



# 데이터 관리

---

Qlik Sense®

3.2

Copyright © 1993-2017 QlikTech International AB. 무단 전재 및 복제를 금합니다.



Copyright © 1993-2017 QlikTech International AB. 무단 전재 및 복제를 금합니다.

Qlik®, QlikTech®, Qlik Sense®, QlikView®, Sense® 및 Qlik 로고는 여러 국가에서 등록된 상표이며, 그렇지 않은 경우 QlikTech International AB에서 상표로 사용됩니다. 여기에 언급된 다른 상표는 해당 소유자의 상표입니다.

---

<b>1 이 문서 정보</b> .....	<b>12</b>
<b>2 데이터 관리</b> .....	<b>13</b>
<b>3 데이터 관리자를 사용하여 앱에서 데이터 관리</b> .....	<b>14</b>
3.1 데이터 테이블 미리 보기 .....	14
3.2 새 데이터 테이블 추가 .....	14
3.3 데이터 테이블 편집 .....	15
3.4 데이터 테이블 삭제 .....	15
3.5 데이터 테이블 연결 관리 .....	15
3.6 변경 적용 및 데이터 다시 로드 .....	16
3.7 데이터 관리자에서 작업 실행 취소 및 다시 실행 .....	17
3.8 컨텍스트네이팅된 테이블 .....	17
컨텍스트네이팅된 테이블 분할 .....	18
3.9 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용 .....	18
3.10 앱에 데이터 추가 .....	19
사용 가능한 데이터 소스 .....	20
기존 데이터 소스의 데이터 추가 .....	20
업로드된 데이터 파일에서 데이터 추가 .....	21
데이터 파일 가져오기 .....	21
앱으로 데이터 로드 .....	22
업데이트된 데이터 파일 가져오기 .....	22
앱으로 데이터 다시 로드 .....	23
새 데이터 소스의 데이터 추가 .....	23
데이터 파일 첨부 및 앱에 데이터 추가 .....	24
여러 데이터 파일을 신속하게 첨부 .....	24
단일 데이터 파일 첨부 .....	25
첨부된 파일 삭제 .....	26
첨부된 파일에서 데이터 다시 로드 .....	26
데이터 필드 선택 .....	27
데이터베이스에서 데이터 선택 .....	27
<b>Microsoft Excel</b> 스프레드시트에서 데이터 선택 .....	28
테이블 파일에서 데이터 선택 .....	29
파일 형식에 맞는 설정 선택 .....	29
구분된 테이블 파일 .....	29
고정 레코드 데이터 파일 .....	30
HTML 파일 .....	31
XML 파일 .....	31
QVD 파일 .....	31
QVX 파일 .....	31
KML 파일 .....	31
이전 단계로 돌아가기(데이터 추가) .....	31
Qlik DataMarket의 데이터 추가 .....	32
Qlik DataMarket 차원, 측정값 및 기간 선택 .....	33
Qlik DataMarket에서 계층 구조 데이터 선택 .....	35

---

부모 및 자식 분기 선택 .....	35
데이터가 없는 부모 분기 선택 .....	36
Qlik DataMarket에서 집계 데이터 선택 .....	36
Qlik DataMarket 검색 .....	38
Qlik DataMarket에서 연결 만들기 .....	39
데이터 셋의 여러 테이블 연결 .....	40
단일 테이블 데이터 셋을 여러 테이블로 변환 .....	41
회사 데이터와 DataMarket 데이터 통합 .....	42
매출과 국내 총생산(GDP) 비교 .....	42
유로로 매출 표시 .....	43
매출에 대한 대체 가능 측정값 만들기 .....	44
3.11 테이블 편집 .....	44
테이블 이름 바꾸기 .....	45
필드 이름 바꾸기 .....	45
다른 테이블에 대한 연결 관리 .....	45
필드 유형 및 표시 형식 변경 .....	46
크로스탭 데이터 피벗 해제 .....	46
데이터 소스에서 테이블 업데이트 .....	46
계산 필드 추가 .....	47
테이블 정렬 .....	47
작업 실행 취소 및 재실행 .....	47
테이블 편집기에서 데이터 연결 .....	47
단일 필드를 사용하여 연결 만들기 .....	48
복합 키 만들기 .....	48
연결 편집 .....	49
연결 해제 .....	49
계산 필드 사용 .....	49
계산 필드 추가 .....	50
계산 필드 편집 .....	50
계산 필드 식에 사용할 수 있는 함수 .....	50
계산 필드 식에 사용할 수 있는 문자열 함수 .....	50
계산 필드 식에 사용할 수 있는 날짜 함수 .....	51
계산 필드 식에 사용할 수 있는 형식 지정 및 해석 함수 .....	53
계산 필드 식에 사용할 수 있는 숫자 함수 .....	54
계산 필드 식에 사용할 수 있는 조건부 함수 .....	54
필드 유형 변경 .....	55
날짜 또는 타임스탬프 필드가 올바르게 인식되었는지 확인 .....	55
날짜 또는 타임스탬프 필드의 표시 형식 변경 .....	56
필드 유형을 지리 데이터로 변경 .....	56
데이터 관리자에서 크로스탭 데이터 피벗 해제 .....	57
크로스탭이란? .....	57
크로스탭을 일반 테이블로 피벗 해제 .....	58
원래 크로스탭 테이블로 되돌리기 .....	58
3.12 데이터 연결 관리 .....	58

권장 사항에 기반하여 모든 테이블 연결 .....	59
두 테이블 연결 .....	59
연결 해제 .....	60
연결 편집 .....	60
데이터 미리 보기 .....	61
가상 키 .....	61
제한 사항 .....	61
<b>4 데이터 로드 스크립트를 사용하여 데이터 로드 .....</b>	<b>63</b>
4.1 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용 .....	63
4.2 데이터 로드 편집기 사용 .....	64
빠른 시작 .....	65
도구 모음 .....	65
기본 도구 모음 .....	66
편집기 도구 모음 .....	66
데이터 로드 편집기에서 데이터 소스에 연결 .....	67
새 데이터 연결 만들기 .....	67
데이터 연결 삭제 .....	68
데이터 연결 편집 .....	68
연결 문자열 삽입 .....	68
데이터 연결에서 데이터 선택 .....	69
스크립트에서 데이터 연결 참조 .....	69
데이터 연결 저장 위치 .....	70
데이터 로드 편집기에서 데이터 선택 .....	70
데이터베이스에서 데이터 선택 .....	70
Microsoft Excel 스프레드시트에서 데이터 선택 .....	71
테이블 파일에서 데이터 선택 .....	72
파일 형식에 맞는 설정 선택 .....	73
스크립트 미리 보기 .....	75
LOAD 문 포함 .....	75
데이터 로드 스크립트 편집 .....	75
명령 및 함수에 대한 구문 도움말 액세스 .....	75
도움말 포털 액세스 .....	76
자동 완성 기능 사용 .....	76
준비된 테스트 스크립트 삽입 .....	76
코드 들여쓰기 .....	76
텍스트 검색 및 바꾸기 .....	77
텍스트 검색 .....	77
텍스트 바꾸기 .....	77
스크립트에 주석 추가 .....	77
주석 처리 .....	78
주석 처리 제거 .....	78
자동 생성 섹션 잠금 해제 .....	78
모든 코드 선택 .....	79
스크립트 코드 분류 .....	79

새 스크립트 섹션 만들기 .....	79
스크립트 섹션 삭제 .....	79
스크립트 섹션 이름 변경 .....	80
스크립트 섹션 재배열 .....	80
데이터 로드 스크립트 디버그 .....	80
디버그 도구 모음 .....	80
출력 .....	81
변수 .....	81
변수를 즐겨찾기로 설정 .....	81
변수 필터링 .....	81
중단점 .....	82
중단점 추가 .....	82
중단점 삭제 .....	82
중단점 사용/사용 안 함 .....	82
로드 스크립트 저장 .....	83
데이터를 로드하는 스크립트 실행 .....	83
데이터 로드 편집기의 키보드 바로 가기 .....	83
키보드 바로 가기 .....	83
4.3 스크립트 구문 및 데이터 구조에 대한 이해 .....	85
추출, 변환 및 로드 .....	85
데이터 로드 문 .....	86
규칙 .....	87
스크립트의 실행 .....	87
필드 .....	87
파생된 필드 .....	87
캘린더 필드 정의 선언 .....	88
<b>Derive</b> 를 사용하여 캘린더에 데이터 필드 매핑 .....	88
시각화에서 파생 날짜 필드 사용 .....	89
필드 태그 .....	89
시스템 필드 태그 .....	89
파생된 필드 태그 .....	90
시스템 필드 .....	90
사용 가능한 시스템 필드 .....	90
시각화에서 시스템 필드 사용 .....	91
필드 이름 바꾸기 .....	91
논리 테이블 .....	92
테이블 이름 .....	93
테이블 레이블 .....	93
논리 테이블 간의 연결 .....	94
<b>Qlik Sense</b> 연결과 <b>SQL</b> 자연 외부 조인 비교 .....	94
연결 필드 내 빈도 정보 .....	95
해결책 .....	95
가상 키 .....	95
가상 키 처리 .....	96

순환 참조에 대한 이해 및 해결 .....	96
순환 참조의 해결 .....	98
테이블 연결 .....	99
자동 연결 .....	99
강제 연결 .....	99
연결 방지 .....	100
이전에 로드한 테이블에서 데이터 로드 .....	100
Resident 또는 선행 LOAD? .....	100
Resident LOAD .....	100
선행 load .....	101
Qlik Sense의 데이터 유형 .....	102
Qlik Sense 내부의 데이터 표현 .....	102
숫자 해석 .....	102
유형 정보가 포함된 데이터 .....	102
유형 정보가 없는 데이터 .....	103
날짜 및 시간 해석 .....	104
달러 기호 확장 .....	105
변수를 사용한 달러 기호 확장 .....	105
파라메타를 사용한 달러 기호 확장 .....	107
표현식을 사용한 달러 기호 확장 .....	107
스크립트에서 인용 부호 사용 .....	108
LOAD 문 내부 .....	108
SELECT 문 내부 .....	108
Microsoft Access 인용 부호 예 .....	108
LOAD 문 외부 .....	108
컨텍스트 외부 필드 참조 및 테이블 참조 .....	109
이름과 리터럴의 차이점 .....	109
숫자와 문자열 리터럴의 차이점 .....	109
문자열에서 작은따옴표 사용 .....	109
데이터의 와일드카드 .....	110
별표 기호 .....	110
OtherSymbol .....	110
NULL 값 처리 .....	111
개요 .....	111
ODBC에서 NULL 값 연결/선택 .....	111
텍스트 파일에서 NULL 값 만들기 .....	112
표현식에서 NULL 값 전파 .....	112
함수 .....	112
산술 및 문자열 연산자 .....	113
관계형 연산자 .....	113
4.4 데이터 및 필드에 대한 지침 .....	114
로드되는 데이터의 양에 대한 지침 .....	114
데이터 테이블 및 필드의 상한 .....	114
로드 스크립트 섹션의 권장 한계 .....	114

숫자 및 시간 서식에 대한 규칙 .....	114
숫자 서식 .....	114
특수 숫자 서식 .....	115
날짜 .....	116
시간 .....	117
타임스탬프 .....	117
4.5 QVD 파일 작업 .....	117
QVD 파일의 용도 .....	118
QVD 파일 만들기 .....	118
QVD 파일에서 데이터 읽기 .....	118
QVD 형식 .....	119
4.6 섹션 액세스를 사용하여 보안 관리 .....	119
스크립트의 섹션 .....	120
섹션 액세스 시스템 필드 .....	120
동적 데이터 축소 .....	122
상속된 액세스 제한 .....	124
<b>5 데이터 소스에 연결 .....</b>	<b>125</b>
5.1 데이터 연결 유형 .....	125
Qlik Sense 기본 제공 커넥터 .....	125
추가적인 Qlik Sense 커넥터 .....	126
타사 커넥터 .....	126
5.2 데이터 연결 저장 위치 .....	126
5.3 제한 사항 .....	126
5.4 파일에서 데이터 로드 .....	126
파일에서 데이터를 로드하는 방법 .....	127
폴더 데이터 연결 설정 .....	127
웹 파일 데이터 연결 설정 .....	128
Microsoft Excel 스프레드시트에서 데이터 로드 .....	128
Microsoft Excel 시트에서 데이터 선택 .....	129
Qlik Sense에서 로드하기 쉽도록 Microsoft Excel 스프레드시트 준비 .....	130
열 머리글 사용 .....	130
데이터 서식 지정 .....	130
영역에 이름 지정 .....	130
비밀번호 보호 제거 .....	130
5.5 데이터베이스에서 데이터 로드 .....	130
ODBC 데이터베이스에서 데이터 로드 .....	131
ODBC .....	131
ODBC 데이터 연결 설정 .....	132
Single Sign-On 로그인 자격 증명 .....	133
ODBC 드라이버 추가 .....	133
64비트 및 32비트 버전 ODBC 구성 .....	133
ODBC 데이터 소스 만들기 .....	133
ODBC 데이터 연결 사용 시 모범 사례 .....	134
ODBC 데이터 연결이 포함된 앱 이동 .....	134

파일 기반 ODBC 데이터 연결에 연결하는 경우의 보안 측면 .....	134
OLE DB .....	134
OLE DB 데이터 연결 설정 .....	134
파일 기반 OLE DB 데이터 연결에 연결하는 경우의 보안 측면 .....	136
데이터베이스 내의 논리 .....	136
5.6 Qlik DataMarket에서 데이터 로드 .....	136
5.7 Direct Discovery를 사용하여 대규모 데이터 셋에 액세스 .....	137
Direct Discovery와 인 메모리 데이터의 차이점 .....	138
인 메모리 모델 .....	138
Direct Discovery .....	139
인 메모리 필드와 Direct Discovery 필드 간 성능 차이 .....	141
인 메모리 데이터와 데이터베이스 데이터의 차이점 .....	142
캐싱 및 Direct Discovery .....	143
Direct Discovery 필드 유형 .....	143
DIMENSION 필드 .....	144
MEASURE 필드 .....	144
DETAIL 필드 .....	144
Direct Discovery에서 지원하는 데이터 소스 .....	145
SAP .....	145
Google Big Query .....	145
MySQL 및 Microsoft Access .....	145
DB2, Oracle 및 PostgreSQL .....	145
Sybase 및 Microsoft SQL Server .....	145
Direct Discovery를 사용할 때의 제한 사항 .....	146
지원되는 데이터 유형 .....	146
보안 .....	146
지원되지 않은 Qlik Sense 기능 .....	146
Direct Discovery에서 다중 테이블 지원 .....	147
Where 절을 사용하여 Direct Discovery 테이블 연결 .....	147
Join On 절을 사용하여 Direct Discovery 테이블 연결 .....	148
Direct Discovery에서 하위 쿼리 사용하기 .....	149
시나리오 1: 집합 크기 작음 .....	150
시나리오 2: 하위 쿼리 사용 .....	150
Direct Discovery 액세스 로깅 .....	151
<b>6 데이터 모델 보기 .....</b>	<b>152</b>
6.1 도구 모음 .....	152
6.2 데이터 모델 뷰어에서 테이블 이동 및 크기 조정 .....	153
테이블 이동 .....	153
테이블 크기 조정 .....	154
6.3 데이터 모델 뷰어에서 테이블 및 필드 미리 보기 .....	154
테이블의 미리 보기 표시 .....	154
필드의 미리 보기 표시 .....	155
6.4 데이터 모델 뷰어에서 마스터 차원 만들기 .....	155
6.5 데이터 모델 뷰어에서 마스터 측정값 만들기 .....	156

<b>7 데이터 모델링의 모범 사례</b> .....	<b>157</b>
7.1 데이터 열을 행으로 변환 .....	157
7.2 데이터 행을 필드로 변환 .....	157
7.3 계층 수준으로 구성된 데이터(예: 조직 체계) 로드 .....	158
7.4 대규모 데이터베이스에서 신규 또는 업데이트된 레코드만 로드 .....	159
7.5 공통 필드를 사용하여 두 테이블의 데이터 결합 .....	159
7.6 간격에 불연속 값 일치 .....	159
7.7 일관성 없는 필드 값 처리 .....	160
7.8 일관성 없는 필드 값 대소문자 처리 .....	160
7.9 특정 지역 관련 데이터를 로드하여 맵으로 데이터 시각화 .....	161
7.10 증분 로드를 통해 새 레코드 및 업데이트된 레코드 로드 .....	162
추가만 .....	162
삽입만(업데이트 또는 삭제는 제외) .....	163
삽입 및 업데이트(삭제는 제외) .....	163
삽입, 업데이트 및 삭제 .....	164
7.11 Join 및 Keep을 사용한 테이블 결합 .....	164
SQL SELECT 문 내의 조인 .....	165
Join .....	165
Keep .....	166
Inner .....	166
Inner Join .....	167
Inner Keep .....	167
Left .....	167
Right .....	168
7.12 조인에 대한 대안으로 매핑 사용 .....	169
7.13 데이터 로드 스크립트에서 교차 표 작업 .....	171
하나의 한정하는 열이 있는 크로스탭 피벗 해제 .....	171
두 개의 한정하는 열이 있는 크로스탭 피벗 해제 .....	172
7.14 일반 데이터베이스 .....	173
7.15 간격과 불연속 데이터의 연결 .....	175
Intervalmatch 예 .....	175
확장된 intervalmatch 구문을 사용하여 느리게 변화하는 차원 문제 해결 .....	176
예제 스크립트: .....	176
7.16 단일 날짜에서 날짜 간격 만들기 .....	178
7.17 계층 구조 데이터 로드 .....	181
7.18 맵 데이터 로드 .....	182
KML 파일의 데이터에서 맵 만들기 .....	183
Excel 파일의 포인트 데이터에서 맵 만들기 .....	184
포인트 데이터 서식 .....	184
데이터 로드 스크립트를 사용하여 단일 열에서 포인트 데이터 로드 .....	185
데이터 로드 스크립트를 사용하여 개별 위도 및 경도 열에서 포인트 데이터 로드 .....	185
표시되는 포인트 수 .....	186

7.19 데이터 정리 .....	186
매핑 테이블 .....	187
규칙: .....	187
매핑 테이블 사용 .....	187
<b>8 문제 해결 — 데이터 로드 .....</b>	<b>189</b>
8.1 SQL Server를 다시 시작한 후 데이터 연결이 작동을 멈춥니다. ....	189
8.2 데이터 로드 시 가상 키 경고 .....	189
8.3 데이터 로드 시 순환 참조 경고 .....	190
8.4 OLE DB 데이터 소스를 사용하는 경우의 데이터 선택 문제 .....	190
8.5 ANSI 이외로 인코딩된 데이터 파일을 사용하는 경우의 문자 집합 문제 .....	190
8.6 커넥터가 작동하지 않습니다. ....	191
커넥터가 제대로 설치되지 않았습니다. ....	191
커넥터가 Qlik Sense에 맞게 조정되어 있지 않습니다. ....	191
8.7 데이터 로드 스크립트가 오류 없이 실행되었으나 데이터가 로드되지 않습니 다. ....	191
세미콜론으로 종결되지 않은 문 .....	191
문자열 내의 작은따옴표 .....	192
8.8 고정된 레코드 파일에서 데이터를 선택하면 열이 예상대로 정렬되지 않습니 다. ....	192
8.9 파일 첨부 시 "잘못된 경로" 오류 메시지 .....	192
8.10 QlikView 문서에서 변환된 앱 로드 시 오류 발생 .....	193
스크립트에서 절대 파일 경로 참조가 사용되었습니다. ....	193
스크립트에서 지원되지 않는 함수 또는 문이 사용되었습니다. ....	193
8.11 ODBC를 통해 Microsoft Excel 파일에 연결하거나 해당 파일에서 데이터를 로 드할 때 문제 발생 .....	193
8.12 파일을 데이터 추가에 끌어서 첨부하는 방식이 작동하지 않습니다. ....	194
8.13 공통 필드가 있는 테이블이 필드 이름에 의해 자동으로 연결되지 않습니다. ...	194
8.14 날짜 필드가 시트 보기에서 날짜 필드로 인식되지 않습니다. ....	194
날짜 서식이 인식되지 않습니다. ....	195
8.15 섹션 액세스 규칙이 변경되었지만 모든 데이터를 볼 수 있습니다. ....	195
8.16 데이터가 포함된 앱의 테이블이 데이터 관리자에 표시되지 않습니다. ....	195
8.17 서버에서 앱을 만드는 모든 사용자에게 대해 데이터 관리자 워크플로가 손상되 었습니다 .....	196
8.18 디스크 공간이 부족합니다. ....	196
8.19 데이터 로드 편집기에 스크립트가 표시되지 않습니다. ....	196

# 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 함수를 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/sense](https://help.qlik.com/sense)에서 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 앱에서 데이터 관리

데이터 관리자를 앱에서 사용하여 자체 데이터 소스 또는 Qlik DataMarket의 데이터를 추가하고 관리할 수 있습니다.

데이터 관리자에는 두 가지 보기가 있습니다.

- ▶ **Associations**

테이블 간의 연결을 만들고 편집할 수 있습니다.

- ' **Tables**

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 **ú** 을 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다. 또한 새 앱을 만들 때 데이터를 추가할 것인지 묻는 메시지가 표시됩니다.

다음 데이터 소스에서 데이터를 추가할 수 있습니다.

<b>연결</b>	사용자 또는 관리자에 의해 정의된 데이터 연결 및 사용자가 이전에 데이터를 선택했던 폴더에서 선택합니다.  사용자에 의해 정의된 데이터 연결 및 사용자가 이전에 데이터를 선택했던 폴더에서 선택합니다.
<b>내 데이터 연결</b>	ODBC 또는 OLE DB 데이터베이스, 데이터 파일, 웹 파일 또는 REST 및 Salesforce 등의 커넥터와 같은 새로운 데이터 소스에서 선택합니다. Salesforce와 같은 새 데이터 소스에서 선택합니다.
<b>Qlik DataMarket</b>	공용 데이터베이스 및 상용 데이터베이스의 정규화된 데이터에서 선택합니다.
<b>파일 첨부</b>	사용자 컴퓨터의 데이터 파일을 업로드한 후 앱에 첨부합니다.

### 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. 달기를 클릭하여 돌아갑니다.

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



데이터 추가를 사용하여 추가한 데이터 테이블만 편집할 수 있습니다. 로드 스크립트를 사용하여 로드한 테이블에서 @ 을 클릭하면 데이터 로드 편집기가 열립니다.  
참조: 데이터 로드 편집기 사용(64페이지)

### 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 Ö 을 클릭합니다.

이제 해당 테이블은 삭제 보류로 표시되며 다음 번에 데이터를 다시 로드할 때 제거됩니다.

B 및 C 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 추가의 데이터 로드 및 마침 옵션을 사용하여 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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

- 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

연결 문제가 발생할 수 있는 데이터를 로드해야 하는 경우 **데이터 추가**의 **데이터 준비** 옵션을 사용하는 것이 바람직합니다.

**Qlik Sense**에서는 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 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 로드 및 마침** 옵션을 사용하면 데이터 프로파일링이 활성화되지 않으며, **Qlik Sense**에서는 공통 필드 이름에 기반하여 자동으로 테이블을 연결합니다.

### 3.6 변경 적용 및 데이터 다시 로드

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

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



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

**데이터 관리자**의 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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

- 삭제 보류  $\ddot{O}$  로 표시된 테이블이 삭제됩니다.
- 업데이트 보류  $\hat{O}$  로 표시된 테이블이 추가, 이름 변경 또는 제거된 필드로 업데이트되거나 테이블의 이름이 변경됩니다.
- 추가 보류  $m$  로 표시된 테이블이 추가됩니다.

다음과 같이 하십시오.

- 데이터 로드를 클릭하여 앱의 변경 내용을 로드합니다.

이제 데이터 관리자에서 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  $\hat{O}$  버튼을 클릭합니다.

### 3.7 데이터 관리자에서 작업 실행 취소 및 다시 실행

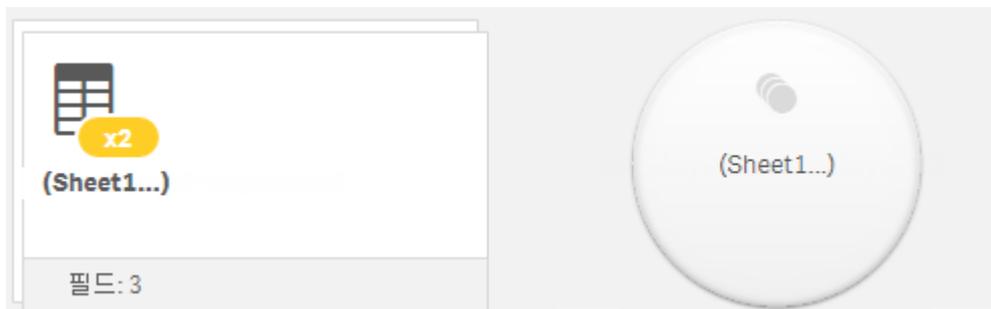
데이터 관리자에서 편집하는 경우, **B** 및 **C** 를 클릭하거나 키보드 바로 가기 **Ctrl + Z** 및 **Ctrl + Y** 를 사용하여 일부 작업을 실행 취소 또는 다시 실행할 수 있습니다.

다음 경우에는 작업 로그가 지워집니다.

- 테이블 개요에서 연결로 이동하는 등 보기를 변경하는 경우
- 데이터를 로드하는 경우
- 데이터 관리자를 닫는 경우

### 3.8 컨케트네이트된 테이블

추가한 두 개 이상의 테이블에서 필드 이름과 필드의 수가 완전히 동일한 경우 Qlik Sense는 자동으로 해당 내용을 한 테이블로 컨케트네이트합니다.



테이블 보기 및 연결 보기의 컨케트네이트된 테이블

참조: 테이블 연결 (99페이지)

### 컨케트네이트된 테이블 분할

컨케트네이트된 단일 테이블을 만들려고 한 경우가 아니면 컨케트네이트된 테이블을 구성 요소 테이블로 분할할 수 있습니다.

다음과 같이 하십시오.

1. 컨케트네이트된 테이블을 선택합니다.
2. → 를 클릭합니다.  
분할 결과로 생성되는 테이블의 미리 보기가 표시됩니다.
3. **분할**을 클릭합니다.

이제 테이블은 해당 구성 요소 테이블로 분할되고 구성 요소 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

### 3.9 데이터 관리자와 데이터 로드 스크립트 간의 상호 작용

**데이터 관리자**에서 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.

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.

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 관리자를 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 앱에 데이터 추가

앱에 데이터를 신속하게 추가할 수 있습니다. **데이터 관리자**를 연 다음 **ú**를 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다. 또한 새 앱을 만들 때 데이터를 추가할 것인지 묻는 메시지가 표시됩니다.

다음과 같은 데이터 소스 그룹을 사용할 수 있습니다.

- **연결**

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

정의한 데이터 소스에서 데이터를 추가할 수 있습니다. **내 데이터 파일**에 업로드된 파일 또는 **Salesforce.com**과 같은 외부 데이터 소스에 대한 커넥터가 여기에 포함될 수 있습니다. 다음에 주의하십시오.

- 파일을 **내 데이터 파일**로 업로드한 후 여러 앱에서 재사용하면 **파일 첨부**를 사용하여 동일한 파일을 여러 앱에 첨부하는 것보다 저장소 공간을 덜 사용합니다.
- **내 데이터 파일**에서 앱으로 데이터를 로드하는 경우 해당 데이터는 압축됩니다. 앱은 데이터 파일을 첨부하는 경우보다 작아집니다.
- 데이터를 앱으로 다시 로드하려면 업데이트된 데이터 파일을 **내 데이터 파일**로 가져온 후 데이터를 앱으로 다시 로드합니다.

- **내 데이터 연결**

이전에 사용하지 않았던 데이터 소스에서 데이터를 선택하고 새 데이터 연결을 만들 수 있습니다.

현재 **Qlik Sense Cloud**에서 지원되는 커넥터 목록을 보려면 계정에 로그인한 후 **업그레이드 버튼**을 클릭합니다.

- **Qlik DataMarket**

**Qlik Sense** 내에서 직접 액세스할 수 있는 외부 소스에서 사용할 준비가 된 최신 데이터에 대한 확장 모음의 데이터를 추가할 수 있습니다. **Qlik DataMarket**은 현재 및 이전 날씨와 인구 통계 데이터, 환율은 물론, 비즈니스, 경제 및 사회적 데이터도 제공합니다.

사용할 수 있는 **Qlik DataMarket** 소스는 **Qlik Sense Cloud** 구독 상황에 따라 달라집니다.

- **파일 첨부**

컴퓨터 또는 네트워크 드라이브의 파일에서 데이터를 추가할 수 있습니다. 파일이 업로드되어 앱에 첨부됩니다.

파일 첨부 기능을 사용하면 컴퓨터의 데이터 파일을 **Qlik Sense Cloud**의 앱에 첨부할 수 있습니다. **내 데이터 파일**에 업로드한 파일에서 단지 데이터를 로드하는 것이 아니라 데이터 파일을 첨부하는 것입니다. 다음에 주의하십시오.

- 데이터 파일을 첨부하거나 첨부 해제하려면 앱이 개인 클라우드의 **내 작업** 영역이나 그룹 **작업 영역**에 있어야 합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

- 첨부된 파일은 해당 파일을 첨부한 앱에서만 사용할 수 있습니다.
- 데이터 파일을 첨부하면 앱 크기가 개인 데이터 저장소 공간에서 데이터를 로드하는 경우보다 커집니다.
- 25MB를 넘는 파일은 첨부할 수 없습니다. 25MB보다 큰 파일의 데이터를 추가하려면 **연결**을 사용하여 개인 데이터 저장소 공간으로 데이터 파일을 가져옵니다. 그 다음, 데이터를 앱에 로드할 수 있습니다.
- 첨부된 데이터 파일과 함께 앱을 이식할 수 있습니다. 앱을 다운로드하고 **Qlik Sense Enterprise** 또는 **Qlik Sense Desktop** 등의 다른 환경으로 이동하면 데이터 파일이 앱에 첨부된 그대로 유지됩니다.
- 앱의 데이터를 업데이트하려면 **파일 첨부**를 사용하여 업데이트된 데이터 파일을 첨부한 후 데이터를 앱으로 다시 로드합니다.

## 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 Desktop**에서만 사용할 수 있습니다. **Qlik Sense**의 서버 설치에서는 이러한 파일을 사용할 수 없습니다.



**Qlik Sense**의 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.

**Qlik Sense Cloud**에서 이미 정의한 외부 데이터 연결에서 데이터를 추가할 수 있습니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 **ú** 을 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **연결**을 클릭합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

4. 사용할 연결을 선택합니다.  
일부 연결은 로드할 테이블 및 필드를 선택할 수 있는 데이터 소스에 직접 연결됩니다. 예를 들어 **Salesforce.com**이나 **IBM DB2**와 같은 데이터베이스에 연결하면 데이터를 선택할 수 있는 소스에 직접 연결됩니다.
5. 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 선택합니다.
  - 웹 파일: 웹 파일의 **URL**을 입력합니다.
  - 기타 데이터 소스: 커백터에서 지정합니다.
6. 로드할 테이블과 필드를 선택합니다.
7. 해당 데이터에서 작업을 계속할지 또는 해당 데이터를 앱으로 로드할지 선택합니다.
  - **데이터 준비**를 클릭하여 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결합니다. 날짜 및 시간 필드는 자동으로 생성되어 **autoCalendar**에 매핑됩니다. 데이터 프로파일링도 수행되어 연결 권장 사항을 제공합니다. 테이블 간에 공통인 모든 필드는 자동으로 한정됩니다.



공통 필드 이름을 기준으로 테이블이 자동으로 연결되지 않으므로 **데이터 관리자**의 **연결 보기**에서 모든 테이블을 연결해야 합니다.

- **데이터 로드 및 마침**을 클릭하여 추가된 데이터를 앱으로 로드한 후 시트로 이동합니다. 이렇게 하면 외부 데이터 소스에서 새로 선택된 데이터를 로드합니다.  
외부 소스에서 선택한 모든 데이터를 다시 로드하려면 **데이터 관리자** 바닥글의 **Ô** 버튼을 사용합니다. 이렇게 하면 선택 내용에 대한 소스에서 현재 데이터를 모두 가져올 수 있습니다. 모든 데이터를 다시 로드하면 새 데이터만 로드하는 것보다 더 오래 걸립니다. 이전에 로드한 데이터가 데이터 소스에서 변경되지 않은 경우 모든 데이터를 다시 로드할 필요는 없습니다.  
공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다.



**데이터 로드 및 마침**을 사용하는 경우 날짜 및 시간 필드는 생성되지 않습니다.

심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.

### 업로드된 데이터 파일에서 데이터 추가

개인 클라우드에 데이터 파일을 업로드한 후 데이터를 앱으로 로드할 수 있습니다.

#### 데이터 파일 가져오기

데이터를 추가하려면 먼저 데이터 파일을 **Qlik Sense Cloud**로 가져와야 합니다.



개인 클라우드에 업로드한 파일은 그룹 작업 영역에서 사용할 수 없습니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

다음과 같이 하십시오.

1. 왼쪽 메뉴에서 **내 데이터 파일**을 클릭한 다음 **파일 가져오기**를 클릭합니다.
2. 데이터 파일을 업로드합니다.

#### 앱으로 데이터 로드

사용자에 의해 이미 정의된 연결에서 데이터를 선택할 수 있습니다. Qlik Sense Cloud에서는 데이터 파일이 포함된 폴더일 수 있습니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 **ú**을 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **연결**을 클릭한 다음 Qlik Sense Cloud 프로필과 이름이 같은 폴더를 선택합니다.
4. 데이터를 추가할 파일을 선택합니다.
5. 로드할 테이블과 필드를 선택합니다.
6. 해당 데이터에서 작업을 계속할지 또는 해당 데이터를 앱으로 로드할지 선택합니다.
  - **데이터 준비**를 클릭하여 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결합니다. 날짜 및 시간 필드는 자동으로 생성되어 autoCalendar에 매핑됩니다. 데이터 프로파일링도 수행되어 연결 권장 사항을 제공합니다. 테이블 간에 공통인 모든 필드는 자동으로 한정됩니다.



공통 필드 이름을 기준으로 테이블이 자동으로 연결되지 않으므로 **데이터 관리자**의 **연결** 보기에서 모든 테이블을 연결해야 합니다.

- **데이터 로드 및 마침**을 클릭하여 추가된 데이터를 앱으로 로드한 후 시트로 이동합니다. 또한 이 작업을 수행하면 기존의 모든 데이터가 데이터 소스에서 다시 로드됩니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다.



**데이터 로드 및 마침**을 사용하는 경우 날짜 및 시간 필드는 생성되지 않습니다.

심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.

#### 업데이트된 데이터 파일 가져오기

앱의 데이터를 업데이트하려면 업데이트된 데이터 파일을 클라우드로 가져온 후 데이터를 앱으로 다시 로드합니다.

다음과 같이 하십시오.

1. **내 데이터 파일**을 클릭한 후 **파일 가져오기**를 클릭합니다.
2. 데이터 파일을 업로드합니다.

#### 앱으로 데이터 다시 로드

다음과 같이 하십시오.

1. 개인 클라우드의 **내 작업** 영역에서 앱을 열고 도구 모음 드롭다운 목록에서 **데이터 추가**를 선택한 다음 **데이터 파일**을 클릭합니다.
2. **Qlik Sense Cloud** 프로필과 이름이 같은 폴더를 선택한 후 원본 파일과 이름이 같은 업데이트된 파일을 선택합니다.
3. 데이터를 다시 로드합니다.

#### 새 데이터 소스의 데이터 추가

새 데이터 소스에서 앱에 데이터를 추가할 수 있습니다. 새 데이터 소스의 데이터를 추가하는 경우, 해당 데이터 소스에 대한 연결이 **연결**에서 만들어지므로 동일한 데이터 소스에서 데이터를 보다 쉽게 추가할 수 있습니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 **ú**을 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **내 데이터 연결**을 클릭합니다.
4. 사용할 데이터 소스의 유형을 선택합니다.
5. 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.
6. 로드할 테이블과 필드를 선택합니다.
7. 해당 데이터에서 작업을 계속할지 또는 해당 데이터를 앱으로 로드할지 선택합니다.
  - **데이터 준비**를 클릭하여 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결합니다. 날짜 및 시간 필드는 자동으로 생성되어 **autoCalendar**에 매핑됩니다. 데이터 프로파일링도 수행되어 연결 권장 사항을 제공합니다. 테이블 간에 공통인 모든 필드는 자동으로 한정됩니다.



공통 필드 이름을 기준으로 테이블이 자동으로 연결되지 않으므로 **데이터 관리자의 연결 보기**에서 모든 테이블을 연결해야 합니다.

- **데이터 로드 및 마침**을 클릭하여 추가된 데이터를 앱으로 로드한 후 시트로 이동합니다. 또한 이 작업을 수행하면 기존의 모든 데이터가 데이터 소스에서 다시 로드됩니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다.



**데이터 로드 및 마침**을 사용하는 경우 날짜 및 시간 필드는 생성되지 않습니다.

심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.

## 데이터 파일 첨부 및 앱에 데이터 추가

사용자 컴퓨터의 데이터 파일을 사용하는 경우 데이터 파일을 업로드한 후 앱에 첨부할 수 있습니다.

첨부된 파일은 해당 파일을 첨부한 앱에서만 사용할 수 있습니다. 다른 앱에서 데이터 파일을 사용하려면 데이터 파일 저장소로 업로드합니다.

원본 데이터 파일에 대한 연결이 없으므로 원본 파일을 업데이트한 경우에는 첨부된 파일을 새로 고쳐야 합니다.



**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.

제한된 데이터를 노출하지 않으려면 앱을 게시하기 전에 섹션 액세스 설정이 있는 모든 첨부 파일을 제거합니다.

참조: 섹션 액세스를 사용하여 보안 관리(119페이지)

앱에 첨부된 최대 총 파일 크기는 미디어 라이브러리에 업로드된 이미지 파일을 포함하여, 200MB입니다.



**파일 첨부**는 *Qlik Sense Desktop*에서 사용할 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 Sense**에서는 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.

참조: 테이블 편집(44페이지)

#### 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.

파일을 앱에 첨부하려면 앱이 개인 클라우드의 **내 작업** 영역이나 **그룹 작업 공간**에 있어야 합니다. 25MB를 넘는 파일은 첨부할 수 없습니다. 25MB를 넘는 파일의 데이터를 추가하려면 해당 파일을 데이터 파일 저장소에 업로드한 후 데이터를 앱으로 로드합니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 **ú** 을 클릭합니다.” 메뉴에서 **데이터 추가**를 클릭할 수도 있습니다.
3. **파일 첨부**를 클릭합니다.
4. 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.

5. 로드할 테이블과 필드를 선택합니다.
6. 해당 데이터에서 작업을 계속할지 또는 해당 데이터를 앱으로 로드할지 선택합니다.
  - **데이터 준비**를 클릭하여 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결합니다. 날짜 및 시간 필드는 자동으로 생성되어 autoCalendar에 매핑됩니다. 데이터 프로파일링도 수행되어 연결 권장 사항을 제공합니다. 테이블 간에 공통인 모든 필드는 자동으로 한정됩니다.



공통 필드 이름을 기준으로 테이블이 자동으로 연결되지 않으므로 **데이터 관리자의 연결 보기**에서 모든 테이블을 연결해야 합니다.

- **데이터 로드 및 마침**을 클릭하여 추가된 데이터를 앱으로 로드한 후 시트로 이동합니다. 또한 이 작업을 수행하면 기존의 모든 데이터가 데이터 소스에서 다시 로드됩니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다.



**데이터 로드 및 마침**을 사용하는 경우 날짜 및 시간 필드는 생성되지 않습니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

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 **ú** 을 클릭합니다.
3. **파일 첨부**를 클릭합니다.
4. 해당 파일을 삭제합니다.

이제 데이터 파일이 앱에서 영구적으로 삭제됩니다.



앱에서 사용되는 첨부된 파일을 삭제하면 **데이터 관리자** 또는 **로드 스크립트**에서 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. 원래 파일을 업데이트하는 경우 앱에 첨부된 파일을 새로 고쳐야 합니다. 그 다음, 업데이트된 데이터를 앱에 로드할 수 있습니다.

다음과 같이 하십시오.

1. 앱을 엽니다.
2. **데이터 관리자**를 연 다음 **ú** 을 클릭합니다.
3. **파일 첨부**를 클릭합니다.
4. 기존 파일을 바꿉니다. 업데이트된 파일은 원래 파일과 이름이 같아야 합니다. 데이터 파일 내용이 새로 고쳐집니다.
5. **데이터 준비**를 클릭합니다.
6. **연결 보기** 또는 **테이블 보기**에서 테이블을 클릭합니다.
7. **ô** 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.

## 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.



현재 Qlik Sense Cloud에서 OLE DB/ODBC 데이터베이스에 연결할 수 없습니다.

## 데이터베이스에서 데이터 선택

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

참조: [ODBC\(131페이지\)](#)

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 끝나면 다음 두 가지 방법 중 하나를 계속 수행할 수 있습니다.
  - **데이터 준비**를 클릭하여 데이터 프로파일링을 계속하고 테이블 관계에 대한 권장 사항을 확인합니다.
  - 선택한 데이터를 그대로 로드하기 위해 **데이터 로드 및 마침**을 클릭하여 데이터 프로

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

파일링 단계를 통과하고 시각화 만들기를 시작합니다. 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.

**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.

<b>Machine:</b>	AEJ12B		
<b>Date:</b>	2015-10-05 09		
<b>Timestamp</b>	<b>Order</b>	<b>Operator</b>	<b>Yield</b>
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.

**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 읽은 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 끝나면 다음 두 가지 방법 중 하나를 계속 수행할 수 있습니다.
  - **데이터 준비**를 클릭하여 데이터 프로파일링을 계속하고 테이블 관계에 대한 권장 사항을 확인합니다.
  - 선택한 데이터를 그대로 로드하기 위해 **데이터 로드 및 마침**을 클릭하여 데이터 프로파일링 단계를 통과하고 시각화 만들기를 시작합니다. 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.

4. 데이터 선택이 끝나면 다음 두 가지 방법 중 하나를 계속 수행할 수 있습니다.
  - **데이터 준비**를 클릭하여 데이터 프로파일링을 계속하고 테이블 관계에 대한 권장 사항을 확인합니다.
  - 선택한 데이터를 그대로 로드하기 위해 **데이터 로드 및 마침**을 클릭하여 데이터 프로파일링 단계를 통과하고 시각화 만들기를 시작합니다. 테이블은 자연 연결, 즉 공통으로 이름이 지정된 필드를 사용하여 연결됩니다.

#### 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

**파일 형식** 구분 또는 고정 레코드로 설정합니다.

선택하면, 데이터 선택 대화 상자가 선택한 파일 형식에 맞춰 조정됩니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

<b>필드 이름</b>	테이블에 <b>포함된 필드 이름</b> 이 있는지, 또는 <b>필드 이름 없음</b> 이 있는지 지정합니다.
<b>구분 기호</b>	테이블 파일에서 사용되는 <b>구분 기호</b> 를 설정합니다.
<b>인용</b>	인용 부호를 처리하는 방법을 지정하기 위해 설정합니다.  <b>없음</b> = 인용 문자가 허용되지 않음  <b>표준</b> = 표준 인용 부호(필드 값의 첫 문자 및 마지막 문자로 인용 부호 사용 가능)  <b>MSQ</b> = 최신 인용 부호(필드에서 복수 행의 콘텐츠 허용)
<b>헤더 크기</b>	테이블 헤더로 생략할 행 수를 설정합니다.
<b>문자 집합</b>	테이블 파일에서 사용되는 문자 집합을 설정합니다.
<b>주석</b>	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. <b>Qlik Sense</b> 에서는 여기에 지정된 문자로 시작하는 행을 로드하지 않습니다.
<b>EOF 무시</b>	데이터에 필드 값의 일부로 파일 끝 문자가 포함되는 경우, <b>EOF 무시</b> 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 위치에** 선택표로 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

<b>필드 이름</b>	테이블에 <b>포함된 필드 이름</b> 이 있는지, 또는 <b>필드 이름 없음</b> 이 있는지 지정합니다.
<b>헤더 크기</b>	<b>헤더 크기</b> 를 테이블 헤더로서 생략할 행 수로 설정합니다.

---

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

<b>문자 집합</b>	테이블 파일에서 사용하는 문자 집합으로 설정됩니다.
<b>탭 크기</b>	테이블 파일에서 하나의 탭 문자가 나타내는 공백 수로 설정합니다.
<b>레코드 줄 크기</b>	테이블 파일에서 하나의 레코드 범위에 해당하는 행 수로 설정합니다. 기본 값은 1입니다.

### HTML 파일

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

### 파일 형식 설정

<b>필드 이름</b>	테이블에 <b>포함된 필드 이름</b> 이 있는지, 또는 <b>필드 이름 없음</b> 이 있는지 지정합니다.
<b>문자 집합</b>	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.

## Qlik DataMarket의 데이터 추가

Qlik DataMarket은 Qlik Sense 내에서 직접 액세스할 수 있는 외부 소스에서 사용할 준비가 된 최신 데이터에 대한 확장 모음을 제공합니다. Qlik DataMarket은 현재 및 이전 날씨와 인구 통계 데이터, 환율은 물론, 비즈니스, 경제 및 사회적 데이터도 제공합니다.

또한 Qlik DataMarket은 Eurostat 데이터베이스에서 Database by themes, Tables by themes, Tables on EU policy, Cross cutting topics 등의 데이터 집합을 제공합니다.

Qlik DataMarket에서 데이터를 추가할 때는 범주를 선택한 후 이러한 범주에서 사용 가능한 데이터의 필드를 필터링합니다. DataMarket 범주에는 대량의 데이터가 포함되어 있으며, 필터링을 통해 데이터를 하위 집합으로 좁혀서 로드되는 데이터의 양을 줄일 수 있습니다.

일부 Qlik DataMarket 데이터는 무료로 사용할 수 있습니다. **프리미엄**으로 표시된 데이터 패키지는 구독 요금을 내면 사용할 수 있습니다.

Qlik DataMarket 데이터를 사용하려면 먼저 사용 약관에 동의해야 합니다. 또한 프리미엄 데이터 패키지에 대한 라이선스를 구매한 경우 해당 패키지의 데이터를 사용하려면 액세스 자격 증명을 입력해야 합니다. 액세스 자격 증명에 적용되면 프리미엄 데이터는 **사용 허가됨**으로 레이블이 지정됩니다.

약관에 동의하고 어떠한 프리미엄 데이터 패키지에 대한 라이선스도 입력하지 않는 경우, 프리미엄 패키지 옆에 라이선스를 구입할 수 있는 **구매** 버튼이 표시됩니다. **구매** 버튼은 **프리미엄** 레이블을 대체합니다.



*Qlik Sense Desktop을 사용할 때는 Qlik DataMarket 사용 약관에 동의할 필요가 없습니다. Qlik Sense Desktop에서는 프리미엄 데이터 셋을 사용할 수 없으므로 액세스 자격 증명도 필요 없습니다.*

다음과 같이 하십시오.

1. 데이터 추가를 클릭합니다.
2. 데이터 소스 선택 단계에서 **Qlik DataMarket**을 클릭하여 Qlik DataMarket 패키지를 표시합니다.



*DataMarket 사용자 인터페이스는 광고를 차단하고 개인정보 보호를 강화하는 Privacy Badger와 같은 브라우저 확장 기능에 의해 차단될 수 있습니다. 해당 확장 기능이 타사의 사용자 추적에 대한 DataMarket 통신에서 오류가 발생한 경우에 이런 문제가 발생합니다. 이런 경우, DataMarket을 차단하는 브라우저 확장 기능의 차단된 사이트 목록에서 Qlik Sense 사이트를 제외시키면 DataMarket에 액세스할 수 있습니다.*

3. 데이터 패키지를 선택합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

목록에서 데이터 패키지(예: **Essentials Free** 또는 **Financial Reports**)를 선택하거나 특정 유형의 데이터를 검색할 수 있습니다. Qlik DataMarket을 검색하려면 데이터 패키지 목록 상단에서 **DataMarket 검색**을 선택하고 **데이터 셋 선택** 단계에서 검색 텍스트 상자에 검색어 또는 구를 입력합니다.

참조: *Qlik DataMarket 검색 (38페이지)*

4. **데이터 셋 선택** 단계에서 데이터 셋을 선택합니다.  
선택한 DataMarket 패키지에 따라서는 패키지 내의 몇몇 하위 범주를 선택하여 데이터를 포함하는 개별 데이터 셋에 연결해야 할 수 있습니다. 데이터 셋은 데이터 셋 설명 옆에 있는 ] 아이콘으로 구별됩니다. ] 아이콘을 클릭하면 데이터 셋의 메타데이터가 표시됩니다. 데이터 셋을 선택하면 해당하는 특정 데이터 모음에 대한 연결이 생성됩니다.
5. **로드할 데이터 선택** 단계의 각 차원, 측정값 및 기간에서 하나 이상의 필터를 선택합니다. 왼쪽 창에 해당 차원, 측정값 및 기간이 목록으로 표시됩니다. 왼쪽 창에서 차원, 측정값 및 또는 기간을 클릭하면 해당 차원, 측정값 또는 기간에 대한 값이 오른쪽 창에 표시됩니다. 왼쪽 열 아래쪽에 로드 크기 표시기가 있어서 현재 선택한 필드에 얼마나 많은 셀이 로드될지 대략적으로 표시합니다. 표시기는 수가 적을 때는 녹색이고, 로드 시간에 확연히 영향을 미칠 수 있는 크기로 수가 늘어나면 노란색이 됩니다. 데이터의 양이 굉장히 많아서 성공적으로 로드할 수 없는 경우 표시기는 빨간색이 됩니다.
6. 해당 데이터에서 작업을 계속할지 또는 해당 데이터를 앱으로 로드할지 선택합니다.
  - **데이터 준비**를 클릭하여 **데이터 관리자**에서 계속 데이터 소스를 추가하고, 데이터를 변환하고, 테이블을 연결합니다. 날짜 및 시간 필드는 자동으로 생성됩니다. 데이터 프로파일링도 수행되어 연결 권장 사항을 제공합니다. 테이블 간에 공통인 모든 필드는 자동으로 한정됩니다.



공통 필드 이름을 기준으로 테이블이 자동으로 연결되지 않으므로 **데이터 관리자의 연결** 보기에서 모든 테이블을 연결해야 합니다.

- **데이터 로드 및 마침**을 클릭하여 추가된 데이터를 앱으로 로드한 후 시트로 이동합니다. 또한 이 작업을 수행하면 기존의 모든 데이터가 데이터 소스에서 다시 로드됩니다. 공통 필드 이름을 기준으로 테이블이 자동으로 연결됩니다.



**데이터 로드 및 마침**을 사용하는 경우 날짜 및 시간 필드는 생성되지 않습니다.

심각한 문제가 발견된 경우는 데이터를 앱으로 로드하기 전에 **데이터 관리자**에서 문제를 해결해야 합니다.

#### Qlik DataMarket 차원, 측정값 및 기간 선택

Qlik DataMarket 데이터 셋에는 차원, 측정값 및 기간이 포함됩니다. 차원은 국가, 연령대 또는 성별과 같은 데이터의 컨텍스트를 정의합니다. 측정값에는 차원의 컨텍스트와 관련된 숫자 데이터가 포함됩니다. 기간은 측정값을 표시할 연도 또는 기타 기간을 지정합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

데이터 셋에는 적어도 하나의 차원과 하나의 측정값이 포함되며, 모두 시간 차원을 가지고 있습니다. 앱에 데이터를 추가하기 전에 먼저 적어도 하나의 차원과 하나의 측정값을 선택하고 기간을 설정해야 합니다. 차원 선택 시, 데이터가 들어 있는 차원을 포함해야 합니다. 데이터가 계층 구조적으로 구성된 경우, 부모 분기에는 데이터가 포함되지 않을 수 있습니다.

일부 차원에는 데이터의 여러 표현이 포함되어 있습니다. 예를 들어 국가 이름으로 지정된 지리적 위치에는 국가 코드 ISO(International Standards Organization)도 포함됩니다. 통화에는 Pound sterling 및 Euro와 같은 일반적인 이름뿐 아니라 해당 ISO 4217 코드(GBP 및 EUR)도 포함됩니다. 차원의 추가 값은 개별적으로 선택할 수 없습니다. 이러한 값은 차원에 대한 설명에 표시됩니다.

#### 로드할 데이터 선택

**Selected development indicators**

**Dimension**

! **Geographical area** 0/222

\_dim\_8so2, World, Region, Country, ISO 3166

**Time**

! Year

**Measure**

! Indicator 0/20

**Geographical area**

---

🔍

▼ World

---

▶ East Asia & Pacific

---

▶ Europe & Central Asia

---

▶ Latin America & Caribbean

---

▶ Middle East & North Africa

---

▼ North America

---

Bermuda

---

Canada

---

United States

---

▶ South Asia

---

▶ Sub-Saharan Africa

---

추가 값은 별도 필드로 로드됩니다.

Geographical area.qlik_datam ...	World	Region	Country	ISO 3166
3	World	Europe & Central Asia		
5	World	East Asia & Pacific		
8	World	North America		
16	World	North America	Canada	CA
41	World	East Asia & Pacific	New Zealand	NZ
k	World	Europe & Central Asia	Austria	AT

일부 데이터 셋에서는 데이터 셋에 하나의 측정값만 포함되므로 측정값을 선택할 필요가 없습니다. 측정값 선택은 선택할 측정값이 둘 이상 있는 경우에만 표시됩니다. 예를 들어, **US per capita personal income by state** 데이터 셋은 해당 데이터 셋에 단 하나의 측정값(1인당 개인 소득)만 있으므로 지역적 차원과 기간만 표시합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

또한 차원 선택이 필요 없는 데이터 셋도 있습니다. 예를 들어, **US federal interest rate** 및 **US consumer price index for urban consumers** 데이터 셋은 해당 데이터 셋에 단 하나의 차원과 하나의 측정값만 있으므로 기간만 선택하면 됩니다. 첫 번째의 경우, 측정값은 **federal interest rate**이고 차원은 **United States**입니다. 두 번째의 경우, 측정값은 **consumer price index**이고 차원은 **United States urban consumers**입니다.

데이터 셋에는 측정값에 수반되는 데이터가 없는 차원만 포함할 수 있습니다. 데이터 셋에는 회사 최고경영자(CEO) 목록 등의 데이터만 포함할 수 있습니다. 이런 경우 차원 내에서 이루어지는 선택이 없으므로 해당 차원이 미리 선택됩니다.

#### Qlik DataMarket에서 계층 구조 데이터 선택

많은 Qlik DataMarket 데이터 셋에는 계층 구조적으로 구조화된 차원 및 측정값이 포함됩니다. 계층 구조적으로 구조화된 DataMarket 데이터 셋에는 2수준 및 3수준 계층 구조가 포함됩니다. 이 계층 구조에서 선택하는 방법은 각 수준의 데이터에 따라 달라집니다.

#### 부모 및 자식 분기 선택

일부 계층 구조에서는 언제든지 모든 수준에서 선택할 수 있습니다. 예를 들어, **Selected development indicators** 데이터 셋에는 3개 수준의 **Geographical area** 차원이 포함됩니다.

#### 로드할 데이터 선택

The screenshot displays the 'Selected development indicators' configuration panel. On the left, the 'Dimension' section shows 'Geographical area' selected with a count of 0/222. Below it, the 'Time' section has 'Year' selected, and the 'Measure' section has 'Indicator' selected with a count of 0/20. On the right, the 'Geographical area' panel is active, showing a search icon and a list of regions with checkboxes: World, East Asia & Pacific, Europe & Central Asia, Latin America & Caribbean, Middle East & North Africa, North America (expanded to show Bermuda, Canada, and United States), South Asia, and Sub-Saharan Africa.

**World, Region** 또는 **Country**에서의 선택은 그 자체로 유효합니다. 가장 높은 수준을 포함하는 모든 선택은 특정 지역 및 국가가 선택되어 있는 경우라도 지역 및 국가의 모든 데이터를 로드합니다. 그러나 지역이 자체적으로 선택된 경우에는 전세계의 지역만 로드됩니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

**World** 및 **North America**를 모두 선택한 경우, 전세계 데이터는 **North America** 데이터와는 별도로 표시됩니다.

**Country**에서 **Canada**를 선택하면 전세계, **North America** 지역 및 **Canada**에 대한 개별 데이터를 얻게 됩니다.

**Country**에서 **Canada**를 선택하고 **North America**는 선택하지 않으면 **North America**에 대한 집계 데이터가 로드되지 않습니다. **Canada**에 대한 데이터만 **North America** 지역에 대해 로드됩니다.

#### 데이터가 없는 부모 분기 선택

일부 데이터 셋에서 계층 구조의 부모 분기에는 데이터가 포함되지 않습니다. 데이터가 없는 부모 분기는 **US social characteristics (by state)** 데이터 셋에서와 마찬가지로 확인란에 꺾인 선이 표시됩니다.

#### 로드할 데이터 선택

The screenshot displays the 'US social characteristics (by state)' data selection interface. On the left, there are three sections: 'Dimension' with 'Geography' selected (1/53), 'Time' with 'Year' selected (Most recent), and 'Measure' with 'Subject' selected (0/164). The main area shows a search bar and a list of subjects with checkboxes. The subjects listed are: 'Ancestry', 'Disability status of the civilian noninstitutionalized population', 'Educational attainment' (which is expanded to show sub-items: 'Population 25 years and over', 'Population 25 years and over - 9th to 12th grade, no diploma', 'Population 25 years and over - Associate's degree', 'Population 25 years and over - Bachelor's degree', 'Population 25 years and over - Graduate or professional degree', 'Population 25 years and over - High school graduate (includes ...)', 'Population 25 years and over - Less than 9th grade', and 'Population 25 years and over - Some college, no degree'), and 'Fertility'.

부모 필드가 선택되면 해당 분기의 모든 자식도 자동으로 선택됩니다. 분기에서 모든 자식이 아닌 일부 자식을 선택하려면 자동 부모 선택에서 개별 필드의 선택을 취소하거나 부모 필드를 선택하지 않고 개별 필드를 선택할 수 있습니다.

#### Qlik DataMarket에서 집계 데이터 선택

일부 Qlik DataMarket 데이터 셋에는 집계 필드를 포함하는 차원이 포함됩니다. 예를 들어, 다음 **World population by country** 데이터 셋에는 집계 총 수 및 남녀별 총합이 있는 **Sex** 차원이 포함되어 있습니다. 전체 차원을 선택(맨 위의 확인란)하면 차원의 모든 필드가 선택됩니다. 그러나 더 낮은 수준에서 개별 필드를 선택할 수 있습니다.

#### 로드할 데이터 선택

**World population by country**

**Dimension**

✓ Country <span style="float: right;">228/228</span>
! Sex <span style="float: right;">0/3</span>
<i>_dim_8q8f, Sex aggregate, Sex</i>
✓ Age group <span style="float: right;">22/22</span>

**Time**

✓ Year <span style="float: right;">Most recent</span>
---

Sex

---

🔍

▼ Total

---

Female

---

Male

**Sex aggregate**만 선택된 경우, 모든 남성 및 여성의 총 수가 데이터에 포함됩니다. 그러나 남성 및 여성에 대한 총합은 포함되지 않습니다. 따라서 남성 및 여성에 대한 개별 데이터를 사용하여 시각화를 만들 수는 없습니다.

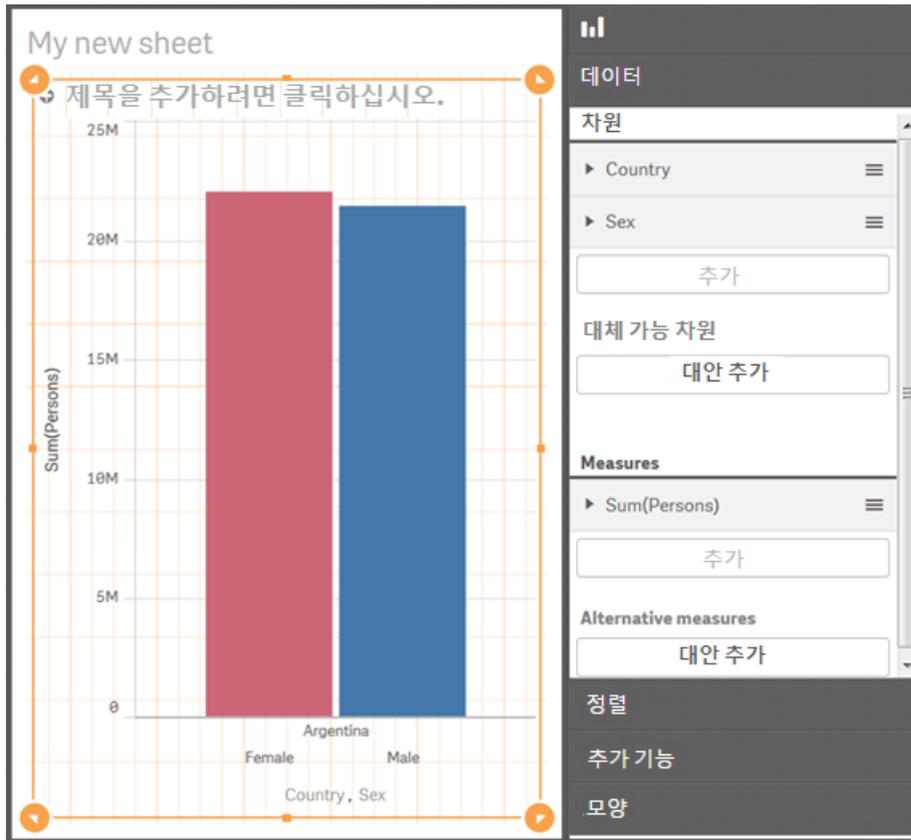


집계 데이터가 있는 여러 필드가 포함된 **World population by country** 등의 데이터 셋으로 작업하는 경우, 해당 집계 필드가 있는 테이블이 직접 연결되어 있지 않은지 확인하십시오. 직접 연결되어 있으면 순환 참조가 생성될 가능성이 있습니다.

남성 및 여성으로 나뉜 데이터를 포함하려면 **Sex**를 선택해야 합니다. 그러면 남성 및 여성 인구 총합을 개별적으로 표시하는 시각화를 만들 수 있습니다. 한 성별에 대한 총합에만 관심이 있는 경우에는 **Male** 또는 **Female**만 선택할 수 있습니다.

명확하게 선택하지 않은 경우에도 집계 필드는 포함됩니다. 또한 집계 필드의 값은 선택한 개별 필드의 값과 동일합니다. **Sex** 데이터의 경우, **Sex aggregate**는 **Female** 필드가 선택된 유일한 **Sex** 필드일 때 여성의 총 수와 동일합니다.

**Sex** 차원을 사용하여 시각화를 작성하는 경우, 데이터는 남성 및 여성의 합계를 개별적으로 표시합니다. 예를 들어, 아르헨티나의 남성 및 여성 인구의 총 수는 막대형 차트에서 두 개의 개별 막대로 표시됩니다.



반대로, 막대형 차트에서 **Sex** 차원 대신 **Sex aggregate** 차원을 사용하는 경우에는 남성 및 여성이 결합된 총 수에 대한 단일 막대를 얻게 됩니다.

**Sex aggregate** 데이터만 선택하고 **Sex**는 선택하지 않은 경우, 또는 두 성별을 개별적으로 선택한 경우에는 시각화에서 성별에 따른 데이터를 표시할 수 없습니다. 위의 막대형 차트에는 위 화면에서처럼 **Female** 및 **Male**에 대한 개별 막대 대신 남성 및 여성의 총 수에 대한 단일 막대가 있습니다.

#### Qlik DataMarket 검색

Qlik DataMarket의 검색 기능은 데이터 패키지 또는 데이터 셋을 검색하지 않고 용어나 구를 검색하여 필요한 데이터를 찾도록 도와줍니다. 입력한 용어나 구는 DataMarket 데이터 셋에 대한 이름 및 설명과 데이터 셋 내의 값에서 검색됩니다.

예를 들어 *europa*이라는 용어를 검색하면 제목에 *Europe*이라는 용어가 있는 모든 데이터 셋 목록을 가져온 다음 해당 용어로 레이블이 지정된 데이터가 포함된 데이터 셋을 가져옵니다. *europa*의 경우 발견된 데이터 셋 중 하나는 해당 **Geographical area** 차원(*Europe & Central Asia*)에 이 용어가 있는 **Selected development indicators**입니다.

DataMarket에서는 입력한 리터럴 용어나 구를 검색하며, 관련 용어나 동의어도 검색합니다. 단수 형식으로 입력된 용어에 대해 해당 복수 형식도 검색됩니다. 예를 들어 용어 *currency* 및 *index*는 단수 형식과 동시에 해당 복수 형식(*currencies* 및 *indices*)도 검색됩니다.

또한 검색 기능은 용어의 어간 또는 어근을 기반으로 일치 항목을 찾습니다. 예를 들어 *production*이라는 용어를 검색하는 경우 이 단어의 어근인 *product*도 검색됩니다.



*DataMarket*에서는 부분 용어를 검색하지 않습니다. 예를 들어 "prod"라는 문자열은 검색 기능이 *Gross Domestic Product*와 같은 구에서 찾는 용어인 *product* 및 *production*의 일부인 경우에도 검색되지 않습니다.

또한 Qlik *DataMarket*에는 동의어 인덱스가 포함되어 있으므로 데이터 컬렉션 또는 데이터 필드의 이름이나 설명에 사용된 정확한 용어를 사용하지 않고도 광범위한 데이터를 찾을 수 있습니다. 예를 들어 **Sex**라는 레이블이 지정된 차원을 사용하는 데이터 셋은 *gender*라는 용어로도 검색됩니다.

*DataMarket* 검색 기능에는 200개가 넘는 동의어 집합이 있습니다. 포함된 동의어의 예를 들면 다음과 같습니다.

- earnings, income
- GBP, pound
- health care, healthcare
- labor, labour
- salary, wages, pay, earnings

검색 결과는 관련성이 가장 높은 순에서 낮은 순으로 표시됩니다. 관련성은 검색 용어를 찾은 위치에 따라 결정됩니다. 데이터 셋 이름 또는 설명에서 찾은 용어는 데이터 셋 값에서 찾은 용어보다 순위가 더 높습니다.

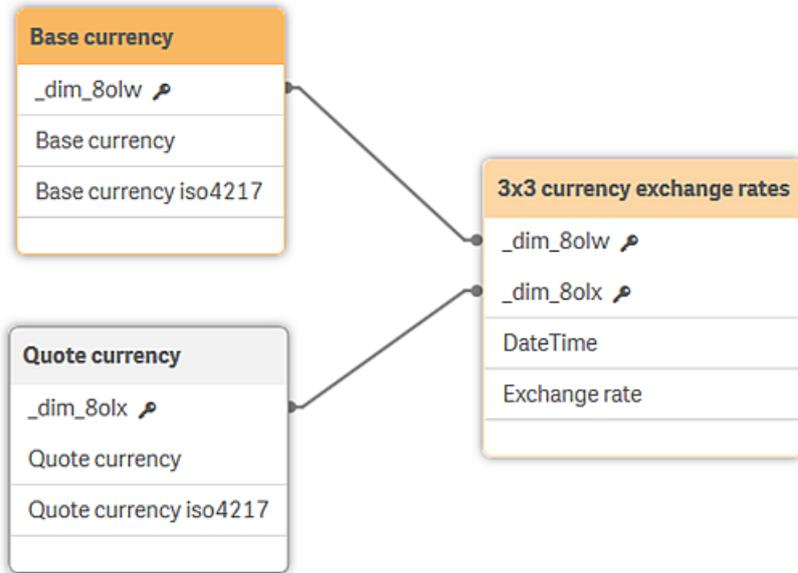
여러 검색 용어를 입력한 경우 결과에 반드시 모든 용어가 포함되지는 않습니다. 용어 중 한 개만 찾은 경우 해당 용어를 포함하는 항목은 검색 결과 중 하나로 반환됩니다. 하지만 입력한 검색 용어를 더 많이 포함하는 항목이 순위가 더 높습니다.

검색에 대해 상위 50개의 결과만 나열됩니다.

검색을 좁히려면 검색 문자열에 용어를 입력할 때 용어 앞에 하이픈을 넣어 검색에서 용어를 제외할 수 있습니다. 예를 들어 검색 용어 앞에 하이픈을 넣어("-unemployment") "US"는 검색하고 unemployment는 제외할 수 있습니다.

#### Qlik *DataMarket*에서 연결 만들기

Qlik *DataMarket* 데이터 셋에서 데이터를 로드하면 해당 데이터가 여러 개의 개별 테이블에 할당되어 있습니다. 이들 테이블은 생성된 키 필드에 의해 연결됩니다. 데이터 셋의 측정값과 기간은 데이터 셋 이름이 할당된 하나의 테이블로 통합됩니다. 차원 필드는 개별 테이블에 할당됩니다. 예를 들어 **3x3 currency exchange rates** 데이터 셋은 세 개의 테이블로 로드됩니다.



일부 차원은 로드할 때 추가적인 필드를 제공합니다. 추가 필드는 차원에 대한 추가적인 표현을 제공합니다. **3x3 currency exchange rates** 데이터 셋에서 통화는 iso4217 표현에 의해서도 나열됩니다. 예:

기준 통화	Euro
기준 통화 iso4217	EUR
가격제시 통화	미국 달러
가격제시 통화 iso4217	USD

국가 및 지역별 인구 데이터가 포함된 데이터 셋은 ISO 3166 코드 등의 지역 이름에 대한 추가적인 차원 표현을 제공합니다.

#### 데이터 셋의 여러 테이블 연결

단일 데이터 셋의 모든 테이블을 측정값 테이블을 통해 연결해야 합니다. 데이터 추가에서 데이터 셋을 로드하면 생성된 키 필드로부터 연결이 자동으로 만들어집니다. 다음 데이터 모델에는 **US population by state, race, sex and age** 데이터 셋이 나와 있습니다. 차원 테이블 (**Age, Sex, Location** 및 **Race**)은 데이터 셋 이름을 가진 측정값 테이블을 통해 연결됩니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리



이러한 연결은 시각화에서 중요한 차원 및 측정값 간의 관계를 해석하는 데 필요합니다. 예를 들어 회사에서 US 인구 데이터를 사용하여 제품 매출과 여러 US 주의 연령대를 비교하려는 경우, 여러 주의 각 그룹 내 사람 수를 구하려면 **Age** 및 **Location** 차원이 측정값 테이블을 통해 연결되어야 합니다.

데이터 셋에 여러 개의 차원 테이블이 있으면 종종 추가적인 연결을 만들 수 있습니다. 예를 들어 집계 필드에는 일반적으로 사용 가능한 연결을 제안하는 동일한 값("Total")이 있습니다. 하지만 그러한 연결이 유용하지는 않는데, 순환 참조를 생성할 수 있습니다.

다중 테이블 구조를 사용하면 데이터를 로드할 때 효율성이 증가하고 데이터 연결을 개선할 수 있습니다.

#### 단일 테이블 데이터 셋을 여러 테이블로 변환

Qlik Sense 3.0 전에는 **DataMarket** 데이터 셋이 단일 테이블로 로드되었습니다. **Qlik Sense 3.0**에서는 이전에 로드된 테이블이 단일 테이블로 남아 있으며 해당 테이블을 사용하는 모든 시각화도 전과 마찬가지로 작동합니다. 그러한 단일 테이블을 여러 테이블로 변환할 수 있으며, 시각화는 수정하지 않고도 계속 작동됩니다. 변환하기 전에 여러 테이블일 때의 효율성에 대해 알아 보십시오.

다음과 같이 하십시오.

1. **데이터 관리자**를 엽니다.
2. 여러 테이블로 전환하려는 **DataMarket** 데이터 셋이 포함된 테이블을 삭제합니다. 해당 테이블이 **삭제 보류**로 표시됩니다.
3. **데이터 추가**를 선택합니다.
4. **Qlik DataMarket**을 선택합니다.
5. 삭제한 테이블에 해당하는 데이터 셋을 선택합니다.  
**로드할 데이터 선택** 페이지가 열리면 이전에 선택한 내용이 표시됩니다. 해당 데이터 셋을 사용하는 시각화도 변경할 예정이 아니라면 선택 내용을 변경하지 마십시오. 선택을 변경하면 해당 데이터 셋을 사용하는 기존 시각화가 더 이상 작동하지 않을 수 있습니다.
6. **데이터 로드 및 마침**을 선택합니다.  
데이터 셋이 여러 테이블에 로드되고 단일 테이블이 삭제됩니다.

여러 테이블로 변환한 데이터 셋을 사용하는 시각화를 점검합니다. 이전에 테이블을 로드했을 때 수행한 선택 중 일부를 추가 또는 제거하여 선택한 데이터를 변경하지 않는 한, 원래 설계된 대로 작동해야 합니다.

### 회사 데이터와 DataMarket 데이터 통합

**Qlik DataMarket** 데이터는 회사 데이터와 통합하여 해당 데이터에 대한 더 폭넓은 컨텍스트를 제공하고 시장에 대한 보다 뛰어난 통찰력을 제공할 수 있습니다.

**DataMarket** 데이터는 다양한 소스에서 가져올 수 있으며, 그로 인해 사용자의 데이터와의 연결성이 즉시 명확해지지 않을 수 있습니다. 따라서 데이터 준비 단계에서 여러 연결을 편집해야 한다는 것을 알 수 있습니다. 예를 들어, 사업을 운영하는 국가의 특정한 특성을 평가하는 것이 중요하다는 사실을 발견할 수 있습니다. 그러나 일부 **DataMarket** 데이터 셋에 있는 국가의 경우 필드에는 사용자의 회사 데이터와 공통되는 값이 충분하지 않으므로 유용한 연결을 만들 수 없습니다. 그러므로 사용자의 데이터와 **DataMarket** 데이터 간의 연결을 조심스럽게 평가해야 합니다.

아래 그림에는 회사 데이터와 **DataMarket** 데이터를 통합하여 의미 있는 **Qlik Sense** 시각화를 만드는 방법이 나와 있습니다.

### 매출과 국내 총생산(GDP) 비교

어떤 국가에서 비즈니스를 운영하고 있거나 시장에 진입하려고 고려하는 중인 경우, 회사의 성과가 해당 국가의 인구 통계 및 경제적 환경에서 예상되는 수준에 맞는지 판단하도록 도와줄 수 있는 여러 가지 지표가 있습니다.

이 예에서는 회사 데이터를 통해 매출 데이터를 국가별로 집계할 수 있습니다. 막대형 차트는 국가별 매출을 비교합니다. 이를 통해 사용자의 회사가 모든 시장에서 어떻게 비즈니스를 해야 할지에 대한 통찰력을 제공할 수 있습니다.

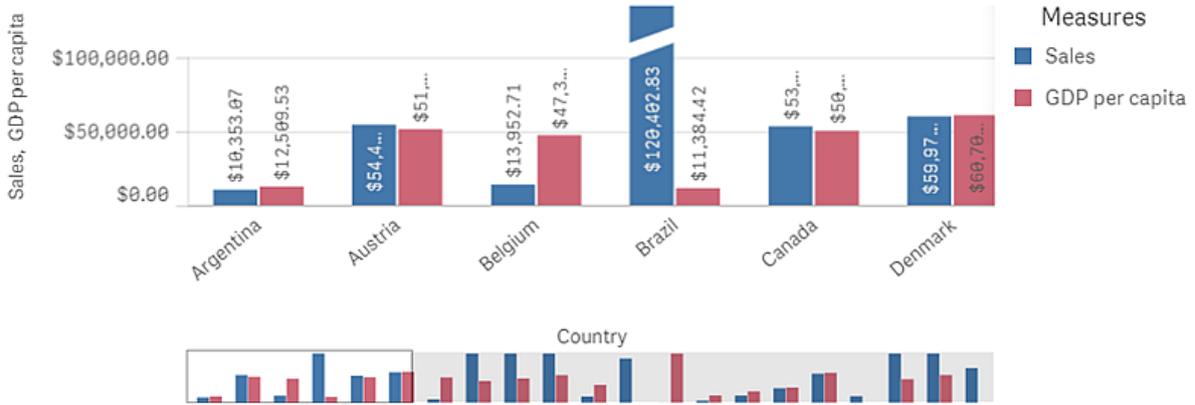
각 국가에서 어떻게 비즈니스를 해야 할지 알아보려면 회사의 매출을 해당 시장이 얼마나 탄탄한지 나타내는 국가 데이터와 비교할 수 있습니다. 예를 들어 해당 국가의 국내 총생산(GDP)과 각 국가의 매출을 비교할 수 있습니다. 또는 매출을 대상 시장의 인구 통계와 비교할 수 있습니다. 회사의 목표 인구 연령대가 21~35세이면 해당 국가에서 얼마나 많은 사람이 이 연령 범주에 속하는지 확인할 수 있습니다. 또는 총 인구의 몇 %가 해당 연령 범주에 속하는지 알아볼 수 있습니다.

**Qlik DataMarket**에는 *Select development indicators*라는 Essentials Free 그룹의 데이터 셋이 포함되며, GDP 성장률, 문맹률, 인터넷 사용자, 총 인구 및 GDP per capita in US dollars를 포함한 여러 가지 경제 측정값을 제공합니다. *Select development indicators*에서 국가 데이터를 연결하기 위해서는 회사 데이터에 **DataMarket** 데이터 셋의 Country 필드와 일치하는 필드가 있어야 합니다. 회사 데이터에 *Select development indicators*보다 많은 국가가 있으면 강력하게 연결되지 않아 유용하지 않을 수 있습니다. 회사 데이터에 *Select development indicators*보다 적은 국가가 있으면 연결은 비교에 유용할 수 있습니다.

회사의 국가 필드와 **DataMarket** 데이터 간에 양호한 연결이 존재한다고 가정하면 GDP per capita in US dollars를 매출 막대형 차트에 추가하여 각 국가에서의 매출을 GDP와 비교할 수 있습니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

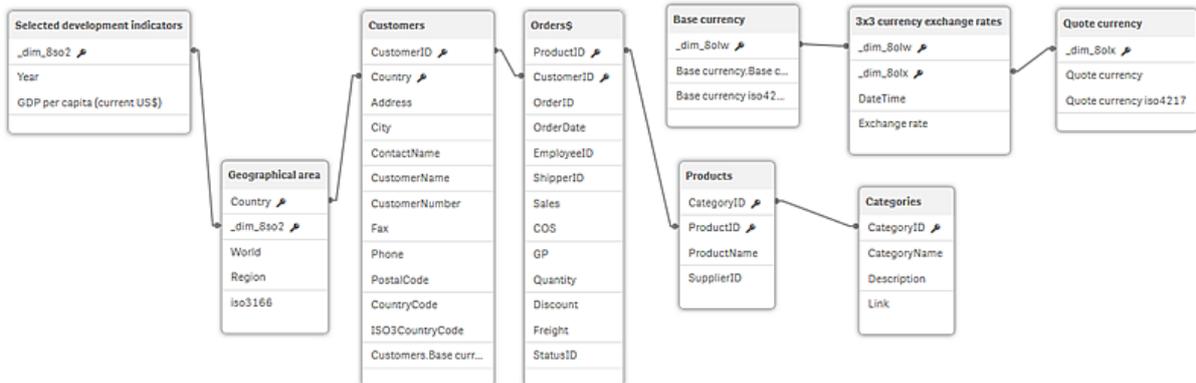
Sales per country compared to per capita GDP



#### 유로로 매출 표시

매출 데이터를 검토하는 경우 일반적으로 사용되는 시각화는 총 매출을 표시하는 KPI 차트입니다. 여기에 나온 예에서는 매출 총액이 US 달러로 표시되는데, 회사 데이터를 기록하는 방법이 US 달러로 지정되어 있기 때문입니다. 하지만 유로를 사용하는 국가의 매출이 높다면 유로로 매출 총액을 표시하는 것이 좋습니다. 이를 KPI에 추가할 수 있습니다. Essentials Free 그룹에서 DataMarket에는 US 달러를 유로로 변환하는 데 사용할 수 있는 *3x3 currency exchange rates*가 포함됩니다.

*3x3 currency exchange rates* 데이터 셋에서 Base currency 필드를 선택하는 경우에는 매출이 기록된 통화인 US 달러만 선택할 수 있습니다. 이 예에서 사용되는 회사 데이터 셋에는 각 고객이 사용하는 통화를 나타내는 Base currency라는 필드가 있습니다. 그러나 여기에는 12개의 서로 다른 통화가 포함되어 있으므로 데이터 관리자에서는 두 필드를 연결할 것을 권장합니다. US 달러 및 유로에 해당하지 않는 회사 데이터의 통화는 달러와 유로의 비교에 방해가 될 수 있으므로 해당 필드에는 No association을 선택해야 합니다. 그러면 데이터 모델이 다음과 같이 나타납니다.



*3x3 currency exchange rates*의 Quote currency는 Euro여야 합니다. KPI 시각화에 이전 데이터가 아닌 현재 환율만 필요하므로 DateTime 선택은 Most recent가 되어야 합니다.

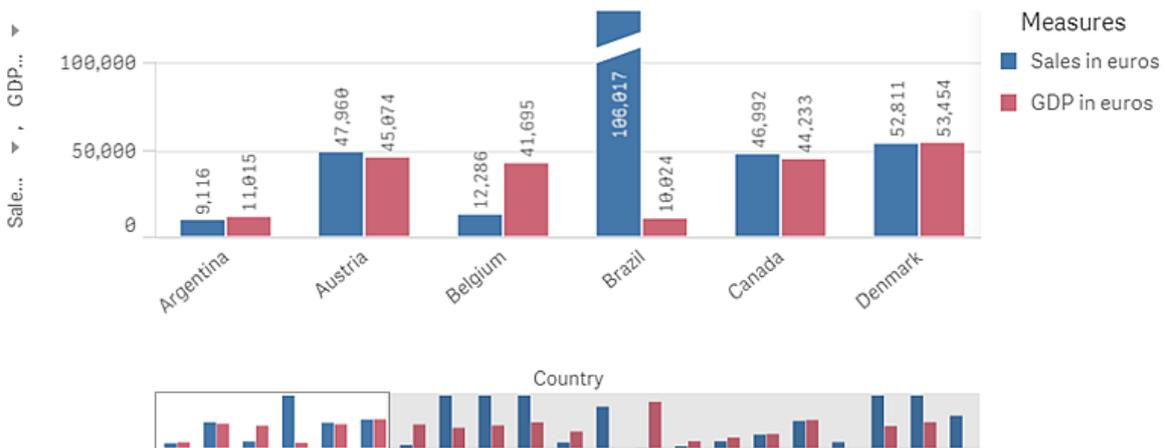


Euro Sales를 구하려면 Sum(Sales)를 *3x3 currency exchange rates* 데이터 셋의 Exchange rate로 곱하면 됩니다.

매출에 대한 대체 가능 측정값 만들기

DataMarket 통화 데이터를 사용하여 매출 막대형 차트를 더욱 개선할 수 있습니다. 이 차트는 매출을 각 국가의 1인당 GDP와 US 달러로 비교하지만, 대체 가능 측정값을 매출 및 GDP에 추가할 수 있습니다. Sum(Sales)는 *3x3 currency exchange rates* 테이블의 Exchange rate로 곱하여 Sales KPI에서처럼 매출을 유로로 표시할 수 있습니다. US 달러로 표시된 1인당 GDP 역시 Exchange rate로 곱할 수 있으며, 그런 다음 막대형 차트에서 매출을 1인당 GDP와 유로로 비교합니다.

Sales per country compared to per capita GDP



Base currency 필드는 연결되지 않아야 하는데, 위에서 언급한 바와 같이 회사 데이터 셋의 Base currency 필드에 12개의 서로 다른 통화가 포함되어 있기 때문입니다. 각 국가별로 환율 계산을 수행하는 경우, 테이블이 연결되어 있으면 각 국가의 기준 통화가 사용됩니다. 그러나 회사 데이터에서는 이들 12개 통화 중 대부분이 매출 값에 전혀 사용되지 않았습니다. US 달러 단위의 매출 값만 포함되어 있습니다. 또한 DataMarket 데이터 셋의 Base currency는 오로지 US 달러이므로 회사 데이터에서 US 달러 외의 Base currency 값을 갖는 국가의 경우 두 테이블이 연결되어 있으면 Sales in euros가 Null 값이 됩니다.

### 3.11 테이블 편집

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

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

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.



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

#### 테이블 이름 바꾸기

**데이터 관리자**에서 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. **Enter**를 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 Sense**에서는 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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

- 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.

- **General**
- **G Date**
- **Ö Timestamp**
- **, Geo data**

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.

#### 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

참조: [데이터 관리자에서 크로스탭 데이터 피벗 해제\(57페이지\)](#)

#### 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.

다음과 같이 하십시오.

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

1. **소스에서 데이터 선택**을 클릭합니다.  
현재 선택이 포함된 데이터 선택 마법사가 열립니다.
2. 선택 사항에서 필요한 변경을 수행합니다.
3. **데이터 준비**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

**B** 및 **C** 를 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.

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

- 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 연결할 테이블 중 하나의 **@** 를 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. **P** 를 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. **F** 가 있는 테이블에서 검색하면 더 쉽게 비교할 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 **로** 표시됩니다. 연결을 편집하거나 해제하는 옵션을 표시하려면 **를** 클릭합니다.

### 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 연결할 테이블 중 하나의 **@** 를 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. **P** 를 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. **P**를 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. **F**가 있는 테이블에서 검색하면 더 쉽게 비교할 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. **연결**을 클릭하여 연결 메뉴를 표시합니다.
2. **@**를 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. **연결**을 클릭하여 연결 메뉴를 표시합니다.
2. **연결 해제**를 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.

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

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



둘 이상의 테이블을 연결한 결과로 생성된 테이블에는 계산 필드를 추가할 수 없습니다.

### 계산 필드 추가

다음과 같이 하십시오.

1. 필드 추가를 클릭하고 **계산 필드**를 선택합니다.  
계산 필드 추가를 위한 편집기가 열립니다.
2. 이름에 계산 필드의 이름을 입력합니다.
3. 표현식에서 계산 필드의 표현식을 정의합니다. 이 작업은 두 가지 방법으로 수행할 수 있습니다.
  - **3 (함수)**, **- (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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

함수	설명
Capitalize	<b>Capitalize()</b> 는 모든 단어의 첫 글자가 대문자인 문자열을 반환합니다.
FindOneOf	<b>FindOneOf()</b> 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 <b>0</b> 이 반환됩니다.
KeepChar	<b>KeepChar()</b> 은 첫 번째 문자열 'text'로 이루어진 문자열 중 두 번째 문자열 "keep_chars"에 포함되지 않는 문자를 뺀 문자열을 반환합니다.
Left	<b>Left()</b> 는 입력 문자열의 첫 번째(가장 왼쪽) 문자로 구성된 문자열을 반환하며 문자 수는 두 번째 인수로 결정됩니다.
Len	<b>Len()</b> 은 입력 문자열의 길이를 반환합니다.
Lower	<b>Lower()</b> 는 입력 문자열의 모든 문자를 소문자로 변환합니다.
LTrim	<b>LTrim()</b> 은 모든 선행 공백이 제거된 입력 문자열을 반환합니다.
Mid	<b>Mid()</b> 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.
PurgeChar	<b>PurgeChar()</b> 은 두 번째 인수('remove_chars')에 나타나는 문자를 제외하고, 입력 문자열('text')에 포함된 문자로 구성된 문자열을 반환합니다.
Replace	<b>Replace()</b> 는 입력 문자열 내에서 특정 부분 문자열의 모든 발견 항목을 다른 부분 문자열로 대체한 후의 문자열을 반환합니다. 이 함수는 비재귀적이며 왼쪽에서 오른쪽으로 작동합니다.
Right	<b>Right()</b> 는 입력 문자열의 마지막(가장 오른쪽) 문자로 구성된 문자열을 반환합니다. 문자 수는 두 번째 인수로 결정됩니다.
RTrim	<b>RTrim()</b> 은 모든 후행 공백이 제거된 입력 문자열을 반환합니다.
TextBetween	<b>TextBetween()</b> 은 구분 기호로 지정된 문자 사이에서 발견되는 입력 문자열의 텍스트를 반환합니다.
Trim	<b>Trim()</b> 은 모든 선행 및 후행 공백이 제거된 입력 문자열을 반환합니다.
Upper	<b>Upper()</b> 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 Sense**는 인수의 숫자 값을 사용하므로, 숫자는 날짜 또는 시간으로 서식이 지정되지 않았을 때도 인수로 유효합니다. 인수가 문자열인 경우와 같이 숫자 값에 해당하지 않는 경우 **Qlik Sense**는 날짜 및 시간 환경 변수에 따라 문자열을 해석하려고 시도합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

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.

함수	설명
<b>addmonths</b>	이 함수는 <b>startdate</b> 를 기준으로 <b>n</b> 개월 후의 날짜, 또는 <b>n</b> 이 음수일 경우는 <b>startdate</b> 를 기준으로 <b>n</b> 개월 전의 날짜를 반환합니다.
<b>addyears</b>	이 함수는 <b>startdate</b> 를 기준으로 <b>n</b> 년 후의 날짜, 또는 <b>n</b> 이 음수일 경우는 <b>startdate</b> 를 기준으로 <b>n</b> 년 전의 날짜를 반환합니다.
<b>age</b>	<b>age</b> 함수는 <b>date_of_birth</b> 에 태어난 사람이 <b>timestamp</b> 의 시간에 (만으로) 몇 살인지 반환합니다.
<b>day</b>	이 함수는 <b>expression</b> 의 분위수가 표준 숫자 해석에 따라 날짜로 해석될 경우 일을 나타내는 정수를 반환합니다.
<b>makedate</b>	이 함수는 연도 <b>YYYY</b> , 월 <b>MM</b> , 일 <b>DD</b> 에서 계산한 날짜를 반환합니다.
<b>maketime</b>	이 함수는 시간 <b>hh</b> , 분 <b>mm</b> , 초 <b>ss</b> 에서 계산한 시간을 반환합니다.
<b>month</b>	이 함수는 환경 변수 <b>MonthNames</b> 로 정의되고 1~12 사이의 정수인 월 이름이 포함된 이중 값을 반환합니다. 월은 표준 숫자 해석에 따라 표현식의 날짜 해석을 통해 계산됩니다.
<b>monthend</b>	이 함수는 <b>date</b> 를 포함하는 월의 마지막 날의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>monthname</b>	이 함수는 해당 월의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 월 ( <b>MonthNames</b> 스크립트 변수에 따라 서식 지정) 및 연도를 보여주는 표시 값을 반환합니다.
<b>monthstart</b>	이 함수는 <b>date</b> 를 포함하는 월의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>quarterend</b>	이 함수는 <b>date</b> 를 포함하는 분기의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>quartername</b>	이 함수는 해당 분기의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 분기의 월 ( <b>MonthNames</b> 스크립트 변수에 따라 서식 지정) 및 연도를 보여주는 표시 값을 반환합니다.
<b>quarterstart</b>	이 함수는 <b>date</b> 를 포함하는 분기의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>week</b>	이 함수는 ISO 8601에 따른 주 번호를 나타내는 정수를 반환합니다. 주 번호는 표준 숫자 해석에 따라 표현식의 날짜 해석을 통해 계산됩니다.
<b>weekend</b>	이 함수는 <b>date</b> 를 포함한 주의 마지막 날짜(일요일)의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>weekname</b>	이 함수는 <b>date</b> 를 포함한 주의 첫 번째 날의 첫 번째 밀리초의 타임스탬프에 해당하는 기본 숫자 값으로 연도 및 주차를 보여주는 값을 반환합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

함수	설명
<b>weekstart</b>	이 함수는 <b>date</b> 를 포함하는 주의 첫 번째 날(월요일)의 첫 번째 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>year</b>	이 함수는 <b>expression</b> 이 표준 숫자 해석에 따라 날짜로 해석될 경우 연도를 나타내는 정수를 반환합니다.
<b>yearend</b>	이 함수는 <b>date</b> 를 포함하는 연도의 마지막 날의 마지막 밀리초의 타임스탬프에 해당하는 값을 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.
<b>yearstart</b>	이 함수는 <b>date</b> 를 포함하는 연도의 첫 번째 날의 시작에 해당하는 타임스탬프를 반환합니다. 기본 출력 형식은 스크립트에 설정된 <b>DateFormat</b> 입니다.

#### 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.

함수	출력(텍스트)	출력(숫자)
<code>Date#('20140831', 'YYYYMMDD')</code>	20140831	41882
<code>Date(41882, 'YYYY.MM.DD')</code>	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.

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

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

함수	설명
<b>Interval#</b>	<b>Interval#()</b> 은 운영 체제에서 기본 설정된 서식 또는 두 번째 인수(지정된 경우)로 지정된 서식을 사용하여 텍스트 표현식을 시간 간격으로 평가합니다.
<b>Money</b>	<b>Money()</b> 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.
<b>Money#</b>	<b>Money#()</b> 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.
<b>Num</b>	<b>Num()</b> 은 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)의 시스템 변수에 설정된 숫자 서식 및 선택적 소수점 구분 기호와 천 단위 구분 기호를 사용하여 표현식을 숫자로 서식 지정합니다.
<b>Num#</b>	<b>Num#()</b> 은 데이터 로드 스크립트 또는 운영 체제에 설정된 숫자 서식을 사용하여 텍스트 문자열을 숫자 값으로 변환합니다. 사용자 지정 소수점 구분 기호 및 천 단위 구분 기호는 옵션 파라메타입니다.
<b>Text</b>	<b>Text()</b> 는 표현식을 숫자로 해석 가능한 경우에도 텍스트로 취급하도록 지정합니다.
<b>Time</b>	<b>Time()</b> 은 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)의 시스템 변수에 설정된 시간 서식을 사용하여 표현식을 시간 값으로 서식 지정합니다.
<b>Time#</b>	<b>Time#()</b> 은 데이터 로드 스크립트 또는 운영 체제(서식 문자열을 지정하지 않은 경우)에 설정된 시간 서식을 사용하여 표현식을 시간 값으로 평가합니다.
<b>Timestamp</b>	<b>TimeStamp()</b> 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.
<b>Timestamp#</b>	<b>TimeStamp#()</b> 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.

함수	설명
<b>floor</b>	<b>Floor()</b> 는 <b>offset</b> 숫자에 의해 시프트된 <b>step</b> 의 가장 가까운 배수로 숫자를 내림합니다.
<b>round</b>	<b>Round()</b> 는 <b>offset</b> 숫자에 의해 시프트된 <b>step</b> 의 가장 가까운 배수로 숫자를 올림 또는 내림한 결과를 반환합니다.

계산 필드 식에 사용할 수 있는 조건부 함수

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

함수	설명
<b>if</b>	<b>if</b> 함수는 함수에 제공된 조건이 <b>True</b> 로 평가되는지, <b>False</b> 로 평가되는지에 따른 값을 반환합니다.
<b>match</b>	<b>match</b> 함수는 첫 번째 파라메타를 이후의 모든 파라메타와 비교하여 일치하는 표현식 수를 반환합니다. 대소문자가 구분됩니다.
<b>wildmatch</b>	<b>wildmatch</b> 함수는 첫 번째 파라메타를 이후의 모든 파라메타와 비교하여 일치하는 표현식 수를 반환합니다. 비교 문자열에는 와일드카드 문자( <b>*</b> 및 <b>?</b> )를 사용할 수 있습니다. 대소문자는 구분되지 않습니다.

## 필드 유형 변경

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

- **General**
- **G Date**
- **Ö Timestamp**
- **, Geo data**

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

테이블 편집기를 열려면 편집할 데이터 테이블에서 @ 를 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 있으면 올바르게 해석되었는지 확인할 수 있습니다.

다음과 같이 하십시오.

1. 필드 머리글 위의  를 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.  
참조: [숫자 및 시간 서식에 대한 규칙\(114페이지\)](#)



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 위에서 **G**(날짜) 또는 **õ**(타임스탬프)를 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 위의 **지리 데이터**를 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 **SWE**입니다.  
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 **France**와 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 사용자에게 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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

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.

#### 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.

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.

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 Year에 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. 이 필드는 적절하게 Month로 이름을 지정할 수 있습니다.
- 데이터 필드로, 파란색으로 표시되어 있습니다. 이 경우, 판매 데이터를 나타내므로 적절히 Sales로 이름을 지정할 수 있습니다.

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

Year	Month	Sales
2008	Jan	45
2008	Feb	65
2008	Mar	78
2008	Apr	12
2008	May	78
2008	Jun	22
2009	Jan	11
2009	Feb	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 두 개의 필드 **Tablename.Attribute** 및 **Tablename.Data**가 있는 행으로 바뀝니다.
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.

### 3.12 데이터 연결 관리

연결 문제가 발생할 수 있는 데이터를 로드해야 하는 경우 **데이터 추가**의 **데이터 준비** 옵션을 사용하는 것이 바람직합니다.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

Qlik Sense에서는 로드할 데이터를 프로파일링하여 테이블 연결을 정정하도록 지원합니다. 기존의 올바른 연결과 잘못된 연결 및 잠재적으로 올바른 연결이 강조 표시되므로 데이터 분석 결과에 기반하여 연결에 사용할 필드를 선택하는 데 도움이 됩니다.

데이터 관리자의 **연결** 보기에서 데이터는 거품으로 묘사되며, 각 거품은 데이터 테이블을 나타내고 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.



대부분의 경우, 모델 보기에서 테이블 연결을 편집하는 것이 훨씬 쉽지만 테이블 편집기에서 **연결** 옵션을 사용하여 단일 테이블 연결을 편집할 수도 있습니다.

참조: 테이블 편집기에서 데이터 연결 (47페이지)

### 권장 사항에 기반하여 모든 테이블 연결

많은 경우, Qlik Sense에서 데이터 분석에 기반한 권장 사항에 따라 추가된 모든 데이터 테이블 간의 연결을 자동으로 만들도록 할 수 있습니다.

다음과 같이 하십시오.

- 연결 보기의 오른쪽 위 모서리에서 **N** 를 클릭합니다.

이제 권장 사항에 따라 테이블이 연결됩니다.

### 두 테이블 연결

어떤 테이블이든 두 개를 연결할 수 있지만 Qlik Sense에서는 성공할 가능성이 가장 높은 연결을 권장합니다.

---

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

권장 사항은 테이블에서 마우스를 클릭하고 눌렀을 때 강조 표시됩니다. 강력하게 권장되는 연결은 녹색으로 표시되고 보통 권장 사항은 주황색으로 표시됩니다.

다음과 같이 하십시오.

- 연결되지 않은 테이블을 녹색 또는 주황색으로 표시된 테이블 중 하나로 끕니다.

이제 테이블이 권장 필드를 사용하여 연결됩니다.

### 테이블을 권장되지 않는 테이블과 연결하고 싶은 경우

권장되지 않는 연결에 테이블을 유지하면 해당 테이블이 빨간색으로 표시됩니다. 사용자 지정 연결을 만들어 계속 테이블을 연결할 수 있습니다.

1. 테이블을 빨간색으로 표시된 테이블로 끕니다.  
테이블 연결 편집기가 열립니다.
2. 왼쪽 테이블에서 연결에 사용할 필드를 선택합니다. 단일 필드 또는 복수의 필드를 사용할 수 있습니다. 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 이름에 입력합니다.  
두 테이블의 기존 필드와 동일한 이름을 사용할 수는 없습니다.
5. 연결을 클릭합니다.

이제 테이블이 사용자 지정 연결을 사용하여 연결됩니다.

## 연결 해제

데이터 모델의 기존 연결이 올바르지 않은 경우 해당 연결을 해제할 수 있습니다.

다음과 같이 하십시오.

- 연결 테이블 중 하나(연결 수가 가장 적고 가장 중요도가 떨어지는 테이블)를 선택하고 연결이 해제될 때까지 다른 테이블에서 멀리 끕니다.

이제 두 테이블이 더 이상 연결되어 있지 않습니다.

## 연결 편집

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

다음과 같이 하십시오.

1. 테이블 쌍 사이의 연결을 선택합니다.  
연결된 필드의 데이터에 대한 미리 보기와 함께 연결 패널이 열립니다.
2. @ 을 클릭합니다.  
하나 이상의 연결 버튼이 표시되며 각각은 데이터 분석 결과에 따라 권장 상태를 색을 나타냅니다. 현재 연결은 회색으로 표시됩니다.
3. 연결 버튼 중 하나를 클릭합니다.

## 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

- 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.  
각 테이블의 단일 필드를 사용하거나 각 테이블에서 둘 이상의 필드를 사용하여 사용자 지정 연결을 정의할 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 하단에서 **j**를 클릭합니다.

테이블의 데이터와 함께 미리 보기 창이 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.

참조: [가상 키\(95페이지\)](#)

다음 경우 중 하나에서 테이블 결과를 추가하는 경우 데이터 추가를 완료했을 때 사용 가능한 유일한 옵션은 **데이터 준비**입니다.

- 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.

### 3 데이터 관리자를 사용하여 앱에서 데이터 관리

---

또한 데이터 관리자는 **데이터 추가**를 통해 추가된 테이블만 분석합니다. 데이터 로드 스크립트를 사용하여 추가된 테이블은 연결 권장 사항에 포함되지 않습니다.

# 4 데이터 로드 스크립트를 사용하여 데이터 로드

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. 로드할 필드와 테이블은 스크립트에서 지정합니다. 스크립트 문과 표현식을 사용하여 데이터 구조를 편집하는 것도 가능합니다.

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. 이 앱은 프로그램 기능의 핵심으로서 주요 특성은 제한 없는 자유로운 데이터 연결 방법, 방대한 차원 수 지원, 빠른 분석 속도, 그리고 작은 크기 등입니다. 해당 앱이 열려 있는 동안 앱은 RAM에 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 관리자 와 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 관리자에서 관리할 수 없습니다.

로드 스크립트에서 정의한 데이터 테이블은 데이터 관리자에서 관리되지 않습니다. 즉, 데이터 개요에서 테이블을 볼 수 있지만 데이터 관리자에서 테이블을 삭제하거나 편집할 수는 없으며, 스크립트를 사용하여 로드된 테이블에는 연결 권장 사항이 제공되지 않습니다.

스크립트 섹션을 추가하고 데이터 관리자에서 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.

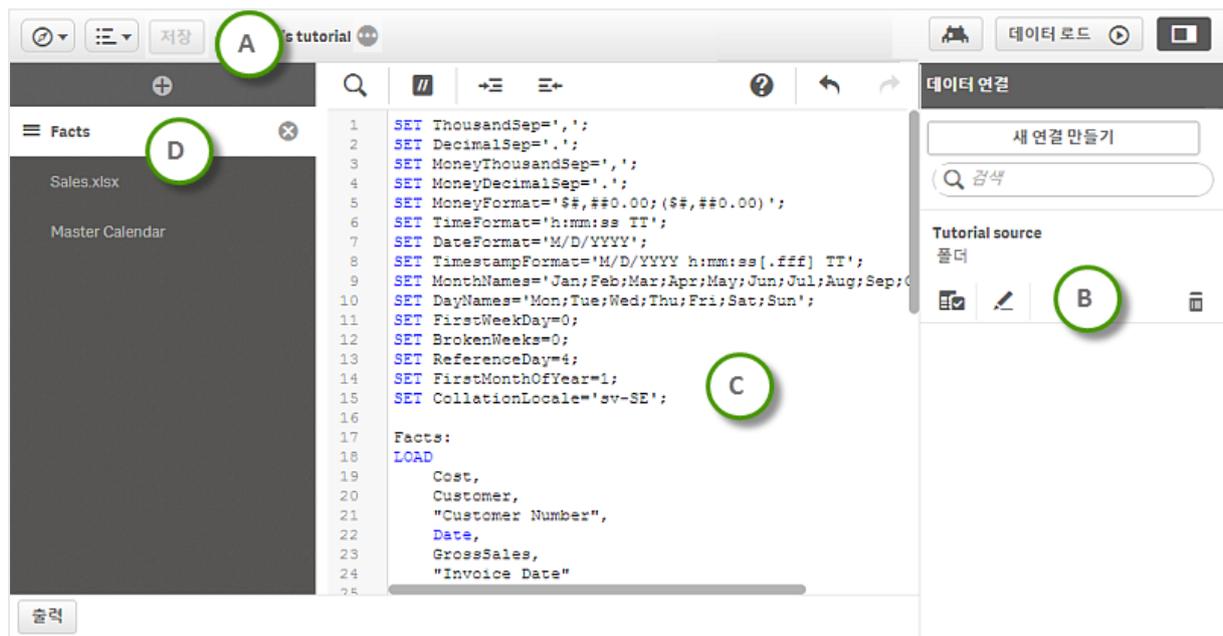
- 스크립트에서 데이터 관리자를 사용하여 추가된 테이블의 이름을 바꾸거나 삭제하는 경우.
- 데이터 관리자를 사용하여 추가된 테이블에서 필드를 삭제하는 경우.
- 데이터 관리자를 사용하여 추가된 테이블과 스크립트에서 로드된 테이블을 연결하는 경우.
- 데이터 관리자를 사용하여 추가된 테이블의 필드에서 **Qualify** 문을 사용하는 경우.
- 스크립트에서 **Resident**를 사용하여 데이터 관리자를 통해 추가된 테이블을 로드하는 경우.
- 생성된 코드 섹션 뒤에 스크립트 코드를 추가하는 경우. 데이터 모델에서의 결과 변경이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

## 4.2 데이터 로드 편집기 사용

이 섹션에서는 데이터 로드 편집기를 사용하여 데이터 모델을 앱으로 로드하는 데 사용할 수 있는 데이터 로드 스크립트를 만들거나 편집하는 방법을 설명합니다.

데이터 로드 스크립트는 앱을 데이터 소스에 연결하고 데이터 소스에서 앱으로 데이터를 로드합니다. 데이터를 로드했다면 분석을 위해 앱에서 사용할 수 있습니다. 데이터 로드 스크립트를 만들고 편집하고 실행하려면 데이터 로드 편집기를 사용하십시오.

스크립트는 수동으로 입력하거나 자동으로 생성할 수 있습니다. 복잡한 스크립트 문의 경우 최소한 일부를 수동으로 입력해야 합니다.



<p><b>A</b></p>	<p>데이터 로드 편집기에서 가장 자주 사용되는 명령인 탐색 메뉴, 전역 메뉴, 저장, u (디버그) 및 데이터 로드<sup>o</sup>가 포함되어 있는 도구 모음입니다. 도구 모음에는 앱의 저장 및 데이터 로드 상태도 표시됩니다.</p>	
<p><b>B</b></p>	<p>데이터 연결에서 공통적으로 사용하는 데이터 소스(데이터베이스 또는 원격 파일)에 대한 바로 가기를 저장할 수 있습니다. 로드할 데이터에 대한 선택을 시작하는 곳이기도 합니다.</p>	

<b>C</b>	<p>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 <b>검색 및 바꾸기, 도움말 모드, 실행 취소 및 재실행</b> 명령이 포함되어 있습니다. 최초의 스크립트에는 일반적으로 편집할 필요가 없는 일부 사전 정의된 지역 변수 설정(예: <code>SET ThousandSep=</code>)이 이미 포함되어 있습니다.</p>	
<b>D</b>	<p>읽고 유지 관리하기 쉽도록 스크립트를 섹션으로 구분합니다. 섹션은 맨 위에서 아래로 실행됩니다.</p> <p><b>데이터 추가</b>를 사용하여 데이터를 추가한 경우, 데이터를 로드하는 데 필요한 스크립트 코드가 포함되어 있는 <b>자동 생성 섹션</b>이라는 이름의 데이터 로드 스크립트 섹션이 생깁니다.</p>	
<b>E</b>	<p><b>출력</b>에는 스크립트 실행 중에 생성된 모든 메시지가 표시됩니다.</p>	

### 빠른 시작

데이터베이스에서 파일 또는 테이블을 로드하려면 **데이터 연결**에서 다음 단계를 완료해야 합니다.

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

**스크립트 삽입**을 통해 선택 대화 상자를 완료하면 **데이터 로드**를 선택하여 데이터 모델을 앱으로 로드할 수 있습니다.



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

### 도구 모음

도구 모음을 사용하면 데이터 로드 스크립트에서 실행 취소/재실행, 디버그, 검색/바꾸기 등의 전역 작업을 수행할 수 있습니다. 또한 **데이터 로드** °를 클릭하여 앱에서 데이터를 다시 로드할 수도 있습니다.

### 기본 도구 모음

”	<p>다음 옵션이 포함된 탐색 메뉴:</p> <p><b>ü 앱 개요</b></p> <p><b>7 데이터 모델 뷰어</b></p> <p><b>^ 허브 열기</b></p> <p>허브가 관리자에 의해 비활성화된 경우 <b>허브 열기</b>가 표시되지 않습니다.</p>
..	<p>다음 옵션이 포함된 메뉴:</p> <p><b>« 데이터 추가</b></p> <p><b>D 도움말</b></p> <p><b>] 정보</b></p>
저장	변경 사항을 저장합니다.
⊞	<p>선택하여 앱 정보를 편집하거나 앱 옵션을 열고 앱 스타일을 지정할 수 있는 앱 정보를 표시하거나 숨깁니다.</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p> 앱이 게시되면 앱 정보를 편집하거나 앱 옵션을 열 수 없습니다.</p> </div>
u	스크립트를 디버깅합니다.
데이터 로드 °	스크립트를 실행하고 데이터를 다시 로드합니다. 다시 로드하기 전에 앱이 자동으로 저장됩니다.
h	<b>데이터 연결</b> 보기를 설정/해제합니다.

### 편집기 도구 모음

F	스크립트의 텍스트를 검색하고 바꿉니다.
...	주석 처리/주석 처리 취소
À	들여쓰기
Ã	내어쓰기

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

<b>D</b>	구문 도움말 모드를 활성화합니다. 도움말 모드에서 편집기의 구문 키워드 (파란색으로 표시)를 클릭하여 자세한 구문 도움말에 액세스할 수 있습니다. <div data-bbox="478 369 1388 470"> 도움말 모드에서는 스크립트를 편집할 수 없습니다.</div>
<b>B</b>	현재 섹션에서 최근 변경 내용을 취소합니다(여러 단계 실행 취소도 가능). <b>Ctrl+Z</b> 를 누르는 것과 같습니다.
<b>C</b>	현재 섹션에서 가장 최근의 <b>실행 취소</b> 를 재실행합니다. <b>Ctrl+Y</b> 를 누르는 것과 같습니다.

### 데이터 로드 편집기에서 데이터 소스에 연결

데이터 로드 편집기에서 데이터 연결은 일반적으로 사용하는 데이터베이스, 로컬 파일 또는 원격 파일과 같은 데이터 소스에 대한 바로 가기를 저장하는 방법을 제공합니다. **데이터 연결**은 영문자순으로 저장된 연결을 표시합니다. 검색/필터 상자를 사용하면 특정 이름 또는 유형을 가진 연결로 목록을 좁힐 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.

 현재는 개인 파일, 첨부 파일 또는 **Qlik Sense Cloud**의 **Qlik DataMarket**에서만 데이터에 연결할 수 있습니다.

### 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 Console**에서 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 **DSN**에서 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 **E** 를 클릭합니다.
2. 연결을 삭제할 것인지 확인합니다.

이제 데이터 연결이 삭제됩니다.



**E** 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. 편집할 데이터 연결에서 **@** 을 클릭합니다.
2. 데이터 연결 세부 정보를 편집합니다. 연결 세부 정보는 연결 유형마다 고유합니다.
3. **저장**을 클릭합니다.

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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

다음과 같이 하십시오.

- 연결 문자열을 추가할 연결에서  $\emptyset$  을 클릭합니다.

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



또한 데이터 연결을 끌어서 스크립트 내의 삽입할 위치에 놓으면 연결 문자열을 삽입할 수 있습니다.

### 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.  $\pm$  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 1: 폴더 데이터 연결에서 파일 로드

이 예에서는 MyData 데이터 연결에 정의된 위치에서 *orders.csv* 파일을 로드합니다.

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

#### 예 2: 하위 폴더에서 파일 로드

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

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

#### 예 3: 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 4: 데이터베이스에서 로드

이 예에서는 MyDataSource 데이터베이스 연결에서 *Sales\_data* 테이블을 로드합니다.

```
LIB CONNECT TO 'MyDataSource';  
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 Desktop**에서 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 로드 편집기의 데이터 연결에서 **±**을 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 편집기의 데이터베이스 연결에서 **±**을 클릭합니다.  
데이터 선택 대화 상자가 표시됩니다.
2. 드롭다운 목록에서 **데이터베이스**를 선택합니다.  
연결을 구성할 때 데이터베이스 이름이 입력되므로 일부 선택 대화 상자에는 **데이터베이스** 드롭다운 목록이 없습니다.
3. 데이터베이스의 **소유자**를 선택합니다.  
**테이블** 목록은 선택된 데이터베이스에서 사용할 수 있는 보기와 테이블로 채워집니다.  
일부 데이터베이스는 데이터 선택 프로세스에서 소유자를 지정할 필요가 없습니다.
4. 테이블을 선택합니다.
5. 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.

- 추가 테이블에서 데이터를 추가하려면 이를 선택합니다.



검색을 통해 필드를 필터링 하면서 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.

- 데이터 선택이 끝나면 다음을 수행합니다.

- 스크립트 삽입을 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 컨케이트네이트됩니다.

다음과 같이 하십시오.

- 데이터 로드 편집기의 적절한 폴더 연결에서 **±** 을 클릭합니다. 데이터 선택 대화 상자가 표시됩니다.
- 데이터를 선택할 첫 번째 시트를 선택합니다. 시트 이름 옆의 상자를 선택하면 시트 내의 모든 필드를 선택할 수 있습니다.
- 해당 시트에 대한 설정이 적절한지 확인하십시오.

**필드 이름** 테이블에 포함된 필드 이름이 있는지, 또는 필드 이름 없음이 있는지 지정합니다. Excel 스프레드시트에서는 일반적으로 첫 행에 포함된 필드 이름이 있습니다. 필드 이름 없음을 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.

<b>Machine:</b>	AEJ12B		
-----------------	--------	--	--

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

<b>Date:</b>	2015-10-05 09		
<b>Timestamp</b>	<b>Order</b>	<b>Operator</b>	<b>Yield</b>
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.

**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 읽은 첫 줄은 세 번째 줄입니다.

- 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.

- 데이터 선택이 완료되면 다음을 수행합니다.

- 스크립트 삽입을 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, **폴더 데이터 연결** 대신 **ODBC 데이터 연결**을 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

- 고정 레코드 길이 파일.
- DIF 파일(Data Interchange Format).

다음과 같이 하십시오.

1. 데이터 로드 편집기의 적절한 폴더 연결에서 **±** 을 클릭합니다.  
데이터 선택 대화 상자가 표시됩니다.
2. **파일 형식**에 적절한 파일 형식이 선택되어 있는지 확인하십시오.
3. 해당 파일에 대한 설정이 적절한지 확인하십시오. 파일 형식이 다르면 파일 설정이 달라집니다.
4. 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.

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

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.

<b>파일 형식</b>	구분 또는 고정 레코드로 설정합니다.  선택하면, 데이터 선택 대화 상자가 선택한 파일 형식에 맞춰 조정됩니다.
<b>필드 이름</b>	테이블에 포함된 필드 이름이 있는지, 또는 필드 이름 없음이 있는지 지정합니다.
<b>구분 기호</b>	테이블 파일에서 사용되는 구분 기호를 설정합니다.
<b>인용</b>	인용 부호를 처리하는 방법을 지정하기 위해 설정합니다.  <b>없음</b> = 인용 문자가 허용되지 않음  <b>표준</b> = 표준 인용 부호(필드 값의 첫 문자 및 마지막 문자로 인용 부호 사용 가능)  <b>MSQ</b> = 최신 인용 부호(필드에서 복수 행의 콘텐츠 허용)
<b>헤더 크기</b>	테이블 헤더로 생략할 행 수를 설정합니다.
<b>문자 집합</b>	테이블 파일에서 사용되는 문자 집합을 설정합니다.

---

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

<b>주석</b>	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.
<b>EOF 무시</b>	데이터에 필드 값의 일부로 파일 끝 문자가 포함되는 경우, <b>EOF 무시</b> 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 위치에** 선택표로 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.

<b>필드 이름</b>	테이블에 <b>포함된 필드 이름</b> 이 있는지, 또는 <b>필드 이름 없음</b> 이 있는지 지정합니다.
<b>헤더 크기</b>	<b>헤더 크기</b> 를 테이블 헤더로서 생략할 행 수로 설정합니다.
<b>문자 집합</b>	테이블 파일에서 사용하는 문자 집합으로 설정됩니다.
<b>탭 크기</b>	테이블 파일에서 하나의 탭 문자가 나타내는 공백 수로 설정합니다.
<b>레코드 줄 크기</b>	테이블 파일에서 하나의 레코드 범위에 해당하는 행 수로 설정합니다. 기본 값은 1입니다.

### HTML 파일

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

<b>필드 이름</b>	테이블에 <b>포함된 필드 이름</b> 이 있는지, 또는 <b>필드 이름 없음</b> 이 있는지 지정합니다.
<b>문자 집합</b>	테이블 파일에서 사용하는 문자 집합을 설정합니다.

### XML 파일

XML 형식으로 저장된 데이터를 로드할 수 있습니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

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 커백터를 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 보기를 클릭하여 숨길 수 있습니다.

## LOAD 문 포함

**LOAD 문 포함**을 선택하면 앞에 **LOAD** 문이 있고 **SELECT** 문을 입력으로 사용하는 **SELECT** 문이 생성됩니다.



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

## 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

### 도움말 포털 액세스

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

- 도구 모음에서 **D** 을 클릭하여 구문 도움말 모드를 시작합니다. 구문 도움말 모드에서 구문 키워드(파란색과 밑줄로 표시)를 클릭하여 구문 도움말에 액세스할 수 있습니다.



구문 도움말 모드에서는 스크립트를 편집할 수 없습니다.

- 키워드 내부 또는 끝에 커서를 놓고 **Ctrl+H**를 누릅니다.

### 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+00**을 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. **À** 를 클릭하여 텍스트를 들여쓰거나(들여쓰기 증가), **Ã** 를 클릭하여 텍스트를 내어쓰니다(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 **F** 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. **ê** 및 **ë** 을 클릭하여 결과를 탐색할 수 있습니다.

도구 모음의 **F** 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 **F** 을 클릭합니다.  
검색 드롭다운 대화 상자가 표시됩니다.
2. 검색 상자에 찾으려는 텍스트를 입력합니다.
3. 바꾸기 상자에 대체 텍스트를 입력하고 **바꾸기**를 클릭합니다.
4. **ë** 을 클릭하여 검색 텍스트의 다음 인스턴스를 찾고 다음 중 하나를 수행합니다.
  - **바꾸기**를 클릭하여 텍스트를 바꿉니다.
  - **ë** 을 클릭하여 다음 항목을 찾습니다.

도구 모음의 **F** 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

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+K**를 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+K**를 누릅니다.

이제 선택한 코드가 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
```

### 자동 생성 섹션 잠금 해제

**데이터 추가**를 사용하여 데이터를 추가한 경우, 데이터를 로드하는 데 필요한 스크립트 코드가 포함되어 있는 **자동 생성 섹션**이라는 이름의 데이터 로드 스크립트 섹션이 생깁니다. 자동 생성된 스크립트는 읽기 전용이지만 잠금 해제할 수 있습니다. 따라서 데이터 모델의 이 부분을 정의하려면 **데이터 관리자**에서 데이터 로드 스크립트로 전환해야 합니다. 이 기능은 앱에서 필요한 데이터 모델에 연결하기 위해 보다 복잡한 데이터 작업을 수행해야 하는 경우에 유용할 수 있습니다.



자동 생성된 스크립트를 잠금 해제하면 더 이상 **데이터 관리자**를 사용하여 이 데이터의 선택 항목을 편집할 수 없습니다. **데이터 관리자**를 사용하여 새 데이터를 추가하는 경우 이 기능은 자동 생성 스크립트 섹션을 새로 만듭니다.

다음과 같이 하십시오.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

1. 이름이 **Auto-generated section**으로 지정된 섹션을 선택합니다.
2. **잠금 해제**를 클릭합니다.

이제 전체 데이터 로드 스크립트를 편집하여 데이터 모델을 조정할 수 있습니다.



해당 섹션의 잠금을 해제한 후 이름을 변경하는 것이 좋습니다.

### 모든 코드 선택

현재 스크립트 섹션의 모든 코드를 선택할 수 있습니다.

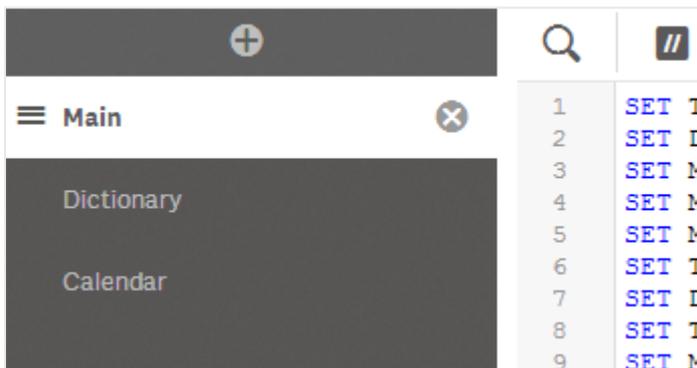
다음과 같이 하십시오.

- **Ctrl+A**를 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 추가**를 사용하여 데이터를 추가한 경우, 데이터를 로드하는 데 필요한 스크립트 코드가 포함되어 있는 **자동 생성 섹션**이라는 이름의 데이터 로드 스크립트 섹션이 생깁니다.

### 새 스크립트 섹션 만들기

새 스크립트 섹션을 삽입하여 코드를 분류할 수 있습니다.

다음과 같이 하십시오.

- **P**를 클릭합니다.

현재 선택한 섹션 뒤에 새 섹션이 삽입됩니다.

### 스크립트 섹션 삭제

섹션 내의 모든 코드를 포함하여 스크립트 섹션을 삭제할 수 있습니다.



스크립트 섹션 삭제는 실행 취소할 수 없습니다.

다음과 같이 하십시오.

- 삭제하려면 섹션 탭 옆의 **E** 를 클릭합니다. 삭제 확인을 해야 합니다. 이제 섹션이 삭제됩니다.

### 스크립트 섹션 이름 변경

스크립트 섹션의 이름을 변경할 수 있습니다.

다음과 같이 하십시오.

1. 섹션 이름을 클릭하고 입력하여 이름을 편집합니다.
2. 끝나면 **Enter**를 누르거나 섹션 바깥쪽을 클릭합니다.

이제 섹션 이름이 변경됩니다.

### 스크립트 섹션 재배열

섹션 순서를 재배열하여 스크립트 실행 순서를 변경할 수 있습니다.

다음과 같이 하십시오.

1. 이동할 섹션을 선택합니다.
2. **o** 드래그 바에 커서를 놓고 섹션을 끌어서 순서를 재배열합니다.

이제 섹션이 재배열되었습니다.

## 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 **u** 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

<b>제한된 로드</b>	<p>이 옵션을 사용하면 각 데이터 소스에서 로드할 데이터 행의 수를 제한할 수 있습니다. 데이터 소스가 큰 경우 실행 시간을 줄이는 데 유용합니다.</p> <p>로드할 행의 수를 입력하십시오.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> 이 옵션은 실제 데이터 소스에만 적용됩니다. 자동으로 생성되며 인라인 로드 등은 제한되지 않습니다.</p> </div>
<b>œ</b>	다음 중단점에 도달할 때까지 디버그 모드에서 실행을 시작하거나 계속합니다.
<b>œ</b>	다음 코드 행으로 넘어갑니다.
<b>-</b>	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.

<b>지우기</b>	모든 출력 메시지를 삭제하려면 이 옵션을 클릭합니다.
<b>모든 텍스트 선택</b>	모든 출력 메시지를 선택하려면 이 옵션을 클릭합니다.
<b>맨 아래로 스크롤</b>	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.

모든 변수 표시	모든 유형의 변수를 표시하려면 이 옵션을 클릭합니다.
시스템 변수 표시	시스템 변수를 표시하려면 이 옵션을 클릭합니다.  시스템 변수는 <b>Qlik Sense</b> 에서 정의하지만 스크립트에서 해당 변수 값을 변경할 수 있습니다.
예약된 변수 표시	예약된 변수를 표시하려면 이 옵션을 클릭합니다.  예약된 변수는 <b>Qlik Sense</b> 에서 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 **Q**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 **Q**을 클릭합니다.
- 중단점 목록에서 중단점 옆의 **E**을 클릭합니다.

또한 **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 **m**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.

다음과 같이 하십시오.

- 데이터 로드 편집기 도구 모음에서 **저장**을 클릭하여 스크립트를 저장하십시오.

이제 스크립트가 저장되었지만 앱에는 이전에 다시 로드했던 데이터가 아직 들어 있으며, 이와 관련된 메시지가 도구 모음에 표시됩니다. 새 데이터로 앱을 업데이트하려면 데이터 로드 편집기 도구 모음에서 **데이터 로드**<sup>o</sup>를 클릭합니다.

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



데이터를 로드하면 스크립트가 앱에 자동 저장됩니다.

### 데이터를 로드하는 스크립트 실행

도구 모음에서 **데이터 로드**<sup>o</sup>를 클릭하면 스크립트를 실행하고 앱에서 데이터를 다시 로드할 수 있습니다. 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 **Windows**에서 작업하는 경우를 기준으로 설명합니다. **Mac OS**는 **Ctrl** 대신 **Cmd**를 사용하십시오.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

바로 가기	동작	Qlik Sense에서 지원	Qlik Sense Desktop에서 지원
Ctrl+P	현재 보기 또는 활성 시트/스토리를 인쇄합니다.	예	예
Ctrl+C	선택한 항목을 클립보드로 복사합니다.	예	예
Ctrl+X	선택한 항목을 잘라내서 클립보드로 복사합니다. <b>Google Chrome</b> 브라우저를 사용하는 경우: 데이터 로드 편집기 또는 수식 편집기에서 아무 것도 선택하지 않고 커서를 행 앞에 놓으면 전체 행이 잘립니다.	예	예
Ctrl+V	가장 최근에 복사한 항목을 클립보드에서 붙여넣습니다.	예	예
Ctrl+Z	작업을 실행 취소합니다. 더 이전의 작업을 실행 취소하려면 반복합니다.	예	예
Ctrl+H	데이터 로드 편집기 또는 수식 편집기에 있을 때 현재 함수 컨텍스트에서 온라인 도움말을 엽니다.	예	예
Ctrl+F	검색 항목 찾기 필드를 표시하거나 열린 경우 이를 숨깁니다.	예	예
Ctrl+S	변경 사항을 저장합니다.	예	예
검색 필드에서 Enter/Return 키	검색 문자열의 다음 인스턴스를 검색합니다.	예	예
바꾸기 필드에서 Enter/Return 키	검색 문자열의 선택된 인스턴스를 바꿉니다.	예	예
검색 또는 바꾸기 필드에서 Esc 키	검색 항목 찾기 필드를 닫습니다.	예	예
Ctrl-shift-Enter/Return	데이터를 다시 로드합니다.	예	예
Alt+F11	스크립트 편집기를 전체 화면으로 확장합니다.	예	예
Alt+F5	디버그 도구를 표시하거나 표시된 경우 숨깁니다.	예	예
Alt+F6	디버그를 실행합니다.	예	예

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

Alt+F7	디버거에서 다음 단계로 진행합니다.	예	예
Alt+F8	디버거를 중지합니다.	예	예
F9	디버그 중단점의 삽입 및 제거를 전환합니다.	예	예
Alt+F9	왼쪽 패널을 표시하거나 표시된 경우 숨깁니다.	예	예
Alt+F10	오른쪽 패널을 표시하거나 표시된 경우 숨깁니다.	예	예
Alt+PgUp	이전 섹션으로 이동합니다.	예	예
Alt+PgDn	다음 섹션으로 이동합니다.	예	예
Alt+Insert	스크립트에 새 섹션을 삽입합니다.	예	예
Ctrl+K	코드 주석 행의 삽입 및 제거를 전환합니다.	예	예
Tab	스크립트에서 행을 들여쓰습니다.	예	예
Shift+Tab	스크립트에서 행을 내어쓰습니다.	예	예
Alt+1	출력 패널을 표시하거나 표시된 경우 숨깁니다.	예	예
Alt+2	변수 패널을 표시하거나 표시된 경우 숨깁니다.	예	예
Alt+3	중단점 패널을 표시하거나 표시된 경우 숨깁니다.	예	예
Ctrl+space	자동 텍스트 문자열로 자동 완성합니다.	예	예
Ctrl+Shift+space	도구 설명을 표시합니다.		

### 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.

##### 1. 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 공급자로부터 제공되는 데이터를 선택하는 데 사용됩니다. **SELECT SQL** 문은 Qlik Sense가 아니라 데이터 공급자가 평가합니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

- **LOAD** 문은 파일, 스크립트에 정의된 데이터, 이전에 로드한 테이블, 웹 페이지, 이후 **SELECT** 문의 결과에서 또는 자동으로 데이터를 생성하여 데이터를 로드하는 데 사용됩니다.

### 2. 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

### 3. 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 일종의 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 공급자로부터 제공되는 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 이들의 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 엔티티입니다. 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

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

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 Sense**에 대한 기본 캘린더 필드 정의는 **데이터 관리자**를 사용하여 로드된 날짜 필드의 **autoCalendar**에 포함됩니다. **앱에 데이터 추가(19페이지)**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 **autoCalendar**를 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

### 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.

태그	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
\$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. <b>HidePrefix</b> 및 <b>HideSuffix</b> 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

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

태그	설명	스크립트에서 편집 가능 여부
\$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 [위도, 경도] 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.

태그	설명
\$axis	\$axis 태그는 필드가 차트의 연속 축에 눈금을 생성하도록 지정하는 데 사용됩니다.
\$qualified	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.
\$simplified	예를 들어, 분기별 데이터를 표시하는 경우 다음 두 개의 필드를 생성할 수 있습니다. <ul style="list-style-type: none"> <li>'Q1'과 같이 분기를 표시하며 \$simplified 태그를 통해 간소화된 필드.</li> <li>'2016-Q1'과 같이 연도 및 분기를 표시하며 \$qualified 태그를 통해 한정된 필드.</li> </ul> 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

필드	설명	스크립트에서 편집 가능 여부
<b>\$Table</b>	로드된 모든 테이블을 포함합니다.	아니요
<b>\$Field</b>	로드된 테이블의 모든 필드를 포함합니다.	아니요
<b>\$Fields</b>	각 테이블의 필드 수를 포함합니다.	아니요
<b>\$FieldNo</b>	테이블에 있는 필드의 위치를 포함합니다.	아니요
<b>\$Rows</b>	테이블에 있는 행의 수를 포함합니다.	아니요

### 시각화에서 시스템 필드 사용

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

시스템 필드는 자산 패널 또는 식 편집기의 필드 목록에 포함되지 않습니다. 시스템 필드를 사용하려면 수동으로 입력하여 참조해야 합니다.

### 자산 패널의 차원에서

**=*\$Field***

### 식 편집기에서

***\$Field***

### 필드 이름 바꾸기

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

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

### 예 1: alias 문 사용

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

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

### 예 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 Sense**는 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.

<b>concatenate</b>	이 테이블이 다른 이름을 가진 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 연결됩니다.
<b>crosstable</b>	이 테이블은 피벗 해제됩니다. 즉, 교차 표 형식에서 열 형식으로 변환됩니다.
<b>generic</b>	이 테이블이 다른 여러 논리 테이블로 분할됩니다.
<b>intervalmatch</b>	테이블(정확히 2개의 열을 포함해야 함)이 지정된 필드 내 불연속 숫자와 연결된 숫자 간격으로 해석됩니다.
<b>join</b>	<b>Qlik Sense</b> 에서는 다른 이름의 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 공통적인 필드를 통해 이 테이블을 조인합니다.
<b>keep</b>	이 테이블이 다른 이름의 테이블 또는 이전에 마지막으로 생성된 논리 테이블과 공통적인 필드로 축소됩니다.
<b>mapping</b>	이 테이블(2개의 열을 포함해야 함)이 다른 테이블과 연결되지 않는 매핑 테이블로 로드됩니다.
<b>semantic</b>	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 **xx**는 숫자입니다. 첫 번째 인라인 테이블의 이름은 **INLINE01**이 됩니다.
4. 자동으로 생성된 테이블은 **AUTOGENERATExx** 형식으로 이름이 지정됩니다. 여기서 **xx**는 숫자입니다. 자동 생성된 첫 번째 테이블의 이름은 **AUTOGENERATE01**이 됩니다.
5. 위의 규칙에 따라 생성된 테이블 이름이 이전 테이블 이름과 충돌하는 경우는 이름이 **-x**로 확장됩니다. 여기서 **x**는 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.

#### 예 1:

Table1:

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

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

#### 예 2: 공백이 포함된 테이블 레이블

[All Transactions]:

```
SELECT * from Transtable;
```

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

```
LOAD Month, sum(Sales) resident [All Transactions] group by Month;
```

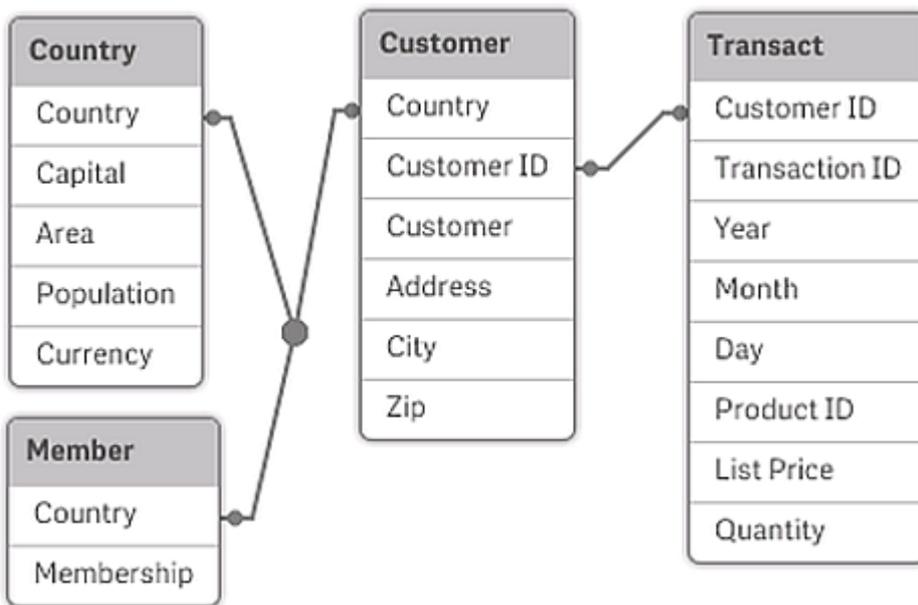
### 논리 테이블 간의 연결

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 Sense**는 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. **SQL**의 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 Sense**가 데이터 빈도를 계산할 때 어느 테이블을 사용해야 할지 인식하는 데 문제가 생깁니다.

**Qlik Sense**는 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 Sense**에 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/a**로 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 Sense**는 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

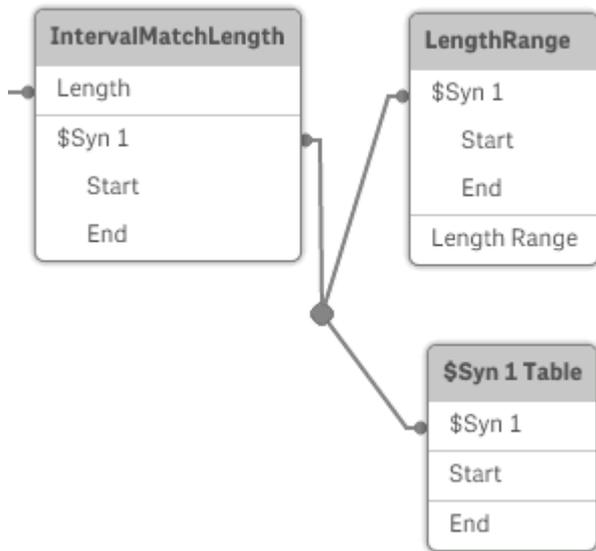
두 개 이상의 데이터 테이블에 두 개 이상의 공통적인 필드가 있는 경우 이는 복합 키 관계를 나타냅니다. **Qlik Sense**에서는 가상 키를 만들어 자동으로 이를 처리합니다. 이 키는 복합 키에서 발생하는 모든 조합을 나타내는 익명 필드입니다.

데이터를 로드할 때 가상 키에 대한 경로를 받는 경우, 데이터 모델 뷰어에서 데이터 구조를 검토하는 것이 좋습니다. 데이터 모델이 정확한지 여부를 확인해 보아야 합니다. 때때로 스크립트에서 오류로 가상 키가 생기는 경우가 있습니다.

복수의 가상 키는 종종 잘못된 데이터 모델의 증상을 보이지만 반드시 그런 것은 아닙니다. 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 Sense**는 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 처리

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

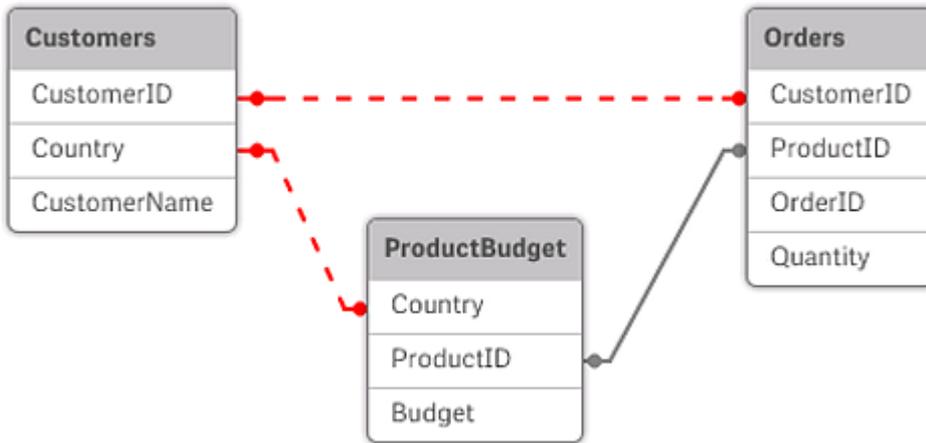
- 두 테이블을 논리적으로 연결하는 필드만 키로 사용하고 있는지 확인하십시오.
  - "Comment", "Remark" 및 "Description" 등의 필드는 관련성 없이 여러 테이블에 존재할 수 있으므로 키로 사용하지 않도록 해야 합니다.
  - "Date", "Company" 및 "Name" 등의 필드는 여러 테이블에 존재하며 동일한 값을 가질 수 있지만 여전히 여러 가지 역할을 가질 수 있습니다(주문 날짜/배송 날짜, 고객 회사/공급 업체). 이런 경우에도 키로 사용하지 않아야 합니다.
- 중복 필드를 사용하지 않도록 해야 합니다. 즉, 필요한 필드만 연결합니다. 예를 들어 날짜가 키로 사용된 경우 둘 이상의 내부 테이블에서 동일한 날짜의 `year`, `month` 또는 `day_of_month`를 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드



순환 참조가 포함된 세 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

NationalTeams		Clubs		Cities	
Country	Team	City	Team	Country	City
Germany	Die Mannschaft	Barcelona	Barcelona	Germany	Hamburg
Italy	Azzurri	Hamburg	Altona	Germany	Munich
Spain	La Roja	Madrid	Real Madrid	Italy	Milano
		Milano	Milan	Italy	Rome
		Munich	Bayern München	Italy	Turin
		Rome	Lazio	Spain	Barcelona
		Turin	Juventus	Spain	Madrid

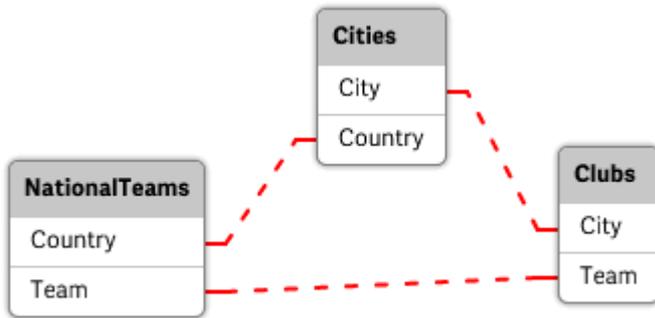
소스 데이터 테이블 보기

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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

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.

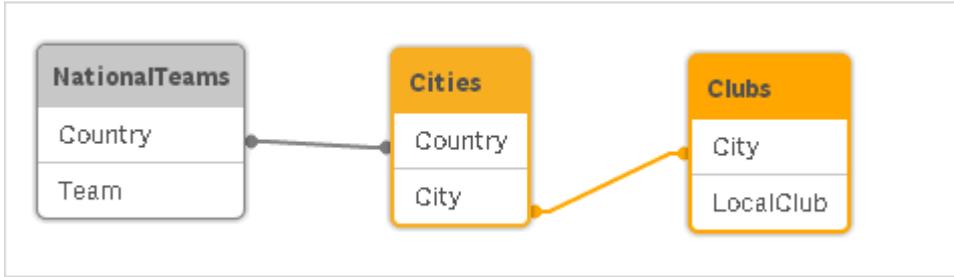
다음과 같이 하십시오.

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.

Country	Team	City	LocalClub
Germany ✓	Die	Hamburg	Altona
Italy	Azzurri	Munich	Barcelona
Spain	La Roja	Barcelona	
		Madrid	

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 컨케트네이트합니다.

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

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

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



이전에 로드한 테이블의 이름을 **concatenate** 문에 지정하지 않은 경우 **concatenate** 접두사는 최근에 만든 테이블을 사용합니다. 따라서 두 문의 순서는 의미가 아닙니다.

### 연결 방지

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

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

```
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. 이 방법으로 원하는 수만큼 **LOAD** 문을 누적할 수 있습니다. 맨 아래의 문을 처음으로 평가한 다음 그 위의 문을 평가하는 방식으로 맨 위의 문이 평가될 때까지 계속 진행됩니다.

**Resident**를 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.



**Join**, **Crosstable** 및 **Intervalmatch** 접두사는 선행 **LOAD**와 연결하여 사용할 수 없습니다.

### 예 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 ... ;
```

### 예 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 ;
```

---

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

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 ;
```

### 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 읽어들이는 여러 데이터 항목이 동일한 숫자 표현을 가지고 있다면 동일한 값으로 취급되며 첫 번째로 발견되는 문자열 표현을 공유하게 됩니다. 예를 들어 숫자 1.0, 1 및 1.000을 순서대로 읽어 온다면 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

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.

1. 소스 데이터베이스에서 데이터가 기록된 방식
2. 숫자, 시간, 날짜 등에 대한 운영 체제 설정 (**제어판**)
3. 스크립트에서 선택적 숫자 해석 변수 사용
4. 스크립트에서 선택적 해석 함수 사용
5. 스크립트에서 선택적 서식 지정 함수 사용
6. 문서 내의 숫자 서식 콘트를

**Qlik Sense**는 입력 데이터를 숫자, 날짜, 시간 등으로 해석하려고 시도합니다. 데이터에 시스템 기본 설정이 사용되는 경우에 한해 **Qlik Sense**에서 해석과 표시 서식 지정을 자동으로 처리하며 사용자가 스크립트 또는 **Qlik Sense**의 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 Sense**는 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 Sense**에서 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 Sense**에서 날짜 또는 시간으로 해석한 데이터는 시각화의 속성 패널에서 다른 날짜 또는 시간 서식으로 변경할 수 있습니다.

---

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

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 지정되지 않지만 원래 서식대로 표시됩니다.

### 날짜 및 시간 해석

**Qlik Sense**는 데이터에서 발견된 각 날짜, 시간 및 타임스탬프를 날짜 일련 번호로 저장합니다. 날짜 일련 번호는 날짜, 시간 및 타임스탬프와 날짜 및 시간 엔티티를 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 Sense**에서 일반 달력과 동일하게 확장된 날짜 시스템을 사용합니다.

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.

#### 예 1:

1997-08-06            다음과 같이 저장됨:    35648  
09:00                다음과 같이 저장됨:    0.375  
1997-08-06 09:00    다음과 같이 저장됨:    35648.375

반대의 경우,

35648    숫자 서식이 'D/M/YY'일 경우 다음과 같이 표시됨:    6/8/97  
0.375    숫자 서식이 'hh.mm'일 경우 다음과 같이 표시됨:    09.00

**Qlik Sense**는 일련의 규칙에 따라 날짜, 시간 및 기타 데이터 유형을 해석하려 시도합니다. 하지만 최종 결과는 여기에 설명된 여러 요소의 영향을 받습니다.

#### 예 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 Sense**로 데이터를 읽을 때의 다양한 표현이 나와 있습니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

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.

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

### 달려 기호 확장

달려 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 시작하고 ')'로 끝나며, 괄호 안의 내용은 텍스트 대체가 이루어지는 방법을 정의합니다. 스크립트 매크로와 혼동을 피하기 위해 지금부터는 매크로 확장을 달려 기호 확장이라고 부르겠습니다.

달려 기호 확장은 다음 중 하나로 사용할 수 있습니다.

- 변수
- 파라메타
- 표현식



달려 기호 확장은 계산할 수 있는 확장 수에 따라 제한됩니다. 1000개가 넘는 확장은 계산되지 않습니다.

### 변수를 사용한 달려 기호 확장

스크립트 또는 표현식 내의 텍스트 대체에 변수를 사용하는 경우 다음과 같은 구문이 사용됩니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

### `$(variablename)`

`$(variablename)`은 변수의 값으로 확장됩니다. `variablename`이 존재하지 않을 경우 확장 결과는 빈 문자열이 됩니다.

숫자 변수 확장의 경우 다음과 같은 구문이 사용됩니다.

### `$(#variablename)`

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 `0`으로 확장됩니다.

```
SET DecimalSep=',';
```

```
LET X = 7/2;
```

달러 기호 확장 `$(X)`은 `3,5`로 확장되고 `$(#X)`은 `3.5`로 확장됩니다.

```
Set MyPath=C:\MyDocs\Files\;
```

```
...
```

```
LOAD * from $(MyPath)abc.csv;
```

데이터는 `C:\MyDocs\Files\abc.csv`에서 로드됩니다.

```
Set CurrentYear=1992;
```

```
...
```

```
SQL SELECT * FROM table1 WHERE Year=$(CurrentYear);
```

`Year=1992`인 행이 선택됩니다.

```
Set vConcatenate = ;
```

```
For each vFile in FileList('\*.txt')
```

```
  Data:
```

```
    $(vConcatenate)
```

```
    LOAD * FROM [$(vFile)];
```

```
    Set vConcatenate = Concatenate ;
```

```
Next vFile
```

이 예에서는 **Concatenate** 접두사를 사용하여 디렉터리의 모든 `.txt` 파일을 로드합니다. 필드가 약간 달라서 자동 연결이 작동하지 않는 경우에 이 작업이 필요할 수 있습니다. 첫 번째 로드에서는 **Concatenate** 접두사를 사용할 수 없으므로 `vConcatenate` 변수가 초기에는 빈 문자열로 설정됩니다. 디렉터리에 `file1.txt`, `file2.txt` 및 `file3.txt`의 3개 파일이 포함되어 있으면 3회 반복 중에 **LOAD** 문이 다음과 같이 확장됩니다.

```
LOAD * FROM[.\file1.txt];
```

```
Concatenate LOAD * FROM[.\file2.txt];
```

```
Concatenate LOAD * FROM[.\file3.txt];
```

### 파라메타를 사용한 달러 기호 확장

파라메타를 달러 기호 확장에 사용할 수 있습니다. 이때 변수에는 **\$1**, **\$2**, **\$3** 등과 같은 공식 파라메타가 포함되어 있어야 합니다. 변수를 확장할 때 파라메타는 쉼표로 구분된 목록으로 지정해야 합니다.

```
Set MUL='$1*$2';  
Set X=$(MUL(3,7)); //은 X에 '3*7'을 반환합니다.
```

```
Let X=$(MUL(3,7)); //은 X에 21을 반환합니다.
```

공식 파라메타의 수가 실제 파라메타의 수를 초과하는 경우 실제 파라메타에 대응하는 공식 파라메타만 확장됩니다. 실제 파라메타의 수가 공식 파라메타의 수를 초과하는 경우 여분의 실제 파라메타는 무시됩니다.

```
Set MUL='$1*$2';  
Set X=$(MUL); //은 X에 '$1*$2'를 반환합니다.
```

```
Set X=$(MUL(10)); //은 X에 '10*$2'를 반환합니다.
```

```
Let X=$(MUL(5,7,8)); //은 X에 35를 반환합니다.
```

파라메타 **\$0**은 호출에서 실제로 전달한 파라메타의 수를 반환합니다.

```
set MUL='$1*$2 $0 par';  
set X=$(MUL(3,7)); //은 X에 '3*7 2 par'을 반환합니다.
```

### 표현식을 사용한 달러 기호 확장

표현식을 달러 기호 확장에 사용할 수 있습니다. 이때 괄호 사이의 내용은 등호로 시작해야 합니다.

```
$(=expression )
```

표현식이 평가되고 값이 확장에 사용됩니다.

```
$(=Year(Today())); //는 현재 연도가 포함된 문자열을 반환합니다.
```

```
$(=Only(Year)-1); //은 선택한 해의 이전 해를 반환합니다.
```

### 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 매우 유용합니다.

```
$(include=C:\Documents\MyScript.qvs);
```

### 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.

	설명	기호	코드 포인트	예
필드 이름	큰따옴표	" "	34	"string"
	대괄호	[ ]	91, 93	[string]
	억음 악센트 기호	``	96	`string`
문자열 리터럴	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, land가 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 작은따옴표 사용

문자열에 작은따옴표를 포함해야 하는 경우 추가적인 작은따옴표를 이스케이프 문자로 사용할 수 있습니다.

문자열 '10 o'clock News'는 10 O'clock News로 해석됩니다.

### 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.

정보 파일에서는 별표 기호를 사용할 수 없습니다. 또한 키 필드(테이블을 조인하는 데 사용되는 필드)에도 별표 기호를 사용할 수 없습니다.

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	Name
1	1234
3	1243
5	1248
7	1299

스크립트에서 위의 첫 번째 테이블이 로드되는 시점 앞에 다음 문을 삽입합니다.

```
SET OTHERSYMBOL=+;
```

*OrderID* 1299를 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.

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.

sin(NULL)      반환 값: NULL

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

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.

isnull(NULL)        반환 값: True (-1)

isnum(NULL)        반환 값: False (0)

### 산술 및 문자열 연산자

이러한 연산자의 항에 NULL이 있으면 NULL이 반환됩니다.

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.

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 서식에 대한 규칙

많은 해석 및 서식 지정 함수에서는 서식 코드를 사용하여 숫자와 날짜의 서식을 설정할 수 있습니다. 이 항목에서는 숫자, 날짜, 시간 또는 타임스탬프의 서식을 지정하는 데 사용되는 규칙을 설명합니다. 이러한 규칙은 스크립트와 차트 함수에 모두 적용됩니다.

#### 숫자 서식

- 특정 자릿수를 지정하려면 각 자리에 "0" 기호를 사용하십시오.
- 가능한 자리를 지정하려면 "#" 기호를 사용하십시오. 서식의 소수점 왼쪽에 # 기호만 있다면 1 미만의 숫자는 소수점으로 시작됩니다.
- 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.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

<b># ##0</b>	숫자를 천 단위 구분 기호가 있는 정수로 표시합니다.
<b>###0</b>	숫자를 천 단위 구분 기호가 없는 정수로 표시합니다.
<b>0000</b>	숫자를 최소 네 자리 이상의 정수로 표시합니다. 예를 들어 숫자 123은 0123으로 표시됩니다.
<b>0.000</b>	소수 자릿수 3개를 사용하여 숫자를 표시합니다.

### 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.

<b>이진수 서식</b>	이진수 서식을 표시하려면 서식 코드가 (bin) 또는 (BIN)으로 시작되어야 합니다.
<b>8진수 서식</b>	8진수 서식을 표시하려면 서식 코드가 (oct) 또는 (OCT)로 시작되어야 합니다.
<b>16진수 서식</b>	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.
<b>10진수 서식</b>	(dec) 또는 (DEC)를 사용하여 10진수 서식을 표시할 수 있지만 불필요합니다.
<b>임의의 기수 서식</b>	2~36 사이의 임의의 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.
<b>로마 숫자 서식</b>	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.

<b>num(199, '(bin)')</b>	반환 값:	11000111
--------------------------	-------	----------

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

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
num(199, '(ROM)' )	반환 값:	CXCIX

### 날짜

다음 기호를 사용하여 날짜의 서식을 지정할 수 있습니다. 임의의 구분 기호를 사용할 수 있습니다.

<b>D</b>	일을 표시하려면 각 자리에 "D" 기호를 사용하십시오.
<b>M</b>	<p>월 번호를 표시하려면 "M" 기호를 사용하십시오.</p> <ul style="list-style-type: none"> <li>한 자리 또는 두 자리의 경우, "M" 또는 "MM"을 사용하십시오.</li> <li>"MMM"은 운영 체제 또는 스크립트의 재정의 시스템 변수 <b>MonthNames</b>에 정의된 문자로 된 짧은 월 이름을 나타냅니다.</li> <li>"MMMM"은 운영 체제 또는 스크립트의 재정의 시스템 변수 <b>LongMonthNames</b>에 정의된 문자로 된 긴 월 이름을 나타냅니다.</li> </ul>
<b>Y</b>	연도를 표시하려면 각 자리에 "Y" 기호를 사용하십시오.
<b>W</b>	<p>요일을 표시하려면 "W" 기호를 사용하십시오.</p> <ul style="list-style-type: none"> <li>"W"는 한 자리의 요일 숫자(예: 월요일=0)를 반환합니다.</li> <li>"WW"는 두 자리의 숫자를 반환합니다(예: 수요일=02).</li> <li>"WWW"를 선택하면 운영 체제 또는 스크립트의 재정의 시스템 변수 <b>DayNames</b>에 정의된 요일 이름의 짧은 버전(예: Mon)이 표시됩니다.</li> <li>"WWW"를 선택하면 운영 체제 또는 스크립트의 재정의 시스템 변수 <b>LongDayNames</b>에 정의된 요일 이름의 긴 버전(예: Monday)이 표시됩니다.</li> </ul>

(날짜 예로 2013년 3월 31일 사용)

<b>YY-MM-DD</b>	날짜를 13-03-31로 표시합니다.
<b>YYYY-MM-DD</b>	날짜를 2013-03-31로 표시합니다.
<b>YYYY-MMM-DD</b>	날짜를 2013-Mar-31로 표시합니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

<b>DD MMMM YYYY</b>	날짜를 31 March 2013으로 표시합니다.
<b>M/D/YY</b>	날짜를 3/31/13으로 표시합니다.
<b>W YY-MM-DD</b>	날짜를 6 13-03-31로 표시합니다.
<b>WWW YY-MM-DD</b>	날짜를 Sat 13-03-31로 표시합니다.
<b>WWWW YY-MM-DD</b>	날짜를 Saturday 13-03-31로 표시합니다.

### 시간

다음 기호를 사용하여 시간의 서식을 지정할 수 있습니다. 임의의 구분 기호를 사용할 수 있습니다.

<b>h</b>	시간을 표시하려면 각 자리에 "h" 기호를 사용하십시오.
<b>m</b>	분을 표시하려면 각 자리에 "m" 기호를 사용하십시오.
<b>s</b>	초를 표시하려면 각 자리에 "s" 기호를 사용하십시오.
<b>f</b>	초의 분위수를 표시하려면 각 자리에 "f" 기호를 사용하십시오.
<b>tt</b>	시간을 오전/오후 형식으로 표시하려면 시간 뒤에 "tt" 기호를 사용하십시오.

(시간 예로 18시 30분 사용)

<b>hh:mm</b>	시간을 18:30으로 표시합니다.
<b>hh.mm.ss.ff</b>	시간을 18.30.00.00으로 표시합니다.
<b>hh:mm:tt</b>	시간을 06:30:pm으로 표시합니다.

### 타임스탬프

위의 날짜 및 시간 표기법과 동일한 표기법이 타임스탬프에 사용됩니다.

(타임스탬프 예로 2013년 3월 31일 18시 30분 사용):

<b>YY-MM-DD hh:mm</b>	타임스탬프를 13-03-31 18:30으로 표시합니다.
<b>M/D/Y hh.mm.ss.ffff</b>	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. QVD 파일에서 데이터를 읽는 속도는 일반적으로 다른 데이터 소스에서 데이터를 읽는 것보다 10-100배 정도 빠릅니다.

---

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

---

QVD 파일은 표준(빠름) 모드와 최적화(매우 빠름) 모드에서 읽을 수 있습니다. 모드 선택은 스크립트 엔진에서 자동으로 결정합니다. 최적화 모드는 모든 로드 파일을 아무런 변형(필드에 적용되는 수식) 없이 읽을 때만 사용할 수 있지만 필드 이름 변경은 허용됩니다. Qlik Sense에서 레코드를 압축 해제하게 만드는 **where** 절은 최적화된 로드 또한 비활성화합니다.

### QVD 파일의 용도

QVD 파일은 다양한 용도로 사용될 수 있습니다. 적어도 네 가지 중요 용도는 쉽게 확인할 수 있습니다. 어느 경우에도 다음 중 둘 이상의 용도를 적용할 수 있습니다.

<b>로드 속도 향상</b>	QVD 파일에서 입력 데이터의 불변하는 또는 느리게 변하는 블록을 버퍼링하면 큰 데이터 셋에 대한 스크립트 실행 속도가 상당히 빨라집니다.
<b>데이터베이스 서버의 부하 감소</b>	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 한번만 로드하면 됩니다. 다른 앱에서도 이 QVD 파일을 통해 동일한 데이터를 사용할 수 있습니다.
<b>여러 앱의 데이터 통합</b>	<b>binary</b> 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.
<b>증분 로드</b>	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.

1. 스크립트에서 **store** 명령을 사용하여 명시적 생성 및 명명. 이전에 읽은 테이블 또는 그 일부를 선택한 위치의 명시적으로 명명한 파일로 내보내도록 스크립트에 명시합니다.
2. 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 생성됩니다.

위의 방법으로 생성된 QVD 파일들은 읽기 속도에 차이가 없습니다.

### QVD 파일에서 데이터 읽기

다음과 같은 방법을 통해 QVD 파일을 읽거나 액세스할 수 있습니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

1. 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.

```
LOAD * from xyz.qvd (qvd);  
LOAD Name, RegNo from xyz.qvd (qvd);  
LOAD Name as a, RegNo as b from xyz.qvd (qvd);
```

2. 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.
3. 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.

1. 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.

2. 바이트 형식의 기호 테이블
3. 비트 형식의 실제 테이블 데이터

## 4.6 섹션 액세스를 사용하여 보안 관리

데이터 로드 스크립트에서 섹션 액세스를 사용하여 보안을 처리할 수 있습니다. 이 방법으로, 하나의 파일만 사용하여 많은 사용자 또는 사용자 그룹을 위해 데이터를 보관할 수 있습니다. Qlik Sense는 인증 및 권한 부여에 섹션 액세스의 정보를 사용하여 사용자만이 본인의 데이터를 볼 수 있도록 데이터의 용량을 동적으로 축소합니다.

보안이 파일 자체에 구축되어 있으므로 다운로드한 파일도 어느 정도까지는 보호됩니다. 단, 보안 요구가 높은 경우에는 파일 다운로드 및 오프라인 사용이 방지되어야 하며, 파일이 Qlik Sense 서버에서만 게시되어야 합니다. 모든 데이터가 한 파일에 보관되므로 이 파일의 크기는 매우 커질 가능성이 높습니다.



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. 제한된 데이터를 노출하지 않으려면 앱을 게시하기 전에 섹션 액세스 설정이 있는 모든 첨부 파일을 제거합니다.



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.



*Qlik Sense Cloud*에서는 현재 섹션 액세스를 지원하지 않습니다.

### 스크립트의 섹션

액세스 제어는 **Qlik Sense**가 일반적으로 데이터를 로드하는 것과 동일한 방식으로 로드한 하나 이상의 보안 테이블을 통해 관리됩니다. 따라서 이러한 테이블을 일반 데이터베이스에 저장할 수 있습니다. 보안 테이블을 관리하는 스크립트 문은 스크립트에서 **Section Access** 문으로 시작하는 액세스 섹션 내에 지정됩니다.

스크립트에 액세스 섹션이 정의되어 있는 경우 앱 데이터를 로드하는 스크립트 부분은 **Section Application** 문으로 시작하는 다른 섹션에 지정해야 합니다.

```
Section Access;  
LOAD * inline  
[ACCESS,USERID  
USER,U ];  
Section Application;  
LOAD... ... from... ...
```

### 섹션 액세스 시스템 필드

액세스 수준은 섹션 액세스 내에서 로드되는 하나 또는 여러 개의 테이블을 통해 사용자에게 할당됩니다. 이러한 테이블에는 여러 가지 사용자 관련 시스템 필드(일반적으로 **USERID** 및 액세스 수준 **ACCESS**를 정의하는 필드)가 포함될 수 있습니다. 모든 섹션 액세스 시스템 필드는 인증 또는 권한 부여에 사용됩니다. 전체 섹션 액세스 시스템 필드에 대한 설명이 아래에 나와 있습니다.

## 4 데이터 로드 스크립트를 사용하여 데이터 로드

**ACCESS** 해당 사용자에게 있어야 하는 액세스 권한을 정의합니다.

지정된 사용자 또는 사용자 그룹에 대해 Qlik Sense 앱에 대한 액세스 권한을 부여할 수 있습니다. 보안 테이블에서 사용자에게 **ADMIN** 또는 **USER** 액세스 수준을 할당할 수 있습니다. 유효한 액세스 수준이 할당되지 않은 사용자는 앱을 열 수 없습니다.

**ADMIN** 권한이 있는 사람은 앱의 모든 데이터에 액세스할 수 있습니다. **USER** 권한이 있는 사람은 보안 테이블에서 정의된 데이터에만 액세스할 수 있습니다.

**USERID** Qlik Sense 사용자 이름에 해당하는 문자열이 포함되어 있습니다. Qlik Sense는 프록시에서 로그인 정보를 가져와서 이 필드의 값과 비교합니다.

**GROUP** Qlik Sense의 그룹에 해당하는 문자열이 포함되어 있습니다. Qlik Sense는 이 그룹에 대해 프록시에서 제공되는 사용자를 해결합니다.



그룹을 사용하여 데이터를 줄이는 경우 **INTERNALISA\_SCHEDULER** 계정 사용자는 **Qlik Management Console** 작업에서 계속 스크립트를 재로드할 수 있어야 합니다.

**OMIT** 이 특정 사용자에게 대해 생략할 필드의 이름이 포함되어 있습니다. 와일드카드가 사용될 수 있으며 필드는 비워둘 수 있습니다. 이 작업을 수행하는 간편한 방법은 하위 필드를 사용하는 것입니다.



기본 데이터 구조가 변경될 수 있으므로 키 필드에 **OMIT**를 적용해서는 안 됩니다. 이는 논리적 성과 계산 비일관성을 만들어낼 수 있습니다.

Qlik Sense는 프록시에서 제공된 사용자를 **UserID**와 비교하고 테이블의 그룹에 대해 사용자를 처리합니다. 액세스가 허용되는 그룹에 사용자가 속하거나 사용자가 일치하는 경우 앱에 대한 액세스 권한을 얻게 됩니다.



섹션 액세스를 설정하여 앱에서 사용자 자신을 잠근 경우에는 데이터 없이 앱을 열고 데이터 로드 스크립트에서 액세스 섹션을 편집할 수 있습니다. 그러려면 데이터 로드 스크립트를 편집하고 다시 로드할 수 있는 액세스 권한이 있어야 합니다.

Qlik Sense의 상징인 동일한 내부 논리가 액세스 섹션에서도 사용되므로 보안 필드는 다른 테이블에 배치될 수 있습니다. 섹션 액세스의 **LOAD** 또는 **SELECT** 문에 나온 모든 필드는 대문자로 작성해야 합니다. 데이터베이스에서 소문자를 포함한 필드 이름은 **LOAD** 또는 **SELECT** 문을 사용하여 필드를 읽기 전에 **Upper** 함수를 사용하여 대문자로 변환하십시오.

와일드카드(\*)는 이 필드의 (나열된) 모든 값, 즉 이 테이블 내의 다른 위치에 나열된 값으로 해석됩니다. 스크립트의 액세스 섹션에서 로드된 테이블의 시스템 필드(**USERID**, **GROUP**) 중 하나에서 사용된 경우 이 필드에 사용 가능한(나열되지 않은) 모든 값으로 해석됩니다.



**QVD 파일에서 데이터를 로드할 경우 *upper* 함수를 사용하면 로딩 속도가 저하됩니다.**



**섹션 액세스를 활성화한 경우 여기에 나열된 섹션 액세스 시스템 필드 이름을 데이터 모델의 필드 이름으로 사용할 수 없습니다.**

이 예에서는 재무 그룹의 사용자만 문서를 열 수 있습니다.

ACCESS	GROUP
USER	Finance

### 동적 데이터 축소

**Qlik Sense**는 섹션 액세스 로그인에 따라 사용자에게 앱의 데이터 일부를 숨길 수 있는 동적 데이터 축소 기능을 지원합니다.

- 필드(열)는 시스템 필드 **OMIT**를 사용하여 숨길 수 있습니다.
- 레코드(행)는 섹션 액세스 데이터를 실제 데이터와 연결하여 숨길 수 있습니다. 표시하거나 제외할 값의 선택은 섹션 액세스 및 섹션 응용 프로그램에 공통적인 이름이 있는 하나 이상의 필드를 가짐으로써 제어됩니다. 사용자 로그인 후 **Qlik Sense**는 섹션 액세스의 필드 선택을 정확하게 동일한 필드 이름(필드 이름은 대문자로 작성되어야 함)으로 섹션 응용 프로그램의 필드에 일치시키려고 시도합니다. 선택이 완료되면 **Qlik Sense**는 이러한 선택에 의해 제외된 모든 데이터를 해당 사용자에게 영구적으로 숨깁니다.



**섹션 액세스에서는 모든 필드 이름 및 필드 값이 기본적으로 대문자로 변환되므로 위에서 설명한 전송에 사용된 모든 필드 이름과 이 필드의 모든 필드 값은 대문자여야 합니다.**



**ADMIN 액세스 권한이 있는 *INTERNAL\SA\_SCHEDULER* 계정 사용자는 *Qlik Management Console* 작업에서 스크립트를 재로드할 수 있어야 합니다.**

### 사용자 ID를 기반으로 데이터 축소

```
section access;  
LOAD * inline [  
ACCESS, USERID,REDUCTION, OMIT  
USER, 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;
```

이제 REDUCTION(대문자) 필드가 섹션 액세스 및 섹션 응용 프로그램 모두에 있습니다(모든 필드 값도 대문자임). 두 필드는 일반적으로 완전히 다르고 별개이지만 섹션 액세스를 사용할 경우 서로 연결되어 사용자에게 표시되는 레코드 수가 축소됩니다.

섹션 액세스에서 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 A는 모든 필드를 볼 수 있지만 REDUCTION=1에 연결된 레코드만 해당됩니다.
- 사용자 B는 REDUCTION=2에 연결된 레코드 중 NUM을 제외한 모든 필드를 볼 수 있습니다.
- 사용자 C는 REDUCTION=3에 연결된 레코드 중 ALPHA를 제외한 모든 필드를 볼 수 있습니다.

### 사용자 그룹을 기반으로 데이터 축소

```
section access;
LOAD * inline [
ACCESS, USERID, GROUP, REDUCTION, OMIT
USER, *, AD_DOMAIN\ADMIN, *,
USER, *, AD_DOMAIN\A, 1,
USER, *, AD_DOMAIN\B, 2, NUM
USER, *, AD_DOMAIN\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 연결된 데이터를 볼 수 있습니다.
- C 그룹에 속한 사용자는 ALPHA 필드를 제외하고 REDUCTION=3에 연결된 데이터를 볼 수 있습니다.
- GROUP1 그룹에 속한 사용자는 모든 필드에서 REDUCTION=3에 연결된 데이터를 볼 수 있습니다.
- INTERNAL\SA\_SCHEDULER 사용자는 어느 그룹에도 속하지 않지만 모든 필드에서 모든 데이터

터를 볼 수 있습니다.



이 행에서 와일드카드(\* 문자)는 **Section access** 테이블 내의 모든 값만 참조합니다. **Section access**의 **REDUCTION** 필드에서 사용할 수 없는 섹션 응용 프로그램에 값이 있는 경우에는 값이 줄어듭니다.

### 상속된 액세스 제한

이진 로드의 경우 액세스 제한이 새로운 Qlik Sense 앱에서 상속됩니다.

## 5 데이터 소스에 연결

데이터 연결은 주로 사용하는 데이터 소스에 빠르게 연결할 수 있는 방법을 제공합니다. 데이터베이스, 로컬 파일, 원격 파일, 웹 파일이 데이터 소스가 될 수 있습니다. 데이터를 선택할 때 새로운 데이터 연결을 만들거나 기존 데이터 연결을 사용할 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 스크립트를 사용하여 데이터 연결에서 데이터를 로드할 수 있습니다. 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.

### 5.1 데이터 연결 유형

**Qlik Sense**에서는 여러 유형의 데이터 소스에 액세스할 수 있는 커넥터를 제공합니다.

이러한 데이터 소스에 액세스할 수 있는 많은 커넥터가 **Qlik Sense**에서 기본 제공되며, 다른 커넥터도 추가할 수 있습니다. 각 데이터 연결 유형에는 구성해야 하는 특정한 설정이 있습니다.

#### Qlik Sense 기본 제공 커넥터

- DSN을 통한 **ODBC** 연결
- Qlik 데이터베이스 커넥터를 사용한 **ODBC** 연결
  - Apache Hive
  - Cloudera Impala
  - IBM DB2
  - Microsoft SQL Server
  - MySQL Enterprise
  - Oracle
  - PostgreSQL
  - Sybase ASE
  - Teradata
- **OLE DB** 데이터베이스 연결
- 로컬 및 네트워크 파일 폴더의 파일
- 웹 URL에 있는 웹 파일
- 개인 또는 그룹 데이터 파일에 업로드한 파일

- **첨부된 파일** 폴더는 삭제하거나 편집할 수 없으며 앱에 업로드 및 첨부된 파일이 포함되어 있습니다. (Qlik Sense Desktop에서는 사용할 수 없음)
- Qlik DataMarket(데이터 로드 편집기에서는 사용할 수 없음)
- Essbase 데이터  
[Qlik 커넥터: Essbase](#) (영어만 해당)
- REST API 연결에서 사용할 수 있습니다 Qlik Sense Cloud Business.
- Salesforce.com 연결

### 추가적인 Qlik Sense 커넥터

이러한 추가 커넥터는 [www.qlik.com](http://www.qlik.com)에서 구할 수 있습니다. 등록된 고객 또는 파트너 계정으로 로그인한 후 **지원**을 선택합니다. Qlik 지원 페이지에서 **고객 다운로드**를 선택한 후 **커넥터** 탭을 선택합니다.

- Qlik의 사용 허가를 받은 ODBC 드라이버를 통한 데이터베이스 연결
- SAP Netweaver

### 타사 커넥터

타사 커넥터를 사용하면 Qlik Sense에서 직접 지원되지 않는 데이터 소스에 연결할 수 있습니다. 타사 커넥터는 QVX SDK를 사용하여 개발하거나 타사 개발자에 의해 제공됩니다. 표준 Qlik Sense 설치에서는 타사 커넥터를 사용할 수 없습니다.

## 5.2 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 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.



**Qlik Sense Desktop** 연결에 해당 연결을 만들 때 입력한 사용자 이름, 비밀번호 및 파일 경로에 대한 모든 상세 정보가 저장되기 때문에, 앱을 다른 사용자와 공유하는 경우 이들 저장된 상세 정보를 일반 텍스트로 사용할 수 있게 됩니다. 앱을 공유하기 위해 설계하는 경우 이를 고려해야 합니다.

## 5.3 제한 사항

데이터 연결의 이름을 'DM'으로 지정할 수는 없습니다. 이 이름은 기본 제공된 Qlik DataMarket 커넥터에 의해 예약되었습니다.

## 5.4 파일에서 데이터 로드

Qlik Sense에서는 다양한 형식의 파일에서 데이터를 읽을 수 있습니다.

- 텍스트 파일. 이 파일에는 필드의 데이터가 쉼표, 탭, 세미콜론 등의 구분 기호로 구분되어 있습니다(쉼표로 구분된 변수(CSV) 파일).
- HTML 테이블.
- Excel 파일(비밀번호로 보호된 Excel 파일 제외).
- XML 파일.
- Qlik 네이티브 QVD 및 QVX 파일.
- 고정 레코드 길이 파일.
- DIF 파일(Data Interchange Format). DIF 파일은 데이터 로드 편집기에서만 로드할 수 있습니다.

다양한 데이터 연결 유형에서 파일을 로드할 수 있습니다.

- 로컬 및 네트워크 파일 폴더
- **첨부된 파일** 폴더는 삭제하거나 편집할 수 없으며 앱에 업로드 및 첨부된 파일이 포함되어 있습니다. (Qlik Sense Desktop에서는 사용할 수 없습니다.)
- 웹 URL에 위치한 파일

### 파일에서 데이터를 로드하는 방법

여러 가지 방법으로 파일에서 데이터를 로드할 수 있습니다.

- **데이터 추가**를 통해 데이터를 추가하는 것이 파일에서 데이터를 로드하는 가장 빠른 방법입니다. 기존 데이터 연결에서 데이터를 로드하거나 즉석에서 새 데이터 소스에 연결할 수 있습니다.
- 데이터 로드 편집기에서 데이터 연결로부터 데이터 선택.  
데이터 로드 편집기에 수동으로 문을 입력하는 대신 **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.

### 폴더 데이터 연결 설정

폴더 연결을 사용하여 로컬 및 네트워크 파일 폴더에서 파일을 로드할 수 있습니다.

설정	설명
경로	<p>데이터 파일이 포함된 폴더의 경로입니다. 다음 중 하나를 수행할 수 있습니다.</p> <ul style="list-style-type: none"> <li>폴더 선택</li> <li>올바른 로컬 경로 입력</li> </ul> <p><b>C:\data\MyData\</b></p> <ul style="list-style-type: none"> <li>UNC 경로 입력</li> </ul> <p><b>\\myserver\filedir\</b></p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  매핑된 네트워크 드라이브를 경로에 사용할 수는 없습니다. </div>
이름	데이터 연결 이름입니다.

## 웹 파일 데이터 연결 설정

웹 파일 데이터 연결을 사용하여 FTP, HTTP 또는 HTTPS 등의 웹 리소스에서 파일을 로드할 수 있습니다. Qlik Sense에서 지원하는 모든 유형의 파일을 사용 가능합니다.

설정	설명
URL	<p>프로토콜 식별자를 포함하여, 연결할 웹 파일에 대한 전체 <b>URL</b>입니다.</p> <p><b>http://unstats.un.org/unsd/demographic/products/socind/Dec.%202012/1a.xls</b></p> <p>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.</p>
이름	데이터 연결 이름입니다.

## Microsoft Excel 스프레드시트에서 데이터 로드

Qlik Sense에서는 Microsoft Excel 스프레드시트에서 데이터를 읽을 수 있습니다. 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 컨케이트네이트되는 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 Sense**에서 테이블이 되며, **Qlik Sense** 앱의 시트가 되지 않습니다.

스프레드시트를 로드하기 전에 **Microsoft Excel**에서 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.

- 필드 이름**      테이블에 포함된 필드 이름이 있는지, 또는 필드 이름 없음이 있는지 지정합니다. **Excel** 스프레드시트에서는 일반적으로 첫 행에 포함된 필드 이름이 있습니다. 필드 이름 없음을 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.

<b>Machine:</b>	AEJ12B		
<b>Date:</b>	2015-10-05 09		
<b>Timestamp</b>	<b>Order</b>	<b>Operator</b>	<b>Yield</b>
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.

- 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 읽은 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.

## 5.5 데이터베이스에서 데이터 로드

다음 커백터를 통해 상용 데이터베이스 시스템의 데이터를 Qlik Sense로 로드할 수 있습니다.

- Microsoft ODBC 인터페이스 또는 OLE DB를 사용하는 커넥터입니다. ODBC를 사용하려면, DBMS를 지원하는 드라이버를 설치해야 하며 **Windows 제어판의 ODBC 데이터 원본 관리자**에서 데이터베이스를 Microsoft ODBC 데이터 소스로 구성해야 합니다.
- 커넥터는 DSN 연결 없이, 사용 허가된 ODBC 드라이버를 통해 데이터베이스에서 데이터를 직접 로드할 수 있도록 특별히 개발되었습니다.

### 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.

Qlik 사용권이 있는 ODBC 드라이버는 다음과 같은 데이터베이스를 지원합니다.

- Apache Hive
- Cloudera Impala
- IBM DB2
- Microsoft SQL Server
- MySQL Enterprise
- Oracle
- PostgreSQL
- Sybase ASE
- Teradata

Microsoft ODBC 인터페이스를 사용하려면 다음을 수행합니다.

1. 액세스하려는 데이터베이스에 대한 ODBC 데이터 소스가 있어야 합니다. 이 데이터 소스는 **Windows 제어판의 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 연결에서 **±**를 클릭하여 데이터 선택 대화 상자를 엽니다.

이제 데이터베이스에서 데이터를 선택하고 데이터를 로드하는 데 필요한 스크립트 코드를 삽입할 수 있습니다.

### ODBC

Qlik Sense에서 ODBC를 통해 DBMS(Database Management System)에 액세스하려면 두 가지 옵션이 있습니다.

- 해당하는 DBMS용 ODBC 드라이버를 설치하고 데이터 소스 DSN을 만듭니다. 여기에 대해서는 이 섹션에서 설명합니다.
- Qlik Sense에서 Microsoft Windows **ODBC 데이터 원본 관리자**를 사용할 필요 없이 데이터 소스를 정의할 수 있는 Qlik ODBC Connector Package의 **Database** 커넥터를 사용합니다. ODBC

Connector Package에서 Qlik 사용 허가된 ODBC 드라이버 중 하나를 통해 데이터베이스에 직접 연결하려면 Qlik Connectors 도움말 사이트에서 Database 커백터에 대한 지침을 참조하십시오.



**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 커백터에 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.

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

설정	설명
사용자 DSN 시스템 DSN	연결할 DSN 유형을 선택합니다. 사용자 DSN 소스의 경우, <b>32비트 연결 사용</b> 을 통해 32비트 드라이버를 사용할지 여부를 지정해야 합니다. 시스템 DSN 연결은 <b>32비트</b> 또는 <b>64비트</b> 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 <b>사용자 이름</b> 및 <b>비밀번호</b> 로 자격 증명을 지정하지 않는 한 엔진 서비스 사용자 자격 증명에 사용됩니다. 이 옵션을 선택하면 엔진 서비스 사용자 또는 <b>사용자 이름 / 비밀번호</b> 자격 증명에 Windows 로그인에 사용되며 뒤이어 현재 사용자 자격 증명을 사용하여 SAML(SAP HANA)에 대한 후속 로그인이 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.

설정	설명
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.

### Single Sign-On 로그인 자격 증명

이 테이블에서는 SAP HANA 데이터 소스에 연결할 때 선택한 **싱글 사인온**과 여러 로그인 자격 증명을 조합한 결과를 설명합니다.

데이터베이스 소스	사용자 이름/비밀번호	설명
SAP HANA	-	엔진 서비스 사용자 자격 증명에 <b>Windows</b> 로그인에 사용된 후 현재 사용자 자격 증명에 <b>SAML</b> 로그인에 사용됩니다.
SAP HANA	채워짐	<b>사용자 이름/비밀번호</b> 자격 증명에 <b>Windows</b> 로그인에 사용된 후 현재 사용자 자격 증명에 <b>SAML</b> 로그인에 사용됩니다.

### ODBC 드라이버 추가

Qlik Sense에서 사용자의 데이터베이스에 액세스할 수 있도록 하려면 사용자 DBMS(DataBase Management System)에 적합한 ODBC 드라이버를 설치해야 합니다. 사용 중인 DBMS에 대한 자세한 내용은 설명서를 참조하십시오.

### 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. 데이터 소스 이름을 지정하고 필요한 파라메타를 설정합니다.
6. **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 포함되어 있으면 관련 ODBC 데이터 소스가 새로운 배포에도 존재해야 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.

### 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

설정	설명
공급자	사용 가능한 공급자 목록에서 <b>공급자</b> 를 선택합니다. 새 연결을 만들 때만 사용할 수 있습니다.

설정	설명
데이터 소스	<p>연결할 데이터 소스의 이름을 입력합니다. 이 이름은 서버 이름이거나, 경우에 따라 데이터베이스 파일의 경로일 수 있습니다. 이 이름은 사용 중인 <b>OLE DB</b> 공급자에 따라 달라집니다. 새 연결을 만들 때만 사용할 수 있습니다.</p> <p>Microsoft Office 12.0 Access Database Engine OLE DB Provider를 선택한 경우, 전체 파일 경로를 포함하여 Access 데이터베이스 파일의 이름을 입력합니다.</p> <p><i>C:\Users\{user}\Documents\Qlik\Sense\Apps\Tutorial source files\Sales.accdb</i></p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  데이터 소스에 대한 연결에 실패하면 경고 메시지가 표시됩니다.         </div>
연결 문자열	<p>데이터 소스에 연결할 때 사용할 연결 문자열입니다. 이 연결 문자열에는 <b>공급자</b> 및 <b>데이터 소스</b>에 대한 참조가 포함되어 있습니다. 연결을 편집할 때만 사용할 수 있습니다.</p>
Windows 통합 보안	<p>이 옵션을 사용하면 Qlik Sense 서비스를 실행하는 사용자의 기존 Windows 자격 증명을 사용할 수 있습니다.</p>
특정 사용자 이름 및 비밀번호	<p>이 옵션을 사용하면 데이터 소스 로그인 자격 증명에 대해 <b>사용자 이름</b> 및 <b>비밀번호</b>를 입력해야 합니다.</p>
사용자 이름	<p>데이터 소스에서 필요한 경우, 연결할 사용자 이름입니다.</p> <p><b>Windows 통합 보안</b>을 사용하는 경우 또는 데이터 소스에서 자격 증명을 필요로 하지 않는 경우는 이 필드를 비워둡니다.</p>
비밀번호	<p>데이터 소스에서 필요한 경우, 연결할 비밀번호입니다.</p> <p><b>Windows 통합 보안</b>을 사용하는 경우 또는 데이터 소스에서 자격 증명을 필요로 하지 않는 경우는 이 필드를 비워둡니다.</p>
로드 데이터베이스 선택...	<p>연결을 테스트하려면 <b>로드</b>를 클릭한 다음 <b>데이터베이스 선택...</b>을 클릭하여 데이터 연결에 사용할 데이터베이스를 선택합니다.</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  데이터 연결에서 데이터를 선택할 때 다른 사용 가능한 데이터 소스의 데이터베이스를 모두 사용할 수 있습니다.         </div>
이름	<p>데이터 연결 이름입니다.</p>

### 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.

### 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.

## 5.6 Qlik DataMarket에서 데이터 로드

Qlik DataMarket에서 외부 소스로부터 데이터를 추가할 수 있습니다. Qlik DataMarket은 Qlik Sense 내에서 직접 액세스할 수 있는 외부 소스에서 사용할 준비가 된 최신 데이터에 대한 확장 모음을 제공합니다. Qlik DataMarket은 현재 및 이전 날씨와 인구 통계 데이터, 환율은 물론, 비즈니스, 경제 및 사회적 데이터도 제공합니다.

또한 Qlik DataMarket은 Eurostat 데이터베이스에서 Database by themes, Tables by themes, Tables on EU policy, Cross cutting topics 등의 데이터 집합을 제공합니다.

일부 Qlik DataMarket 데이터는 무료로 사용할 수 있습니다. **프리미엄**으로 표시된 데이터 패키지는 구독 요금을 내면 사용할 수 있습니다.

Qlik DataMarket 데이터를 사용하려면 먼저 사용 약관에 동의해야 합니다. 또한 프리미엄 데이터 패키지에 대한 라이선스를 구매한 경우 해당 패키지의 데이터를 사용하려면 액세스 자격 증명을 입력해야 합니다. 액세스 자격 증명에 적용되면 프리미엄 데이터는 **사용 허가됨**으로 레이블이 지정됩니다.

약관에 동의하고 어떠한 프리미엄 데이터 패키지에 대한 라이선스도 입력하지 않는 경우, 프리미엄 패키지 옆에 라이선스를 구입할 수 있는 **구매** 버튼이 표시됩니다. **구매** 버튼은 **프리미엄** 레이블을 대체합니다.



*Qlik Sense Desktop을 사용할 때는 Qlik DataMarket 사용 약관에 동의할 필요가 없습니다. Qlik Sense Desktop에서는 프리미엄 데이터 셋을 사용할 수 없으므로 액세스 자격 증명도 필요 없습니다.*

일부 Qlik DataMarket 데이터는 Qlik Sense Cloud Basic 계정을 통해 무료로 사용할 수 있습니다. Qlik Sense Cloud Plus 계정으로 업그레이드하면 추가적인 데이터를 사용할 수 있습니다.



**DataMarket** 사용자 인터페이스는 광고를 차단하고 개인정보 보호를 강화하는 **Privacy Badger**와 같은 브라우저 확장 기능에 의해 차단될 수 있습니다. 해당 확장 기능이 타사의 사용자 추적에 대한 **DataMarket** 통신에서 오류가 발생한 경우에 이런 문제가 발생합니다. 이런 경우, **DataMarket**을 차단하는 브라우저 확장 기능의 차단된 사이트 목록에서 **Qlik Sense** 사이트를 제외시키면 **DataMarket**에 액세스할 수 있습니다.

Qlik DataMarket 데이터는 별도로 실험하거나 자체 데이터와 통합할 수 있습니다. Qlik DataMarket에서 내부 데이터가 증가하면 더욱 풍부한 탐색이 가능합니다.

Qlik DataMarket 데이터는 이 데이터가 파생된 소스에서 현재 상태입니다. 소스 데이터의 업데이트 빈도는 유동적입니다. 날씨와 시장 데이터는 일반적으로 하루에 최소 한 번 업데이트되는 반면, 공공 인구 통계 데이터는 대개 연 단위로 업데이트됩니다. 실업률, 가격 지수 및 무역 등과 같은 대부분의 거시 경제 지표가 매월 게시됩니다. 모든 업데이트는 일반적으로 동일한 날짜에 Qlik DataMarket에서 사용할 수 있습니다.

Qlik Sense에서 선택한 데이터는 영구적이므로 데이터 모델을 다시 로드할 때마다 Qlik DataMarket에서 사용 가능한 최신 데이터로 로드됩니다.

대부분의 Qlik DataMarket 데이터는 전 세계 및 국가별로 모두 제공됩니다. 예를 들어, 전 세계 인구 데이터는 200개 이상의 국가 및 지역에 대해 제공됩니다. 또한 Qlik DataMarket에서는 미국 및 유럽 국가 내 주 및 지역에 대한 다양한 데이터를 제공합니다.

### 5.7 Direct Discovery를 사용하여 대규모 데이터 셋에 액세스

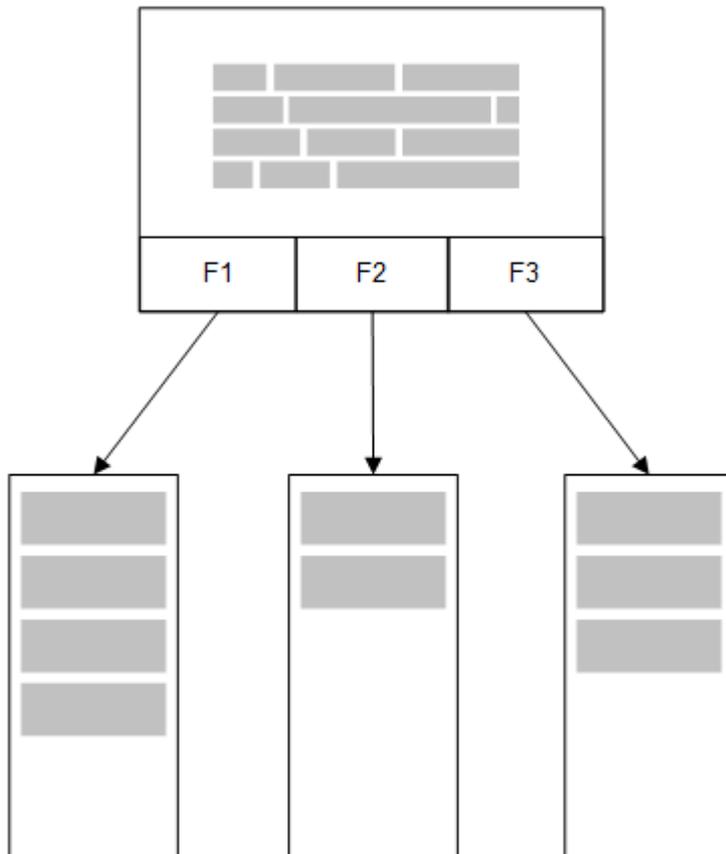
Direct Discovery는 인 메모리 데이터와 더 큰 데이터 셋을 매끄럽게 연결하는 집계 쿼리를 통해 추가 소스 데이터에 대한 액세스를 제공함으로써 Qlik Sense 인 메모리 데이터 모델의 연관 기능을 확장합니다. Direct Discovery는 현업 사용자가 제한 없이 빅 데이터 소스 연관 분석을 수행할 수 있는 역량을 강화합니다. 인 메모리 및 Direct Discovery 데이터를 선택하여 Qlik Sense의 연결 색과 같은 녹색, 흰색, 회색으로 데이터 셋 간의 연결을 확인할 수 있습니다. 시각화는 두 데이터 셋의 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.

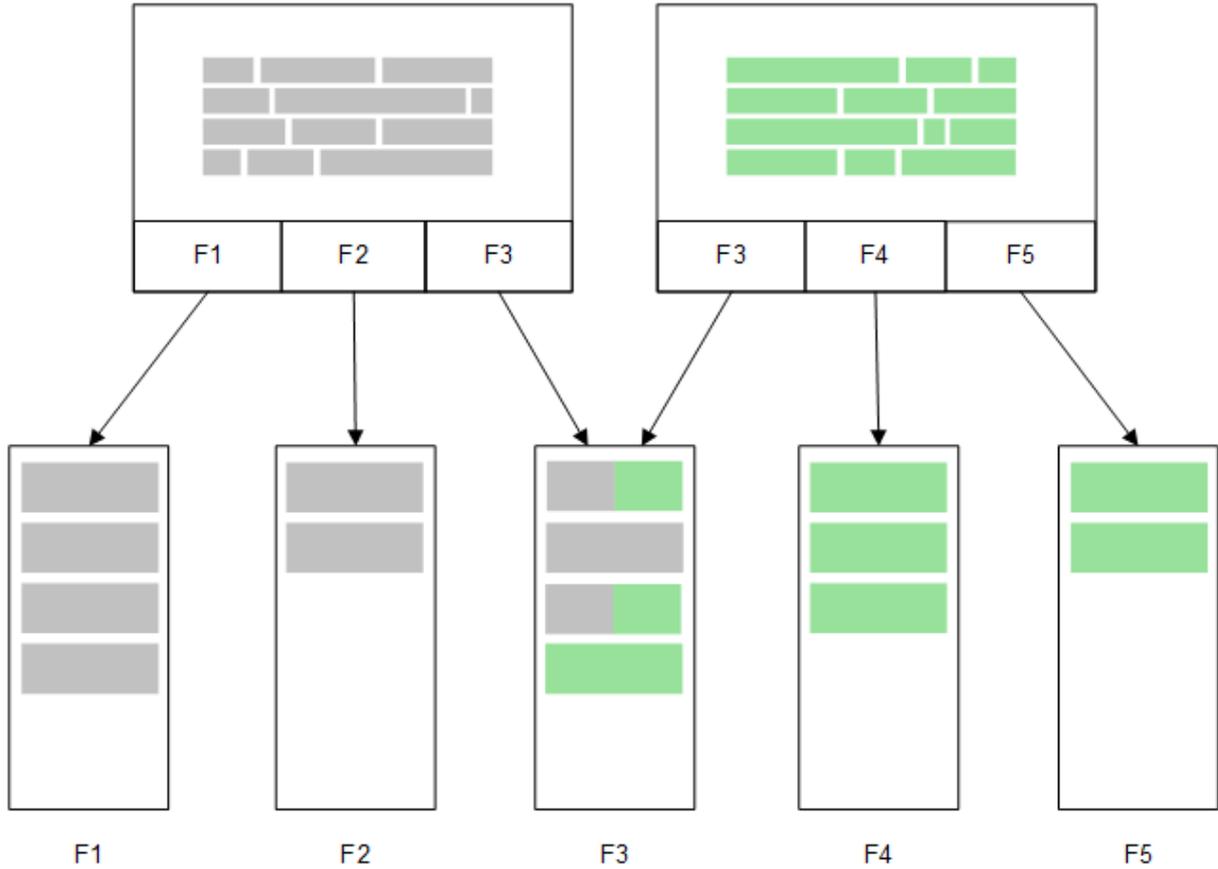
## Direct Discovery와 인 메모리 데이터의 차이점

### 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.

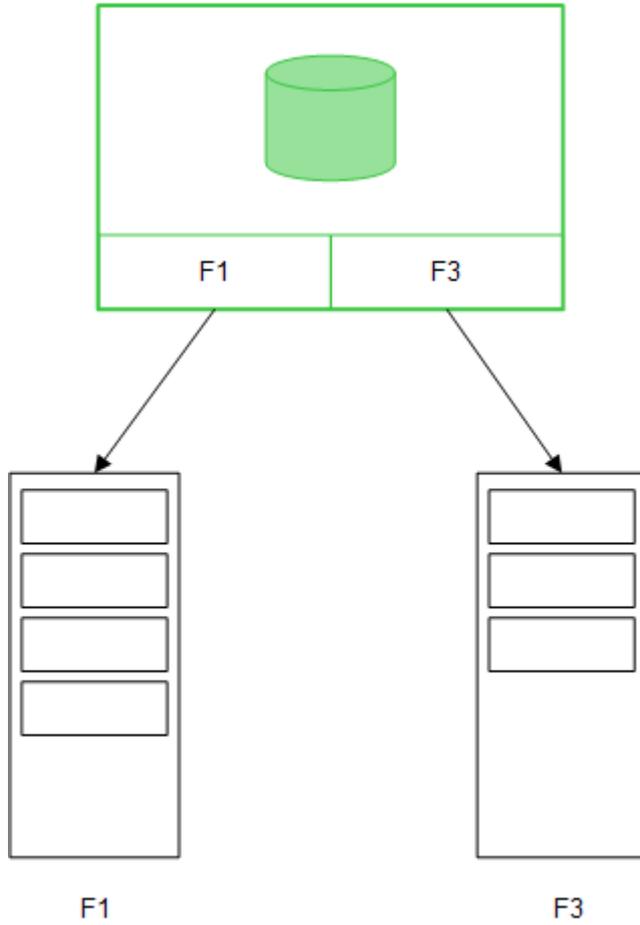


두 번째 관련 테이블이 메모리에 로드되면 공통 필드를 공유하게 되고, 해당 테이블이 새 고유 값을 공통 필드에 추가하거나 기존 값을 공유할 수 있습니다.

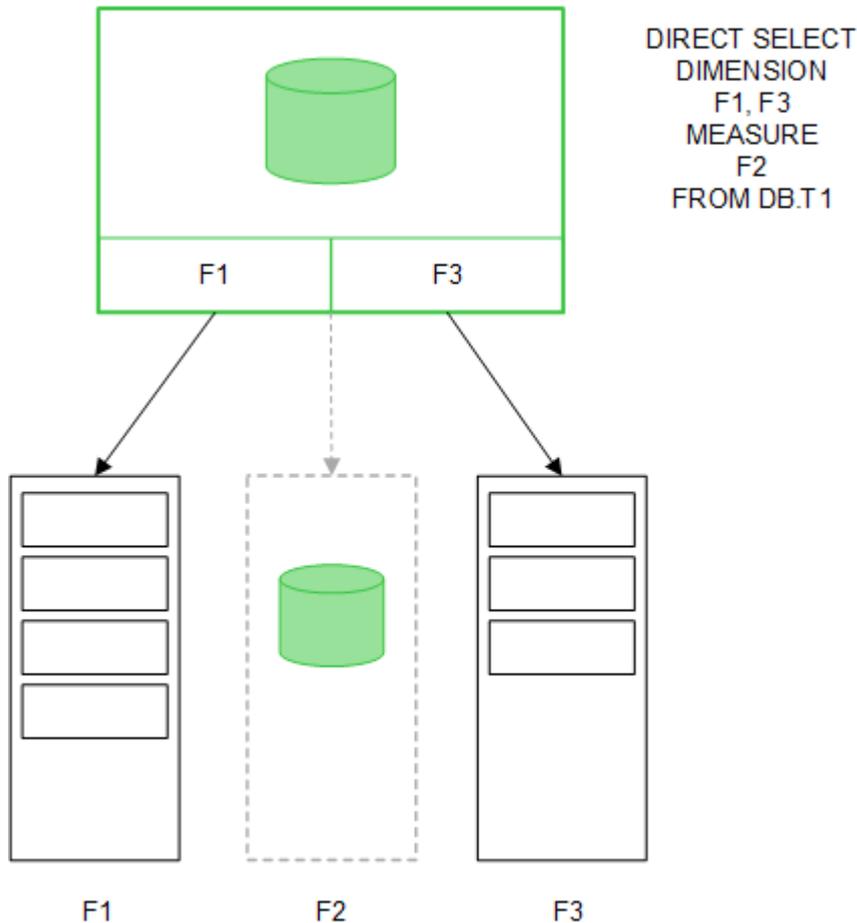


### Direct Discovery

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*(143페이지)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 QUERY**는 인 메모리 데이터와 연결할 때 대/소문자를 구분합니다. Direct Discovery는 데이터베이스의 대/소문자 구분 여부에 따라 소스 데이터베이스에서 데이터를 선택합니다. 데이터베이스가 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.

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.

ColumnA	ColumnB
'1'	space_before
'1'	no_space
'1 '	space_after
'2'	two

표준 Qlik Sense 인 메모리에 데이터가 있는 **필터 창**에서 ColumnA에 대해 "1"을 선택하면 처음 3개 행이 연결됩니다.

'	space_before
'1'	
'1'	no_space
'1'	space_after
'	

**필터 창**에 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 상태를 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.

**DirectCacheSeconds** 시스템 변수를 사용하면 캐싱에 시간 한계를 설정할 수 있습니다. 시간 한계에 도달하면 Qlik Sense가 캐시에서 이전 선택 내용에 대해 생성된 **Direct Discovery** 쿼리 결과를 지웁니다. 그리고 Qlik Sense가 소스 데이터에서 선택 내용을 쿼리하고 지정된 시간 한계에 대한 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 할 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. **DIMENSION** 필드는 개체 표현식에 사용할 수 없습니다.

아래 테이블에는 **Direct Discovery** 필드 유형의 특성과 사용법이 요약되어 있습니다.

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 Discovery** DIMENSION 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 집계 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. **DETAIL** 필드는 **Count** 집계에서만 사용할 수 있으며, **테이블**에서만 볼 수 있습니다. **DETAIL**로 지정된 필드에는 일반적으로 주석과 같이 의미 있는 방식으로 집계할 수 없는 데이터가 포함됩니다.

어느 필드나 **DETAIL** 필드로 지정할 수 있습니다.

## Direct Discovery에서 지원하는 데이터 소스

Qlik Sense Direct 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 소스가 지원됩니다.
- 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 열을 쉼표가 아니라 공백으로 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 및 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 PostgreSQL과 함께 사용할 수 있지만, 이 소스에서 사용된 인용 문자에 따라 **set** 변수에 다음과 같은 파라메타가 필요할 수 있습니다.

```
SET DirectIdentifierQuoteChar='\"';
```

### Sybase 및 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='[]';
```

## Direct Discovery를 사용할 때의 제한 사항

### 지원되는 데이터 유형

Direct Discovery에서는 모든 데이터 유형이 지원되지만 Qlik Sense에서 특정 소스 데이터 서식을 정의해야 하는 경우도 있을 수 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.
- 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** 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.

## Direct Discovery에서 다중 테이블 지원

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.  
참조: *Direct Discovery에서 하위 쿼리 사용하기(149페이지)*
- 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.

```

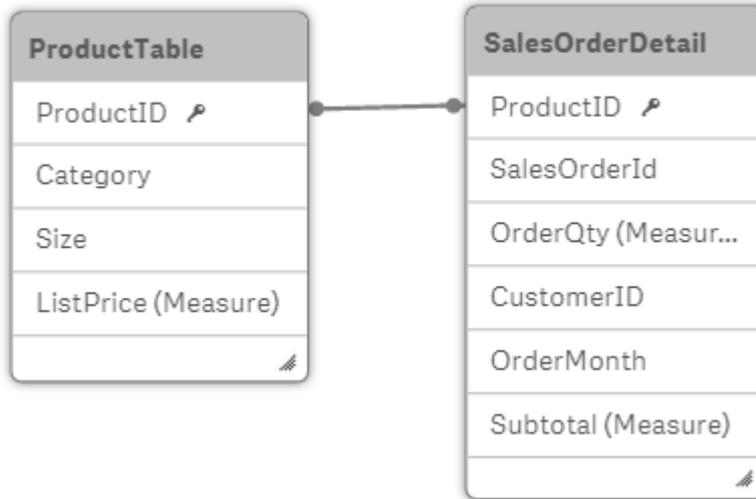
Sales_Order_Header_Join:
DIRECT QUERY
DIMENSION
    AW2012.Sales.Customer.CustomerID as CustomerID,
    AW2012.Sales.SalesOrderHeader.SalesPersonID as SalesPersonID,
    AW2012.Sales.SalesOrderHeader.SalesOrderID as SalesOrderID,
    ProductID,
    AW2012.Sales.Customer.TerritoryID as TerritoryID,
    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);

```

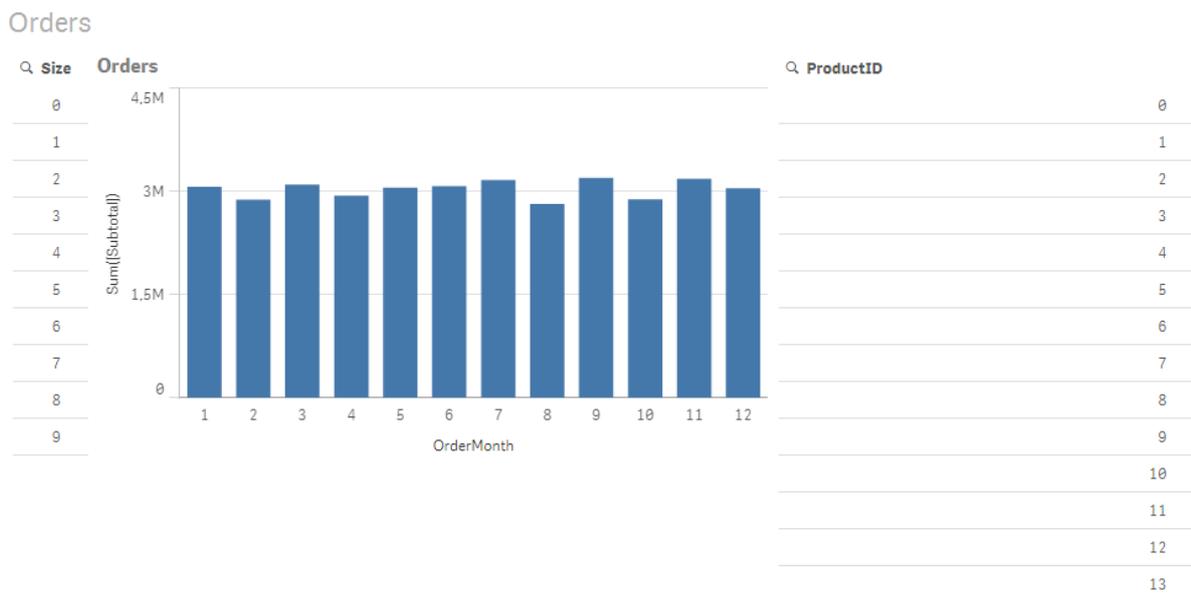
## Direct Discovery에서 하위 쿼리 사용하기

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.



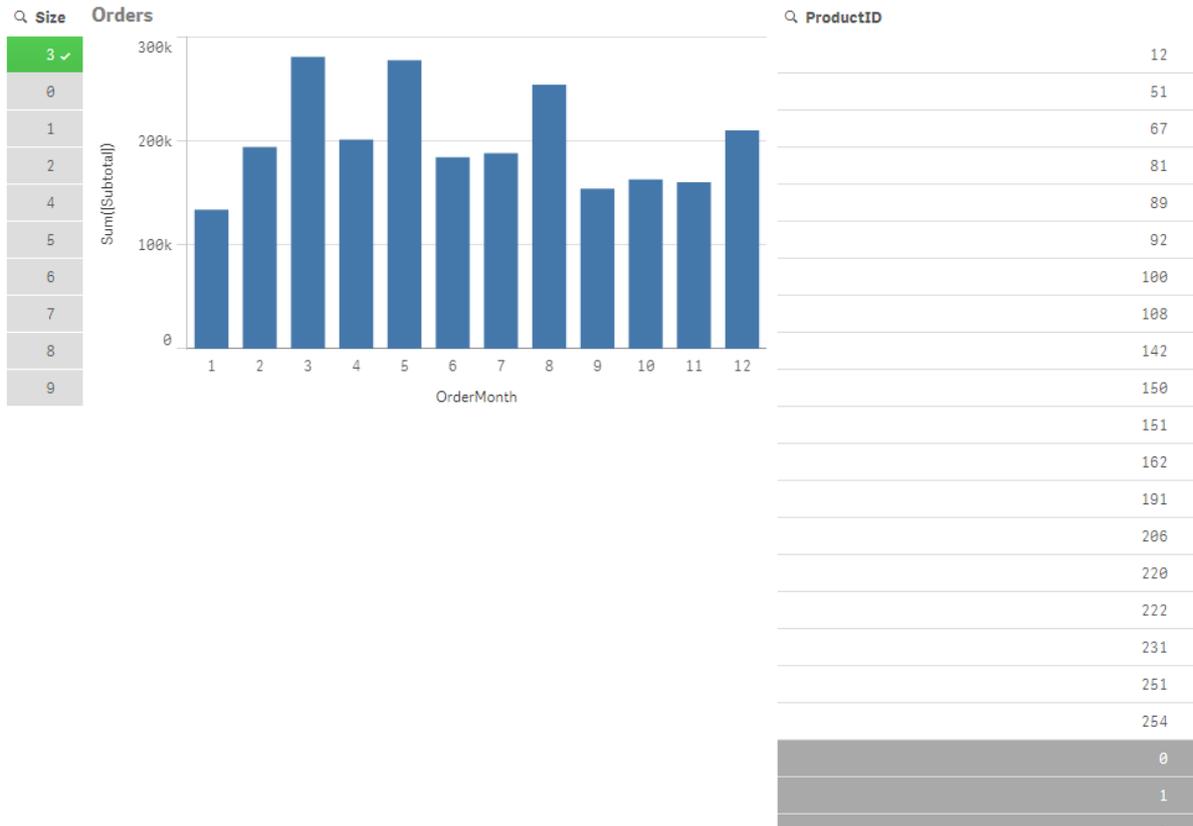
OrderMonth를 차원으로, Sum(Subtotal)을 측정값으로 사용하는 차트 및 Size를 선택하기 위한 필터 상자를 만듭니다.



## 시나리오 1: 집합 크기 작음

이 시나리오의 경우 제품 테이블에 포함된 고유 제품의 수가 266개로, 적습니다. **Size**에서 선택을 하면 **Direct Discovery**는 선택한 크기(이 경우 19개 제품)와 일치하는 제품 ID가 포함된 **WHERE ProductID IN** 절을 사용하여 데이터를 반환하기 위한 **SQL** 문을 생성합니다.

### Orders



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 **Size**를 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.

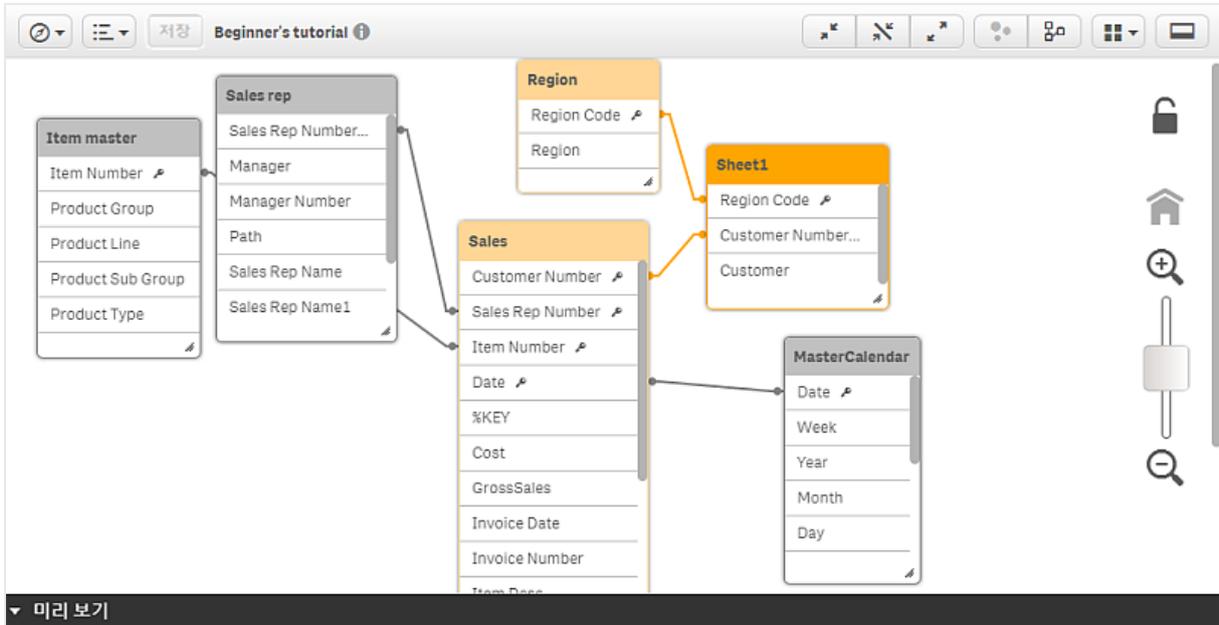
해결책은 **DirectEnableSubquery**를 true로 설정하여 Qlik Sense에서 대신 하위 쿼리를 만들도록 하는 것입니다. 생성된 **SQL** 문은 다음과 같습니다.

```
SELECT ProductID, month([OrderDate]), SUM(OrderQty), SUM(SubTotal)
FROM SalesTable
```



## 6 데이터 모델 보기

데이터 모델 뷰어는 앱의 데이터 구조에 대한 개요를 제공합니다. 데이터 모델 뷰어에서 테이블 및 필드의 데이터를 미리 볼 수 있습니다. 또한 즉석에서 차원 및 측정값을 만들 수도 있습니다.



데이터 모델 뷰어에서 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.

Y, Z 를 클릭하거나 슬라이더를 사용하여 확대/축소 수준을 변경할 수 있습니다. 확대/축소 수준을 1:1로 복원하려면 ü 를 클릭합니다.

### 6.1 도구 모음

데이터 모델 뷰어의 화면 맨 위에 있는 도구 모음에는 다음과 같은 도구가 있습니다.

”	<p>다음 옵션이 포함된 탐색 메뉴:</p> <ul style="list-style-type: none"> <li>ü 앱 개요</li> <li>, 데이터 관리자</li> <li>Ï 데이터 로드 편집기</li> <li>^ 허브 열기</li> </ul> <p>허브가 관리자에 의해 비활성화된 경우 <b>허브 열기</b>가 표시되지 않습니다.</p>
---	--

..	다음 옵션이 포함된 메뉴:  « 데이터 추가  D 도움말  ) 정보
저장	변경 사항을 저장합니다.
⊞	선택하여 앱 정보를 편집하거나 앱 옵션을 열고 앱 스타일을 지정할 수 있는 앱 정보를 표시하거나 숨깁니다.
t	테이블 이름만 표시되도록 모든 테이블을 축소합니다.
â	테이블 이름 및 다른 테이블에 대한 연결이 있는 모든 필드를 표시하도록 모든 테이블의 크기를 줄입니다.
s	모든 필드가 표시되도록 모든 테이블을 확장합니다.
+	내부 테이블 보기 - 가상 필드가 포함된 Qlik Sense 데이터 모델입니다.
7	소스 테이블 보기 - 소스 데이터 테이블의 데이터 모델입니다.
ì	다음 옵션이 포함된 레이아웃 메뉴:  ì 그리드 레이아웃  ó 자동 레이아웃  õ 레이아웃 복원
j	미리 보기 창을 열고 닫습니다.

## 6.2 데이터 모델 뷰어에서 테이블 이동 및 크기 조정

### 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 ì 아래에서 옵션을 사용하여 레이아웃을 자동으로 배열할 수 있습니다.

- ì 그리드 레이아웃 - 그리드에 테이블을 배치합니다.
- ó 자동 레이아웃 - 창에 맞도록 테이블을 배치합니다.
- õ 레이아웃 복원 - 데이터 모델 뷰어를 마지막으로 열었을 때의 레이아웃 상태로 복원합니다.

## 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.

**t 모두 축소** - 테이블 이름만 표시되도록 모든 테이블을 최소화합니다.

**å 연결된 필드 표시** - 테이블 이름 및 다른 테이블에 대한 연결이 있는 모든 필드를 표시하도록 모든 테이블의 크기를 줄입니다.

**s 모두 확장** - 테이블의 모든 필드가 표시되도록 모든 테이블을 최대화합니다.

## 6.3 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 신속하게 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 **j** 을 클릭합니다.
- **미리 보기** 헤더를 클릭합니다.



*Direct Discovery* 데이터는 미리 보기 **Å** 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.

미리 보기					
<b>Item master</b>			<b>Preview of data</b>		
Rows	827	Item Number	Product Group	Product Line	Product Sub Group
Fields	5	10001	Beverages	Drink	Juice
Keys	1	10002	Beverages	Drink	Flavored Drinks
Tags	\$key, \$numeric, \$integer\$ascii, \$text	10003	Beverages	Drink	Flavored Drinks
		10004	Beverages	Drink	Soda
		10005	Beverages	Drink	Soda
					Carbonated Beverages

## 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.

Product Group		Preview of data			
Density	100%	Item Number	Product Group	Product Line	Product Sub Group
Subset ratio	100%	10001	Beverages	Drink	Juice
Has duplicates	true	10002	Beverages	Drink	Flavored Drinks
Total distinct values	15	10003	Beverages	Drink	Flavored Drinks
Present distinct values	15	10004	Beverages	Drink	Soda
Non-null values	827	10005	Beverages	Drink	Soda
Tags	\$ascii, \$text				

- **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.

## 6.4 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 및 태그를 추가합니다.
4. **차원 추가**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 **A**로 표시됩니다.

### 6.5 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.

## 7 데이터 모델링의 모범 사례

이 섹션에서는 데이터의 구조와 연결하려는 데이터 모델에 따라 Qlik Sense 앱에 데이터를 로드할 수 있는 다양한 방법에 대해 설명합니다.

### 7.1 데이터 열을 행으로 변환

다음과 같은 데이터가 있는 경우 별도의 필드에 판매 수치를 표시하고 싶습니다.

Year	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.

Year	Quarter	Sales
2013	Q1	34
2013	Q2	54
2013	Q3	53
2013	Q4	52
2014	Q1	47
...	...	...

### 7.2 데이터 행을 필드로 변환

아래와 비슷한 세 개의 필드가 포함된 일반적인 테이블이 있는 경우 각 특성을 개별 테이블로 표시하고 싶습니다.

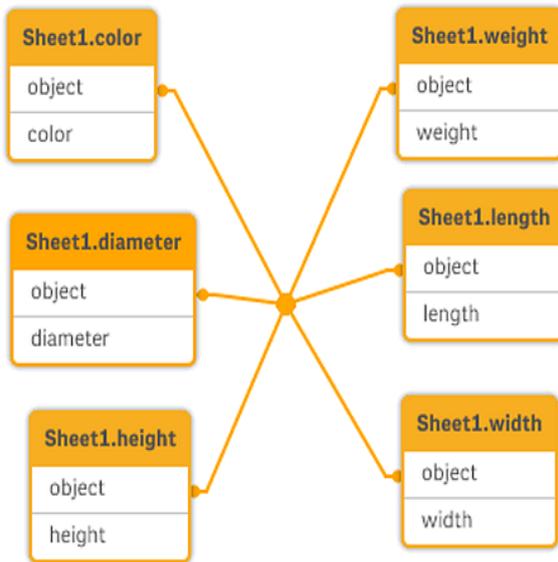
Object	Attribute	Value
ball	color	red
ball	diameter	25
ball	weight	3
box	color	56

Object	Attribute	Value
box	height	30
box	length	20
box	width	25

제안 조치

**Generic** 로드 점두사를 사용하여 일반 데이터 모델을 만듭니다.

다음과 같은 데이터 모델을 얻게 됩니다.



### 7.3 계층 수준으로 구성된 데이터(예: 조직 체계) 로드

다음과 같이 인접한 노드 테이블에 데이터가 저장되어 있습니다.

NodeID	ParentNodeID	Title
1	-	General manager
2	1	Country manager
3	2	Region manager

제안 조치

**Hierarchy** 점두사를 통해 데이터를 로드하여 확장 노드 테이블을 만듭니다.

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

## 7.4 대규모 데이터베이스에서 신규 또는 업데이트된 레코드만 로드

대량의 레코드가 포함된 데이터베이스가 있는 경우 앱의 데이터를 갱신하기 위해 전체 데이터베이스를 다시 로드하고 싶지 않습니다. 신규 또는 업데이트된 레코드만 로드하고 데이터베이스에서 삭제된 레코드는 제거하고 싶습니다.

### 제안 조치

QVD 파일을 사용하여 증분 로드 솔루션을 구현하십시오.

## 7.5 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.

## 7.6 간격에 불연속 값 일치

불연속 숫자 값 테이블(Event)이 있고 이를 하나 이상의 간격(Start 및 End)에 일치시키고 싶습니다.

Time	Event	Comment	Start	End	Order
00:00	0	Start of shift 1			
01:18	1	Line stop	01:00	03:35	A
02:23	2	Line restart 50%	02:30	07:58	B
04:15	3	Line speed 100%	03:04	10:27	C
08:00	4	Start of shift 2	07:23	11:43	D
11:43	5	End of production			

### 제안 조치

**IntervalMatch** 접두사를 사용하여 **Time** 필드를 **Start** 및 **End**에 의해 정의된 간격과 연결하십시오.

시작 및 종료로 간격이 명확하게 정의되어 있지 않고 아래 테이블에서처럼 변경 타임스탬프만 있는 경우, 간격 테이블을 만들어야 합니다.

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

## 7.7 일관성 없는 필드 값 처리

여러 테이블에 일관성 없이 이름이 지정된 필드 값이 포함된 데이터가 있습니다. 예를 들어 한 테이블에는 **Country**에 **US** 값이 있고 다른 테이블에는 **United States**이 포함되어 있습니다. 이런 경우 연결이 수행되지 않습니다.

Table 1		Table 2	
Country	Region	Country	Population
US	Maryland	United States	304
US	Idaho	Japan	128
US	New York	Brazil	192
US	California	China	1333

### 제안 조치

매핑 테이블을 사용하여 데이터 정리 작업을 수행하십시오. 그러면 필드 값을 비교하여 올바른 연결을 수행할 수 있습니다.

## 7.8 일관성 없는 필드 값 대소문자 처리

여러 테이블에 일관성 없이 서식 지정된 필드 값이 포함된 데이터가 있습니다. 예를 들어 한 테이블에는 **Type**에 **single** 값이 있고 다른 테이블의 동일한 필드에는 **Single**이 포함되어 있습니다. 이런 경우 **Type** 필드에 **single**과 **Single** 값을 모두 포함하게 되면 연결이 수행되지 않으므로 대소문자 처리가 중요합니다.

Table 1		Table 2	
Type	Price	Type	Color
single	23	Single	Red
double	39	Single	Blue
		Double	White
		Double	Black

### 제안 조치

데이터 추가를 사용하여 데이터를 로드한 경우에는 데이터 관리자에서 이를 해결할 수 있습니다.

다음과 같이 하십시오.

1. 데이터 관리자의 테이블 편집기에서 **Table2**를 엽니다.
2. **Type** 필드의 이름을 **Table2.Type**으로 변경합니다.  
 데이터 추가를 사용하여 테이블을 추가하고 **데이터 준비** 옵션을 사용한 경우에는 해당 필드가 자동 연결을 방지하기 위해 이미 **Table2.Type**으로 이름이 지정되어 있을 수 있습니다. 이런 경우는 이 절차에서 두 테이블을 연결합니다.
3. 표현식 `Lower(Table2.Type)`을 사용하여 계산 필드를 만들고 **Type**으로 이름을 지정합니다.
4. **데이터 로드**를 클릭합니다.

이제 **Table1** 및 **Table2**는 **single**과 **double**처럼 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.

- **Single**처럼 모든 값의 첫 문자를 대문자로 표시하려면 대신 **Table1**의 계산된 **Type** 필드를 만들고 표현식 `capitalize(Table1.Type)`을 사용합니다.
- **SINGLE**처럼 모든 값을 대문자로 표시하려면 두 테이블 모두에서 계산된 **Type** 필드를 만들고 표현식 `upper(Table1.Type)` 및 `upper(Table2.Type)`을 각각 사용합니다.

## 7.9 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.

## 7.10 증분 로드를 통해 새 레코드 및 업데이트된 레코드 로드

앱에 지속적으로 업데이트되는 데이터베이스 소스의 데이터가 대량으로 포함된 경우 전체 데이터셋을 다시 로드하려면 시간이 오래 걸릴 수 있습니다. 이런 경우, 다른 데이터는 모두 이미 앱에서 사용할 수 있으므로 해당 데이터베이스의 새 레코드나 업데이트된 레코드만 로드할 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. (연산의 복잡성에 따라 Qlik Sense 엔진에서 자동으로 선택한 방법이 적용됨) 최적화 모드는 표준 모드보다 10배 정도 빠르며, 기존의 방법으로 데이터베이스를 로드하는 것보다 100배 정도 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
```

### 7.11 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

**Join**을 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 **Concatenate** 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 Sense**의 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 Sense**에 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 Sense**에 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 Sense**에 저장되기 전에 두 테이블을 공통 교집합으로 축소하도록 지정됩니다.

이 예에서는 소스 테이블 **Table1** 및 **Table2**:

Table1		Table2	
A	B	A	C
1	aa	1	xx
2	cc	4	yy
3	ee		

**Inner** 예제 소스 테이블을 사용합니다.

## Inner Join

먼저 테이블에 대한 **Inner Join**을 수행하여 두 테이블에 존재하는 유일한 레코드인 하나의 행만 포함된 **VTable**을 만들고 두 테이블에서 데이터가 결합되도록 합니다.

```
VTable:
SELECT * from Table1;
inner join SELECT * from Table2;
```

**VTable**

A	B	C
1	aa	xx

*Inner Join의 예*

## Inner Keep

대신 **Inner Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 물론 공통 필드 **A**를 통해 연결됩니다.

```
VTab1:
SELECT * from Table1;
VTab2:
inner keep SELECT * from Table2;
```

<b>VTab1</b>		<b>VTab2</b>								
<table border="1" style="margin: 0 auto;"> <thead> <tr> <th style="text-align: center;">A</th> <th style="text-align: center;">B</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">aa</td> </tr> </tbody> </table>	A	B	1	aa		<table border="1" style="margin: 0 auto;"> <thead> <tr> <th style="text-align: center;">A</th> <th style="text-align: center;">C</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">xx</td> </tr> </tbody> </table>	A	C	1	xx
A	B									
1	aa									
A	C									
1	xx									

*Inner Keep의 예*

## Left

데이터 로드 스크립트에서 **Join** 및 **Keep** 점두사 앞에 **left** 점두사가 올 수 있습니다.

이 점두사가 **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**:

<b>Table1</b>		<b>Table2</b>														
<table border="1" style="margin: 0 auto;"> <thead> <tr> <th style="text-align: center;">A</th> <th style="text-align: center;">B</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">aa</td> </tr> <tr> <td style="text-align: center;">2</td> <td style="text-align: center;">cc</td> </tr> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">ee</td> </tr> </tbody> </table>	A	B	1	aa	2	cc	3	ee		<table border="1" style="margin: 0 auto;"> <thead> <tr> <th style="text-align: center;">A</th> <th style="text-align: center;">C</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">xx</td> </tr> <tr> <td style="text-align: center;">4</td> <td style="text-align: center;">yy</td> </tr> </tbody> </table>	A	C	1	xx	4	yy
A	B															
1	aa															
2	cc															
3	ee															
A	C															
1	xx															
4	yy															

*Left* 예제 소스 테이블을 사용합니다.

먼저 테이블에 대한 **Left Join**을 수행하여 **Table1**의 모든 행이 포함되고 **Table2**의 해당 행에 있는 필드와 결합된 **VTable**을 만듭니다.

```
VTable:
SELECT * from Table1;
left join SELECT * from Table2;
```

**VTable**

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

*Left Join*의 예

대신 **Left Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 물론 공통 필드 **A**를 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

*Left Keep*의 예

## 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 Sense**에 저장되기 전에 첫 번째 테이블을 두 번째 테이블과의 공통 교집합으로 축소하도록 지정됩니다.

이 예에서는 소스 테이블 **Table1** 및 **Table2**:

A	B
1	aa
2	cc
3	ee

A	C
1	xx
4	yy

Right 예제 소스 테이블을 사용합니다.

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;
    
```

A	B	C
1	aa	xx
4	-	yy

Right Join의 예

대신 **Left Keep**을 수행해도 두 테이블이 생깁니다. 두 테이블은 물론 공통 필드 A를 통해 연결됩니다.

```

VTab1:
SELECT * from Table1;
VTab2:
right keep SELECT * from Table2;
    
```

A	B
1	aa

A	C
1	xx
4	yy

Right Keep의 예

## 7.12 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

## 7 데이터 모델링의 모범 사례

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.

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.

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

### 7.13 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 **Year**와 월별 판매 데이터 행렬이 있습니다.

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 - Jun**으로 표시되는 특성 필드는 노란색으로 표시되었습니다. 이 필드는 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.

Year	Month	Sales
2008	Jan	45

Year	Month	Sales
2008	Feb	65
2008	Mar	78
2008	Apr	12
2008	May	78
2008	Jun	22
2009	Jan	11
2009	Feb	23
...	...	...

### 두 개의 한정하는 열이 있는 크로스탭 피벗 해제

이 경우에는 왼쪽에 한정하는 열이 두 개 있고 뒤이어 행렬 열이 나옵니다.

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** 점두사의 세 번째 파라메타로 지정할 수 있습니다.

```
crosstable (Month, sales, 2) LOAD * from ex2.xlsx;
```

이는 Qlik Sense에서 다음과 같은 결과를 만듭니다.

Salesman	Year	Month	Sales
A	2008	Jan	45
A	2008	Feb	65
A	2008	Mar	78
A	2008	Apr	12

Salesman	Year	Month	Sales
A	2008	May	78
A	2008	Jun	22
A	2009	Jan	11
A	2009	Feb	23
...	...	...	...

## 7.14 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 **ball**과 **box**라는 두 개체를 포함하는 일반 데이터베이스입니다. **color** 및 **weight**와 같은 일부 특성은 두 개체에 공통적이지만 **diameter**, **height** 및 **length**와 같은 다른 특성은 공통적인 특성이 아니라는 것을 확실히 알 수 있습니다.

### GenericTable

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

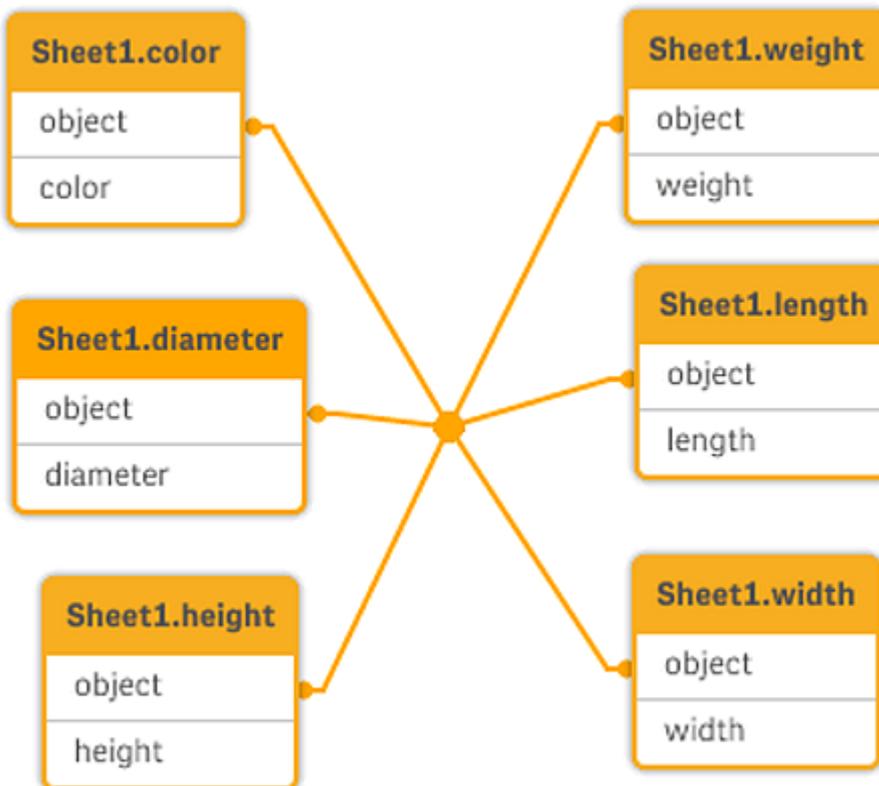
한편, 이러한 특성 중 여러 가지가 특정 개체에는 해당되지 않기 때문에 각 특성에 고유한 열을 지정하는 방식으로 데이터를 저장하는 것은 적합하지 않을 것입니다.

그러나 **length**, **color** 및 **weight**를 하나로 섞어서 표시하면 알아보기 어려울 것입니다.

이 데이터베이스를 표준 방법으로 **Qlik Sense**에 로드하고 테이블의 데이터를 표시하면 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 위한 구문은 간단합니다.

```
Generic SELECT* from GenericTable;
```

일반 데이터베이스를 로드할 때 **LOAD** 또는 **SELECT** 문 중 어느 것을 사용해도 관계가 없습니다.

## 7.15 간격과 불연속 데이터의 연결

**LOAD** 또는 **SELECT** 문에 **intervalmatch** 접두사를 사용하면 불연속 숫자 값을 하나 이상의 숫자 간격과 연결할 수 있습니다. 이는 아래 예와 같은 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?

*Table OrderLog*

Start	End	Order
01:00	03:35	A
02:30	07:58	B
03:04	10:27	C
07:23	11:43	D

*Table EventLog*

Time	Event	Comment
00:00	0	Start of shift 1
01:18	1	Line stop
02:23	2	Line restart 50%
04:15	3	Line speed 100%
08:00	4	Start of shift 2
11:43	5	End of production

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.

Time	Event	Comment	Order	Start	End
0:00	0	Start of shift 1	-	-	-
1:18	1	Line stop	A	1:00	3:35
2:23	2	Line restart 50%	A	1:00	3:35
4:15	3	Line speed 100%	B	2:30	7:58
4:15	3	Line speed 100%	C	3:04	10:...
8:00	4	Start of shift 2	C	3:04	10:...
8:00	4	Start of shift 2	D	7:23	11:...
11:43	5	End of production	D	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.

**intervalmatch**를 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 연결을 설정하려면 별도의 **LOAD** 또는 **SELECT** 문(위 예에서 첫 번째 **SELECT** 문)으로 추가 필드와 간격 필드를 함께 읽어야 합니다.
- 간격은 항상 완료된 상태입니다. 즉, 간격에는 종료 지점이 포함되어야 합니다. 제한이 숫자가 아닐 경우 간격이 무시되며(정의되지 않음), 제한이 **NULL**일 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 생깁니다.

Key	FirstDate	Team
000110	2011-01-21	Southwest
000110		Northwest
000120		Northwest
000120	2013-03-05	Southwest
000120	2013-03-05	Northwest
000120	2013-01-06	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 이 키 테이블로 대체된 후 삭제됩니다.

*Transact*에서 데이터를 로드하면 다음과 같은 테이블이 생깁니다.

Key	Name	Date	Sales
000110	Spengler Aaron	2009-08-18	100
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.

Key	Team	Name	FirstDate	LastDate	Date	Sales
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

## 7.16 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.

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

이 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.

이 스크립트 예제에서는 인라인 로드를 통해 In\_Rates 테이블을 만들었습니다. 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
];
```

다음과 같이 하십시오.

1. 작업할 시간 범위를 결정합니다. 범위의 시작은 데이터의 첫 번째 날짜 이전이어야 하며, 범위의 끝은 마지막 날짜 이후여야 합니다.

```
Let vBeginTime = Num('1/1/2013');
Let vEndTime = Num('1/3/2013');
Let vEpsilon = Pow(2,-27);
```

2. 소스 데이터를 로드하되, 빈 날짜는 이전 항목에서 정의한 범위의 시작으로 변경합니다. 변경 날짜는 "From Date"로 로드되어야 합니다. 먼저 Currency에 따라 테이블을 정렬한 다음, "From Date"에 따라 내림차순으로 정렬하여 최근 날짜가 맨 위에 오도록 합니다.

```
Tmp_Rates:
LOAD Currency, Rate,
    Date(If(ISNum([Change Date]), [Change Date], $(#vBeginTime))) as FromDate
Resident In_Rates;
```

3. 데이터에서 "To Date"를 계산하기 위한 두 번째 단계를 실행합니다. 현재 레코드에 이전 레코드와 다른 통화는 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 빼고, 이 값을 현재 레코드의 "To Date"로 사용해야 합니다.

```
Rates:
LOAD Currency, Rate, FromDate,
    Date(If( Currency=Peek(Currency),
        Peek(FromDate) - $(#vEpsilon),
        $(#vEndTime)
    )) as ToDate
Resident Tmp_Rates
Order By Currency, FromDate Desc;
```

4. 입력 테이블 및 임시 테이블을 삭제합니다.
 

```
Drop Table Tmp_Rates;
```

아래에 나온 스크립트는 다음과 같은 방식으로 소스 테이블을 업데이트합니다.

Currency	Rate	FromDate	ToDate
EUR	8.45	15/02/2013	vEndTime
EUR	8.69	28/01/2013	14/02/2013 23:59:59
EUR	8.59	vBeginTime	28/01/2013 23:59.59
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

스크립트를 실행하면 테이블의 간격이 정확하게 표시됩니다. 데이터 모델 뷰어의 **미리 보기** 섹션을 사용하여 결과 테이블을 봅니다.

**Preview of data**

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	27/01/2013
USD	6.30	03/02/2013	01/03/2013
USD	6.56	10/01/2013	02/02/2013
USD	6.50	01/01/2013	09/01/2013

이 테이블은 이후 **Intervalmatch** 메서드를 사용하여 기존 날짜와 비교하는 데 사용할 수 있습니다.

전체 Qlik Sense 스크립트는 다음과 같습니다.

```

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;
    
```

## 7.17 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.

NodeID	ParentNodeID	Title
1	-	General manager
2	1	Region manager
3	2	Branch manager
4	3	Department manager

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);
```

NodeID	ParentNodeID	Title	Title1	Title2	Title3	Title4
1	-	General manager	General manager	-	-	-
2	1	Region manager	General manager	Region manager	-	-
3	2	Branch manager	General manager	Region manager	Branch manager	-
4	3	Department manager	General manager	Region manager	Branch manager	Department manager

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.

## 7.18 맵 데이터 로드

맵 시각화를 만들려면 앱의 데이터에 연결된 지리 데이터에 액세스해야 합니다. 지리 데이터는 다음 중 한 가지 형식으로 제공됩니다.

- 도시와 같은 단일 위치에 대한 지오포인트 데이터(위도, 경도)
- 지역이나 국가를 나타내는 영역 데이터(지오포인트의 다각형)

데이터 추가를 사용하는 경우 Qlik Sense에서 데이터에 도시 또는 국가가 포함되어 있는지 인식하려고 시도합니다. 성공한 경우 지리 정보가 포함된 새 필드가 자동으로 생성됩니다.



**데이터 추가 사용 시 데이터 로드 및 마침 대신 데이터 준비를 선택해야 합니다. 데이터 로드 및 마침을 선택하면 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 **<data field>\_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.**

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.

인식된 지리 엔터티가 앱 데이터에 포함되어 있지 않거나, 데이터 로드 스크립트를 사용하여 데이터를 로드하는 경우 대신 개별 파일에서 지리 데이터를 로드해야 합니다.

**Excel** 또는 **KML** 파일에서 포인트 데이터를 로드하거나, **KML** 파일에서 영역 데이터를 로드할 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. 빈 필드를 로드해도 문제는 없지만, 맵 시각화를 사용하여 작업할 때에는 빈 필드가 있는 것이 불편할 수 있습니다. 데이터가 없는 차원을 맵에 추가하면 시각적 출력이 생성되지 않아서 사용자를 혼란시킬 수 있습니다. 따라서 데이터를 로드하기 전에 데이터 선택 대화 상자에서 빈 데이터 필드를 모두 선택 해제해야 합니다.



**포인트 레이어 맵의 값의 최대 수는 3333입니다.**

## KML 파일의 데이터에서 맵 만들기

일반적으로 KML 파일에는 포인트 데이터, 영역 데이터 또는 둘 모두 포함됩니다. 다음 스크린샷에서는 *FloridaCounties.Point* 필드에 데이터가 포함되어 있지 않으므로 데이터 선택 대화 상자에서 해당 필드가 선택 해제된 것을 볼 수 있습니다. 필드를 선택 해제하면 데이터가 없는 맵 차원을 만들 위험을 무릅쓰지 않아도 됩니다.

**FloridaCounties.kml에서 데이터 선택** 스크립트 숨기기

테이블:  파일 형식: KML

필드:

<input checked="" type="checkbox"/> FloridaCounties.Name	<input type="checkbox"/> FloridaCounties.Point	<input checked="" type="checkbox"/> FloridaCounties.Area
Citrus		[[[-82.426628,28.694908],[,-82.4
Brevard		[[[-80.493599,28.413055],[,-80.4
Franklin		[[[-85.021421,29.823213],[,-85.0
Broward		[[[-80.29704,26.334356],[,-80.29
Hamilton		[[[-83.055898,30.619483],[,-83.0
Jackson		[[[-85.432857,30.703505],[,-85.4
Okeechobee		[[[-80.982471,27.238772],[,-80.9
Monroe		[[[-80.894429,24.852241],[,-80.8
Duval		[[[-81.216712,30.401676],[,-81.2

```

LOAD
  FloridaCounties.Name,
  FloridaCounties.Area
FROM 'lib://Tutorial source/FloridaCounties.kml'
(kml, Table is [Florida Counties KML]);

```

취소 스크립트 삽입

스크립트 삽입을 클릭하면 아래와 같은 스크립트가 생성됩니다.

```

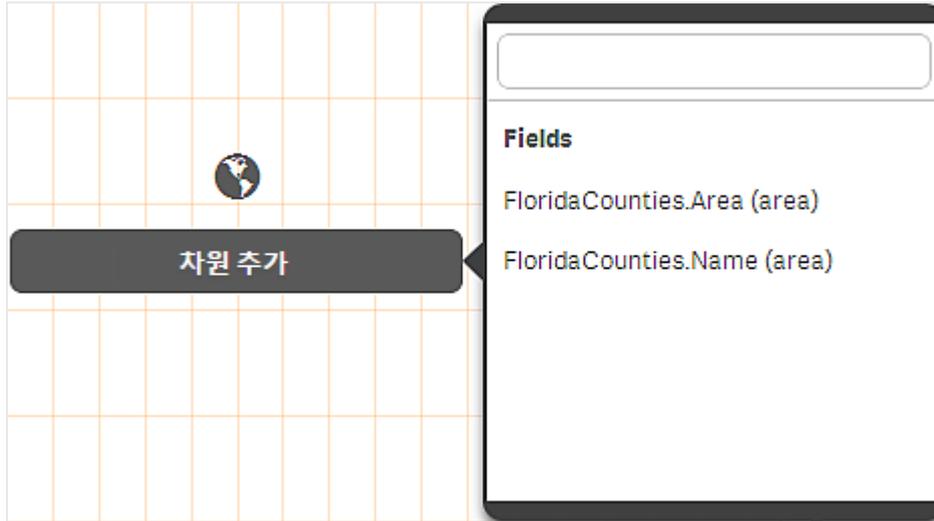
LOAD
  FloridaCounties.Name,
  FloridaCounties.Area
FROM 'lib://data 7/FloridaCounties.kml'
(kml, Table is [Florida Counties KML]);

```

빈 필드의 선택이 해제되지 않은 경우는 스크립트에 다음 문자열도 포함됩니다.

```
FloridaCounties.Point,
```

스크립트를 실행하여 시트에 맵 차트를 추가하면 *FloridaCounties.Area (area)* 또는 *FloridaCounties.Name (area)*를 차원으로 추가할 수 있습니다.



어느 필드에서든 동일한 시각적 결과(국가가 포함된 맵)를 얻게 되었지만 마우스로 국가를 가리키는 경우(터치 장치에서는 탭하는 경우), 차이가 생깁니다. *FloridaCounties.Name (area)*를 선택하면 국가 이름이 도구 설명으로 표시되고, *FloridaCounties.Area (area)*를 선택하면 영역 데이터가 표시됩니다. 의심의 여지 없이 국가 이름에 더 흥미롭습니다. 또한, *FloridaCounties.Area* 필드를 로드하는 것은 *FloridaCounties.Name* 필드를 로드하는 것보다 훨씬 더 느립니다.

선택적으로, 측정값을 추가하고 측정값 기준 색 지정을 통해 여러 국가 간의 측정값에 차이를 줄 수도 있습니다.



**KML 파일에 포인트 데이터나 영역 데이터가 모두 없으면 해당 파일에서 데이터를 로드할 수 없습니다. KML 파일이 손상된 경우는 오류 메시지가 표시되며 데이터를 로드할 수 없게 됩니다.**

## Excel 파일의 포인트 데이터에서 맵 만들기

Excel 파일의 포인트 데이터(좌표)를 사용하여 맵을 만들 수 있습니다.

### 포인트 데이터 서식

Excel 파일에서 포인트 데이터를 읽을 수 있습니다. 두 가지 형식이 지원됩니다.

- 포인트 데이터는 하나의 열에 저장됩니다. 각 포인트는 *x* 및 *y* 좌표의 배열 [*x*, *y*]로 지정됩니다. 특정 지역 관련 좌표를 사용하면 [경도, 위도]에 해당합니다. 이 형식을 사용하는 경우에는 \$geopoint;로 포인트 데이터 필드에 태그를 지정해야 합니다.
- 포인트 데이터는 두 개의 열에 저장되는데, 하나는 위도에 대한 열이고 다른 하나는 경도에 대한 열입니다.
  - 데이터 추가와 데이터 준비 옵션을 함께 사용하여 테이블을 로드한 경우, 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 만들어야 합니다.

다음 예에서는 파일에 회사의 사무소 위치에 대한 동일한 정보가 두 가지 다른 형식으로 포함되어 있다고 가정합니다.

데이터 로드 스크립트를 사용하여 단일 열에서 포인트 데이터 로드 Excel 파일에는 각 사무소에 대해 다음과 같은 내용이 있습니다.

- 사무소 (Office)
- 위치 (Location)
- 직원 수 (Employees)

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 이는 \$geopoint로 필드에 태그를 지정하여 포인트 데이터 필드로 인식되도록 하는 데 필요합니다. office 필드도 \$geoname으로 태그를 지정하여 지오포인트 이름으로 인식되도록 해야 합니다. **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.

데이터 로드 스크립트를 사용하여 개별 위도 및 경도 열에서 포인트 데이터 로드 Excel 파일에는 각 사무소에 대해 다음과 같은 내용이 있습니다.

- 사무소 (Office)
- 위도 (Latitude)
- 경도 (Longitude)
- 직원 수 (Employees)

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 필드도 \$geoname으로 태그를 지정하여 지오포인트 이름으로 인식되도록 해야 합니다. **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.

### 표시되는 포인트 수

성능상의 이유로, 표시할 수 있는 포인트의 수는 3333개로 제한됩니다. 그러나 단일 맵에서 1000개를 넘는 값을 표시하면 가독성에 영향을 줄 수 있습니다. 선택을 수행하여 표시할 데이터 양을 줄일 수 있습니다.

## 7.19 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 States**로 나열됩니다.

컨텍스트네이트된 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 컨텍스트네이트되며, 매핑 테이블의 첫째 열 값과 필드 값을 비교할 *Country* 필드가 포함되어 있습니다. 필드 값 **US**, **U.S.**, **United States**가 발견되고 매핑 테이블의 둘째 열 값인 **USA**로 대체됩니다.

자동 매핑은 Qlik Sense 테이블에 필드가 저장되기 전 일련의 이벤트에서 마지막으로 수행됩니다. 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. 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.

## 8 문제 해결 — 데이터 로드

이 섹션에서는 Qlik Sense에서 데이터를 로드하고 모델링할 때 발생할 수 있는 문제에 대해 설명합니다.

### 8.1 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.

### 8.2 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.

## 8.3 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

## 8.4 OLE DB 데이터 소스를 사용하는 경우의 데이터 선택 문제

### 가능한 원인

OLE DB 데이터 연결에서 데이터를 선택할 수 없는 경우, 연결 구성 방법을 점검해 보아야 합니다.

### 제안 조치

다음과 같이 하십시오.

1. 연결 문자열이 올바르게 설계되었는지 확인하십시오.
2. 적절한 로그온 자격 증명을 사용하는지 확인하십시오.

참조: *데이터 소스에 연결(125페이지)*

## 8.5 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.

## 8.6 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.

≤ [커넥터 다운로드 및 설치](#) (영어만 해당)

커넥터가 **Qlik Sense**에 맞게 조정되어 있지 않습니다.

### 가능한 원인

데이터를 선택할 수 있으려면 **QlikView** 커넥터가 **Qlik Sense**에 맞게 조정되어야 합니다.

### 제안 조치 (QVX SDK를 사용하여 커넥터를 자체 개발한 경우)

다음과 같이 하십시오.

- 데이터를 선택하려면 인터페이스를 사용하여 커넥터를 **Qlik Sense**에 맞게 조정해야 합니다.

≤ [Qlik Sense 사용](#) (영어만 해당)

### 제안 조치 (커넥터가 제공된 경우)

다음과 같이 하십시오.

- 커넥터 공급업체에 연락하여 **Qlik Sense**에 맞는 커넥터를 구하십시오.

## 8.7 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.

### 세미콜론으로 종결되지 않은 문

#### 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.

## 8.8 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.

- 필드 이름에서 **필드 이름 없음**을 선택합니다.
- 필드 나누기를 선택합니다.
- 열이 헤더에 맞춰 정렬될 때까지 **탭 크기**의 설정을 증가시킵니다.
- 적절한 열 위치를 클릭하여 필드 나누기를 삼입합니다.
- 데이터 미리 보기를 선택합니다.
- 필드 이름에서 **포함된 필드 이름**을 선택합니다.

이제 열이 적절하게 정렬되고 각 필드가 올바른 필드 이름을 갖습니다.

## 8.9 파일 첨부 시 "잘못된 경로" 오류 메시지

---

### 가능한 원인

파일 이름이 너무 길습니다. Qlik Sense에서는 파일 이름을 최대 171자까지 지원합니다.

### 제안 조치

파일 이름을 172자보다 적은 문자가 포함된 이름으로 바꾸십시오.

## 8.10 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.

## 8.11 ODBC를 통해 Microsoft Excel 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.

## 8.12 파일을 데이터 추가에 끌어서 첨부하는 방식이 작동하지 않습니다.

파일을 **Windows** 탐색기에서 Qlik Sense의 데이터 추가 파일 첨부 대화 상자로 끌어서 첨부하려고 해도 파일이 업로드되지 않습니다.

### 가능한 원인

파일은 ZIP 보관 파일에 저장됩니다. 해당 보관 파일이 **Windows** 탐색기에서 폴더처럼 표시되더라도 Qlik Sense의 ZIP 보관 파일에서 개별 파일을 첨부하는 것은 불가능합니다.

### 제안 조치

파일을 첨부하기 전에 ZIP 보관 파일에서 추출하십시오.

## 8.13 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 마지막 데이터 추가 대화 상자에서 데이터 준비 옵션을 선택했습니다. 이 옵션은 테이블 간에 공통인 모든 필드 이름을 자동으로 한정합니다. 예를 들어 데이터 준비를 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 로드를 클릭합니다.

## 8.14 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 G로 표시되지 않습니다.

### 가능한 원인

테이블을 추가할 때 마지막 데이터 추가 대화 상자에서 데이터 로드 및 마침 옵션을 선택했습니다.

이 옵션을 사용하면 인식되는 날짜 및 타임스탬프 필드가 정상적으로 기능하지만 자산 패널 및 기타 필드 목록에서 **G**로 표시되지 않으며 확장된 속성 필드를 사용할 수 없습니다.

### 제안 조치

**데이터 관리자**를 열고 **데이터 로드**를 클릭합니다.

이제 모든 날짜 및 타임스탬프 필드가 시트 보기의 자산 패널에서 **G**로 표시됩니다. 여전히 **G**로 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 로깅 설정 및 일반적인 날짜 서식에 기반하여 날짜 필드를 자동으로 인식하지만 경우에 따라서는 입력 서식을 지정해야 할 수 있습니다.

### 제안 조치

**데이터 관리자**를 열고 날짜로 인식되지 않는 필드가 포함된 테이블을 편집합니다. 필드는 대부분 일반 필드처럼 **G**로 표시됩니다. 필드 유형을 필드 데이터와 일치하는 입력 서식을 사용하는 **날짜** 또는 **타임스탬프**로 변경합니다.

## 8.15 섹션 액세스 규칙이 변경되었지만 모든 데이터를 볼 수 있습니다.

---

섹션 액세스 규칙이 스크립트에서 수정되었고, 해당 스크립트가 실행되어 활성 데이터 축소를 통해 데이터를 로드했지만 앱에서 모든 데이터를 볼 수 있습니다.

### 가능한 원인

섹션 액세스 규칙은 앱이 열려 있을 때만 적용됩니다. 스크립트를 실행하면 섹션 액세스 규칙이 동일한 세션의 데이터에 적용되지 않습니다(예: 다른 탭에서 시트를 연 경우).

### 제안 조치

수정된 섹션 액세스 규칙을 적용하려면 앱을 다시 여십시오.

## 8.16 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 현재 사용되는 데이터 소스에 액세스할 수 있어야 합니다.

## 8.17 서버에서 앱을 만드는 모든 사용자에게 대해 데이터 관리자 워크플로가 손상되었습니다

사용자가 데이터 관리자에서 데이터 추가 또는 데이터 로드를 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 파일이 대량으로 누적되거나 손상되면 오류가 발생할 수 있습니다.

**제안 조치**

QVD 파일이 포함된 폴더를 삭제합니다. Qlik Sense 서버에서 캐시는 다음 위치에 있습니다.

```
C:\ProgramData\Qlik\Sense\Apps\DataPrepAppCache
```

Qlik Sense Desktop에서 캐시는 다음 위치에 있습니다.

```
C:\Users\\Documents\Qlik\Sense\Apps\DataPrepAppCache
```

## 8.18 디스크 공간이 부족합니다.

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.

```
C:\ProgramData\Qlik\Sense\Apps\DataPrepAppCache
```

Qlik Sense Desktop에서 캐시는 다음 위치에 있습니다.

```
C:\Users\\Documents\Qlik\Sense\Apps\DataPrepAppCache
```

## 8.19 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 파일로 이동한 후 **include** 변수를 사용하여 런타임 시 스크립트에 삽입하는 것을 고려해 보십시오.