21		2011-02-03	300
000110	Southwest	Spengler Aaron	2011-01-21		2011-05-05	400
000120	Northwest	Ballard John		2013-01-05	2011-06-04	500
000120	Southwest	Ballard John	2013-01-06	2013-03-04	2013-01-20	600
000120	Southwest	Ballard John	2013-03-05		2013-03-10	700
000120	Southwest	Ballard John	2013-03-05		2013-03-13	800
000120	Southwest	Ballard John	2013-03-05		2013-09-21	900

10.16 단일 날짜에서 날짜 간격 만들기

시간 간격은 시작 및 끝을 명시적으로 지정하여 저장되지 않는 경우가 있습니다. 대신 하나의 필드(타임스탬프 변경)를 통해 유추됩니다.

복수의 통화에 대한 환율은 아래 테이블과 같을 수 있습니다. 각 환율 변경은 각각의 행에서 새 전환율에 의해 이루어집니다. 또한 테이블에는 첫 번째 변경이 이루어지기 전의 초기 전환율에 해당하는 빈 날짜가 있는 행이 포함됩니다.

환율

Currency	Change Date	Rate
EUR	-	8.59

Currency	Change Date	Rate
EUR	28/01/2013	8.69
EUR	15/02/2013	8.45
USD	-	6.50
USD	10/01/2013	6.56
USD	03/02/2013	6.30

위 테이블은 일련의 겹치지 않는 기간을 정의합니다. 시작 데이터는 *Change Date*라고 하고, 끝 날짜는 다음 간격의 시작에 의해 정의됩니다. 그러나 끝 날짜가 각 행의 열에 명확하게 저장되지 않으므로 해당 열을 만들어서 새 테이블이 간격의 목록이 되도록 해야 합니다.

다음과 같이 하십시오.

1. 새 앱을 만들고 이름을 지정합니다.
2. **데이터 로드 편집기**에서 새 스크립트 섹션을 추가합니다.
3. 다음 인라인 테이블을 추가합니다. *Change Date* 열의 날짜는 현지 날짜 서식과 동일한 서식이어야 합니다.
```
In_Rates:
LOAD * Inline [
Currency,Change Date,Rate
EUR,,8.59
EUR,28/01/2013,8.69
EUR,15/02/2013,8.45
USD,,6.50
USD,10/01/2013,6.56
USD,03/02/2013,6.30
];
```
4. 작업할 시간 범위를 결정합니다. 범위의 시작은 데이터의 첫 번째 날짜 이전이어야 하며, 범위의 끝은 마지막 날짜 이후여야 합니다.
스크립트의 맨 위에 다음을 추가합니다
```
Let vBeginTime = Num('1/1/2013');
Let vEndTime = Num('1/3/2013');
Let vEpsilon = Pow(2,-27);
```
5. 소스 데이터를 로드하되, 빈 날짜는 이전 항목에서 정의한 범위의 시작으로 변경합니다. 변경 날짜는 "From Date"로 로드되어야 합니다.
먼저 *Currency*에 따라 테이블을 정렬한 다음, "From Date"에 따라 내림차순으로 정렬하여 최근 날짜가 맨 위에 오도록 합니다.
In_Rates 테이블 뒤에 다음을 추가합니다.
```
Tmp_Rates:
LOAD Currency, Rate,
    Date(If(IsNum([Change Date]), [Change Date], $(#vBeginTime))) as FromDate
Resident In_Rates;
```
6. 데이터에서 "To Date"를 계산하기 위한 두 번째 단계를 실행합니다. 현재 레코드에 이전 레코드와 다른 통화가 있는 경우, 새로운 통화의 첫 번째 레코드(및 마지막 간격)이므로 1단계에서 정의한 범위의 끝 날짜를 사용해야 합니다. 동일한 통화인 경우는 이전 레코드의 "From Date"에서 약간의 시간을 빼

고, 이 값을 현재 레코드의 "To Date"로 사용해야 합니다.
Tmp_Rates 테이블 뒤에 다음을 추가합니다.

```
Rates:
LOAD Currency, Rate, FromDate,
    Date(If( Currency=Peek('Currency'),
        Peek('FromDate') - $(#vEpsilon),
        $(#vEndTime)
        )) as ToDate
    Resident Tmp_Rates
    Order By Currency, FromDate Desc;

Drop Table Tmp_Rates;
```

스크립트는 다음과 같이 표시되어야 합니다.

```
Let vBeginTime = Num('1/1/2013');
Let vEndTime = Num('1/3/2013');
Let vEpsilon = Pow(2,-27);

In_Rates:
LOAD * Inline [
Currency,Change Date,Rate
EUR,,8.59
EUR,28/01/2013,8.69
EUR,15/02/2013,8.45
USD,,6.50
USD,10/01/2013,6.56
USD,03/02/2013,6.30
];

Tmp_Rates:
LOAD Currency, Rate,
    Date(If(IsNum([Change Date]), [Change Date], $(#vBeginTime))) as FromDate
Resident In_Rates;

Rates:
LOAD Currency, Rate, FromDate,
    Date(If( Currency=Peek('Currency'),
        Peek('FromDate') - $(#vEpsilon),
        $(#vEndTime)
        )) as ToDate
    Resident Tmp_Rates
Order By Currency, FromDate Desc;

Drop Table Tmp_Rates;
```

스크립트는 다음과 같은 방식으로 소스 테이블을 업데이트합니다.

업데이트된 소스 테이블

Currency	Rate	FromDate	ToDate
EUR	8.45	15/02/2013	vEndTime
EUR	8.69	28/01/2013	14/02/2013 23:59:59

Currency	Rate	FromDate	ToDate
EUR	8.59	vBeginTime	28/01/2013 23:59:99
USD	6.30	03/02/2013	vEndTime
USD	6.56	10/01/2013	2/02/2013 23:59:59
USD	6.50	vBeginTime	9/01/2013 23:59:59

앱에서 테이블은 다음과 같이 나타납니다.

데이터 미리 보기

Currency	Rate	FromDate	ToDate
EUR	8.45	15/02/2013	01/03/2013
EUR	8.69	28/01/2013	14/02/2013
EUR	8.59	01/01/2013	28/01/2013
USD	6.30	03/02/2013	01/03/2013
USD	6.56	10/01/2013	2/02/2013
USD	6.50	01/01/2013	9/01/2013

이 테이블은 이후 IntervalMatch 메서드를 사용하여 기존 날짜와 비교하는 데 사용할 수 있습니다.

10.17 계층 구조 데이터 로드

불균형 *n*수준 계층 구조는 데이터에서 지역 또는 조직 차원 등을 나타내는 데 자주 사용됩니다.

일반적으로 이러한 유형의 계층 구조는 인접 노드 테이블, 즉 각 레코드가 한 노드에 해당하고 부모 노드에 대한 참조를 포함하는 필드를 가지는 테이블에 저장됩니다.

노드 테이블

NodeID	ParentNodeID	Title
1	-	일반 관리자
2	1	지역 관리자
3	2	지점 관리자
4	3	부서 관리자

이러한 테이블에서 노드는 한 레코드에만 저장되지만 가질 수 있는 자식의 수에는 제한이 없습니다. 물론 테이블은 노드의 특성을 나타내는 추가 필드를 포함할 수 있습니다.

인접 노드 테이블은 유지 관리가 간편하지만 일상적인 작업에 사용하기는 어렵습니다. 쿼리와 분석에는 다른 표시 형식이 사용됩니다. 확장 노드 테이블은 일반적인 표시 형식 중 하나로, 계층 구조의 각 수준이 별도의 필드에 저장됩니다. 확장 노드 테이블의 수준은 트리 구조 등에서 손쉽게 사용할 수 있습니다. **hierarchy** 키워드를 데이터 로드 스크립트에 사용하여 인접 노드 테이블을 확장 노드 테이블로 변환할 수 있습니다.

```
Hierarchy (NodeID, ParentNodeID, Title, 'Manager') LOAD
    NodeID,
    ParentNodeID,
    Title
FROM 'lib://data/hierarchy.txt' (txt, codepage is 1252, embedded labels, delimiter is ',',
msq);
```

확장된 노드 테이블

NodeID	ParentNodeID	Title	Title1	Title2	Title4	Title4
1	-	일반 관리자	일반 관리자	-	-	-
2	1	지역 관리자	일반 관리자	지역 관리자	-	-
3	2	지점 관리자	일반 관리자	지역 관리자	지점 관리자	-
4	3	부서 관리자	일반 관리자	지역 관리자	지점 관리자	부서 관리자

확장 노드 테이블의 한 가지 문제는 수준 필드를 검색이나 선택에 사용하기 어렵다는 것인데, 어떤 수준을 검색하거나 선택할지에 대한 사전 지식이 필요하기 때문입니다. 이 문제를 해결하는 데 사용할 수 있는 다른 표시 형식으로는 상위 테이블이 있습니다. 이 표시를 브리지 테이블이라고도 합니다.

상위 테이블에는 데이터에 있는 각 자식-상위 관계에 해당하는 레코드 하나가 포함되어 있습니다. 여기에는 자식과 더불어 상위에 해당하는 키와 이름이 포함되어 있으며, 이는 모든 레코드가 특정 노드가 속한 노드를 설명함을 의미합니다. 데이터 로드 스크립트에서 **hierarchybelongsto** 키워드를 사용하면 인접 노드 테이블을 상위 테이블로 변환할 수 있습니다.

10.18 자체 맵 데이터 로드

맵 시각화를 만들려면 앱의 데이터에 연결된 지리 데이터에 액세스해야 합니다.

Qlik Sense는 다음을 사용할 수 있습니다.

- 맵 레이어에서 위치를 배치할 필드의 이름 데이터
- WGS-84의 지오포인트(위도 및 경도)를 포함하는 필드
- KML 파일과 같은 지리 데이터 소스의 지오포인트 또는 다각형 또는 선을 포함하는 필드
- GeoJSON, LineString 또는 MultiLineString 형식의 지리 데이터를 포함하는 필드
- WGS-84가 아닌 좌표를 포함하는 필드(사용자 지정 맵을 기본 맵으로 사용하는 경우)

데이터 프로파일링이 활성화된 **데이터 관리자**에서 맵 데이터를 로드할 때 데이터 프로파일링 서비스는 국가 이름, 도시 이름, 위도 및 경도 필드를 식별하고 해당 도형을 새 필드에 로드합니다. **데이터 로드 편집기**에서 편의를 위해 선택적으로 좌표 필드를 단일 필드로 결합할 수 있습니다.

맵 시각화의 필드에 대해 지원되는 이름 데이터

맵 시각화는 필드의 이름 데이터를 사용하여 맵 레이어에서 위치를 배치할 수 있습니다. 사용할 수 있는 위치 유형은 다음과 같습니다.

- 대륙 이름
- 국가 이름
- ISO alpha 2 국가 코드
- ISO alpha 3 국가 코드
- 주 또는 지역 이름과 같은 1차 행정 구역 이름
- 2차 행정 구역 이름
- 3차 행정 구역 이름
- 4차 행정 구역 이름
- 도시, 마을 또는 기타 거주 장소의 이름
- 우편 번호
- IATA 공항 코드
- ICAO 공항 코드

위치의 가용성은 국가마다 다릅니다. 명명된 위치를 사용할 수 없는 경우 좌표 또는 영역 데이터를 위치로 사용합니다.

Qlik Sense 허용된 방법론과 모범 사례를 사용하는 알려진 현장 리더로부터 얻은 맵 및 위치 데이터를 사용하여 매핑 내에 테두리를 표시하고 국가 이름을 지정합니다. Qlik Sense 사용자가 자신들이 별도로 사용하던 배경 맵을 통합할 수 있도록 유연성을 제공합니다. 표준 맵이 맞지 않는 경우 Qlik Sense에서는 고객이 제공한 배경 맵, 테두리 및 영역을 로드하는 옵션을 제공합니다.

KML 파일에서 포인트 및 영역 데이터 로드

데이터 관리자 및 **데이터 로드 편집기**에서 KML 파일의 데이터를 맵에 추가할 수 있습니다. 기본적으로, 데이터를 포함하지 않은 경우에도 모든 필드가 데이터 선택 대화 상자에서 선택되어 있습니다. KML 파일에는 예를 들어 영역 데이터를 포함할 수 있지만 포인트 데이터는 없습니다. 비어 있는 포인트 또는 영역 필드가 포함된 KML 파일의 데이터를 Qlik Sense에 추가할 때 아무 데이터가 없는 맵 차원을 만드는 위험을 초래하지 않고 비어 있는 필드를 제외할 수 있습니다.

KML 필드에서 맵 레이어로 필드를 추가할 때 이름 필드에 의미 있는 이름 데이터가 있으면 레이어의 차원으로 추가해야 합니다. 그런 다음 영역 또는 포인트 필드는 **위치 필드**로 추가해야 합니다. 레이어에서 데이터를 시각화하는 방법과 이름 필드에서 텍스트를 도구 설명으로 표시하는 방법에는 차이가 없습니다.

KML 파일에 포인트 데이터, 선 데이터 또는 영역 데이터가 없으면 해당 파일에서 데이터를 로드할 수 없습니다. KML 파일이 손상된 경우는 오류 메시지가 표시되며 데이터를 로드할 수 없게 됩니다.

데이터 프로파일링을 사용하여 맵 데이터 로드

데이터 프로파일링이 활성화된 상태에서 **데이터 관리자**에서 **데이터 추가**를 사용하여 지리 데이터를 로드할 경우 Qlik Sense는 데이터에 다음이 포함되어 있는지 인식하려고 시도합니다.

- 데이터의 국가 및 도시 이름
- 도시와 같은 단일 위치에 대한 지오포인트 데이터(위도, 경도)
- 지역이나 국가를 나타내는 영역 데이터(지오포인트의 다각형)

성공한 경우 지리 정보가 포함된 새 필드가 자동으로 생성됩니다.

> ***데이터 추가****를 사용하는 경우 데이터 프로파일링을 활성화해야 합니다. 이는 기본 선택입니다. 데이터 프로파일링을 비활성화하면 지리 데이터가 검색되지 않고 지리 정보를 포함하는 새 필드가 만들어지지 않습니다.*

데이터를 준비하는 동안 도시가 인식된 경우 새 필드에 지오포인트가 포함되고, 국가가 인식된 경우 새 필드에 영역 다각형 데이터가 포함됩니다. 이 필드를 **<데이터 필드>_GeoInfo**라고 합니다. 예를 들어 도시 이름이 포함된 **Office**라는 필드가 데이터에 있는 경우 **Office_GeoInfo**라는 지오포인트가 있는 필드가 만들어집니다.

> *Qlik Sense에서는 데이터의 하위 집합을 분석하여 도시 또는 국가가 포함된 필드를 인식합니다. 일치율이 75% 미만인 경우 지리 정보가 포함된 필드가 생성되지 않습니다. 필드가 지리 데이터로 인식되지 않으면 수동으로 필드 유형을 지리 데이터로 변경할 수 있습니다.*
>
> *자세한 내용은 필드 유형 변경 (page 53)를 참조하십시오.*

지리 정보가 포함된 필드는 **연결** 미리 보기 패널 또는 **테이블** 보기에 지오포인트나 다각형 데이터를 표시하지 않습니다. 대신 데이터는 일반적으로 *[GEO DATA]*로 표시됩니다. 이렇게 하면 **연결** 및 **테이블** 보기가 표시되는 속도가 향상됩니다. 하지만 **시트** 보기에서 시각화를 만드는 경우 데이터를 사용할 수 있습니다.

포인트 데이터 로드 및 서식 지정

포인트 데이터(좌표)를 사용하여 맵을 만들 수 있습니다. 두 가지 형식이 지원됩니다.

- 포인트 데이터는 두 개의 필드에 저장되는데, 하나는 위도에 대한 필드이고 다른 하나는 경도에 대한 필드입니다. 포인트 레이어의 **위도** 및 **경도** 필드에서 필드를 포인트 레이어에 추가할 수 있습니다. 필요한 경우 이를 단일 필드로 결합할 수 있습니다. 단일 필드로 결합하려면 다음을 수행합니다.
  - 데이터 프로파일링이 활성화된 상태에서 **데이터 추가** 기능을 사용하여 테이블을 로드하면 위도 및 경도 필드가 인식되고 지오포인트 필드가 자동으로 만들어집니다.
  - 데이터 로드 스크립트를 사용하여 데이터를 로드한 경우, `GeoMakePoint()` 함수를 사용하여 *[x, y]* 형식의 포인트 데이터가 있는 단일 필드를 만들 수 있습니다.
    자세한 내용은 *예: 데이터 로드 스크립트를 사용하여 개별 위도 및 경도 열에서 포인트 데이터 로드 (page 262)*를 참조하십시오.
- 포인트 데이터는 하나의 필드에 저장됩니다. 각 포인트는 *x* 및 *y* 좌표의 배열 *[x, y]*로 지정됩니다. 특정 지역 관련 좌표를 사용하면 *[경도, 위도]*에 해당합니다.
  이 형식을 사용하여 **데이터 로드 편집기**에서 데이터를 로드하는 경우 포인트 데이터 필드에 `$geopoint;`로 태그를 지정하는 것이 좋습니다.
  자세한 내용은 다음을 참조하십시오. *예: 데이터 로드 스크립트를 사용하여 단일 열에서 포인트 데이터 로드 (page 262).*

다음 예에서는 파일에 회사의 사무소 위치에 대한 동일한 정보가 두 가지 다른 형식으로 포함되어 있다고 가정합니다.

예: 데이터 로드 스크립트를 사용하여 개별 위도 및 경도 열에서 포인트 데이터 로드

Excel 파일에는 각 사무소에 대해 다음과 같은 내용이 있습니다.

- 사무소
- 위도
- 경도
- 직원 수

로드 스크립트는 다음과 같이 표시될 수 있습니다.

```
LOAD
    Office,
    Latitude,
    Longitude,
    Employees
FROM 'lib://Maps/Offices.xls'
(biff, embedded labels, table is (Sheet1$));
```

`Latitude` 및 `Longitude` 필드의 데이터를 결합하여 포인트에 대한 새 필드를 정의합니다.

스크립트를 실행하여 맵 시각화를 만듭니다. 맵에 포인트 차원을 추가합니다.

LOAD 명령 위에 다음 문자열을 추가하여 스크립트에서 `Location` 차원을 만들 수 있습니다.

```
LOAD *, GeoMakePoint(Latitude, Longitude) as Location;
```

`GeoMakePoint()` 함수는 경도 데이터와 위도 데이터를 함께 조인합니다.

`Office` 필드에 `$geoname`으로 태그를 지정하여 지오포인트 이름으로 인식되도록 하는 것이 좋습니다. **LOAD** 명령의 마지막 문자열 다음에 아래의 줄을 추가합니다.

```
            TAG FIELDS Office WITH $geoname;
```

완성된 스크립트는 다음과 같습니다.

```
LOAD *, GeoMakePoint(Latitude, Longitude) as Location;
LOAD
    Office,
    Latitude,
    Longitude,
    Employees
FROM 'lib://Maps/Offices.xls'
(biff, embedded labels, table is (Sheet1$));

            TAG FIELDS Office WITH $geoname;
```

스크립트를 실행하여 맵 시각화를 만듭니다. 맵에 포인트 차원을 추가합니다.

예: 데이터 로드 스크립트를 사용하여 단일 열에서 포인트 데이터 로드

Excel 파일에는 각 사무소에 대해 다음과 같은 내용이 있습니다.

- 사무소
- 위치
- 직원 수

로드 스크립트는 다음과 같이 표시될 수 있습니다.

```
LOAD
    Office,
    Location,
    Employees
FROM 'lib://Maps/Offices.xls'
(biff, embedded labels, table is (Sheet1$));
```

Location 필드에는 포인트 데이터가 포함되며 이 필드에 $geopoint로 태그를 지정하여 포인트 데이터 필드로 인식되도록 하는 것이 좋습니다. Office 필드에 $geoname으로 태그를 지정하여 지오포인트 이름으로 인식되도록 하는 것이 좋습니다. **LOAD** 명령의 마지막 문자열 다음에 아래의 줄을 추가합니다.

```
            TAG FIELDS Location WITH $geopoint;
            TAG FIELDS Office WITH $geoname;
```

완성된 스크립트는 다음과 같습니다.

```
LOAD
    Office,
    Location,
    Employees
FROM 'lib://Maps/Offices.xls'
(biff, embedded labels, table is (Sheet1$));
 TAG FIELDS Location WITH $geopoint;
 TAG FIELDS Office WITH $geoname;
```

스크립트를 실행하여 맵 시각화를 만듭니다. 맵에 포인트 차원을 추가합니다.

10.19 데이터 정리

서로 다른 테이블에서 데이터를 로드할 때 필드 값의 의미가 같더라도 이름이 항상 일관성 있게 지정되는 것은 아니므로 주의해야 합니다. 일관성 결여는 귀찮은 문제일 뿐만 아니라 연결에 방해가 되므로 해결할 필요가 있습니다. 필드 값을 비교하는 매핑 테이블을 만들면 이 문제를 매끄럽게 해결할 수 있습니다.

매핑 테이블

mapping load 또는 **mapping select**를 통해 로드된 테이블은 일반 테이블과는 다르게 처리됩니다. 이러한 테이블은 별도의 메모리 영역에 저장되며 스크립트 실행 중 매핑 테이블로만 사용됩니다. 스크립트 실행이 완료되면 자동으로 삭제됩니다.

규칙:

- 매핑 테이블에는 두 개의 열, 즉 비교 값을 포함하는 첫째 열과 원하는 매핑 값을 포함하는 둘째 열이 있어야 합니다.

- 두 열에는 이름을 지정해야 하지만 이름은 중요하지 않습니다. 열 이름은 일반 내부 테이블의 필드 이름과 연결되지 않습니다.

매핑 테이블 사용

국가 목록이 나열된 여러 테이블을 로드할 때 동일한 한 국가가 다양한 이름으로 나오는 경우를 볼 수 있습니다. 이 예에서 U.S.A.는 US, U.S., United States로 나열됩니다.

컨케트네이트된 테이블에 미국을 의미하는 레코드가 세 가지로 다르게 포함되지 않도록 하려면 그림에 나오는 것과 비슷한 테이블을 만들고 매핑 테이블로 로드합니다.

전체 스크립트는 다음과 같습니다.

```
CountryMap:
Mapping LOAD x,y from MappingTable.txt
(ansi, txt, delimiter is ',', embedded
labels);
Map Country using CountryMap;
LOAD Country,City from CountryA.txt
(ansi, txt, delimiter is ',', embedded labels);
LOAD Country, City from CountryB.txt
(ansi, txt, delimiter is ',', embedded labels);
```

mapping 문은 *MappingTable.txt* 파일을 *CountryMap* 레이블이 있는 매핑 테이블로 로드합니다.

map 문을 통해 이전에 로드된 매핑 테이블 *CountryMap*을 사용하여 *Country* 필드를 매핑할 수 있습니다.

LOAD 문은 *CountryA* 및 *CountryB* 테이블을 로드합니다. 이 테이블은 동일한 필드 집합이 있기에 컨케트네이트되며, 매핑 테이블의 첫째 열 값과 필드 값을 비교할 *Country* 필드가 포함되어 있습니다. 필드 값 US, U.S., United States가 발견되고 매핑 테이블의 둘째 열 값인 *USA*로 대체됩니다.

자동 매핑은 Qlik Sense테이블에 필드가 저장되기 전 일련의 이벤트에서 마지막으로 수행됩니다. 일반적인 **LOAD** 또는 **SELECT** 문의 이벤트 순서를 대략적으로 정리하면 다음과 같습니다.

1. 표현식 평가
2. as로 필드의 이름 바꾸기
3. alias로 필드의 이름 바꾸기
4. 테이블 이름 한정(해당하는 경우)
5. 필드 이름이 일치하는 경우 데이터 매핑

즉, 매핑은 표현식 실행 도중 필드 이름이 발견될 때마다 수행되는 것이 아니라 Qlik Sense 테이블의 필드 이름으로 값이 저장될 때 수행됩니다.

매핑을 비활성화하려면 **unmap** 문을 사용합니다.

표현식 수준에서 매핑을 수행하려면 **applymap** 함수를 사용합니다.

부분 문자열 수준에서 매핑을 수행하려면 **mapsubstring** 함수를 사용합니다.

11 Insight Advisor의 논리 모델 사용자 지정

비즈니스 논리를 사용하여 쿼리에서 통찰력을 생성할 때 Insight Advisor가 앱 데이터를 해석하는 방식을 사용자 지정할 수 있습니다.

비즈니스 논리는 Insight Advisor이 데이터를 해석하는 방식과 데이터 모델의 값에 대한 대체 용어를 처리하는 방식을 정의합니다.

- Insight Advisor 검색 기반 분석
- Insight Advisor Chat
- Associative Insights

Insight Advisor는 Qlik cognitive engine 및 학습한 선례에 따라 데이터 모델에서 필드의 관계와 사용을 이해합니다. 필요한 경우 앱의 Insight Advisor에서 사용되는 논리 모델을 사용자 지정할 수 있습니다. 사용자가 Insight Advisor 쿼리 시 사용할 수 있는 대체 용어를 Insight Advisor에서 처리할 수 있도록 비즈니스 논리에 어휘를 추가할 수 있습니다.

Qlik Sense에서 다음 비즈니스 논리 영역을 사용자 지정할 수 있습니다.

- **논리 모델**: 모델에서 사용되는 필드 및 그룹을 사용자 지정하고, 패키지를 만들고, 필드 계층을 정의하고, 동작을 설정합니다.
  비즈니스 논리를 사용하여 Insight Advisor의 논리 모델 구축 (page 265) 비즈니스 논리를 사용하여 Insight Advisor의 논리 모델 구축 (page 265)
- **어휘**: 용어를 추가하고 이를 데이터의 필드 및 값에 연결하여 사용자가 Insight Advisor 쿼리 시 대체 용어를 사용할 수 있도록 합니다.
  Insight Advisor용 어휘 만들기 (page 281)

비즈니스 논리 옵션은 **준비** 탭에서 사용할 수 있습니다.

August 2022부터 비즈니스 논리를 포함한 Insight Advisor는 더 이상 Qlik Sense Desktop에서 지원되지 않습니다. November 2022에 Insight Advisor는 새로운 환경으로 업그레이드됩니다. Qlik Sense Enterprise on Windows에서만 사용할 수 있습니다. Qlik Sense Desktop에서 비즈니스 논리와 Insight Advisor를 계속 사용하려는 사용자는 August 2022로 업그레이드하면 안 됩니다.

11.1 **비즈니스 논리**를 사용하여 Insight Advisor의 논리 모델 구축

Insight Advisor은 학습한 선례를 기반으로 하는 논리 모델을 사용하여 통찰력과 관련된 필드를 확인합니다. **비즈니스 논리**를 사용하여 앱의 고유한 논리 모델을 정의할 수 있습니다.

Insight Advisor는 Qlik cognitive engine 및 학습한 선례에 따라 데이터 모델에서 필드의 관계와 사용을 이해합니다. 필요한 경우 Insight Advisor 결과를 개선하기 위해 논리 모델을 사용자 지정할 수 있습니다. **준비** 탭의 **비즈니스 논리** 아래 **논리 모델**에서 논리 모델을 사용자 지정할 수 있습니다.

앱에 비즈니스 논리가 활성화되면 앱의 선례 기반 학습이 비활성화됩니다.

논리 모델 이해하기

앱의 논리 모델은 Insight Advisor이 시각화를 생성할 때 사용하는 개념적 모델입니다. 앱의 기본 데이터 모델로부터 구축되며 각 앱에는 단일 논리 모델이 있습니다. 필드 및 마스터 항목은 논리 모델의 핵심 구성 요소입니다. 필드 및 마스터 항목은 그룹으로 구성됩니다. 그룹은 필드 또는 마스터 항목 간의 개념적 연결 또는 관계를 나타냅니다. 논리 모델에는 그룹 간의 가능한 관계에 대한 정보도 포함되어 있습니다.

논리 모델은 Insight Advisor의 작동 방법에 직접적인 영향을 미칩니다. 예를 들어 사용자가 추세 분석을 표시할 필드를 선택하면 Insight Advisor은 기본 캘린더 그룹의 일부인 날짜 필드를 찾으려고 합니다. *판매* 필드인 경우 Insight Advisor은 *주문 날짜*와 같은 필드의 우선 순위를 *직원 생일*보다 높게 지정합니다.

비즈니스 논리는 시스템이 자연어 쿼리의 필드를 선택하는 방법에도 영향을 줍니다. 예를 들어 *제품 이름* 및 *제품 코드* 필드는 단일 그룹으로 그룹화됩니다. 쿼리에서 *'show me sales by product'*가 사용되는 경우 *'product'*에 *Product Name*을 사용하는 것이 해당 그룹에 좋습니다.

비즈니스 논리에 사용되는 기본 논리 모델은 스타 스키마입니다. 비즈니스 논리를 사용하면 스타 스키마가 최적이 아닌 경우 앱에 대해 다른 모델링을 구성할 수 있습니다. 또한 비즈니스 논리는 어느 정도 가산적인 측정값 또는 매우 큰 측정 테이블을 포함하는 논리 모델에서 집계를 제한하는 데 도움이 될 수 있습니다. 이를 통해 Insight Advisor에서 앱 데이터 탐색이 향상됩니다.

논리 모델 사용자 지정

앱의 논리 모델을 사용자 지정하기 위해 **논리 모델**은 다음 섹션으로 구분됩니다.

- **개요 개요**에서 비즈니스 논리에 대한 요약이 제공됩니다. **필드 및 그룹**, **패키지**, **계층 구조** 또는 **동작**에 대한 카드를 클릭하면 해당 섹션이 열립니다.
- **필드 및 그룹**: **필드 및 그룹**을 사용하면 논리 모델에서 필드 및 마스터 항목이 속하는 그룹을 정의할 수 있습니다.
- **패키지**: **패키지**를 사용하면 관련 그룹의 모음을 만들 수 있습니다. 이렇게 하면 동일한 패키지에 없는 그룹이 함께 사용되는 것을 방지할 수 있습니다.
- **계층 구조 계층 구조**를 사용하면 그룹 간의 드릴다운 관계를 정의할 수 있습니다.
- **동작**: **동작**을 사용하면 필드 간의 선호 또는 거부 관계를 지정할 수 있습니다. 동작은 필요에 따라 선택하도록 할 수도 있습니다.
- **캘린더 기간**: **캘린더 기간**을 사용하여 Insight Advisor에 대한 기본 분석 기간을 만들 수 있습니다.

논리 모델을 사용자 지정하려면 다음을 수행합니다.

1. 앱의 비즈니스 논리를 사용자 지정할 수 있도록 합니다.
2. 필드와 그룹을 정의합니다.
   필드 및 그룹 정의 (page 267)
3. 필요한 경우 패키지에 그룹을 추가합니다.
   패키지로 논리 모델 범위 설정 (page 271)
4. 필요한 경우 그룹 간의 계층 구조를 정의합니다.
   계층 구조로 드릴다운 분석 만들기 (page 272)

5. 필요한 경우 동작을 적용합니다.
   논리 모델에 동작 적용 (page 272)
6. 필요한 경우 캘린더 기간을 만듭니다.
   캘린더 기간으로 분석 기간 정의 (page 275)

논리 모델을 기본값으로 재설정할 수 있습니다. 비즈니스 논리를 임시로 비활성화할 수도 있습니다.

사용자 지정 비즈니스 논리 활성화

다음과 같이 하십시오.

1. 앱에서 **준비**를 클릭하고 **논리 모델**을 선택합니다.
2. **계속**을 클릭합니다.

이제 앱에 대해 사용자 지정 비즈니스 논리가 활성화되었습니다. 선행 기반 학습은 비활성화되었습니다.

비즈니스 논리 재설정

논리 모델을 기본 모델로 재설정할 수 있습니다. 재설정하면 사용자 지정 비즈니스 논리가 비활성화되고 Insight Advisor에서 선례 기반 학습이 활성화됩니다.

다음과 같이 하십시오.

1. **논리 모델**에서 **기본값으로 재설정**을 클릭합니다.
2. **확인**을 클릭합니다.

비즈니스 논리 비활성화

사용자 지정 논리 모델을 비활성화할 수 있습니다. 비즈니스 논리 재설정과 달리 사용자 지정 비즈니스 논리는 나중에 다시 활성화할 수 있습니다. 비즈니스 논리가 비활성화되어 있지만 앱의 기본 비즈니스 논리를 사용할 수 있습니다.

다음과 같이 하십시오.

1. **논리 모델**에서 **논리 비활성화**를 클릭합니다.
2. **확인**을 클릭합니다.

필드 및 그룹 정의

필드 및 그룹은 비즈니스 논리에서 논리 모델의 주요 구성 요소입니다. 필드 및 마스터 항목이 속한 그룹을 정의하여 Insight Advisor에서 사용해야 하는 방법을 정의할 수 있습니다.

논리 모델의 **필드 및 그룹** 섹션에서 비즈니스 논리가 필드 및 그룹을 처리하는 방법을 정의합니다. 필드 및 마스터 항목을 함께 그룹화하여 Insight Advisor 분석에서 관계를 나타낼 수 있습니다. 그룹의 각 항목의 경우 Insight Advisor이 분석에서 이를 처리하는 방법을 정의할 수 있습니다.

그룹을 사용하면 필드와 마스터 항목을 관련된 개념적 그룹으로 구성할 수 있습니다. 예를 들어 로드된 데이터 모델의 테이블에 관계없이 고객과 관련된 모든 필드를 단일 그룹으로 그룹화할 수 있습니다. Insight Advisor은 이 정보를 사용하여 시각화에 함께 표시할 필드를 결정합니다. 또한 그룹을 사용해 패키지를 만들어 관련 그룹만 함께 사용하도록 통찰력 범위를 제한할 수 있습니다. 다음과 같이 세 가지 그룹이 있습니다.

- **차원**: 차원 그룹은 일반적으로 차원으로 분류되는 필드로 구성됩니다. 또한 차원 그룹에는 측정값 또는 날짜로 분류된 필드가 포함될 수 있습니다.
- **측정값**: 측정값 그룹은 관련된 측정값 필드로 구성됩니다. 측정값만 측정값 그룹에 속할 수 있습니다.
- **캘린더**: 캘린더 그룹에는 논리 모델의 시간 차원이 포함됩니다. 캘린더 그룹에는 차원만 포함될 수 있으며 하나 이상의 임시 필드(예: 시간, 타임스탬프 또는 연도)가 있어야 합니다. 캘린더 그룹은 캘린더를 정의하는 별도의 필드(예: *연도*, *월*, *일*)가 있는 경우에 유용합니다. 다른 데이터 관련 필드(예: *회계분기* 또는 *회계년도*)도 그룹화할 수 있습니다.

기본적으로 필드와 마스터 항목은 그룹별로 정렬됩니다. 이 기능을 비활성화하면 **그룹 내** 추가 필드가 테이블에 추가됩니다.

그룹을 정의하는 것 외에도 그룹에 속한 개별 필드 및 마스터 항목의 속성을 정의할 수도 있습니다.

그룹 만들기

그룹을 만든 후에는 그룹 유형을 변경할 수 없습니다. 그룹의 이름을 바꾸고 그룹에서 필드를 추가하거나 제거할 수 있습니다. 그룹을 삭제할 수도 있습니다. 그룹을 삭제하면 그룹의 모든 항목이 그룹 해제됩니다.

기존 그룹을 편집하려면 그룹 행에서 또는 **그룹 내** 열의 그룹 이름 뒤에서 을 클릭합니다.

다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. 그룹 이름을 입력합니다.
3. 그룹 유형을 선택합니다.
4. **사용 가능한 필드**에서 그룹에 필드를 추가합니다.
5. **만들기**를 클릭합니다.

필드 및 마스터 항목 정의

필드 및 그룹은 앱의 필드 및 마스터 항목(해당 그룹과 함께)을 포함하는 테이블로 구성됩니다. 해당 항목 행의 열 값을 조정하여 필드 및 마스터 항목의 속성을 편집할 수 있습니다. 추가 옵션을 보려면 행에 있는 •••을 클릭합니다. 다음을 수행할 수 있습니다.

- **항목 이동**: 항목을 다른 그룹으로 이동합니다.
- **동작 만들기**: 현재 항목이 속한 그룹에 대한 동작을 만듭니다.
- **그룹 해제**: 현재 그룹에서 항목을 제거합니다. 그룹 해제된 항목은 Insight Advisor에서 제외됩니다.

여러 항목을 동일하게 변경하기 위해 행 확인란을 사용하여 여러 행을 선택할 수도 있습니다. 이 방법으로 선택하면 표 위에서 속성 설정 및 옵션을 사용할 수 있습니다.

필드 및 마스터 항목 속성

이 테이블은 다음 열로 나뉩니다.

- 이름
- 그룹
- 가시성
- 분류
- 데이터 값 조회
- 기본 집계
- 유리한 추세
- 즐겨찾기
- 전체 집계
- 기본 기간 단위

이름에는 필드 이름이 나열됩니다. **그룹**에는 필드가 속한 그룹 이름이 나열됩니다. 다음 섹션에서는 다른 필드의 다양한 값과 설정에 대해 설명합니다. ▦을 클릭하여 테이블에 표시할 열을 설정할 수 있습니다.

가시성

가시성은 Insight Advisor에서 항목을 사용할 수 있는지 제어합니다. 다음과 같이 두 가지 가능한 값이 있습니다.

- **표시**: Insight Advisor에서 항목을 사용할 수 있습니다.
- **숨겨짐**: Insight Advisor에서 항목을 사용할 수 없습니다. 숨겨진 필드는 데이터 값 조회에 대해 활성화하지 않아야 합니다.

보이는 부분만 표시를 선택하여 **필드 및 그룹**에서 모든 숨겨진 항목을 숨길 수 있습니다.

분류

분류에서 기본 역할을 정의하여 해당 속성에 따라 분석에서 수행할 수 있습니다. 다음 유형을 사용하여 필드 및 그룹을 분류할 수 있습니다.

- **차원**: 차원으로만 사용되는 필드입니다.
- **측정값**: 측정값으로만 사용되는 필드입니다.
- **부울**: 이진 차원입니다.
- **날짜**: 날짜를 포함하는 임시 차원입니다.
- **타임스탬프**: 타임스탬프를 포함하는 임시 필드입니다.
- **연도**: 연도 데이터를 포함하는 임시 차원입니다.
- **주**: 주 데이터를 포함하는 임시 차원입니다.
- **분기**: 분기 데이터를 포함하는 임시 차원입니다.
- **월**: 월 데이터를 포함하는 임시 차원입니다.
- **weekDay**: 요일에 대한 데이터를 포함하는 임시 차원이며, 짧은 형식(월, 화), 긴 형식(월요일, 화요일) 또는 1-7 사이의 숫자입니다.
- **monthDay**: 월의 날짜를 나타내는 1-31 사이의 숫자를 포함하는 임시 차원입니다.
- **yearDay**: 연중 날짜 수를 포함하는 임시 차원입니다(1-366 사이).
- **시간**: 시간 데이터를 포함하는 임시 차원입니다.
- **이메일**: 이메일 주소를 포함하는 차원입니다.
- **주소**: 주소를 포함하는 차원입니다.

- **국가**: 국가 이름을 포함하는 차원입니다.
- **stateProvince**: 주 및 도와 같은 첫 번째 수준의 관리 영역을 나타내는 차원입니다.
- **도시**: 도시를 나타내는 차원입니다.
- **geoPoint**: 지리적 포인트 데이터를 포함하는 차원입니다.
- **geoPolygon**: 지리적 폴리곤 데이터를 포함하는 차원입니다.
- **geographical**: 국가 또는 지역과 같은 지리적 위치를 나타내는 차원입니다.
- **postalCode**: 우편 번호를 포함하는 차원입니다.
- **경도**: 경도 데이터를 포함하는 차원입니다.
- **위도**: 위도 데이터를 포함하는 차원입니다.
- **백분율**: 고용률 또는 인플레이션과 같은 백분율 값을 나타내는 측정값 필드입니다.
- **화폐**: 수익, 비용 또는 급여와 같은 금전적 측정값입니다.
- **순서**: 값에 고유한 순서가 있는 차원입니다.
- **임시**: 시간 관련 차원입니다.

데이터 값 조회

데이터 값 조회는 자연어 쿼리를 수행할 때 Insight Advisor이 필드에서 값을 조회할 수 있는지 여부를 제어합니다.

데이터 값 조회를 활성화한 필드 수를 줄이면 가양성 결과를 방지하고 쿼리 시간을 줄일 수 있습니다. 예를 들어 데이터 모델에 이름이 포함된 세 개의 필드가 있을 수 있습니다. *이름*, *성* 및 *전체 이름*. 세 개의 필드 모두에 대해 **데이터 값 조회**가 활성화된 경우 사용자가 *`Drew`*를 검색하면 세 필드 모두에서 혼란스러운 결과를 얻을 수 있습니다.

*측정값 및 숨겨진 필드에 대해 **데이터 값 조회**를 비활성화해야 합니다.*

기본 집계

기본 집계는 Insight Advisor에서 측정값에 대한 표준 집계를 설정합니다. 다음 집계를 사용할 수 있습니다.

- **sum**
- **avg**
- **min**
- **max**
- **count**
- **countDistinct**

필드에 기본 집계가 있는 경우 Insight Advisor은 측정값으로 사용할 때 항상 해당 집계를 적용합니다.사용자는 통찰력 차트를 편집하여 Insight Advisor에서 집계를 다른 유형으로 변경할 수 있습니다.

기본 집계는 마스터 항목에 할당할 수 없습니다.

유리한 추세

측정값에 대해 원하는 추세가 증가하거나 감소하는 경우 유리한 추세가 설정됩니다. 기본적으로 측정값에 대한 유리한 추세는 증가하는 것입니다. 다음 추세를 사용할 수 있습니다.

- **위로**
- **아래로**

유리한 추세는 순위 결과에 대해 Insight Advisor가 만든 시각화에 사용됩니다.

즐겨찾기
즐겨찾기는 측정값을 Insight Advisor에 대한 특정 관심 분야로 표시합니다 **생성**과 같이 Insight Advisor는 사용자 쿼리 또는 선택 없이 시각화를 생성할 때 이 측정값을 더 자주 사용합니다.

전체 집계
전체 집계는 Insight Advisor가 마스터 측정값과 관련된 쿼리에 대한 내러티브에서 사용할 집계를 스스로 결정할 수 없을 때 사용해야 하는 집계를 Insight Advisor에 나타냅니다. 일부 마스터 측정값은 복잡한 표현식을 사용하므로 Insight Advisor는 내러티브에서 마스터 측정값의 기여도를 계산할 때 사용할 집계를 모를 수 있습니다. 전체 집계를 설정하면 Insight Advisor에서 이러한 복잡한 마스터 측정값에 대한 내러티브를 계산할 수 있습니다.

마스터 측정값에 대한 전체 집계만 설정할 수 있습니다.

기본 기간 단위
Insight Advisor에서 분석을 생성할 때 캘린더 그룹의 날짜 필드와 함께 사용할 월 또는 연도와 같은 기본 기간 단위를 설정할 수 있습니다. 단위는 자동 캘린더에서 파생되거나 동일한 캘린더 그룹에서 단위가 포함된 필드를 선택할 수 있습니다.

기본 캘린더 기간 동작과 함께 기본 캘린더 설정을 사용할 수 있습니다. 필드 및 마스터 항목에 대해 설정된 기본 캘린더 속성은 그룹에 대해 설정된 기본 캘린더 기간 동작에 의해 무효화됩니다.

패키지로 논리 모델 범위 설정

패키지를 사용하면 논리 모델에서 특정 관심 영역에 대한 통찰력 범위를 정의하고 제한할 수 있습니다.

패키지는 관련 그룹의 모음입니다. Insight Advisor은 결과를 생성할 때 동일한 패키지의 그룹만 함께 사용합니다. 그룹은 여러 패키지에 속할 수 있습니다. 논리 모델 패키지는 **패키지**에서 만들고 관리합니다.

예를 들어 데이터 모델에는 별도의 비즈니스 영역의 테이블이 포함될 수 있습니다. 테이블 간에 연결이 있는 경우 *Sales*의 날짜 필드가 *Support*의 날짜 필드가 있는 통찰력 차트에 사용될 수 있습니다. *Sales* 그룹과 *Support* 그룹을 별도의 패키지에 배치하면 해당 비즈니스 영역에 대한 통찰력이 관련 패키지의 필드만 사용하도록 할 수 있습니다.

패키지 사용은 선택 사항입니다. 패키지를 정의하지 않으면 모든 필드와 그룹이 분석을 위해 동일한 패키지에 있는 것으로 간주됩니다.

•••을 클릭하거나 **편집** 또는 **삭제**를 선택하여 패키지를 편집 또는 삭제할 수 있습니다.

패키지 만들기

다음과 같이 하십시오.

1. **패키지 만들기**를 클릭합니다.
2. 패키지 이름을 입력합니다.
3. **사용 가능한 그룹**에서 패키지에 그룹을 추가합니다.

   패키지에는 하나 이상의 측정값 그룹이 포함되어야 합니다.
4. **만들기**를 클릭합니다.

계층 구조로 드릴다운 분석 만들기

논리 모델에서 계층 관계를 갖는 그룹을 정의하면 Insight Advisor이 지정된 계층 구조를 따라 세부 분석을 제공하고 그룹 간의 종속성을 더 잘 감지할 수 있습니다.

논리 모델 계층 구조는 **계층 구조**에서 만들고 관리합니다. 계층 구조는 비즈니스 논리에 대해 선택 사항입니다.

계층 구조를 통해 권장 사항(예: *제품 범주*, *제품 하위 범주* 및 *제품*에 따른 *판매액*)에서 드릴다운 분석을 수행할 수 있습니다. 또한 계층 구조를 사용하면 Insight Advisor이 권장 사항에서 지리적 항목이 혼합되지 않도록 할 수 있습니다. 예를 들어 *공급업체 시*를 포함하는 *공급업체 국가*에 대한 계층 구조를 만들고 *고객 국가* 및 *고객 시*에 대한 계층 구조를 만들면 *공급업체 국가*와 *고객 시*와 같은 잘못된 조합으로 사용되는 것을 방지할 수 있습니다.

계층 구조는 선택 시 종속성을 감지하기 위해 Associative Insights에서도 사용됩니다. 예를 들어 통찰력 차트에서 *Sweden*을 선택하면 *City* 필드를 사용하는 다른 차트에서 스웨덴 도시 값만 선택할 수 있습니다.

•••을 클릭하거나 **편집** 또는 **삭제**를 선택하여 계층 구조를 편집 또는 삭제할 수 있습니다.

계층 구조 만들기

다음과 같이 하십시오.

1. **계층 구조 만들기**를 클릭합니다.
2. 계층 구조 이름을 입력합니다.
3. 계층 구조의 최하위 레벨에서 최상위 레벨 순으로 계층 구조에 그룹을 추가합니다.
4. **만들기**를 클릭합니다.

논리 모델에 동작 적용

동작을 사용하면 측정값 그룹과 다른 그룹 간의 선호 또는 거부 관계를 설정할 수 있습니다. 또한 동작을 사용하여 Insight Advisor에서 값을 선택하도록 할 수 있습니다.

분석에 항상 함께 사용하거나 분석에 함께 사용할 수 없는 그룹이 있을 수 있습니다. 또한 Insight Advisor에서 그룹의 필드를 사용할 때 선호하는 필드 값이 항상 선택되어 있도록 할 수 있습니다. 동작을 사용하면 그룹별로 이러한 기본 설정을 지정할 수 있습니다. 논리 모델 동작은 **동작**에서 만들고 관리합니다.

논리 모델에서 다음 동작을 사용할 수 있습니다.

- 필수 선택
- 선호 관계
- 거부 관계
- 기본 캘린더 기간

•••을 클릭하거나 **편집** 또는 **삭제**를 선택하여 동작을 편집 또는 삭제할 수 있습니다.

필수 선택

필수 선택 동작을 사용하면 그룹의 필드를 사용할 때 포함되어야 하는 필드 값을 지정할 수 있습니다. 동작에는 여러 필수 선택 항목이 포함될 수 있습니다.

예를 들어 *국가*, *인구* 및 *연도*에 대한 필드가 있을 수 있습니다. Insight Advisor에서 권장 사항을 생성할 때 *국가* 및 *인구*를 사용하는 차트를 얻을 수 있지만 차트에 모든 연도의 데이터 합계가 포함됩니다.필요한 선택 동작을 사용하여 *인구* 또는 *국가*에 대한 통찰력을 생성할 때 현재 연도를 대신 사용하도록 통찰력 결과를 제한할 수 있습니다.

선호 관계

선호 관계 동작을 사용하여 Insight Advisor에서 통찰력을 생성할 때 더 자주 함께 사용해야 하는 그룹을 선택할 수 있습니다. 선호 관계를 지정하면 Insight Advisor에서 결과를 생성할 때 선호 그룹을 사용합니다. 선호 관계는 모호성이 있을 수 있는 경우에 유용합니다.

선호 관계를 사용하여 그룹을 다른 그룹과 함께 사용할 수도 있습니다. 분석에서는 모든 그룹 선택이 동일할 때만 선호하는 그룹을 선택합니다. 다른 그룹이 더 적합한 분석에서는 선호되지 않는 그룹이 대신 사용될 수 있습니다.

예를 들어 네 개의 그룹이 있습니다.

- *판매*
- *Customer*
- *Product*
- *(판매 직원)*

*판매*는 *고객*과 선호 관계를 가지고 있습니다. *판매*의 세부 분석의 경우 선호 관계에 따라 다른 그룹에서 *고객*이 선택됩니다. *판매*의 추세 분석의 경우 *제품*이 분석에 대신 사용될 수 있습니다.

거부 관계

거부 관계 동작은 Insight Advisor이 통찰력을 생성할 때 선택한 그룹을 함께 사용하지 못하도록 합니다. 이는 동일한 패키지의 일부 그룹이 분석에 함께 사용하는 것이 유용하지 않을 때 사용할 수 있습니다. 거부 관계를 사용하여 앱 성능에 영향을 줄 수 있는 그룹이 함께 사용되지 않도록 차단할 수도 있습니다. 스타 스키마 데이터 모델을 사용하면 분석을 복잡하게 만드는 일 대 다 및 다 대 일 관계가 있는 그룹이 있을 수 있습니다. 거부 관계는 이러한 그룹 간에 연결하지 못하도록 할 수 있습니다.

거부 관계가 있는 그룹의 통찰력을 사용자가 특별히 요청하는 경우 거부 관계가 무시됩니다. 예를 들어 *판매*와 *공급업체*에는 거부 관계가 있습니다. *'show me sales'*를 쿼리할 경우 *Sales* 및 *Supplier*를 포함하는 분석이 생성되지 않습니다. *'show me sales by supplier'*를 쿼리할 경우 *Sales* 및 *Supplier*를 포함하는 결과가 생성됩니다.

기본 캘린더 기간

기본 캘린더 기간 동작은 선택한 그룹의 시각화에서 캘린더 기간을 기본 기간으로 사용하도록 할당합니다. 기본 캘린더 기간은 Insight Advisor이 해당 그룹에서 필드의 시각화를 만들 때마다 적용됩니다. 그룹에는 단일 기본 캘린더 기간이 있을 수 있습니다. 기본 캘린더 기간 동작은 날짜 필드에 대해 설정된 기본 기간 단위보다 우선합니다.

캘린더 기간은 일반적으로 다음과 같은 방식으로 분석에 적용됩니다.

- 팩트 및 순위 분석은 캘린더 기간에서 현재 또는 처음 선택한 기간을 사용합니다. 캘린더 기간에서 이전 기간 또는 두 번째로 선택한 기간이 비교를 위해 분석에 추가됩니다.
- 추세 분석 및 유사한 분석은 캘린더 기간의 집계된 단위만 사용합니다.
- 다른 분석은 캘린더 기간에서 현재 또는 처음 선택한 기간을 사용합니다.

예를 들어 클라이언트 만족도 데이터가 포함된 *클라이언트 만족도* 그룹이 있을 수 있습니다. 기본적으로 대부분의 앱 사용자는 이번 달의 데이터에만 관심이 있습니다. 이번 달에 대한 캘린더 기간을 만들고 이를 기본 캘린더 기간으로 지정하면 *클라이언트 만족도*의 Insight Advisor 시각화에 이번 달의 데이터만 표시됩니다.

선택한 분석 유형에 대한 기본 캘린더 기간 무시

기본 캘린더 기간의 일부 또는 전체를 무시하도록 분석 유형을 구성할 수 있습니다. 이는 기본 캘린더 기간을 보다 구체적으로 적용하는 방법을 제어하는 데 유용합니다. 예를 들어, 팩트 분석에서 비교를 원하지 않고 현재 기간만 보기를 원할 수 있습니다.

기간 1(이전 기간 또는 두 번째로 선택한 기간), 기간 2(현재 기간 또는 첫 번째로 선택한 기간) 또는 캘린더 기간의 단위를 무시하도록 분석 유형을 선택할 수 있습니다. 기간 1을 무시하면 기간 2만 표시되도록 두 기간을 비교하는 분석에서 이전 기간 사용이 중지됩니다. 기간 2 또는 단위를 무시하면 캘린더 기간이 해당 분석 유형과 함께 기본적으로 사용되지 않습니다.

동작 만들기

다음과 같이 하십시오.

1. **동작 만들기**를 클릭합니다.
2. **적용 대상**에서 그룹을 선택합니다.
3. 동작 유형 선택
4. 다음 중 하나를 수행합니다.
    - **선호 관계** 또는 **거부 관계**를 구성하는 경우 해당 관계를 적용할 그룹을 선택합니다.
    - **필수 선택**을 구성하는 경우 이것이 단일 값인지 선택합니다. **요구 사항**에서 필드와 필수 값을 선택합니다.
      다른 항목 추가를 클릭하여 필요한 선택을 더 추가할 수 있습니다.

- **기본 캘린더 기간**을 구성하는 경우 해당 그룹에서 캘린더 그룹 및 캘린더 기간을 선택합니다. 선택적으로 **기간 1 무시**, **기간 2 무시** 및 **단위 무시**에서 기본 캘린더 기간의 예외가 있어야 하는 분석 유형을 지정합니다.

5. **만들기**를 클릭합니다.

캘린더 기간으로 분석 기간 정의

캘린더 그룹의 캘린더 기간을 정의하여 Insight Advisor이(가) 시각화를 만들 때 사용하는 시간 프레임을 제어할 수 있습니다.

캘린더 기간은 Insight Advisor이(가) 논리적 모델의 분석에서 사용할 관심 있는 기간을 정의합니다. 예를 들어 모든 시간에 대한 총 판매액을 표시하는 KPI는 현재 월의 총 판매액을 표시하는 KPI보다 덜 유용할 수 있습니다. 현재 월에 대한 캘린더 기간을 만들어 총 판매액과 함께 사용할 기본 기간으로 설정할 수 있습니다. 논리적 모델에 대한 캘린더 기간은 **캘린더 기간**에서 만들고 관리합니다.

캘린더 기간은 단일 기간 또는 상대적 기간의 비교를 정의합니다. 예를 들어, 캘린더 기간을 만들어 다음을 수행할 수 있습니다.

- 이번 달의 값만 표시
- 이번 달과 지난 달 비교
- 현재 분기를 작년 현재 분기와 비교

Insight Advisor은(는) 차트를 만들 때 캘린더 기간을 사용합니다. 캘린더 기간은 추가 분석 유형을 가능하게 합니다. **통찰력**에서 차트를 편집할 때 캘린더 기간을 선택하고 적용할 수 있습니다. Insight Advisor가 해당 그룹의 필드 및 마스터 항목에 대한 차트를 만들 때 그룹에서 사용할 기본 캘린더 기간을 설정할 수 있습니다. 기본 캘린더 기간은 **동작**에서 설정됩니다.

캘린더 기간 만들기

필드 및 그룹에 정의된 캘린더 그룹에서 캘린더 기간을 만듭니다. 데이터 로드 스크립트에서 `autoCalendar` 함수를 사용하는 필드를 사용하거나 필드 없이 캘린더 기간을 만들 수 있습니다.

> *사용자 지정 캘린더가 `autoCalendar`로 선언된 동일한 필드를 사용하는 경우 사용자 지정 캘린더에서 **자동 캘린더 사용** 옵션을 사용할 수 있습니다.*

자동 캘린더를 사용하여 캘린더 기간 만들기

자동 캘린더를 사용하여 캘린더 기간을 만들 때 분석 단위를 선택합니다. 예를 들어, 단위는 연도의 월 또는 연도의 분기입니다. 그런 다음 가장 최근 값을 사용하도록 선택하거나 비교를 만들 수 있습니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. 캘린더 그룹을 선택한 다음 **자동 캘린더 사용**을 선택합니다.
3. 캘린더 기간의 이름을 입력합니다.
4. 캘린더 기간 단위를 선택합니다.

5. 다음 중 하나를 수행합니다.
    - 기간 단위 필드에 가장 최근 값을 사용하려면 **마지막으로 정렬된 값 사용**을 선택합니다.
    - 비교 캘린더 기간을 만들려면 다음 중 하나를 수행합니다.
        - **비교**에서 비교 기간을 선택합니다. Insight Advisor이 현재 기간이 아닌 마지막 전체 기간을 사용해야 하는 경우 **마지막 완료 기간**을 선택합니다.
        - **사용자 지정**의 **오프셋**에서 분석 기간을 선택한 다음 **비교 오프셋**에서 비교 기간을 선택합니다.
          오프셋 및 **비교 오프셋**은 숫자 값을 사용합니다. 현재 기간의 경우 0입니다.
          예를 들어, 연도의 분기를 기간 단위로 사용한 경우 0이 현재 기간이 되고 3이 이전 3개 분기가 됩니다.
6. **만들기**를 클릭합니다.

사용자 지정 캘린더를 사용하여 캘린더 기간 만들기

로드 스크립트에서 autoCalendar를 사용하지 않고 캘린더 기간을 만들 수 있습니다. 예를 들어, 사용자 지정 캘린더가 있을 수도 있고 아니면 autoCalendar에 없는 사용자 지정 캘린더 플래그를 사용하고 싶을 수 있습니다.

사용자 지정 캘린더를 사용하여 캘린더 기간을 만들려면 캘린더 그룹을 선택하고 집계할 시간 필드를 선택합니다. 그런 다음 가장 최근 값을 사용하도록 선택하거나 비교를 만듭니다. 다음 두 가지 비교 방법이 지원됩니다.

- **상대**: 상대적인 이전 기간 수가 포함된 필드를 사용하는 상대 비교입니다. 상대 기간 필드는 집계된 필드의 각 값에 대한 현재 날짜로부터 상대 기간을 정의하는 숫자 값을 포함해야 합니다. 그런 다음 분석을 위해 오프셋 및 비교 오프셋을 선택할 수 있습니다.
  예를 들어, 이번 달과 지난 달의 비교를 수행하려면 *Date*를 집계 필드로 선택합니다. 다음으로 *MonthsAgo*를 상대 기간 필드로 선택합니다. *MonthsAgo*는 몇 개월 전인지 평가하는 표현식을 사용하고 *Date*의 각 값은 현재 월의 날짜입니다. 현재 월이 7월이면 7월의 값은 0, 6월의 값은 1, 5월의 값은 2가 됩니다. 그런 다음 오프셋으로 0을 선택하고 비교 오프셋은 1을 선택합니다.
- **플래그**: 플래그 비교는 별도로 플래그 지정된 두 개의 기간을 정의하는 두 필드를 사용합니다.
  예를 들어, 첫 번째 분기와 두 번째 분기가 플래그 비교에 사용된 두 기간일 수 있습니다. 비교 필드는 집계된 필드 값이 플래그 지정된 기간 내에 있음을 나타내는 이진 값을 포함해야 합니다.
  분기 1과 분기 2를 비교하려면 *Date*를 집계된 필드로 선택합니다. 그런 다음 플래그 필드로 *InQuarter1* 및 *InQuarter2*를 선택할 수 있습니다. 이 필드는 값이 분기 1 또는 분기 2 내에 있는지 없는지를 평가하는 표현식을 사용합니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. 캘린더 그룹을 선택한 다음 **자동 캘린더 사용** 확인란을 선택 해제합니다.
3. 캘린더 기간의 이름을 입력합니다.
4. 사용할 집계된 날짜 필드를 선택합니다.

5. 다음 중 하나를 수행합니다.
    - 기간 단위 필드에 가장 최근 값을 사용하려면 **마지막으로 정렬된 값 사용**을 선택합니다.
    - 비교 캘린더 기간을 만들려면 다음 중 하나를 수행합니다.
        - **상대**에서 **상대 기간 전**의 경우 **집계된 날짜**에서 선택한 필드의 상대 기간 데이터를 포함하는 로드 스크립트에 정의된 필드를 선택합니다. **오프셋**에서 분석할 기간을 설정한 다음 **비교 오프셋**에서 비교할 기간을 설정합니다.
        오프셋 및 **비교 오프셋**은 숫자 값을 사용합니다. 현재 기간의 경우 0입니다.
        - **플래그**에서 현재 기간에 대한 플래그를 포함하는 필드를 선택한 다음 비교 기간에 대한 플래그를 포함하는 필드를 선택합니다.
6. **만들기**를 클릭합니다.

제한 사항

비즈니스 논리 캘린더 기간에는 다음 제한 사항이 있습니다.

- 기본 캘린더 기간은 표현식에 특정 기간을 포함하는 시간 기반 마스터 측정값에 적용되지 않습니다.

단계별 - 사용자 지정 캘린더를 사용하여 캘린더 기간 만들기

이 단계별 연습에서는 사용자 지정 캘린더 필드 및 플래그를 사용하여 캘린더 기간을 만드는 방법을 보여줍니다.

캘린더 기간은 자동 캘린더를 사용하거나 데이터의 개별 날짜/시간 필드를 사용하여 만들 수 있습니다. 이진 데이터가 포함된 필드를 사용하여 비교 분석을 위한 기간을 플래그 지정할 수도 있습니다.

사용자 지정 캘린더 데이터로 만들 수 있는 캘린더 기간에는 세 가지 종류가 있습니다.

- 마지막으로 정렬된 값: 캘린더 기간에 선택한 집계 필드의 가장 최근 기간이 마지막으로 정렬되어 표시된 값입니다. 순위 분석과 같은 비교를 지원하는 Insight Advisor 분석에서 마지막으로 정렬된 값은 이전 기간도 표시합니다.
- 상대 비교: 상대 캘린더 기간은 현재 날짜의 상대 데이터 기간이 포함된 필드를 사용합니다. 현재 또는 이전 기간과 과거 기간을 비교한 내용을 제공합니다.
- 플래그 비교: 플래그는 부울로 분류되어 이진 데이터를 포함하는 필드를 사용하여 비교 분석을 위해 두 기간에 플래그를 지정합니다.

상대적 기간 비교를 보여 주기 위해 이 앱의 데이터 소스에는 미래 날짜에 대한 데이터가 포함됩니다. 로드 스크립트는 현재 날짜까지 데이터 소스의 데이터를 로드합니다. 이미지에 표시된 기간이 변경되므로 이미지의 결과가 사용자의 결과와 다를 수 있습니다.

시작하기

패키지 예를 다운로드하고 압축을 풉니다.

캘린더 기간 예 앱

QVF 파일에는 다음 데이터 파일이 포함됩니다.

- TutorialCustomCalendarData.xlsx
  캘린더 기간을 통해 가능한 상대적 비교를 보여 주기 위해 *TutorialCustomCalendarData.xlsx*에는 미래 날짜에 대한 데이터가 포함됩니다. 앱 로드 스크립트는 로드될 때 현재 날짜에 대한 앱 내 데이터를 업데이트합니다.

Qlik Sense에 있는 QVF 파일을 가져와서 앱에 XLSX 파일을 첨부합니다. 앱을 가져오고 데이터 파일을 앱에 첨부했으면 **데이터 로드 편집기**에서 앱 데이터를 로드합니다.

데이터 예

이 예에서 사용된 데이터는 다음 로드 스크립트로 로드됩니다.

```
Sales: LOAD City, Country, Customer, OrderDate, ProductID, Quantity, Sales, "Q4-2018", "Q1-2019", "Q2-2019", "Q3-2019", "Q4-2019", "Q1-2020", "Q2-2020", "Q3-2020", "Q4-2020", "Q1-2021", "Q2-2021", "Q3-2021", Month([OrderDate]) AS [Month], Year([OrderDate]) AS [Year], Day([OrderDate]) AS [Day], Dual(Year(OrderDate)&'-'&Month(OrderDate), monthstart(OrderDate)) AS [YearMonth], 12*Year(Today())+Month(Today())-12*Year(OrderDate)-Month(OrderDate) AS [MonthsAgo]

FROM [lib://AttachedFiles/TutorialCustomCalendarData.xlsx]

(ooxml, embedded labels, table is Sales) Where OrderDate <= Today(1);
```

로드 스크립트는 *연도*, *월* 및 *일*에 대해 별도의 필드를 만듭니다. 이 세 필드는 다음 계산된 필드를 만드는 데 사용됩니다.

- *YearMonth*, 연도 및 월 정보가 있습니다. 이 예에서 집계를 위한 기본 필드입니다.
- *MonthsAgo*, 특정 월에서 현재 날짜까지의 개월 수를 계산합니다.

이 데이터에는 데이터에서 처리되는 여러 분기에 대한 여러 필드도 포함됩니다. 이러한 필드에는 *Sales* 테이블의 각 값이 속한 회계 분기를 나타내는 이진 데이터가 포함됩니다.

작업

이 연습에서는 세 가지 종류의 캘린더 기간을 만드는 과정으로 안내합니다.

- 자동 캘린더 없이 사용자 지정 캘린더 기간 만들기
- 상대적인 달력 기간 만들기
- 플래그 비교 캘린더 기간 만들기

마지막으로 정렬된 값을 사용하여 캘린더 기간 만들기

첫 번째 캘린더 기간의 경우 마지막으로 정렬된 값을 사용하여 *YearMonth*에 대한 캘린더 기간을 만듭니다.

다음과 같이 하십시오.

1. 예 앱에서 **준비**를 클릭합니다.
2. **비즈니스 논리**에서 **논리 모델**을 선택합니다.
3. **캘린더 기간 만들기**를 클릭합니다.
4. *OrderDate*를 선택합니다.
5. **캘린더 기간 이름**에 *Last sorted month*를 입력합니다.
6. **집계 날짜**에서 *YearMonth*를 선택합니다.

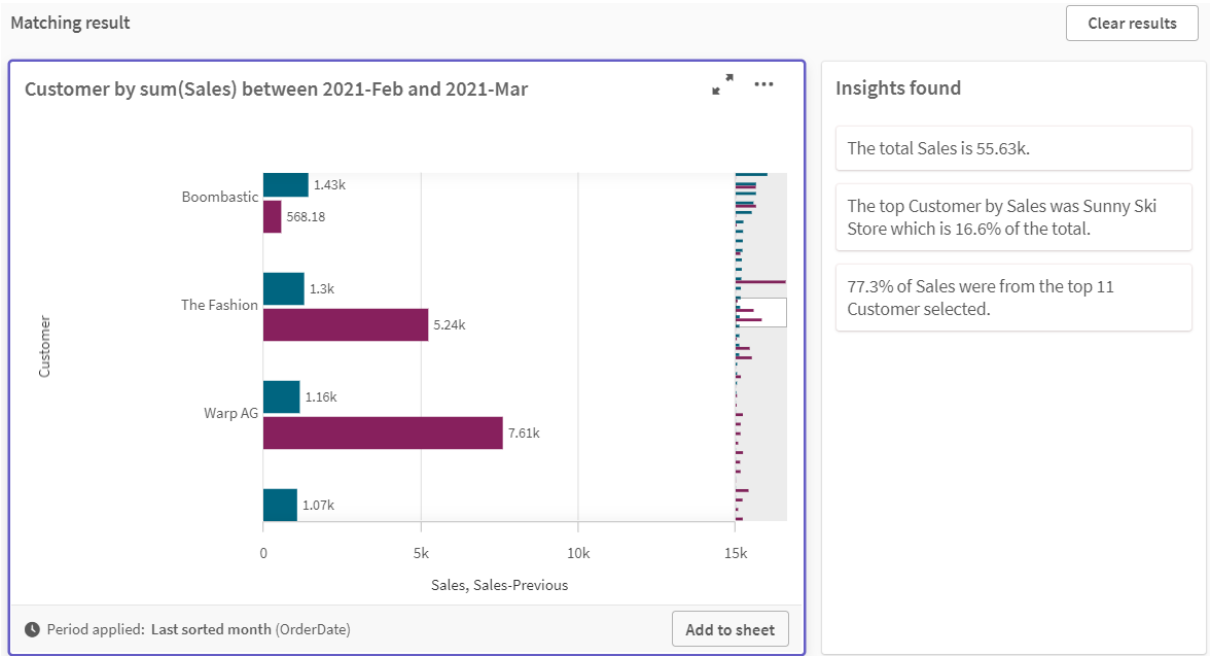
7. **최근 정렬된 값 사용**을 선택합니다.
8. **만들기**를 클릭합니다.

결과

통찰력으로 이동하여 *show me sales by customer*를 검색합니다.

sum(Sales) by customer 차트를 선택하고 분석 기간 *Last sorted month*를 적용합니다. 차트가 업데이트되어 현재 월과 이전 월 간의 비교를 표시합니다.

Customer by sum(Sales) between 2021-Feb and 2021-Mar에 적용된 Last sorted month



상대적인 비교 캘린더 기간 만들기

다음으로 상대적인 캘린더 기간을 만듭니다. 상대적인 캘린더 기간에는 다음이 필요합니다.

- 기간(연도, 월, 분기 등)을 포함하는 집계 필드입니다.
- 해당 필드에서 현재 날짜까지의 상대적 날짜 위치를 포함하는 필드입니다.

그런 다음 이러한 필드에서 오프셋을 정의합니다. 오프셋은 비교할 두 기간에 대한 현재 날짜와 선택한 기간의 상대적인 차이입니다. 현재 또는 이전 기간(**오프셋**에서 0 또는 1로 설정)을 과거 기간 최대 12개 기간 전(**오프셋 비교**에서 1~12의 숫자로 설정)과 비교할 수 있습니다.

이 캘린더 기간의 경우 *YearMonth*를 집계 필드로 사용하고 *MonthsAgo*를 상대 기간 필드로 사용합니다. 이번 달과 작년 같은 달을 비교하려고 합니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. *OrderDate*를 선택합니다.
3. **캘린더 기간 이름**에 *This month to this month last year*를 입력합니다.

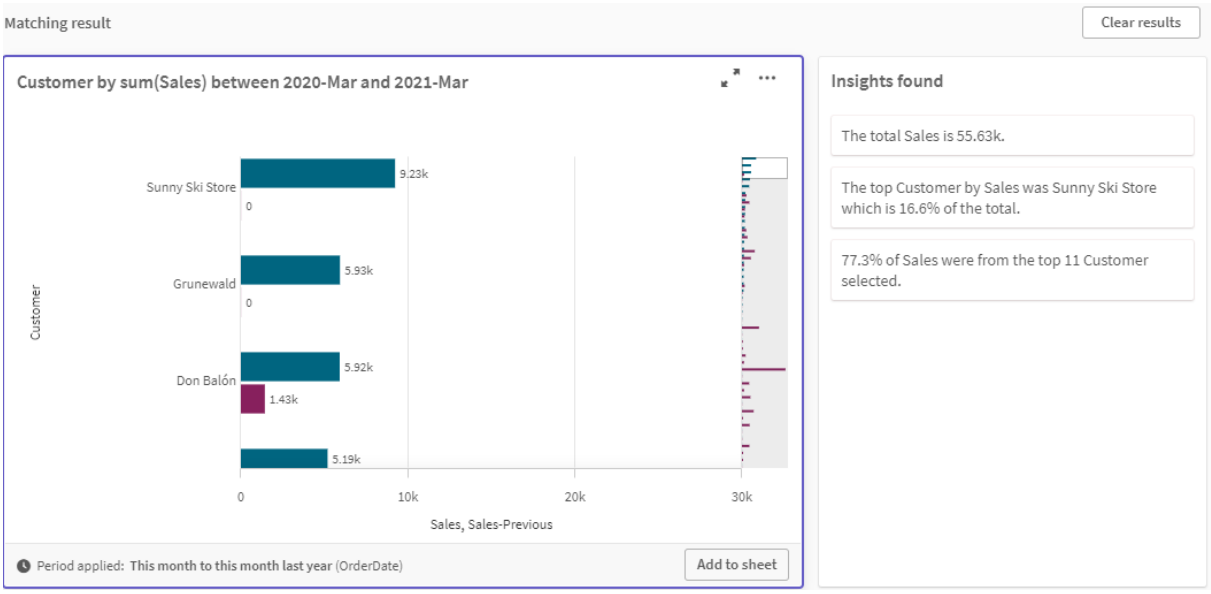
4. **집계 날짜**에서 *YearMonth*를 선택합니다.
5. **상대적인 기간 전**에서 *MonthsAgo*를 선택합니다.
6. **오프셋**에서 0을 선택합니다.
7. **오프셋 비교**에서 12를 선택합니다.
8. **만들기**를 클릭합니다.

결과

통찰력으로 이동하여 *show me sales by customer*를 검색합니다.

sum(Sales) by customer 차트를 선택하고 분석 기간 *This month to this month last year*를 적용합니다. 차트가 업데이트되어 현재 월과 현재 월 간의 비교가 표시됩니다.

2020년 3월과 2021년 3월 간의 sum(Sales) 기준 고객



플래그 비교 캘린더 기간 만들기

플래그 비교 캘린더 기간은 두 개의 필드를 사용하여 집계된 날짜 필드에서 분석할 두 개의 별도의 기간에 플래그를 지정합니다.

예 앱 데이터에는 다른 회계 분기에 대한 별도의 필드가 있습니다. 각 필드에는 해당 날짜가 분기에 있는지 여부를 나타내는 이진 데이터가 있습니다. 이를 *YearMonth*와 함께 사용하여 플래그 비교 캘린더 기간을 만듭니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. *OrderDate*를 선택합니다.
3. **캘린더 기간 이름**에 *Q4 to Q3*를 입력합니다.
4. **집계 날짜**에서 *YearMonth*를 선택합니다.
5. **플래그**를 클릭합니다.

6. **현재 기간 플래그**에서 *Q4-2020*을 선택합니다.
7. **기간 비교 플래그**에서 *Q3-2020*를 선택합니다.
8. **만들기**를 클릭합니다.

결과

통찰력으로 이동하여 *show me sales by customer*를 검색합니다.

sum(Sales) by customer 차트를 선택하고 분석 기간 *Q4 to Q3*을 적용합니다. 차트가 업데이트되어 2020년 4분기와 2020년 3분기 간의 비교가 표시됩니다.

Q4와 Q3 간의 sum(Sales) 기준 고객

11.2 Insight Advisor용 어휘 만들기

Insight Advisor용 어휘를 만들 수 있습니다. 이를 통해 특정 용어에 사용할 분석을 정의하고 Insight Advisor에 대한 동의어를 정의할 수 있습니다.

준비 탭에서 **비즈니스 논리** 아래 **어휘**에서 어휘를 만듭니다. 어휘를 사용하면 자연어 쿼리 기능이 향상됩니다. 예를 들어 다음과 같은 목적으로 어휘를 사용할 수 있습니다.

- 필드, 마스터 항목 및 값에 대한 대체 이름을 추가합니다.
- 코딩된 값의 이름을 정의합니다.
- 특정 용어 또는 쿼리와 함께 사용할 Insight Advisor 분석 유형을 정의합니다.

어휘는 두 가지 종류의 어휘를 지원합니다.

- **동의어**: 동의어를 사용하면 대체 용어를 추가할 수 있습니다.
  자세한 내용은 *Insight Advisor에 동의어 추가 (page 282)*을 참조하십시오.

- **사용자 지정 분석**: 사용자 지정 분석을 사용하면 선택한 용어와 함께 사용해야 하는 분석을 정의할 수 있습니다.
  자세한 내용은 *Insight Advisor에 사용자 지정 분석 추가 (page 283)*을 참조하십시오.

제한 사항

어휘에는 다음 제한 사항이 있습니다.

- 어휘는 Associative Insights에서 지원되지 않습니다.
- 어휘는 날짜로 분류된 필드에 대한 용어 생성을 지원하지 않습니다.
- 다음 단어는 Qlik Sense에서 사용하도록 예약되어 있으며 어휘 용어로 사용할 수 없습니다.
  - 앱
  - 응용 프로그램
  - 차원
  - Dim
  - 측정값
  - Msr
- Insight Advisor는 어휘 용어를 분류 정리하거나 어간을 제거하지 않습니다. 예를 들어 용어가 *open*인 경우 *opened* 또는 *opening*을 사용한 쿼리를 *open*을 입력한 것으로 처리하지 않습니다. 마찬가지로 용어가 *opened*인 경우 *open* 및 *opening*은 용어로 인식되지 않습니다.
- 또한 어휘 용어가 필드 값으로 존재하고 쿼리에 해당 값과 공유 용어가 포함된 필드가 포함되면 Insight Advisor는 어휘 용어를 사용하지 않고 쿼리가 필드를 포함하므로 쿼리를 처리할 때 필드 값을 대신 사용합니다.

Insight Advisor에 동의어 추가

Insight Advisor용 어휘를 만들 수 있습니다. 이렇게 하면 데이터 모델에 없는 용어 및 값을 정의하여 쿼리에 사용할 수 있습니다.

준비 탭에서 **비즈니스 논리** 아래 **어휘**에서 어휘를 만들 수 있습니다. 어휘를 사용하면 자연어 쿼리 기능이 향상됩니다. 예를 들어 어휘를 사용하여 다음을 정의할 수 있습니다.

- 필드, 마스터 항목 및 값의 대체 이름
  예: *Income* 필드의 경우 *Earnings*, *Proceeds* 및 *Revenue*.
- 코딩된 값의 이름
  예: 의학 분류 코드의 이름
- 필드의 값 그룹에 대한 이름
  예: 나이가 포함된 필드의 명명된 연령 범위 정의
- 데이터에 없는 일반 머리글자 및 약어
  예: *year*의 경우 *yr* 또는 *number*의 경우 *num*.

어휘를 만들 때 해당 어휘와 관련된 용어를 정의합니다. 그런 다음 용어를 앱의 필드 및 마스터 항목에 연결합니다. 경우에 따라 용어를 선택한 필드 및 마스터 항목의 개별 값에 연결할 수 있습니다. 필드의 값에 대해 용어를 정의할 때 다음 조건을 사용할 수 있습니다.

- 보다 큼
- 보다 크거나 같음
- 보다 작음
- 보다 작거나 같음
- In
- 속하지 않음
- 범위에 있음

필드의 값에 따라 필드에 사용할 수 있는 조건이 달라집니다.

어휘는 앱의 언어로 생성됩니다. 각 언어의 어휘는 별개입니다.

Qlik Sense에서는 어휘가 영어, 프랑스어, 러시아어 및 스페인어로 지원됩니다.

다음과 같이 하십시오.

1. 앱에서 **준비**를 클릭하고 **어휘**를 선택합니다.
2. 언어 드롭다운에서 언어를 선택합니다.
3. **용어 만들기**를 클릭합니다.
4. 어휘에 대한 용어를 입력합니다.
5. 어휘가 적용되는 필드 또는 마스터 항목을 선택합니다.
6. 필요한 경우 조건 및 조건을 적용할 값을 선택합니다.
7. **만들기**를 클릭합니다.

제한 사항

동의어에는 다음 제한 사항이 있습니다.

- 마스터 항목 및 임시 필드는 조건을 지원하지 않습니다.

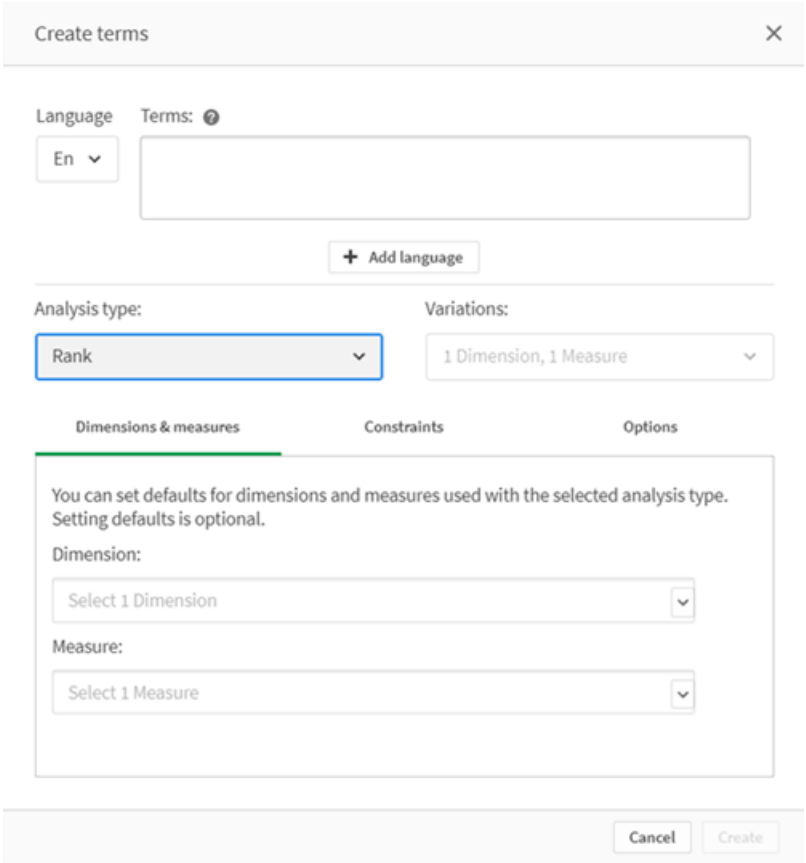
Insight Advisor에 사용자 지정 분석 추가

특정 용어에 사용할 특정 분석을 정의할 수 있습니다. 이를 통해 사용자가 Insight Advisor에서 받는 분석을 제어하고 선호하는 분석을 만들 수 있습니다.

사용자 지정 분석을 사용하면 특정 용어에 대한 응답을 정의하고 Insight Advisor이 반환하는 분석을 제어할 수 있습니다. 사용자 지정 분석에 대해 여러 용어를 정의할 수 있습니다. 사용자 지정 분석을 위해 다른 언어로 다른 용어를 정의할 수도 있습니다.

Qlik Sense에서는 어휘가 영어, 프랑스어, 러시아어 및 스페인어로 지원됩니다.

사용자 지정 분석



사용자 지정 분석을 만들 때 분석에 용어를 추가한 다음 분석 유형을 선택합니다. 그런 다음 기본 차원 또는 측정값을 지정할 수 있습니다. 기본값이 설정되지 않은 경우 Insight Advisor은 사용자 쿼리를 기반으로 분석에 사용할 적절한 차원 또는 측정값을 결정합니다.

선택적으로 분석 유형에 대한 제약 조건을 설정하여 범위를 제한할 수 있습니다. 예를 들어 특정 측정값을 초과하는 값만 분석에 포함할 수 있습니다.

필드의 값에 따라 필드에 사용할 수 있는 조건이 달라집니다.

분석 유형에 사용할 수 있는 차트 유형 중에서 분석에 사용할 차트 유형을 선택할 수 있습니다. 차트 유형을 선택하면 다른 차트가 선택한 데이터에 더 적합하다고 결정하지 않는 한 사용자 지정 분석은 항상 해당 차트 유형을 사용합니다. 차트가 적용되도록 선택할 수 있으므로 쿼리에 더 적합한 차트가 있는 경우에도 차트 유형만 사용됩니다. 예를 들어 원형 차트를 적용할 수 있습니다. 검색에서 200개의 고유 값이 포함된 차원을 사용하는 경우 데이터에 대한 최적의 차트가 아니더라도 원형 차트가 사용됩니다. 일부 분석 유형을 사용하면 차트에 대한 정렬 옵션을 설정하고 선택적으로 표시되는 값 수에 대한 제한을 적용할 수 있습니다.

분석 유형

다음 표에는 사용자 지정 분석을 수행할 때 사용할 수 있는 분석 유형이 요약되어 있습니다. 일부 분석에서는 선택할 수 있는 여러 가지 변형이 있습니다. 이러한 변형은 분석에 사용되는 필드의 종류를 결정합니다.

분석 유형

분석 유형	아이콘	변형
이상	기록 데이터의 패턴을 분석하여 스파이크 변칙을 식별합니다.	1개 측정값, 1개 임시 차원
분석	여러 차원으로 측정값을 분석합니다.	2~3개 차원, 1개 측정값
클러스터 (K-평균)	**KMeans2D** 함수를 사용하여 결과를 클러스터링하여 임의 차원에서 두 측정값을 비교합니다.	1개 차원, 2개 측정값
비교	1개 또는 2개 차원에서 2~4개의 측정값을 비교합니다. 두 가지 변형이 지원됩니다. • 1개 차원, 2~4개의 측정값 • 2개 차원, 2~3개의 측정값	1개 차원, 2~4개의 측정값 2개 차원, 2~3개의 측정값
기여	누적 기여도를 사용하여 순위를 만듭니다(파레토 차트).	1개 차원, 1개 측정값
상관 관계	최대 2개의 선택적 차원에 대한 두 측정값 간의 상대적 상관 관계를 보여 줍니다.	0~2개 차원, 2개 측정값
팩트	집계된 합계를 표시합니다.	측정값 1개
맵 분포	여러 지역적 차원으로 측정값을 분석합니다.	0~2개 측정값, 1-2개 지역적 차원
상호 정보	선택한 항목과 대상 간의 통계적 종속성을 표시합니다. 종속성 표시기 범위는 0%(종속성 없음)에서 100%(강력한 종속성)입니다. 하나의 필드(측정값 또는 차원)를 대상으로 선택한 다음 1~10개의 차원 또는 측정값을 동인으로 선택할 수 있습니다. 상호 정보는 무작위로 선택된 데이터 샘플을 사용합니다. 동일한 필드 또는 선택 항목에 대한 이 분석 유형의 결과는 결과적으로 다를 수 있습니다.	2~11개 필드
개요	여러 차원별 측정값 개요를 표시합니다.	1~2개 차원, 1개 측정값
기간 대비 기간	현재 기간과 이전 기간의 측정값을 비교합니다.	측정값 1개
프로세스 제어(평균)	계산된 두 제어 제한 사이에서 측정값 성능 표시를 보여 줍니다.	1개 측정값, 1개 임시 차원
프로세스 제어(롤링 평균)	지난 7개 기간 동안 계산된 두 제어 제한 사이에서 측정값 성능 표시를 보여 줍니다.	1개 측정값, 1개 임시 차원

분석 유형	아이콘	변형
순위	선택적 그룹화를 사용한 측정값으로 차원 값의 순위를 지정합니다.	1개 차원, 1개 측정값
순위가 매겨진 그룹	시간에 따라 측정값에 대한 하나 이상의 차원에 대한 중첩된 순위를 표시합니다.	3개 차원, 1개 측정값
추세	집합 크기가 작은 차원별로 선택적으로 분석된 시간 경과에 따른 측정값의 성능을 표시합니다. 두 가지 변형이 지원됩니다. • 0~1개 차원, 1개 측정값, 1개 날짜/시간 차원 • 2~3개 측정값, 1개 날짜/시간 차원	0~1개 차원, 1개 측정값, 1개 임시 차원 2~3개 측정값, 1개 임시 차원
연간 누계	차원을 기준으로 측정값(작년 및 올해의 결과)을 분석합니다.	1개 차원, 1개 측정값, 1개 임시 차원

사용자 지정 분석 만들기

다음과 같이 하십시오.

1. 앱에서 **준비**를 클릭하고 **어휘**를 선택합니다.
2. **사용자 지정 분석**을 클릭합니다.
3. **용어 만들기**를 클릭합니다.
4. 언어 드롭다운에서 언어를 선택하고 사용자 지정 분석을 위한 용어를 입력합니다.
5. 선택적으로 다른 언어를 추가하고 해당 언어에 대한 용어를 추가합니다.
6. **분석 유형**에서 분석 유형을 선택합니다.
7. 가능한 경우 사용할 변형을 선택합니다.
8. 분석을 위한 차원 및 측정값을 지정하려는 경우 **데이터**에서 사용할 기본 필드를 선택합니다.
9. 분석에 제약 조건을 적용하려면 **제약 조건**에서 필드, 조건 및 값을 선택합니다.
10. **차트**의 **차트 유형**에서 사용자 지정 분석에 사용할 기본 차트를 선택적으로 선택합니다.
    다른 차트가 더 적절할 때도 이 차트 유형을 사용하려면 **차트 선택 적용**을 선택합니다.
11. 분석 유형에 사용할 수 있는 경우 차트의 정렬 및 제한을 설정합니다.
12. **만들기**를 클릭합니다.
13. Insight Advisor Chat에 앱을 사용할 수 있는 경우 탐색 막대에서 앱 이름을 클릭한 다음 **Insight Advisor 사용** 아래에서 **허브에서**를 껐다가 켭니다.

제한 사항

사용자 지정 분석에는 다음과 같은 제한 사항이 있습니다.

- 다음 시나리오에서는 **차트 선택 적용** 설정을 따르지 않습니다.
    - 선택한 차트 유형이 지원하는 것보다 지원되지 않는 수의 필드가 사용자 지정 분석에 추가되는 경우.

예를 들어, 일부 차트는 2개의 차원을 지원하지만 3개의 차원은 지원하지 않습니다. 2개의 차원으로 세부 분석을 만들고 mekko 차트를 적용하면 예상대로 작동합니다. 3개의 차원으로 세부 분석을 만들고 mekko 차트를 선택하면 mekko 차트는 3개의 차원을 지원하지 않으므로 사용되지 않습니다.

  - 사용자가 차원 제한을 초과하도록 하는 제약 조건을 차원에 추가하는 경우.
    예를 들어 차원 범주 및 제품을 사용하여 세부 분석을 만들고 mekko 차트를 선택하고 적용합니다. 이는 예상대로 작동합니다. 차원 국가에 대한 제약 조건을 추가하면 이제 차트에 3개의 차원이 사용되므로 mekko 차트가 적용되지 않습니다.
  - 분산 폭포형 차트를 선택했지만 적용된 캘린더 기간이 없는 경우.

- Insight Advisor Chat에서 후속 질문은 사용자 지정 분석에서 작동하지 않습니다. 예를 들어 *sales summary*, 사용자 지정 분석을 쿼리한 다음 *what about profit*과 같은 후속 질문을 하면 Insight Advisor Chat은 새로운 질문으로 처리됩니다.
- 로드 스크립트에 `$hidden` 태그가 지정된 필드는 논리 모델에서 표시되도록 설정되어 있어도 사용자 지정 분석에서 사용할 수 없습니다.

11.3 자습서 - Insight Advisor에서 데이터 해석 방법 사용자 지정

자습서에 오신 것을 환영합니다. 이 자습서에서는 Insight Advisor에서 Qlik Sense 데이터 해석 방법을 사용자 지정하는 것을 소개합니다.

데이터를 앱에 로드한 후 Insight Advisor에서 데이터를 해석하는 방법을 사용자 지정할 수 있습니다. Insight Advisor에서 선례 기반 학습을 통해 데이터 모델의 관계를 해석하는 방법을 배울 수 있습니다. 데이터 모델에 익숙하고 필드 사용 방법을 알고 있는 경우 대신 비즈니스 논리에서 Insight Advisor에 대한 논리 모델을 정의할 수 있습니다. 이를 통해 Insight Advisor에서 앱 사용자에게 가장 관련성이 높은 결과를 제공하도록 할 수 있습니다.

학습 내용

이 자습서를 완료한 후에는 비즈니스 논리를 사용하여 앱의 논리 모델을 정의하는 데 익숙해져야 합니다. 비즈니스 논리가 Insight Advisor 결과에 미치는 영향을 이해하게 됩니다.

이 자습서의 대상

Qlik Sense 기본 사항에 익숙해야 합니다. 예를 들어 데이터를 로드하고 앱을 만들고 시트에 시각화를 만들 수 있어야 합니다.

데이터 관리자에 대한 액세스 권한이 필요하며 Qlik Sense Enterprise on Windows에서 데이터를 로드할 수 있어야 합니다.

준비 사항

이 앱을 다운로드하고 Qlik Sense에 QVF 파일을 업로드합니다.

통찰력 및 비즈니스 논리 자습서 앱

이 앱은 데이터 파일 *TutorialData.xlsx*를 포함합니다. 앱 Qlik Sense를 업로드합니다. 앱을 열고 데이터 로드 편집기로 이동합니다. **데이터 로드**를 클릭하여 앱 데이터를 로드합니다.

캘린더 기간을 통해 가능한 상대적 비교를 보여 주기 위해 TutorialData.xlsx에는 미래 날짜에 대한 데이터가 포함됩니다. 앱 로드 스크립트는 로드될 때 현재 날짜에 대한 앱 내 데이터를 업데이트합니다. 이 자습서의 스크린샷은 데이터를 로드하는 날짜에 따라 다릅니다.

이 자습서의 단원

이 자습서의 항목은 차례로 완료할 수 있도록 설계되었습니다. 그러나 중단했다가 언제든지 다시 돌아올 수 있습니다. 이 자습서의 스크린샷은 Qlik Sense SaaS에서 만든 것입니다. 이 자습서는 Qlik Sense SaaS에 초점을 맞추고 있으므로 다른 환경에서 Qlik Sense를 사용하는 경우 약간의 차이가 발생할 수 있습니다.

- *Insight Advisor 및 비즈니스 논리란? (page 288)*
- *사용자 지정 논리 모델 활성화 (page 291)*
- *필드 및 그룹 사용자 지정 (page 293)*
- *패키지 구성 (page 300)*
- *계층 구조 정의 (page 301)*
- *캘린더 기간 구성 (page 304)*
- *동작 구성 (page 308)*
- *어휘 만들기 (page 311)*

추가 자료 및 리소스

- Qlik에서는 보다 자세한 정보를 알아볼 수 있도록 다양한 리소스를 제공합니다.
- Qlik 온라인 도움말을 사용할 수 있습니다.
- Qlik Continuous Classroom에서 교육(무료 온라인 과정 포함)이 제공됩니다.
- Qlik Community에서 토론 포럼, 블로그 등을 찾을 수 있습니다.

Insight Advisor 및 비즈니스 논리란?

Insight Advisor은 Qlik Sense 기능의 모음입니다. Insight Advisor은 데이터 모델을 구축하고, 시각화를 만들고, 데이터를 분석합니다. Insight Advisor에서 비즈니스 논리를 사용하여 데이터를 해석할 수 있습니다.

Insight Advisor

Insight Advisor은 다음 Qlik Sense 기능으로 구성됩니다.

- Insight Advisor 검색 기반 분석: 검색 기반 분석은 앱의 **분석** 탭에 있는 **통찰력**에서 사용할 수 있습니다. 검색 기반 분석을 통해 자연어 검색 또는 선택한 필드 및 마스터 항목을 기반으로 하는 시각화가 만들어집니다. Insight Advisor 검색 기반 분석을 통해 잠재적인 관심 분야 차트를 생성할 수도 있습니다.
- Insight Advisor Chat: Insight Advisor Chat은 대화형 분석을 위한 채팅 기반 인터페이스입니다. Insight Advisor Chat 그러면 Insight Advisor Chat에서 관련 시각화를 반환합니다.
- Associative Insights: Associative Insights를 통해 맹점을 알아내고 놓쳤을 수 있는 관계를 표시할 수 있습니다. Associative Insights는 측정값에 대해 선택 값과 제외된 값의 기여를 비교할 수 있습니다.

- 차트 제안: 차트 제안을 사용하여 시트를 편집할 때 데이터 필드를 선택할 수 있으며 Qlik Sense에서 차원, 측정값 및 시각화 유형을 선택하도록 할 수 있습니다. 제안된 차트는 변경 내용에 따라 자동으로 조정됩니다. 초점이 있는 속성 집합으로 제안된 시각화를 사용자 지정할 수 있습니다.
- 권장 연결: Insight Advisor은 **데이터 관리자**의 **연결** 보기에서 데이터 테이블 간 연결을 권장할 수 있습니다. **권장 연결** 패널을 통해 이러한 권장 사항을 보고 적용할 수 있습니다.

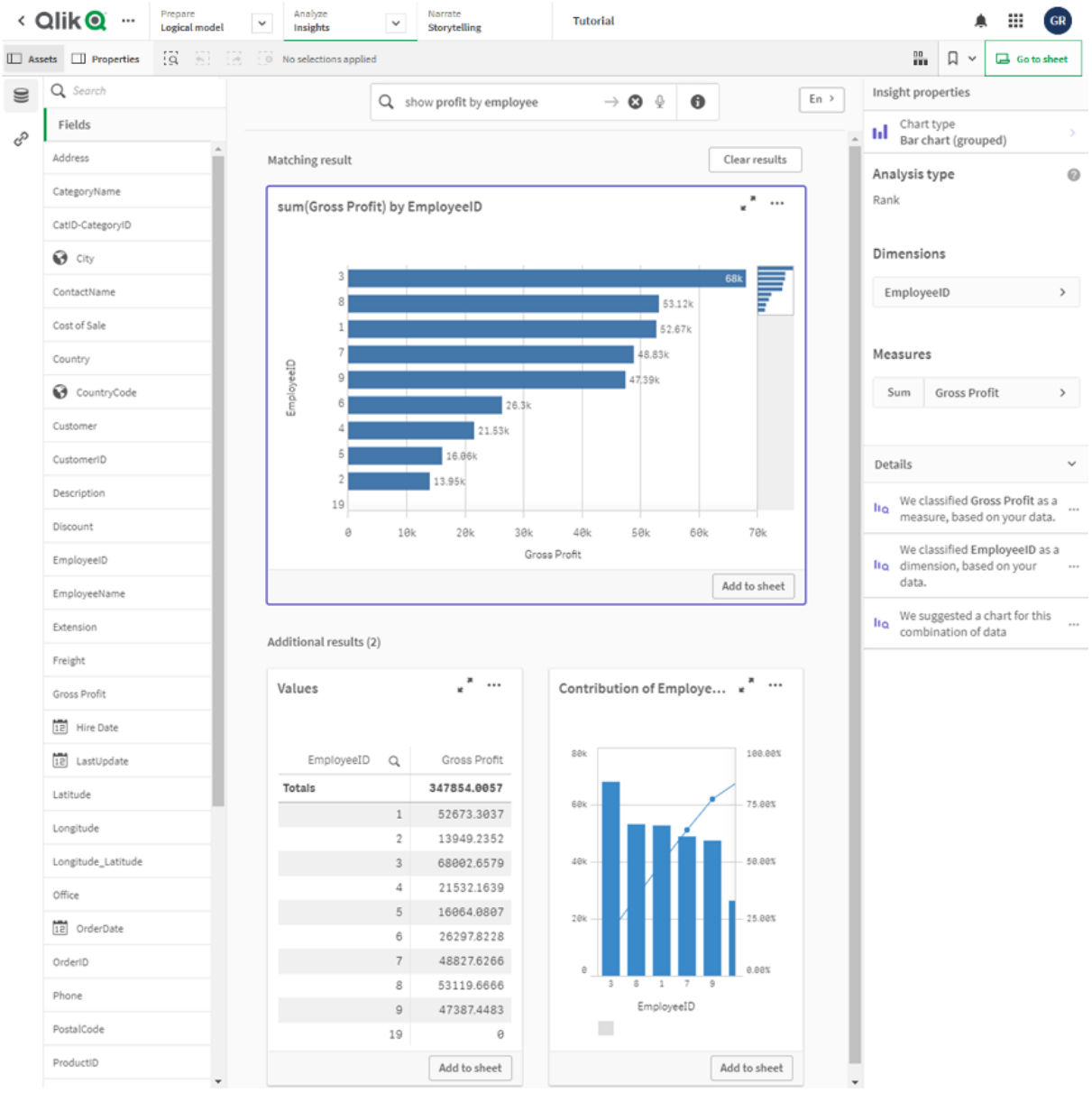
이 자습서는 비즈니스 논리를 사용하여 검색 기반 분석을 개선하는 데 중점을 두었습니다.

Insight Advisor 검색 기반 분석

Insight Advisor 검색 기반 분석은 앱의 **분석** 탭에 있는 **통찰력**에서 사용할 수 있습니다. **통찰력**에서 *what is profit by employee for 2020*과 같은 자연어 검색을 입력합니다. 특정 데이터 자산을 선택할 수도 있습니다. 그런 다음 Insight Advisor은 앱에서 차트를 생성하거나 일치하는 차트를 찾습니다. 앱을 만드는 경우 Insight Advisor 검색 기반 분석을 통해 답하려는 질문을 기반으로 앱을 만들 수 있습니다. 앱을 사용하는 경우 Insight Advisor을 통해 질문에 답할 수 있는 올바른 시각화를 앱에서 찾거나 사용자에 맞게 만들어 찾을 수 있습니다.

예를 들어 각 직원이 얻은 수익을 표시하는 일부 시각화를 보려고 합니다. Qlik Sense에서 자습서 앱을 엽니다. **통찰력**에서 *직원별 수익 표시*를 검색합니다. Insight Advisor에서 결과를 생성합니다.

show profit by employee 결과



*EmployeeID*별 *총수익*의 합계와 일부 추가 결과를 표시하는 하나의 일치 결과가 있습니다.이 차트를 앱의 시트에 추가할 수 있습니다.

이러한 결과는 찾고 있던 것과 일치했지만 선택한 필드를 개선할 수 있습니다. 데이터 모델에서 *직원*과 일치하는 필드를 선택하려고 할 때 Insight Advisor이 *EmployeeID*를 선택했습니다. 그러나 *EmployeeName* 필드가 분석에 더 유용합니다. 차트의 *총수익* 합계도 전체 시간에 대한 것입니다. 이것은 잘못된 것은 아니지만 사용자는 기본적으로 최신 데이터를 보려고 할 수 있습니다.

통찰력 속성에서 해당 차원을 선택하여 *EmployeeName*을 선호한다고 Insight Advisor에게 알려줄 수 있습니다.이 선례는 사용자별로 학습됩니다. 그러나 앱 개발자는 데이터 모델을 해석하고 해당 필드를 사용하는 방법을 Insight Advisor에 정확히 표시하는 모델을 만들 수 있습니다. Insight Advisor은 앱을 사용하는 모든 사용자에 대해 이 논리 모델을 사용합니다. 비즈니스 논리를 사용하여 이를 수행할 수 있습니다.

비즈니스 논리

비즈니스 논리는 데이터 모델을 해석하는 일부 Insight Advisor 기능에서 사용하는 논리 모델을 정의하는 데이터 준비의 선택적 부분입니다. 비즈니스 논리는 두 가지 주요 기능으로 구성됩니다.

- **논리 모델**: 시각화를 생성할 때 사용되는 앱의 데이터 모델입니다.
- **어휘**: 자연어 쿼리에 대한 대체 용어

Insight Advisor은 다음 Insight Advisor 기능에서 비즈니스 논리를 사용합니다.

- Insight Advisor 검색 기반 분석
- Insight Advisor Chat
- Associative Insights

논리 모델

논리 모델은 Insight Advisor이 시각화를 만들 때 사용하는 개념적 모델입니다. 앱의 데이터 모델로부터 구축되며 각 앱에는 단일 논리 모델이 있습니다. 필드 및 마스터 항목은 이 모델의 핵심 구성 요소입니다. 필드 및 마스터 항목은 그룹으로 구성됩니다. 그룹은 필드 또는 마스터 항목 간의 개념적 연결 또는 관계를 나타냅니다. 이를 정의한 다음 다른 관계와 처리 동작을 지정할 수 있습니다. 여기에는 다음 사항이 포함됩니다.

- **패키지**: **패키지**를 사용하면 관련 그룹의 모음을 만들 수 있습니다. 이렇게 하면 동일한 패키지에 없는 그룹이 함께 사용되는 것을 방지할 수 있습니다.
- **계층 구조 계층 구조**를 사용하면 그룹 간의 드릴다운 분석 관계를 정의할 수 있습니다.
- **캘린더 기간**: **캘린더 기간**을 사용하여 Insight Advisor에 대한 기본 분석 기간을 만들 수 있습니다.
- **동작**: **동작**을 사용하면 필드 간의 선호 또는 거부 관계를 지정할 수 있습니다. 동작은 필수 선택 사항과 기본 캘린더 기간을 적용할 수도 있습니다.

어휘

비즈니스 논리 어휘를 사용하면 자연어 쿼리 기능이 향상됩니다. 어휘를 사용하면 데이터 모델에 없는 용어 및 값을 정의하여 자연어 쿼리에 사용할 수 있습니다. 특정 용어가 포함된 쿼리에 사용할 분석 유형을 설정할 수도 있습니다. 예를 들어 다음과 같은 목적으로 어휘를 사용할 수 있습니다.

- 필드, 마스터 항목 및 값에 대한 대체 이름을 추가합니다.
  예: *Income* 필드의 경우 *Earnings*, *Proceeds* 및 *Revenue*.
- 코딩된 값의 이름을 정의합니다.
  예: 의학 분류 코드의 이름
- 데이터에 없는 일반 머리글자 및 약어
  예: *year*의 경우 *yr* 또는 *number*의 경우 *num*.
- 특정 용어 또는 쿼리와 함께 사용할 Insight Advisor 분석 유형을 정의합니다.
  예를 들어 쿼리에 *regional view*라는 용어가 포함된 경우 맵 분포 분석 유형이 Insight Advisor에서 항상 사용되도록 설정합니다.

사용자 지정 논리 모델 활성화

Insight Advisor에서 데이터 해석 방법을 사용자 지정하는 첫 번째 단계에서 비즈니스 논리 모델을 활성화합니다.

활성화되면 Qlik Sense는 데이터 모델을 기반으로 비즈니스 논리에 대한 논리 모델을 만듭니다. 그런 다음 이 논리 모델을 사용자 지정할 수 있습니다.

> ***논리 비활성화/논리 활성화****를 클릭하여 논리 모델을 켜기 및 끄기로 전환할 수 있습니다. 이는 모델이 적용되지 않은 결과에 적용된 모델과 결과를 비교할 때 유용합니다.*

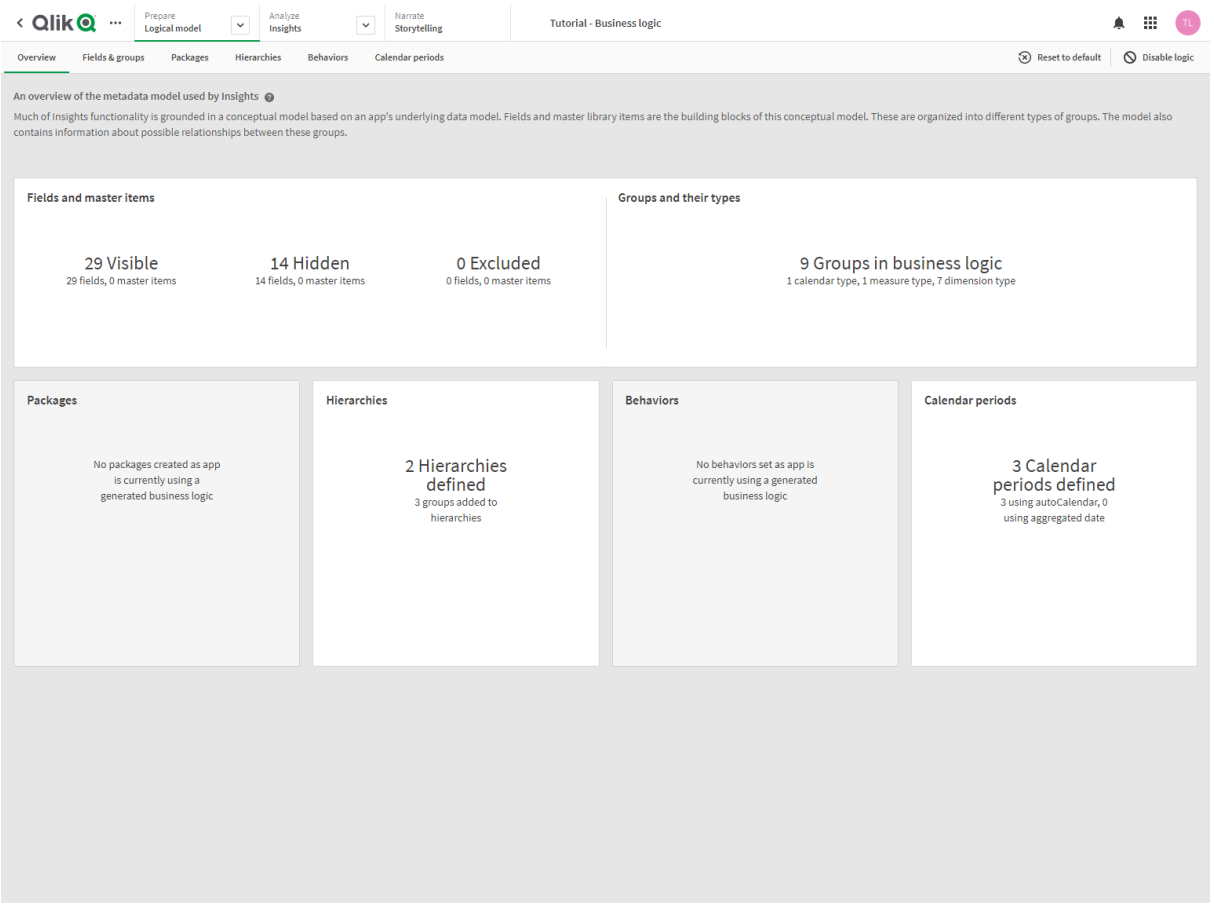
다음과 같이 하십시오.

1. 자습서 앱에서 **준비**를 클릭합니다.
2. **비즈니스 논리**에서 **논리 모델**을 선택합니다.
3. **계속**을 클릭합니다.

결과

비즈니스 논리는 데이터 모델을 분석하고 그룹, 계층 구조 및 동작을 자동으로 만듭니다. **개요**에서 새로운 논리 모델을 볼 수 있습니다.

개요*의 새로운 논리 모델*



Qlik Sense가 분석에서 14개의 필드를 숨겼으므로 Insight Advisor에서 더 이상 사용할 수 없습니다. 관련 필드를 함께 연결하여 데이터 모델에서 9개의 그룹을 만들었습니다. 드릴다운 분석을 위해 그룹을 연결하여 2개의 계층 구조를 정의했습니다. 분석에 사용할 기간을 제공하여 3개의 캘린더 기간을 정의했습니다.

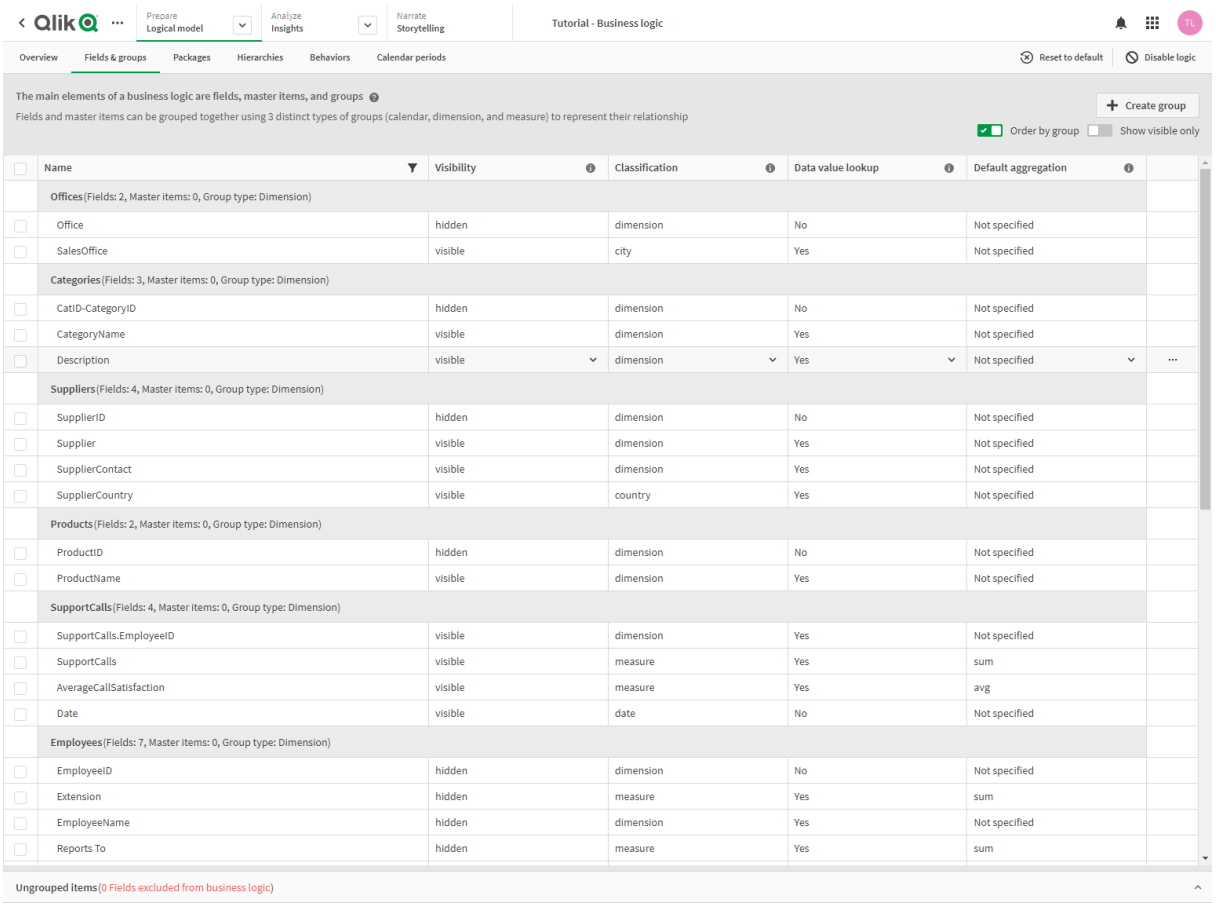
다음으로 기본 필드 및 그룹을 수정하여 이 기본 논리 모델을 편집합니다.

필드 및 그룹 사용자 지정

다음 단계에서 논리 모델을 사용자 지정하여 필드와 그룹을 정의합니다. 필드와 그룹은 논리 모델의 기본 수준입니다. 다른 모든 논리 모델 요소는 이러한 그룹을 사용합니다.

사용자 지정 논리 모델을 활성화하면 Qlik Sense는 데이터 모델에서 기본 그룹을 자동으로 만듭니다. 또한 모든 필드의 속성을 설정합니다. 이러한 그룹 및 필드 속성 중 일부를 조정해야 합니다. Qlik Sense는 데이터 모델을 가장 잘 해석하지만, 어떤 필드를 숨겨야 하는지 또는 어떤 필드를 측정해야 하는지 등에 대해서는 항상 정확한 것은 아닙니다.

필드 및 그룹*(기본 그룹 있음)*



다음과 같이 세 가지 그룹이 있습니다.

- **차원**: 차원 그룹은 차원으로 분류되는 관련 필드로 구성됩니다. 또한 차원 그룹에는 측정값 또는 날짜로 분류된 필드가 포함될 수 있습니다.
- **측정값**: 측정값 그룹은 관련된 측정값 필드로 구성됩니다. 측정값만 측정값 그룹에 속할 수 있습니다.
- **캘린더**: 캘린더 그룹에는 논리 모델의 시간 차원이 포함됩니다. 캘린더 그룹에는 차원만 포함될 수 있으며 하나 이상의 임시 필드(예: 날짜, 타임스탬프 또는 연도)가 있어야 합니다.

각 그룹에는 다음 속성이 있는 필드가 있습니다.

- **가시성**: 필드가 Insight Advisor에 표시되는지 여부를 정의합니다. 숨겨진 필드는 사용자가 해당 필드를 검색할 때와 같이 분석에 계속 사용할 수도 있습니다.
- **분류**: 필드가 분석에서 수행하는 기본 역할을 정의합니다. 분류는 **차원**과 같이 광범위하거나 **도시**와 같이 구체적일 수 있습니다.
- **데이터 값 조회**: 이 필드에서 특정 값을 검색할 수 있는지 여부를 제어합니다. 데이터 값 조회를 활성화한 필드 수를 줄이면 가양성 결과를 방지하고 쿼리 시간을 줄일 수 있습니다.
- **기본 집계**: Insight Advisor에서 측정값에 대한 표준 집계를 설정합니다. 필드에 기본 집계가 있는 경우 Insight Advisor은 측정값으로 사용할 때 항상 해당 집계를 적용합니다. 사용자는 통찰력 차트를 편집하여 Insight Advisor에서 집계를 다른 유형으로 변경할 수 있습니다.
- **유리한 추세**: 유리한 추세가 측정값이 상승하는 것인지 하락하는 것인지 대해 정의합니다.
- **전체 집계**: Insight Advisor가 복잡한 표현식이 있는 마스터 측정값과 관련된 쿼리에 사용할 집계를 스스로 결정할 수 없을 때 사용해야 하는 집계를 정의합니다. 마스터 측정값에 대한 전체 집계만 설정할 수 있습니다.
- **즐겨찾기**: **생성**과 같이 Insight Advisor에서 쿼리 또는 필드 선택 없이 분석을 생성할 때 분석에서 더 자주 사용해야 하는 측정값을 정의합니다.
- **기본 기간 단위**:분석에서 날짜 필드와 함께 사용할 기본 단위를 설정합니다.

을 클릭하여 테이블에 표시할 열을 설정할 수 있습니다.

필드를 그룹 해제할 수 있습니다. 그룹 해제된 필드는 그룹에 속하지 않으며 분석에서 제외됩니다.

필드와 그룹을 정의하는 이유

필드와 그룹을 정의하면 필드 및 마스터 항목이 처리되는 방법의 기본값을 설정할 수 있습니다. 필드를 정의할 때 Insight Advisor에서 처리하는 방법의 기본 설정을 지정합니다. 예:

- 이 필드를 Insight Advisor에서 분석에 사용할 수 있어야 합니까?
- 이 필드를 차원 또는 측정값으로 사용해야 합니까? 어떤 종류의 차원 또는 측정값입니까?
- Insight Advisor이 검색에서 필드의 개별 값을 조회할 수 있습니까?
- 측정값으로 사용할 때 필드에 대한 기본 집계는 어떻게 해야 합니까?

그룹을 정의할 때 Insight Advisor에서 어떤 필드가 밀접하게 관련되어 있고 분석에 사용되어야 하는지 표시합니다. 그룹은 패키지, 캘린더 기간 및 동작 등의 다른 비즈니스 논리 기능을 만드는 데 사용됩니다.

필드 및 그룹 사용자 지정

먼저 가시성을 변경하고 몇 가지 분류를 조정하기 위해 일부 필드 속성을 정리합니다. 그런 다음 그룹에서 일부 필드의 그룹을 해제합니다. 이러한 필드를 사용하여 다음과 같은 새 그룹을 만듭니다.

- *Customer*
- *SalesCity*
- *SalesCountry*
- *SupportCalls*
- *SupportDate*

일부 필드는 그룹 해제된 상태로 둡니다. Insight Advisor은 모든 분석에서 그룹 해제된 필드를 사용하지 않습니다. 따라서 ID 및 GeoInfo 필드와 같이 분석과 관련 없는 필드가 제거됩니다.

필드 속성 사용자 지정

다음과 같이 하십시오.

1. **필드 및 그룹**으로 이동합니다.
2. *Employees*에서 다음 필드를 조정합니다.
    - *Extension*의 경우 다음 속성을 설정합니다.
        - **가시성**: **표시**
        - **분류**: **차원**
        - **데이터 값 조회**: **아니요**
        - **기본 집계**: **지정되지 않음**
    - *EmployeeName*의 경우 다음 속성을 설정합니다.
        - **가시성**: **표시**
        - **분류**: **차원**
        - **데이터 값 조회**: **예**
        - **기본 집계**: **지정되지 않음**
    - *Hire Date*의 경우 다음 속성을 설정합니다.
        - **가시성**: **표시**
    - *Reports To*의 경우 다음 속성을 설정합니다.
        - **가시성**: **숨김**
        - **분류**: **차원**
        - **데이터 값 조회**: **예**
        - **기본 집계: 지정되지 않음**
    - *Title*의 경우 다음 속성을 설정합니다.
        - **가시성**: **표시**
    - *Year Salary*의 경우 다음 속성을 설정합니다.
        - **가시성**: **표시**
        - **분류**: **화폐**
        - **데이터 값 조회**: **아니요**
        - **기본 집계**: **합계**

필드 그룹 해제

다음과 같이 하십시오.

1. *Categories*에서 *CatID-CategoryID*를 선택합니다.
2. *Suppliers*에서 *SupplierID*를 선택합니다.
3. *SupportCalls*에서 다음을 선택합니다:
    - *SupportCalls*
    - *AverageCallSatisfaction*
4. *Products*에서 *ProductID*를 선택합니다.

5. *Employees*에서 *EmployeeID*를 선택합니다.
6. *Sales Data_ENT*에서 다음 필드를 선택합니다:
    - *Address*
    - *City*
    - *ContactName*
    - *Country*
    - *CountryCode*
    - *Customer*
    - *CustomerID*
    - *Latitude*
    - *Longitude*
    - *Longitude_Latitude*
    - *OrderDate*
    - *OrderID*
    - *Phone*
    - *PostalCode*:
    - *ShipperID*
7. **그룹 해제**를 클릭합니다.

Customer 그룹 만들기

이 그룹을 사용하면 고객 정보와 관련된 차원 필드를 그룹화할 수 있습니다.

다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. **그룹 이름**의 경우 *Customer*를 입력합니다.
3. **그룹 유형**의 경우 **차원**을 선택합니다.
4. 그룹에 다음 필드를 추가합니다:
    - *Address*
    - *ContactName*
    - *Customer*
    - *Phone*
    - *PostalCode*
5. **만들기**를 클릭합니다.

SalesCity 그룹 만들기

이 그룹을 사용하면 도시 정보와 관련된 차원 필드를 그룹화할 수 있습니다. 나중에 계층을 만들 때 이것을 사용합니다.

다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. **그룹 이름**의 경우 City를 입력합니다.
3. **그룹 유형**의 경우 **차원**을 선택합니다.
4. 그룹에 다음 필드를 추가합니다:
    - *City*
    - *Longitude_Latitude*
5. **만들기**를 클릭합니다.

SalesCountry 그룹 만들기

이 그룹을 사용하면 국가 정보와 관련된 차원 필드를 그룹화할 수 있습니다. 나중에 계층을 만들 때 이것을 사용합니다.

다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. **그룹 이름**의 경우 *Country*를 입력합니다.
3. **그룹 유형**의 경우 **차원**을 선택합니다.
4. 그룹에 다음 필드를 추가합니다:
    - *Country*
    - *CountryCode*
5. **만들기**를 클릭합니다.

SupportCalls 그룹 만들기

차원 그룹으로 분류되었으므로 이전 *SupportCalls* 그룹에서 모든 필드를 제거했습니다. 필드를 사용하여 측정값 그룹을 만듭니다.

다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. **그룹 이름**의 경우 *SupportCalls*를 입력합니다.
3. **그룹 유형**의 경우 **측정값**을 선택합니다.
4. 그룹에 다음 필드를 추가합니다:
    - *AverageCallSatisfaction*
    - *SupportCalls*
5. **만들기**를 클릭합니다.

SupportDate 그룹 만들기

마지막으로 지원 호출 데이터에서 *Date* 필드에 대한 캘린더 그룹을 만듭니다.

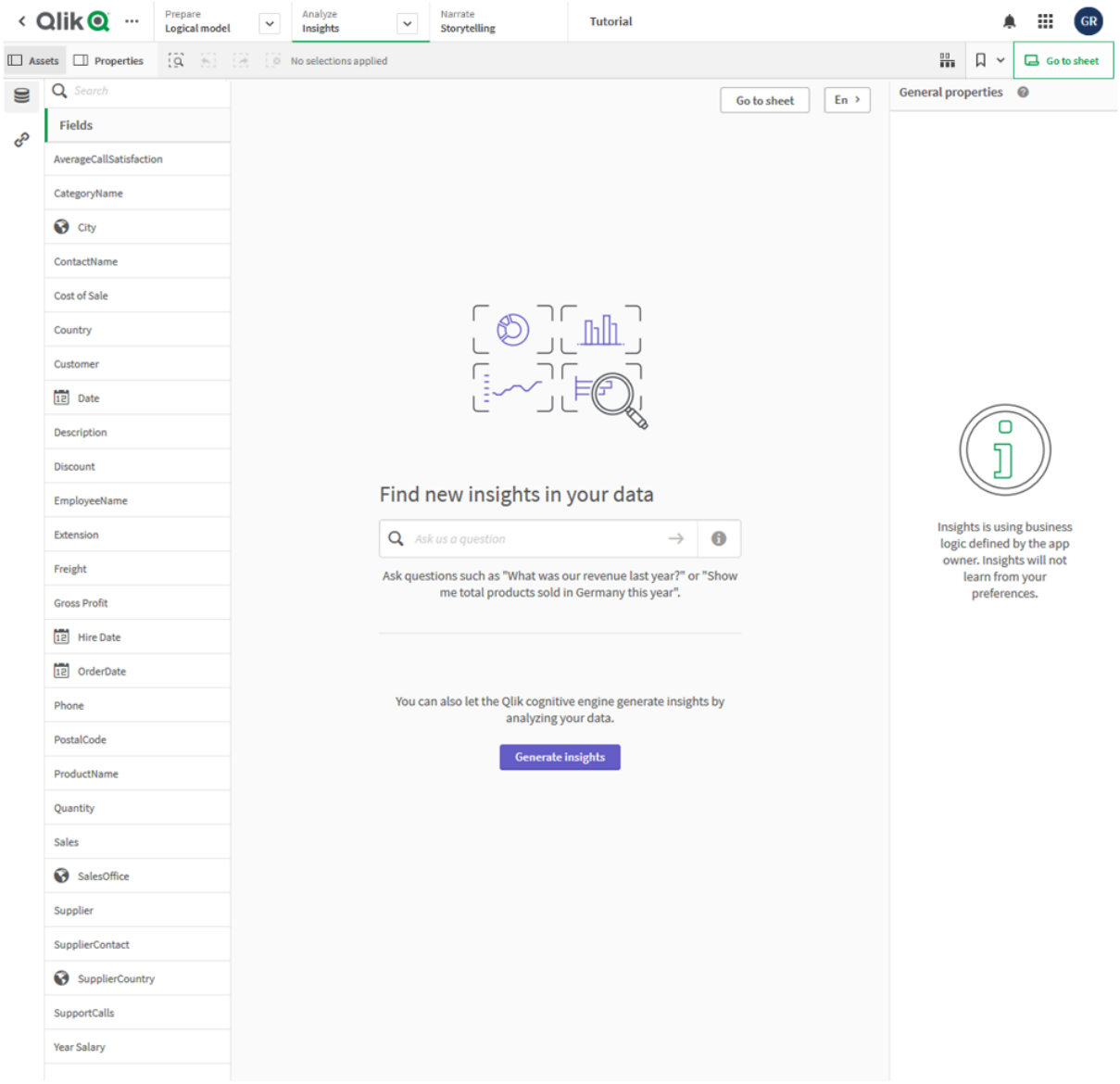
다음과 같이 하십시오.

1. **그룹 만들기**를 클릭합니다.
2. **그룹 이름**의 경우 *SupportDate*를 입력합니다.
3. **그룹 유형**의 경우 **캘린더**를 선택합니다.
4. 그룹에 다음 필드를 추가합니다:
    - *Date*
5. **만들기**를 클릭합니다.

결과

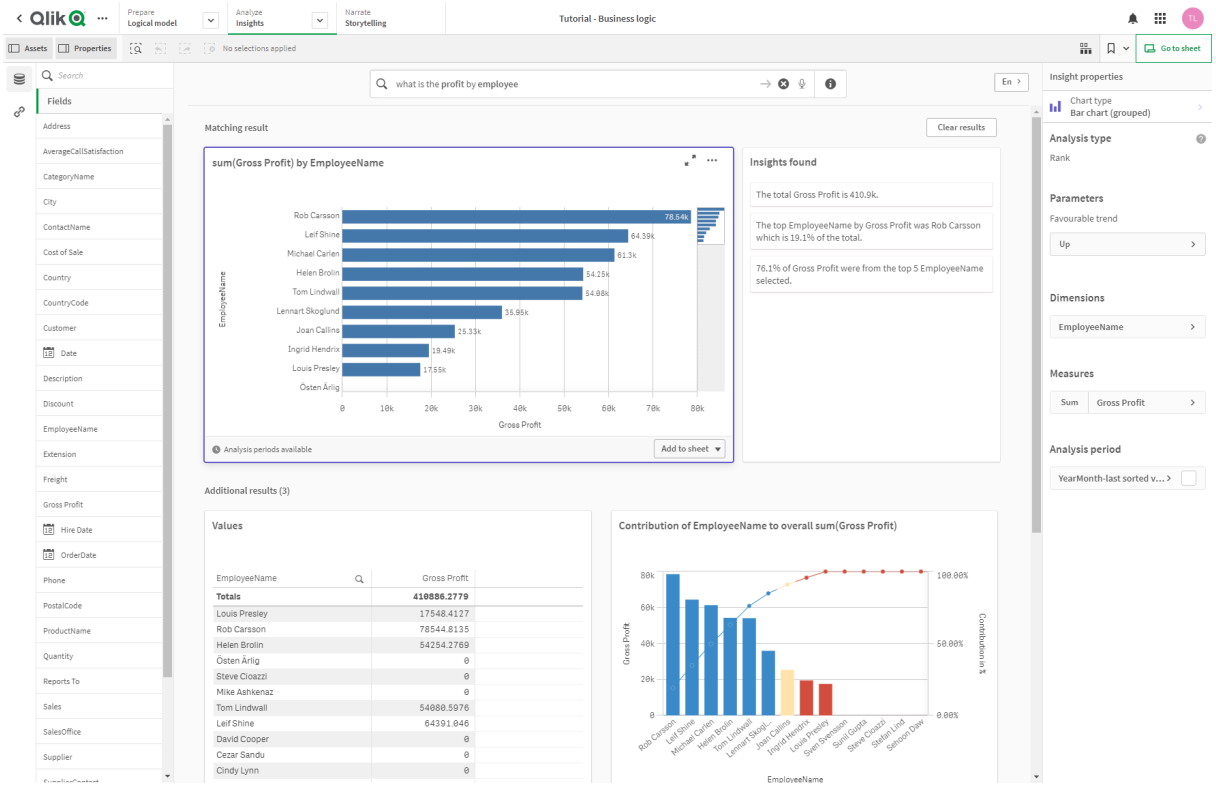
분석 탭의 **통찰력**으로 이동하여 **자산** 패널을 확인합니다. **자산**에 있는 필드 수가 크게 감소했습니다. 예를 들어 ID 필드는 더 이상 존재하지 않습니다.

통찰력*에서* ***자산*** *패널에 있는 필드*



이제 *직원별 수익*을 검색합니다. 이번에는 *EmployeeID* 대신 Insight Advisor이 시각화에서 차원으로 사용하는 데 더 유용한 필드인*EmployeeName*을사용합니다.

직원별 수익에 대한 쿼리 결과



패키지 구성

다음 단계에서 패키지부터 논리 모델의 다른 요소를 사용하기 시작합니다.

패키지는 분석에 함께 사용해야 하는 그룹의 모음입니다. 이렇게 하면 관련 필드만 Insight Advisor에서 함께 사용됩니다. 그룹은 여러 패키지에 속할 수 있습니다. 패키지는 논리 모델의 선택적 부분입니다.

패키지를 만드는 이유

패키지는 분석에 함께 사용해야 하는 그룹의 범위를 정의합니다. 예를 들어 논리 모델에는 조직의 여러 영역을 포함하는 그룹이 있을 수 있습니다. 이러한 그룹을 혼합하는 것이 통찰력 차트에 항상 좋은 것은 아닙니다.

예를 들어 자습서 앱에는 판매 및 공급업체 데이터와 관련이 없는 지원 호출 처리와 관련된 데이터가 있습니다. **통찰력**으로 이동하여 **자산** 패널에서 *AverageCallSatisfaction*을 선택하면 지원 통화 데이터가 판매 데이터 필드와 혼합된 결과를 얻을 수 있습니다.

패키지를 사용하면 판매 및 공급업체 데이터를 분석을 위해 함께 유지하고 지원 호출 데이터를 분석을 위해 함께 유지하여 Insight Advisor이 데이터를 함께 사용하지 않도록 할 수 있습니다.

패키지 만들기

판매 관련 그룹 및 지원 관련 그룹에 대한 패키지를 만듭니다. 두 패키지 모두 관련 *Employees* 및 *Offices* 그룹을 포함합니다.

Support 패키지 만들기

다음과 같이 하십시오.

1. **패키지**로 이동합니다.
2. **패키지 만들기**를 클릭합니다.
3. **패키지 이름**에 *Support*를 입력합니다.
4. 패키지에 다음 그룹을 추가합니다:
    - *SupportCalls*
    - *SupportDate*
    - *Employees*
    - *Offices*
5. **만들기**를 클릭합니다.

Sales 패키지 만들기

다음과 같이 하십시오.

1. **패키지** 섹션으로 이동합니다.
2. **패키지 만들기**를 클릭합니다.
3. **패키지 이름**에 *Sales*를 입력합니다.
4. 패키지에 다음 그룹을 추가합니다:
    - *Categories*
    - Customer
    - *SalesCity*
    - *SalesCountry*
    - *Employees*
    - *Offices*
    - *Sales_OrderDate*
    - *Products*
    - *Sales*
    - *Suppliers*
5. **만들기**를 클릭합니다.

결과

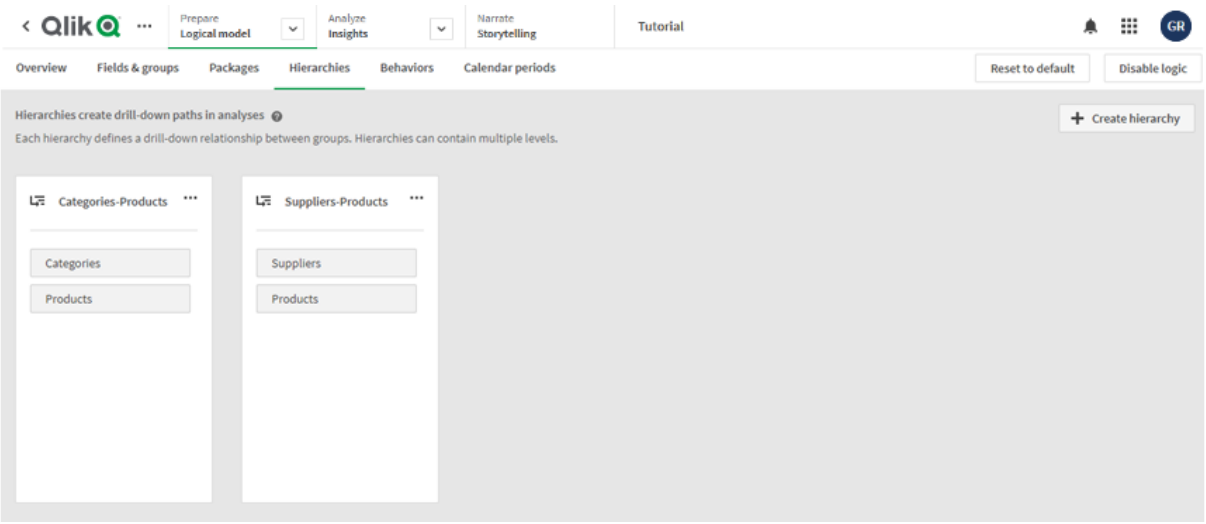
통찰력으로 돌아갑니다. **자산**에서 *AverageCallSatisfaction*을 선택합니다. 이제 생성된 통찰력 차트는 *Support* 패키지의 일부가 아닌 그룹의 필드를 사용하지 않습니다.

계층 구조 정의

다음으로 논리 모델에 계층 구조를 추가합니다. **계층 구조**는 선택적인 비즈니스 논리 기능입니다. 그룹 간의 드릴다운 관계를 정의합니다.

비즈니스 논리를 활성화하면 Qlik Sense가 데이터 모델에서 일부 계층 구조를 자동으로 만들 수 있습니다. **계층 구조**로 이동하면 비즈니스 논리가 두 개의 계층 구조를 만들었음을 알 수 있습니다.

논리 모델의 계층 구조

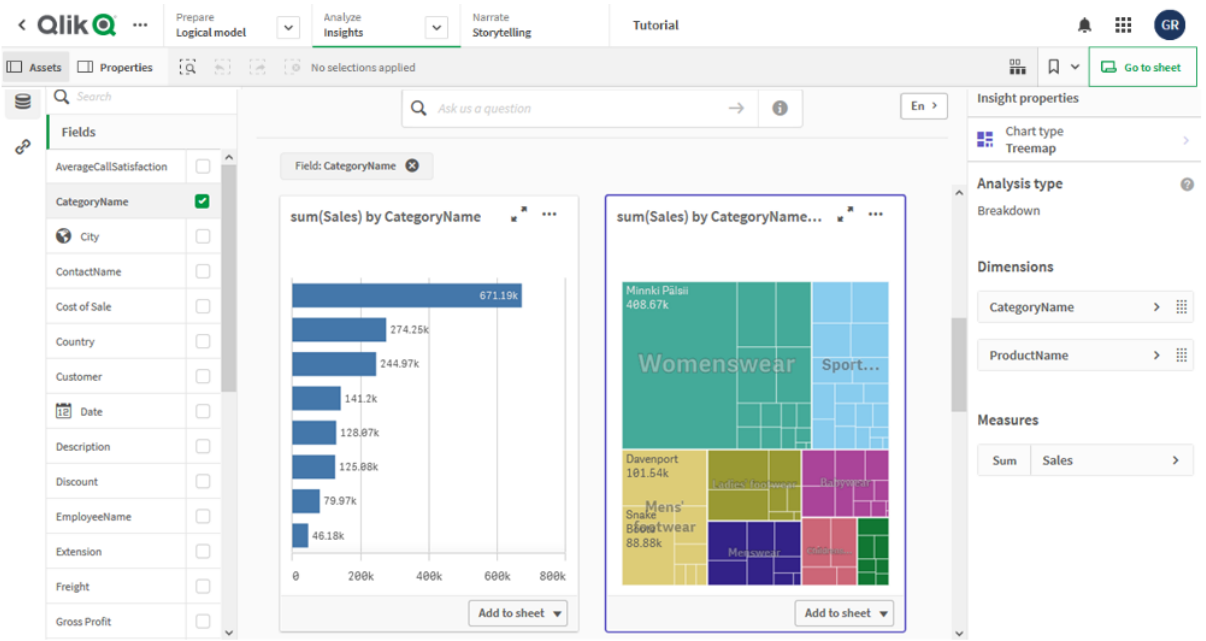


계층 구조는 분석에서 서로를 분할하는 데 사용할 수 있는 그룹을 나타냅니다. 예를 들어 Qlik Sense가 만든 두 계층 구조는 논리 모델에서 두 드릴다운 관계를 올바르게 식별합니다.

- *Category* 필드의 데이터는 *Products* 필드의 데이터로 분할할 수 있습니다.
- *Suppliers* 필드의 데이터는 *Products* 필드의 데이터로 분할할 수 있습니다.

통찰력으로 이동하여 *CategoryName*을 선택하면 Insight Advisor에는 *ProductName*별로 *CategoryName*을 분할하는 트리 맵이 포함됩니다.

*카테고리 및 제품 분할 **통찰력** 차트*



계층 구조는 생성된 차트에서 드릴다운 선택을 활성화하지 않습니다. 이를 위해서는 드릴다운 차원 마스터 항목이 필요합니다. 예를 들어 국가를 선택한 후 도시로 드릴다운하는 맵 차트를 원하는 경우 해당 드릴다운 차원 마스터 항목을 만들어야 합니다.

계층 구조를 정의하는 이유

계층 구조를 사용하면 논리 모델에서 그룹 간 드릴다운 관계를 정의할 수 있습니다. 이를 통해 Insight Advisor는 트리 맵과 같이 데이터를 분할하는 분석을 만들 수 있습니다. 또한 계층 구조는 드릴다운 분석에 선호하는 그룹이 함께 사용되도록 합니다.

데이터 모델에는 국가 데이터와 도시 데이터 사이에 지리적 계층 구조가 있습니다. **통찰력**으로 이동하여 *Country* 필드에 대한 차트를 생성하는 경우 생성된 결과에 *City* 필드가 사용되지 않습니다. 논리 모델에 새로운 계층 구조를 추가하여 이 문제를 해결할 수 있습니다.

SalesCountry 및 *SalesCity*에 대한 계층 구조 정의

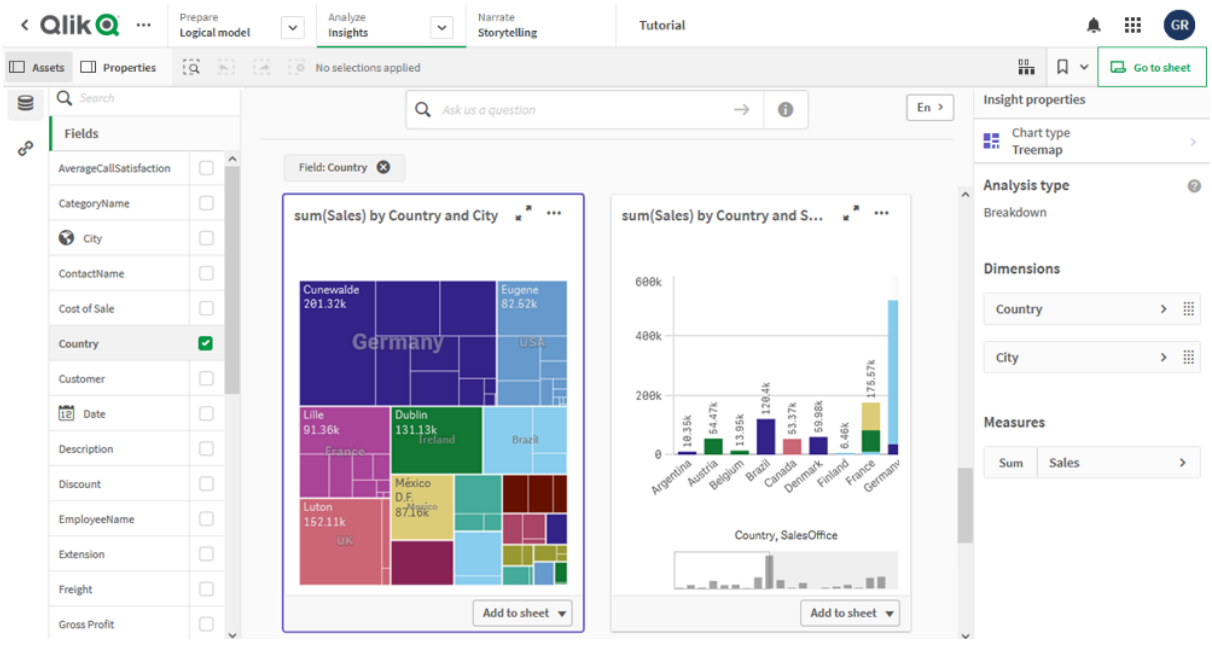
다음과 같이 하십시오.

1. **계층 구조 만들기**를 클릭합니다.
2. **이름**에 *Country-City*를 입력합니다.
3. **사용 가능한 그룹**에서 *SalesCountry*를 선택하고 **>**를 클릭합니다.
4. **사용 가능한 그룹**에서 *SalesCity*를 선택하고 **>**를 클릭합니다.
5. **만들기**를 클릭합니다.

결과

통찰력으로 이동합니다. 자산 패널에서 *Country*를 선택합니다. 이제 *sum(Sales) by Country and City*를 표시하는 트리 맵을 포함하여 *City-Country* 계층 구조를 반영하는 결과가 있습니다.

Country에 대한 새로운 Insight Advisor 결과



캘린더 기간 구성

캘린더 기간에서는 캘린더 그룹을 사용하여 Insight Advisor의 기본 분석 기간을 만들 수 있습니다.

캘린더 기간은 Insight Advisor 분석의 분석 기간을 정의하는 데 사용됩니다. 캘린더 그룹에서 캘린더 기간을 만듭니다. 데이터에 따라 비즈니스 논리는 데이터 모델에서 캘린더 기간을 자동으로 만들 수 있습니다. **캘린더 기간**으로 이동하면 캘린더 기간이 이미 생성된 것을 볼 수 있습니다. 캘린더 기간은 현재 날짜 기준의 기간을 포함하여 상대적일 수 있습니다. 예를 들어 현재 월을 포함하는 상대 기간을 만들 수 있습니다. 캘린더 기간은 데이터의 마지막에 기록된 값을 사용할 수도 있습니다. 예를 들어 비즈니스 논리를 활성화할 때 생성된 세 개의 캘린더 기간은 모두 마지막으로 기록된 값을 사용하여 지난 연도, 월 및 분기에 대한 캘린더 기간을 만듭니다. 앱 사용자는 **통찰력**에서 캘린더 기간을 통찰력 차트에 적용할 수 있습니다.

동작에서 그룹의 기본 캘린더 기간을 정의할 수도 있습니다. 이렇게 하면 기본적으로 Insight Advisor이 해당 그룹의 필드에 대해 차트를 만들 때 해당 캘린더 기간을 사용합니다.

캘린더 기간을 정의하는 이유

캘린더 기간은 데이터에 대한 특정 분석 기간을 만들므로 유용합니다. Insight Advisor에 *in 2020* 또는 *for December*와 같이 특정 시간 프레임이 지정되지 않은 경우 사용 가능한 전체 데이터를 사용합니다. 캘린더 기간을 제공하면 앱 사용자가 특정 기간의 데이터를 보다 쉽게 보고 비교할 수 있습니다.

예를 들어 제품별 판매량을 보려고 합니다. **통찰력**으로 이동하여 *show me sales by product*를 입력합니다. 지난 달과 비교한 이번 달의 데이터 또는 이전 분기와 현재 분기의 비교와 같은 특정 판매 기간을 보려면 Insight Advisor에 다시 요청해야 합니다. 캘린더 기간을 만들면 Insight Advisor에 다시 요청하지 않고 질문에 대한 이 기간을 볼 수 있습니다. 비즈니스 논리를 통해 이미 데이터에서 지난 달, 분기 및 연도에 대해 마지막으로 기록된 값 캘린더 기간이 만들어졌습니다. 그러나 상대적인 기간은 없습니다. 이번 달과 작년 같은 달에 대한 비교를 만들 수 있습니다. 현재 분기(x)와 2분기 전 분기(x-2)를 비교할 수도 있습니다.

Month comparison 캘린더 기간 만들기

이 캘린더 기간은 데이터의 최근 달에 대한 결과를 표시하고 작년 동일한 달과 비교합니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. *Sales_Order Date*를 선택합니다.
3. **캘린더 기간 이름**에 *Month comparison*을 입력합니다.
4. **상세 캘린더 기간**에 대해 **월**을 선택합니다.
5. **기간 비교**에서 **연도별**을 선택합니다.
6. **마지막 완료 기간**을 선택합니다.
7. **만들기**를 클릭합니다.

분기 비교 캘린더 기간 만들기

이 캘린더 기간은 현재 분기(x)에 대한 결과를 표시하고 이를 2분기 전 분기(x-2)와 비교합니다.

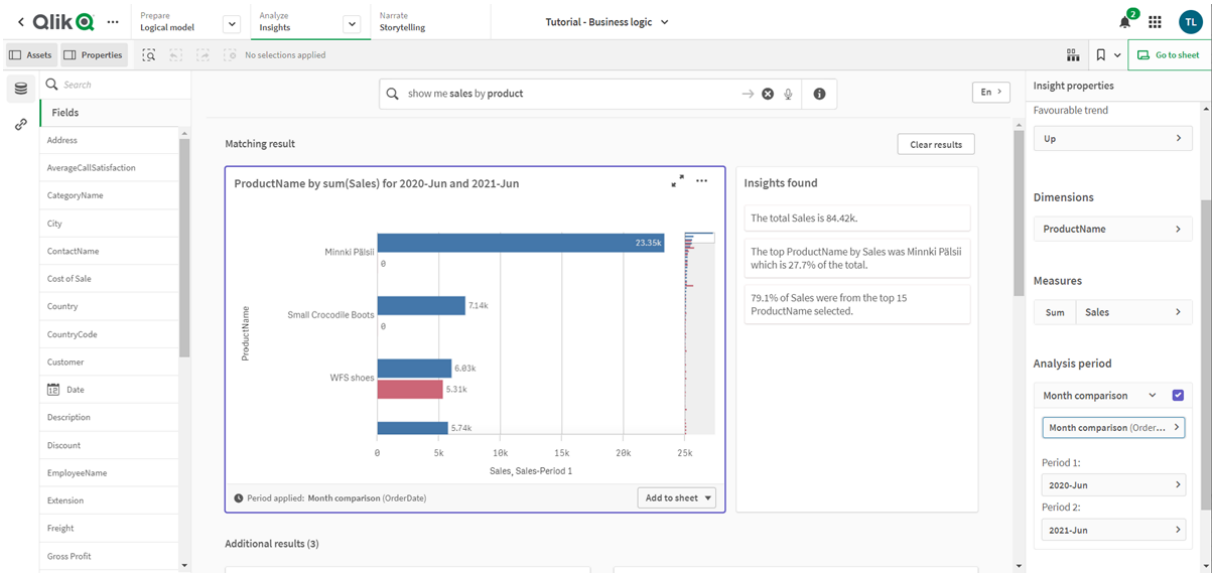
다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. *Sales_Order Date*를 선택합니다.
3. **캘린더 기간 이름**에 *Quarter comparison*을 입력합니다.
4. **상세 캘린더 기간**에서 **분기**를 선택합니다.
5. **사용자 지정**을 클릭합니다.
6. **오프셋**에서 *0*을 선택합니다.
7. **오프셋 비교**에서 *2*를 선택합니다.
8. **만들기**를 클릭합니다.

결과

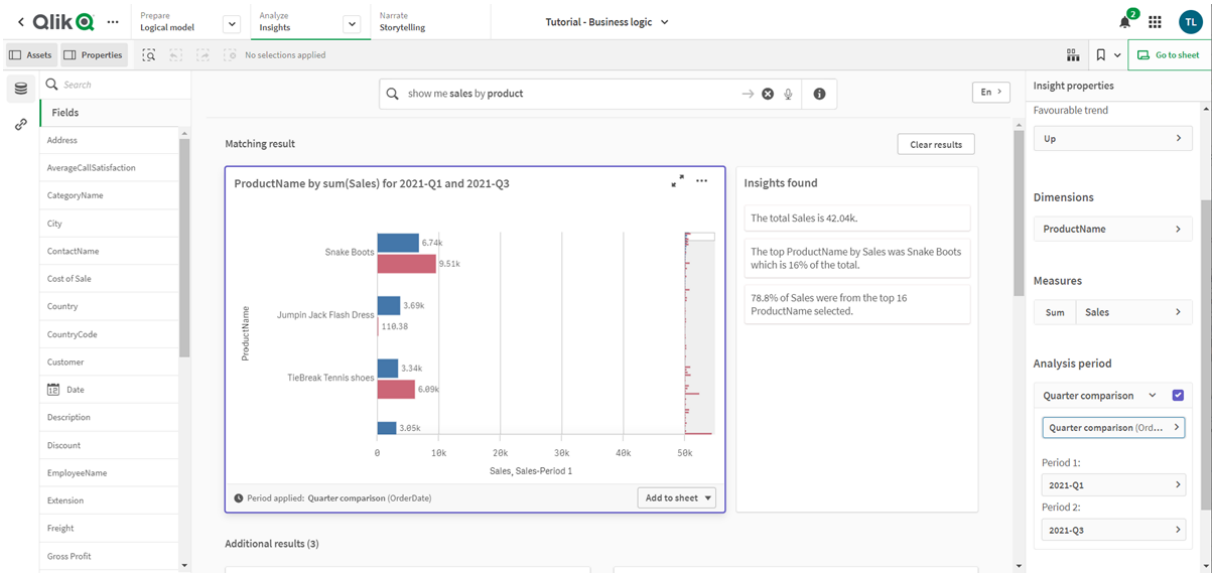
통찰력으로 이동하여 *show me sales by product*를 검색합니다. 일치하는 결과를 선택합니다. **통찰력 속성**에 사용 가능한 새 속성인 **분석 기간**이 있습니다. 이를 선택하고 *월 비교*를 선택하여 차트에 기간을 적용합니다.

통찰력 차트에 적용된 월 비교 캘린더 기간



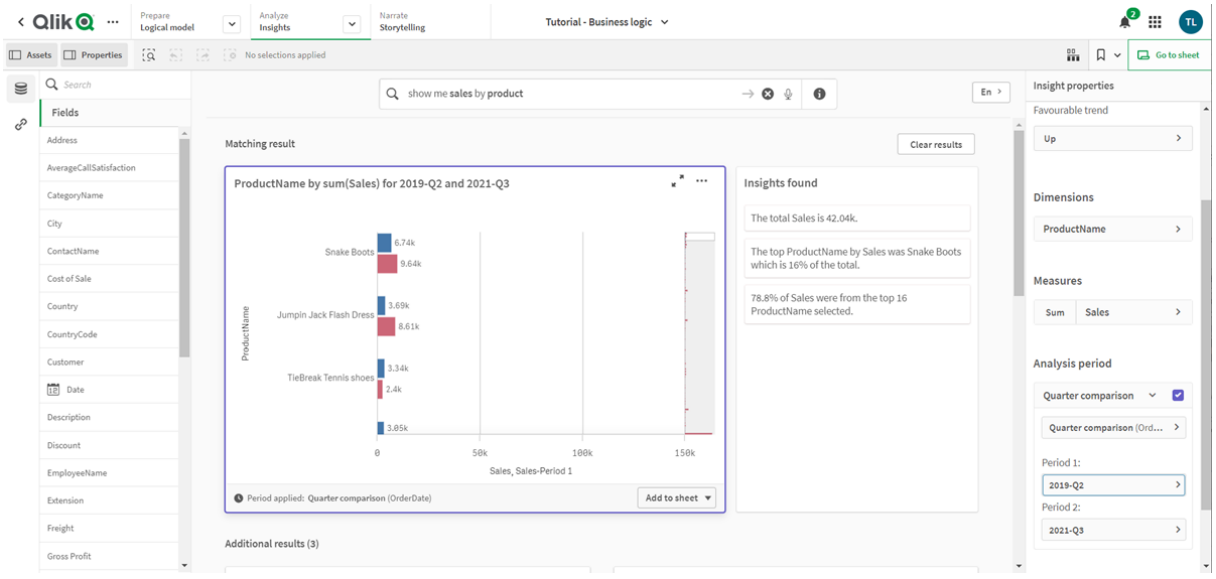
다음으로 *Quarter comparison* 캘린더 기간을 적용합니다.

통찰력 차트에 적용된 Quarter comparison 캘린더 기간



기간 1 및 **기간 2**의 값을 변경하여 Insight Advisor의 개별 차트에 대한 분석 기간의 기간을 수동으로 조정할 수도 있습니다. 예를 들어 **기간 1**에서 값을 *2019-Q2*로 변경합니다. 새로 정의된 기간 동안 차트가 업데이트 됩니다.

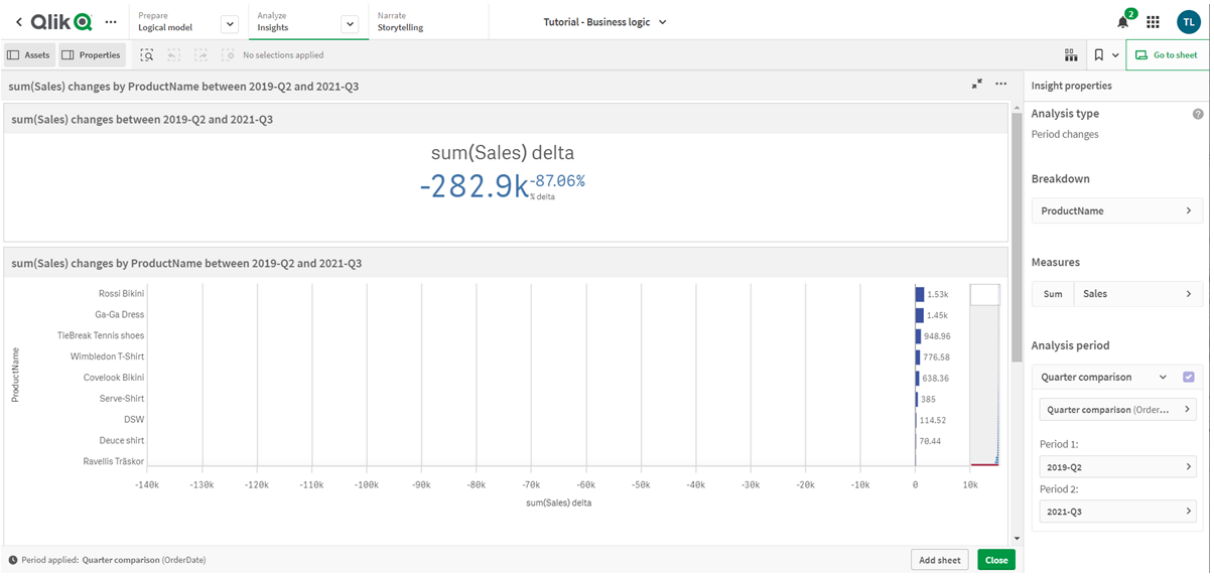
통찰력 차트에 적용된 Quarter comparison 캘린더 기간, 이제 2019-Q2 및 2021-Q2 사용



캘린더 기간을 사용할 수 있는 경우 Insight Advisor에서 기간 분석, 기간 대비 기간 및 목표 대비 기간 성과와 같은 추가 분석 유형을 사용할 수 있습니다. 이는 기간 및 해당 변경 내용을 분석하기 위한 차트 모음을 제공합니다.

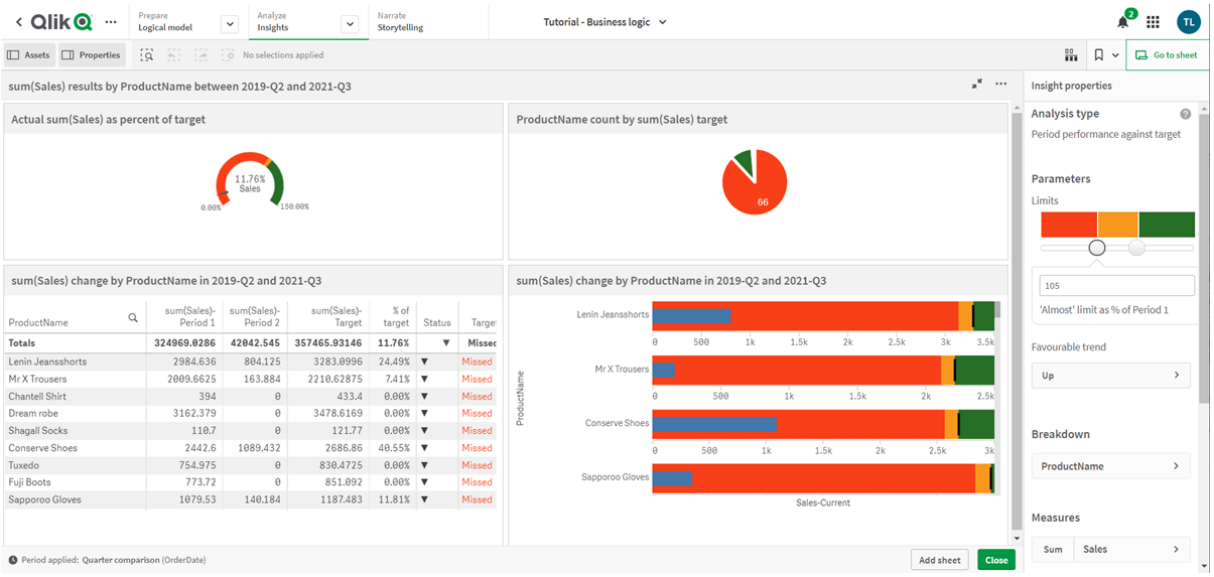
분석 기간에서 **분석 탐색**을 클릭합니다. 선택할 수 있는 두 가지 분석이 있습니다. 먼저 **기간 변경**을 선택합니다. 기간 변경은 두 기간 사이의 sum(Sales) 델타 변경에 대한 KPI 및 막대형 차트를 표시합니다. 차원의 수가 작은 경우 변동 폭포형 차트가 막대형 차트 대신 표시됩니다.

기간 변경 분석



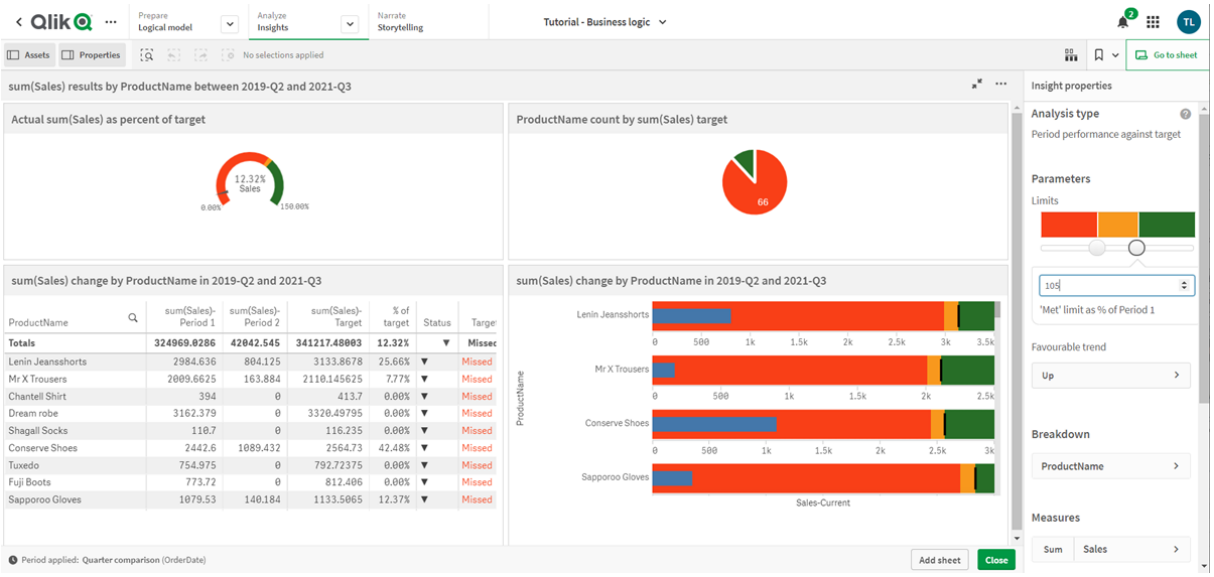
닫기를 클릭하고 막대형 차트로 다시 스크롤합니다. **분석 탐색**을 클릭하고 이제 **자세한 기간 분석**을 선택합니다. 이 기능은 목표 대비 기간 성과를 제공합니다. 이 기간 동안 목표가 충족되었는지 측정하는 4개의 차트가 표시됩니다. 매개 변수를 조정하여 **분석 속성**에서 목표를 충족한 항목, 거의 충족한 항목 및 충족하지 못한 항목을 정의할 수 있습니다.

목표 대비 기간 성과 분석



매개 변수에서 목표 누락 제한을 *100*으로 조정하고 목표 충족 제한을 *105*로 조정하십시오. 자세한 기간 분석이 새 목표에 대해 업데이트됩니다.

업데이트된 목표 대비 기간 성과 분석



표시된 대로 캘린더 기간을 사용하면 Insight Advisor에 관심 있는 앱 사용자가 서로 다른 기간을 빠르게 볼 수 있습니다.

동작 구성

이제 동작을 만듭니다. 동작을 통해 측정값 그룹이 사용되는 방식에 대한 사용자 지정 개별 범위를 설정할 수 있습니다.

동작을 통해 다음을 수행할 수 있습니다.

- 측정값 그룹이 있는 필드에서 필요한 값을 선택하도록 합니다.
- 측정값 그룹과 함께 사용할 그룹을 설정합니다.
- 측정값 그룹과 함께 사용하지 않을 그룹을 설정합니다.
- 측정값 그룹과 함께 기본적으로 사용할 캘린더 기간을 설정합니다.

동작은 선택 사항입니다.

동작을 구성하는 이유

패키지와 마찬가지로, 동작을 통해 그룹을 함께 사용하는 방법에 대한 범위를 설정할 수 있습니다. 동작을 통해 개별 측정값 그룹이 다른 그룹과 함께 사용되는 방식에 대한 좁은 범위를 설정할 수 있습니다.

예를 들어 **통찰력**의 *Sales* 그룹에서 필드를 검색할 때 기본적으로 *Gross Profits*에 대한 측정값 결과는 항상 집계됩니다. 최근 기간을 적용한 경우 더 유용할 것입니다. 기본적으로 캘린더 기간 중 하나를 사용하는 동작을 만들 수 있습니다.

기본 캘린더 기간 동작 만들기

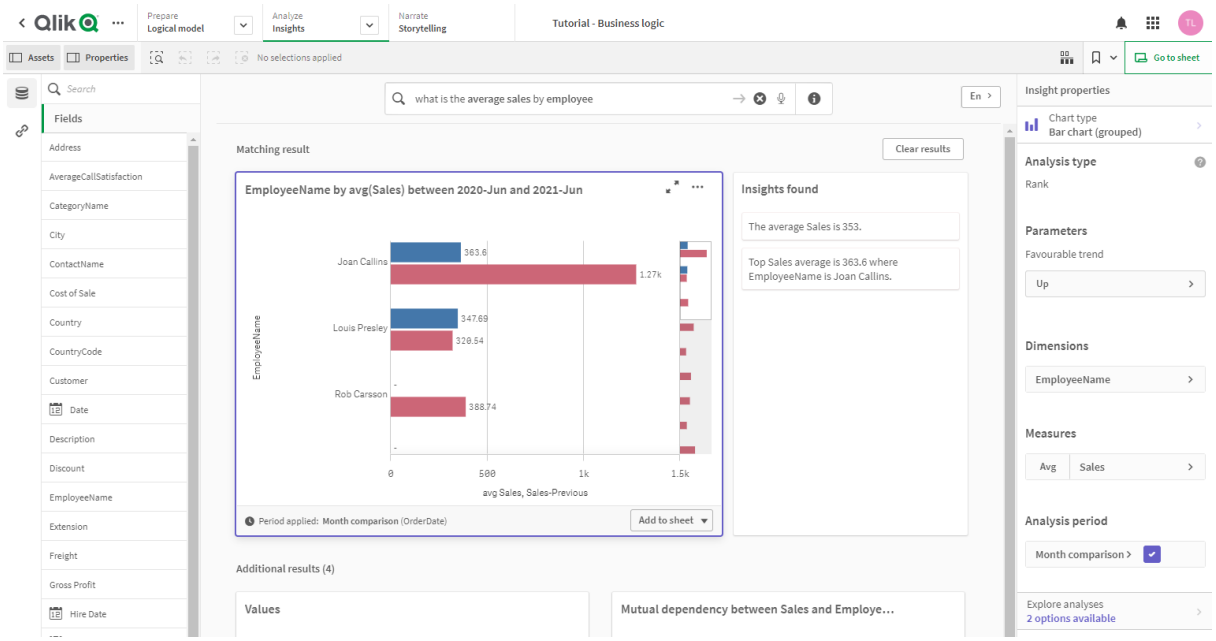
다음과 같이 하십시오.

1. **동작 만들기**를 클릭합니다.
2. **적용 대상**에서 *Sales*를 선택합니다.
3. **동작 유형**에서 **기본 캘린더 기간**을 선택합니다.
4. **캘린더 그룹**에서 *Sales_OrderDate*를 선택합니다.
5. **기간**에서 *Month comparison*을 선택합니다.
6. **만들기**를 클릭합니다.

결과

통찰력으로 이동하여 *what is the average sales by employee*를 검색합니다. 이제 일치하는 결과에 각 직원의 2020년 12월 매출과 2019년 11월 매출을 비교하는 막대형 차트가 표시됩니다. 비교를 지원하지 않는 차트의 경우 대신 2020년 12월의 결과가 표시됩니다.

Insight Advisor에서 기본 캘린더 기간을 사용한 통찰력 차트가 표시됩니다.



기간 대비 기간 분석에 기본 캘린더 기간을 사용하도록 동작 구성

기간 대비 기간 분석에 적용되는 그룹에 대한 기본 캘린더 기간 동작을 설정할 수도 있습니다. **기간 대비 기간** 분석 유형은 꺾은선형 차트를 사용하여 현재 기간과 이전 기간의 측정값을 비교합니다. 논리 모델의 측정값을 포함하는 그룹에 대해 설정된 기본 캘린더 기간이 필요합니다.

먼저 *Quarter over quarter* 캘린더 기간을 만듭니다.

다음과 같이 하십시오.

1. **캘린더 기간 만들기**를 클릭합니다.
2. *Sales_Order Date*를 선택합니다.
3. **캘린더 기간 이름**에 *Quarter over quarter*를 입력합니다.
4. **상세 캘린더 기간**에서 **분기**를 선택합니다.
5. **최근 정렬된 값 사용**을 선택합니다.
6. **만들기**를 클릭합니다.

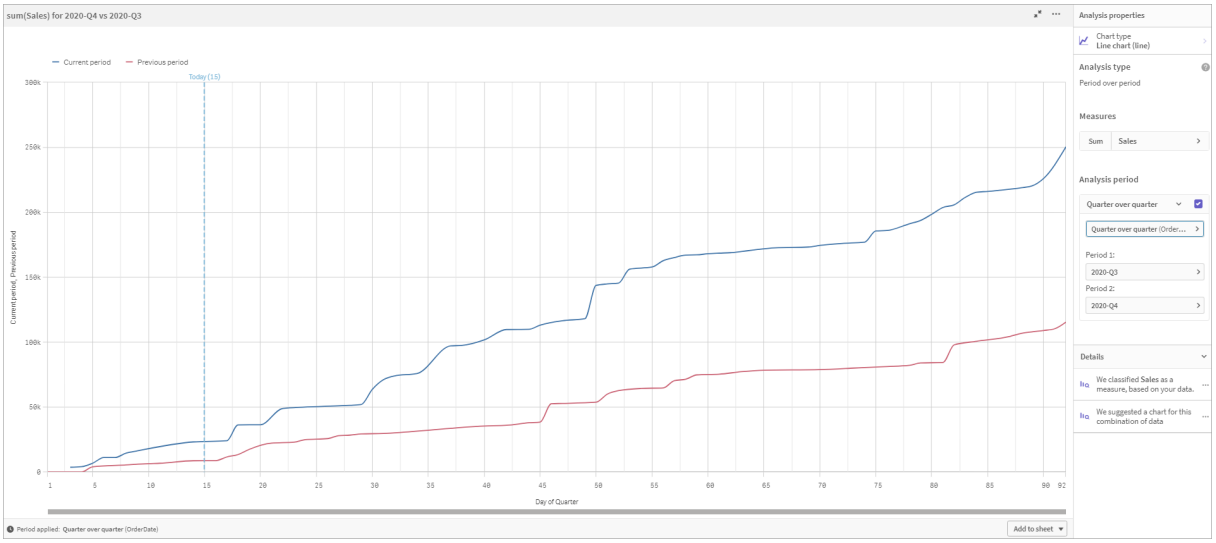
이제 **동작** 탭에서 이 캘린더 기간을 *Sales* 그룹에 대한 기본값으로 설정합니다.

다음과 같이 하십시오.

1. **동작 만들기**를 클릭합니다.
2. **적용 대상**에서 *Sales*를 선택합니다.
3. **동작 유형**에서 **기본 캘린더 기간**을 선택합니다.
4. **캘린더 그룹**에서 *Sales_OrderDate*를 선택합니다.
5. **기간**에서 *Quarter over quarter*를 선택합니다.
6. **만들기**를 클릭합니다.

검색으로 이동하여 *show me sales*를 검색합니다. KPI 아래에 나타나는 첫 번째 차트는 **기간 대비 기간** 분석입니다. 이 차트(*sum (Sales) for 2020-Q4 vs 2020-Q3*)를 열고 속성 **분석 기간**으로 이동합니다. 이를 선택하고 가장 최근 기간을 이전 기간과 비교하기 위해 기본 캘린더 기간 *Quarter over quarter*가 올바르게 선택되었는지 확인합니다. 이 자습서에서 해당 기간은 2020-Q3(기간 1) 및 2020-Q4(기간 2)입니다. 이러한 선택은 드롭다운에서 다른 분기를 선택하여 변경할 수 있습니다. 참조선은 현재 캘린더 기간 단위의 현재 날짜를 나타내며 이 경우 분기입니다.

***꺾은선형 차트**에 적용된 기간 대비 분석*



분석 기간에서 이 기본 캘린더 기간을 변경할 수 있습니다.

어휘 만들기

어휘를 사용하면 Insight Advisor에 대한 용어를 추가할 수 있습니다. 이는 데이터에 없는 필드 및 값에 대한 동의어로 서버에 사용되거나 사용자 지정 분석을 정의할 수 있습니다. 어휘에는 사용자 지정 논리 모델이 필요하지 않습니다.

어휘를 사용하면 동의어 및 사용자 지정 분석을 정의할 수 있습니다. 동의어를 사용하면 용어를 추가하여 데이터 모델에서 값 또는 값 선택을 정의할 수 있습니다. 사용자 지정 분석을 사용하면 사용자 지정 질문 또는 부분 질문의 응답을 정의하고 Insight Advisor에서 반환된 특정 결과를 나타낼 수 있습니다.

어휘를 만드는 이유

어휘는 자연어 쿼리를 위한 앱을 준비할 핵심 도구입니다. Insight Advisor은 모든 자연어 쿼리를 데이터의 필드 이름 및 값에 연결하려고 합니다. 앱 사용자가 입력할 수 있는 모든 검색어를 알 수는 없습니다. 어휘를 사용하면 용어를 필드 및 필드 값에 연결하여 이러한 용어 차이를 해소할 수 있습니다.

예를 들어 자습서 앱에 여러 swimwear 제품이 있습니다. 이러한 항목은 데이터에서 *Beachwear* 카테고리에 있습니다. **통찰력**에서 *swimwear*를 검색하면 결과가 표시되지 않습니다.마찬가지로 *footwear*를 검색하여 남성용 및 여성용 신발에 대한 결과를 얻을 수 없습니다.

사용자 지정 분석은 검색어를 기반으로 앱 소비자가 받기를 원하는 분석을 알고 있을 때 유용합니다. 예를 들어 앱 사용자가 지역 데이터를 맵으로 보는 것을 선호한다는 것을 알고 있습니다. 사용자 지정 분석을 사용하여 사용자가 검색에 *지역*을 포함할 때 맵이 제공되는지 확인할 수 있습니다.

beachwear 어휘 만들기

다음과 같이 하십시오.

1. **준비**에서 **어휘**를 클릭합니다.
2. **동의어**에서 **용어 만들기**를 클릭합니다.
3. **용어**에 다음 용어를 입력하십시오.
    - *swimwear*
    - *swimsuits*
4. **적용 대상**에서 *CategoryName*을 선택합니다.
5. **조건**에서 **In**을 선택합니다.
6. **In**에서 다음 값을 선택합니다.
    - *Beachwear*
7. **만들기**를 클릭합니다.

footwear 어휘 만들기

다음과 같이 하십시오.

1. **동의어**에서 **용어 만들기**를 클릭합니다.
2. **용어**에서 *footwear*를 입력합니다.
3. **적용 대상**에서 *CategoryName*을 선택합니다.

4. **조건**에서 **In**을 선택합니다.
5. **In**에서 다음 값을 선택합니다.
    - *men's footwear*
    - *women's footwear*
6. **만들기**를 클릭합니다.

지역 사용자 지정 분석 만들기

이 사용자 지정 분석의 경우 측정값을 지정하지 않습니다. 이렇게 하면 Insight Advisor에서 누군가가 쿼리에서 *regional*을 사용하는 방법을 기반으로 측정값을 선택할 수 있습니다.

다음과 같이 하십시오.

1. **사용자 지정 분석**을 클릭합니다.
2. **용어 만들기**를 클릭합니다.
3. **용어**에서 *regional*을 입력합니다.
4. **분석**에서 **분할(지역 관련)**을 선택합니다.
5.
6. **데이터**의 **지리**에서 *Country*를 선택합니다.
7. **차트**를 클릭합니다.
8. **차트 유형**에서 **맵**을 선택합니다.
9. **만들기**를 클릭합니다.

결과

통찰력으로 이동합니다. *what is the average freight for footwear*를 검색합니다. 이제 남성 및 여성 신발에 대한 결과를 받습니다.

다음으로 *gross profit for swimwear in 2019*를 검색합니다. 이제 beachwear를 검색하는 동안 beachwear에 대한 결과를 얻을 수 있습니다.

이제 *show me regional sales*를 검색합니다. 이제 국가별 판매의 맵 분포를 얻습니다.

감사합니다.

이제 이 자습서가 끝났습니다. Qlik Sense에서 비즈니스 논리 및 Insight Advisor에 대한 많은 지식을 얻으셨기 바랍니다.

12 문제 해결 - 데이터 로드

이 섹션에서는 Qlik Sense에서 데이터를 로드하고 모델링할 때 발생할 수 있는 문제에 대해 설명합니다.

12.1 파일을 **데이터 추가**에 끌어서 첨부하는 방식이 작동하지 않습니다.

파일을 Qlik Sense의 **데이터 추가/파일 첨부** 대화 상자로 끌어 놓아서 첨부하려고 해도 파일이 업로드되지 않습니다.

가능한 원인

파일은 ZIP 보관 파일에 저장됩니다. Qlik Sense에서는 ZIP 보관 파일의 개별 파일을 첨부할 수 없습니다.

제안 조치

파일을 첨부하기 전에 ZIP 보관 파일에서 추출하십시오.

12.2 ANSI 이외로 인코딩된 데이터 파일을 사용하는 경우의 문자 집합 문제

ODBC 데이터 연결을 사용하는 경우 ANSI 이외로 인코딩된 데이터 파일에서 문자 인코딩과 관련된 문제가 발생할 수 있습니다.

가능한 원인

ODBC 데이터 연결이 문자 집합 인코딩에 대한 모든 기능을 제공하지는 않습니다.

제안 조치

다음과 같이 하십시오.

- 가능하면 문자 코드 처리를 위한 많은 옵션을 지원하는 폴더 데이터 연결을 사용하여 데이터 파일을 가져오십시오. Microsoft Excel 스프레드시트 또는 텍스트 데이터 파일을 로드하는 경우에는 가장 좋은 옵션일 수 있습니다.

12.3 데이터 로드 시 순환 참조 경고

가능한 원인

둘 이상의 테이블을 로드한 경우 두 필드 간에 둘 이상의 연결 경로가 있어서 데이터 구조에서 루프를 생성하는 방식으로 테이블이 연결될 수 있습니다.

제안 조치

12.4 고정된 레코드 파일에서 데이터를 선택하면 열이 예상대로 정렬되지 않습니다.

가능한 원인

파일은 열을 패딩하는 데 탭 문자를 사용합니다. 일반적으로 선택 대화 상자에서 **필드 나누기**를 선택한 경우 필드 머리글이 예상한 데이터에 따라 정렬되지 않는 것을 볼 수 있습니다.

이 경우 탭 문자는 일반적으로 수많은 문자와 동등합니다.

제안 조치

다음과 같이 하십시오.

1. **필드 이름**에서 **필드 이름 없음**을 선택합니다.
2. **필드 나누기**를 선택합니다.
3. 열이 헤더에 맞춰 정렬될 때까지 **탭 크기**의 설정을 늘립니다.
4. 적절한 열 위치를 클릭하여 필드 나누기를 삽입합니다.
5. **데이터 미리 보기**를 선택합니다.
6. **필드 이름**에서 **포함된 필드 이름**을 선택합니다.

이제 열이 적절하게 정렬되고 각 필드가 올바른 필드 이름을 갖습니다.

12.5 커넥터가 작동하지 않습니다.

데이터 로드 편집기에서 별도로 설치된 커넥터에 대한 데이터 연결을 만들려고 시도했지만 연결이 실패하거나 기존 연결의 레이블이 알 수 없음으로 지정됩니다.

커넥터가 제대로 설치되지 않았습니다.

가능한 원인

커넥터가 설치 지침에 따라 제대로 설치되지 않았습니다. 앱이 다중 노드 사이트에서 커넥터를 사용하는 경우 해당 커넥터를 모든 노드에 설치해야 합니다.

제안 조치

다음과 같이 하십시오.

- 커넥터가 사이트의 모든 노드에 지침에 따라 설치되어 있는지 확인하십시오.

커넥터가 Qlik Sense에 맞게 조정되어 있지 않습니다.

가능한 원인

데이터를 선택할 수 있으려면 QlikView 커넥터가 Qlik Sense에 맞게 조정되어야 합니다.

제안 조치 (QVX SDK를 사용하여 커넥터를 자체 개발한 경우)

다음과 같이 하십시오.

- 데이터를 선택하려면 인터페이스를 사용하여 커넥터를 Qlik Sense에 맞게 조정해야 합니다.

제안 조치 (커넥터가 제공된 경우)

다음과 같이 하십시오.

- 커넥터 공급업체에 연락하여 Qlik Sense에 맞는 커넥터를 구하십시오.

12.6 SQL Server를 다시 시작한 후 데이터 연결이 작동을 멈춥니다.

가능한 원인

SQL Server에 대한 데이터 연결을 만든 후 SQL Server를 다시 시작하면 데이터 연결이 작동을 멈추고 데이터를 선택할 수 없습니다. Qlik Sense에서 SQL Server에 대한 연결이 끊어졌으며 다시 연결할 수 없습니다.

제안 조치

Qlik Sense:

다음과 같이 하십시오.

- 앱을 닫고 허브에서 다시 엽니다.

Qlik Sense Desktop:

다음과 같이 하십시오.

1. 모든 앱을 닫습니다.
2. Qlik Sense Desktop을 다시 시작합니다.

12.7 데이터 로드 편집기에 스크립트가 표시되지 않습니다.

데이터 로드 편집기를 열면 편집기의 내용이 비어 있고 스크립트를 편집할 수 없습니다.

가능한 원인

스크립트에는 수많은 중첩 if 문 등과 같이 매우 복잡한 구조가 포함되어 있습니다.

제안 조치

*/debug/dle_safe_mode*를 URL에 추가하여 안전 모드에서 데이터 로드 편집기를 여십시오. 그러면 구문 강조 표시 기능과 자동 완성 기능이 비활성화되지만 스크립트를 편집하고 저장할 수는 있습니다.

스크립트에서 복잡한 부분을 별도의 텍스트 파일로 옮긴 후 ***include*** *변수를 사용하여 런타임 시 스크립트에 삽입하는 것을 고려해 보십시오.*

12.8 데이터 로드 스크립트가 오류 없이 실행되었으나 데이터가 로드되지 않습니다.

스크립트가 구문 오류 또는 로드 오류 없이 실행되었지만 데이터가 예상대로 로드되지 않습니다. 일반적인 권장 사항은 스크립트를 통한 단계별 디버그를 활성화하고 실행 결과를 시험하는 것이지만, 오류에 대한 다음과 같은 몇 가지 공통 원인을 확인해 보십시오.

문이 세미콜론으로 종결되지 않았습니다.

가능한 원인

문을 세미콜론으로 종결하지 않았습니다.

제안 조치

다음과 같이 하십시오.

- 모든 문을 세미콜론으로 종결하십시오.

문자열 내의 작은따옴표

가능한 원인

SET 변수 문 등에서 문자열에 작은따옴표가 포함되어 있습니다.

제안 조치

다음과 같이 하십시오.

- 문자열에 작은따옴표가 포함되는 경우 추가적인 작은따옴표로 둘러싸야 합니다.

12.9 데이터가 포함된 앱의 테이블이 데이터 관리자에 표시되지 않습니다.

Qlik Sense 3.0 이전 버전에서 만든 앱을 열 때 데이터 관리자에 테이블이 표시되지 않고 앱에 아무 데이터도 포함되어 있지 않다는 메시지가 표시됩니다.

가능한 원인

Qlik Sense 3.0 이상의 향상된 데이터 모델에서는 데이터 프로파일링 및 준비를 완료하려면 데이터를 다시 로드해야 합니다.

제안 조치

데이터 관리자에서 **데이터 로드**를 클릭합니다. 앱에서 사용되는 데이터 소스에 액세스할 수 있어야 합니다.

12.10 서버에서 앱을 만드는 모든 사용자에 대해 데이터 관리자 워크플로가 손상되었습니다.

사용자가 **데이터 관리자**에서 **데이터 추가** 또는 **데이터 로드**를 사용하려고 하거나 브라우저에서 앱을 새로 고칠 때 오류가 발생합니다.

가능한 원인

데이터 관리자는 QVD 파일을 사용하여 로드된 데이터를 캐시합니다. 이러한 파일은 더 이상 사용하지 않는 경우 자동으로 삭제되지만 대량으로 누적되거나 손상되면 오류가 발생할 수 있습니다.

제안 조치

QVD 파일이 포함된 폴더를 삭제합니다. Qlik Sense 서버에서 캐시는 다음 위치에 있습니다.

<Qlik Sense 공유 폴더>\Apps\DataPrepAppCache

Qlik Sense Desktop에서 캐시는 다음 위치에 있습니다.

C:\Users\<사용자 이름>\Documents\Qlik\Sense\Apps\DataPrepAppCache

12.11 OLE DB 데이터 소스를 사용하는 경우의 데이터 선택 문제

가능한 원인

OLE DB 데이터 연결에서 데이터를 선택할 수 없는 경우 연결 구성 방법을 확인해야 합니다.

제안 조치

다음과 같이 하십시오.

1. 연결 문자열이 올바르게 설계되었는지 확인합니다.
2. 적절한 로그온 자격 증명을 사용하는지 확인합니다.

12.12 날짜 필드가 시트 보기에서 날짜 필드로 인식되지 않습니다.

날짜 또는 타임스탬프 데이터가 포함된 필드가 있지만 시트 보기에서 날짜 필드로 인식되지 않습니다. 즉, 자산 패널 및 기타 필드 목록에서 12으로 표시되지 않습니다.

테이블 추가 시 데이터 프로파일링이 비활성화되었습니다.

가능한 원인

테이블을 추가할 때 **데이터 추가** 버튼 옆의 •••에서 데이터 프로파일링을 사용하지 않도록 설정했습니다.

이 옵션을 사용하면 인식되는 날짜 및 타임스탬프 필드가 정상적으로 기능하지만 자산 패널 및 기타 필드 목록에서 12으로 표시되지 않으며 확장된 속성 필드를 사용할 수 없습니다.

제안 조치

데이터 관리자를 열고 **데이터 로드**를 클릭합니다.

이제 모든 날짜 및 타임스탬프 필드가 시트 보기의 자산 패널에서 12으로 표시됩니다. 여전히 12으로 표시되지 않으면 필드 데이터가 날짜로 인식되지 않는 서식을 사용하는 것일 수 있습니다.

날짜 서식이 인식되지 않습니다.

가능한 원인

테이블을 로드했을 때 날짜 필드의 입력 서식이 인식되지 않습니다. 일반적으로 Qlik Sense에서는 로캘 설정 및 일반적인 날짜 서식에 기반하여 날짜 필드를 자동으로 인식하지만 경우에 따라서는 입력 서식을 지정해야 할 수 있습니다.

제안 조치

데이터 관리자를 열고 날짜로 인식되지 않는 필드가 포함된 테이블을 편집합니다. 필드는 대부분 일반 필드처럼 [ ]으로 표시됩니다. 필드 유형을 필드 데이터와 일치하는 입력 서식을 사용하는 **날짜** 또는 **타임스탬프**로 변경합니다.

12.13 파일 첨부 시 **"잘못된 경로"** 오류 메시지 발생

가능한 원인

파일 이름이 너무 깁니다. Qlik Sense에서는 파일 이름을 최대 171자까지 지원합니다.

제안 조치

파일 이름을 172자보다 적은 문자가 포함된 이름으로 바꾸십시오.

12.14 QlikView 문서에서 변환된 앱 로드 시 오류 발생

두 제품 간의 차이로 인해 QlikView 문서에서 변환된 앱을 다시 로드할 때 오류 메시지가 표시될 수 있습니다.

스크립트에서 절대 파일 경로 참조가 사용되었습니다.

가능한 원인

로드 스크립트가 Qlik Sense 표준 모드에서는 지원되지 않는 절대 경로를 사용하는 파일을 참조합니다. 오류 메시지의 예는 "Invalid Path" 및 "LOAD statement only works with lib:// paths in this script mode"입니다.

제안 조치

다음과 같이 하십시오.

- 모든 파일 참조를 Qlik Sense의 데이터 연결에 대한 **lib://** 참조로 바꿉니다.

스크립트에서 지원되지 않는 함수 또는 문이 사용되었습니다.

가능한 원인

데이터 로드 편집기에서 스크립트를 실행할 때 구문 오류가 표시되는 경우에는 Qlik Sense에서 지원되지 않는 QlikView 스크립트 문이나 함수 사용과 관련되어 있을 수 있습니다.

제안 조치

다음과 같이 하십시오.

- 잘못된 문을 제거하거나 올바른 문으로 바꾸십시오.

12.15 Microsoft Excel: 데이터 관리자 또는 데이터 로드 편집기에서 파일의 데이터를 로드할 수 없습니다.

가능한 원인

Excel 스프레드 시트에서 **창 고정** 또는 화면 **분할**이 활성화되어 있고 스프에 빈 셀이 있습니다.

제안 조치

창 고정 또는 화면 **분할**을 비활성화하거나 스프레드시트를 지운 후 데이터를 다시 로드합니다.

12.16 Microsoft Excel: ODBC를 통해 파일에 연결하거나 해당 파일에서 데이터를 로드할 때 문제 발생

가능한 원인

Microsoft Excel 파일에 대한 ODBC 데이터 연결을 설정하거나 ODBC 데이터 연결을 통해 Microsoft Excel 파일에서 데이터를 로드하는 경우 문제가 발생할 수 있습니다. 일반적으로는 Windows의 ODBCDSN 구성으로 인한 문제이거나 연결된 ODBC 드라이버로 인한 문제입니다.

제안 조치

Qlik Sense에서는 기본적으로 Microsoft Excel 파일 로드를 지원합니다. 가능하면 ODBC 데이터 연결을 Microsoft Excel 파일이 포함된 폴더에 연결하는 폴더 데이터 연결로 바꾸십시오.

12.17 디스크 공간이 부족합니 다.

시스템의 디스크 공간이 부족할 수 있는 원인은 몇 가지가 있는데 데이터 관리자의 QVD 파일에 로드된 데이터 캐시 방법이 그 이유 중 하나입니다.

제안 조치

QVD 파일이 포함된 폴더를 삭제합니다. Qlik Sense 서버에서 캐시는 다음 위치에 있습니다.

<Qlik Sense 공유 폴더>\Apps\DataPrepAppCache

Qlik Sense Desktop에서 캐시는 다음 위치에 있습니다.

C:\Users\<사용자 이름>\Documents\Qlik\Sense\Apps\DataPrepAppCache

12.18 데이터 로드 시 가상 키 경고

여러 개의 파일을 로드한 경우 데이터 로드 후에 가상 키가 만들어졌다는 경고 메시지가 표시될 수 있습니다.

가능한 원인

두 테이블에 둘 이상의 공통 필드가 포함된 경우 Qlik Sense에서는 링크 문제를 해결하기 위해 가상 키를 만듭니다.

제안 조치

많은 경우 링크에 의미가 있으면 가상 키에 대해 아무 것도 할 필요가 없지만, 데이터 모델 뷰어에서 데이터 구조를 검토하는 것이 좋습니다.

12.19 공통 필드가 있는 테이블이 필드 이름에 의해 자동으로 연결되지 않습니다.

데이터 추가를 사용하여 둘 이상의 테이블이 추가되었습니다. 이들 테이블에 공통 필드 이름을 가진 필드가 있지만 자동으로 연결되어 있지는 않습니다.

가능한 원인

테이블을 추가할 때 **데이터 추가** 대화 상자에서 데이터 프로파일링을 활성화하도록 기본 옵션을 유지했습니다. 이 옵션은 테이블 간에 공통인 모든 필드 이름을 자동으로 한정합니다. 예를 들어 이 옵션을 사용하여 공통 필드 F1이 있는 테이블 A와 테이블 B를 추가하는 경우 테이블 A에서는 필드 이름이 F1이 되고 테이블 B에서는 필드 이름이 B.F1이 됩니다. 따라서 테이블은 자동으로 연결되지 않습니다.

제안 조치

데이터 관리자를 열고 **연결** 보기를 선택합니다. 이제 데이터 프로파일링 권장 사항에 기반하여 테이블을 연결할 수 있습니다.

테이블을 연결했으면 **데이터 로드**를 클릭합니다.