

Zelfstudie - Scriptgebruik voor beginners

Qlik Sense®

August 2023

Copyright © 1993-2023} QlikTech International AB. Alle rechten voorbehouden.



1 Welkom bij deze zelfstudie!	4
1.1 Wat u gaat leren	4
1.2 Voor wie is deze zelfstudie bedoeld	4
1.3 Inhoud pakket	4
1.4 Lessen van deze zelfstudie	5
1.5 Meer informatie en bronnen	5
2 Scriptgebruik in de editor voor laden van gegevens	6
2.1 Editor voor laden van gegevens	6
2.2 Scripteditor	7
Help bij syntaxis openen voor opdrachten en functies	7
Opmerkingen toevoegen aan het script	8
Code laten inspringen	8
Een voorbereid testscript invoegen	8
3 De opdrachten LOAD en SELECT	9
4 Gegevens selecteren en laden	10
5 Velden hernoemen	19
6 Gegevens verminderen	22
7 Gegevens transformeren	28
7.1 Resident LOAD	28
7.2 Voorafgaande LOAD	32
8 Aaneenschakelen	34
8.1 Automatisch aaneenschakelen	34
8.2 Geforceerd aaneenschakelen	37
8.3 Aaneenschakelen voorkomen	38
9 Cirkelreferenties	40
9.1 Cirkelreferenties oplossen	41
10 Synthetische sleutels	43
10.1 Synthetische sleutels oplossen	45
11 Gegevens in een app gebruiken	47
11.1 Een diagram toevoegen	47
11.2 Dimensies en metingen toevoegen	47
Dimensies maken en toevoegen	48
Metingen maken en toevoegen	48
11.3 Hartelijk dank!	52

1 Welkom bij deze zelfstudie!

Welkom bij deze zelfstudie waarin u leert apps te bouwen met Qlik Sense.

Voordat u kunt beginnen met het maken van visualisaties in uw app in Qlik Sense, moet u uw gegevens laden. Als u weet hoe u load-scripts moet gebruiken, kunt u uw gegevens voorbereiden en manipuleren zodra u deze in uw app laadt.

U gebruikt Gegevensbeheer of de editor voor laden van gegevens om gegevens te laden. U gebruikt de editor voor laden van gegevens bij het maken, bewerken en uitvoeren van een load-script voor gegevens.

1.1 Wat u gaat leren

Nadat u deze zelfstudie hebt voltooid, bent u vertrouwd geraakt met het laden van gegevens met behulp van scripts, het bewerken van scripts en het transformeren van gegevens.

1.2 Voor wie is deze zelfstudie bedoeld

U bent vertrouwd met de basisprincipes van Qlik Sense. Met andere woorden, u hebt al eens apps en visualisaties gemaakt.

U moet toegang hebben tot de editor voor laden van gegevens en u moet toestemming hebben om gegevens te laden in Qlik Sense Enterprise on Windows.

1.3 Inhoud pakket

Het zip-pakket dat u hebt gedownload bevat de volgende gegevensbestanden die u nodig hebt om de zelfstudie af te ronden:

- *Customers.xlsx*
- *Dates.xlsx*
- *Region.txt*
- *Sales.xlsx*

Dit pakket bevat daarnaast ook een kopie van de app *Zelfstudie scriptgebruik*. U kunt de app uploaden naar uw hub.

We raden u aan om de app zelf te bouwen en dat te doen zoals beschreven in de zelfstudie, zodat u hier zoveel mogelijk van leert. U moet daarnaast de app uploaden en deze aan de gegevensbestanden koppelen om de gegevensloads te laten functioneren, zoals beschreven in de zelfstudie.

Als u echter tegen problemen aanloopt, kan de app u helpen deze op te lossen. We hebben aangegeven welke scriptsegmenten aan welke les gekoppeld zijn.

1.4 Lessen van deze zelfstudie

U hebt 3 tot 4 uur nodig om deze zelfstudie af te ronden, afhankelijk van uw ervaring met Qlik Sense. De onderwerpen zijn bedoeld om achtereenvolgens door te nemen. U kunt echter op elk gewenst moment stoppen om de zelfstudie later weer te hervatten. Er zijn gelukkig geen toetsen.

- Inleiding tot het laden van gegevens
- LOAD- en SELECT-opdrachten
- Gegevens selecteren en laden
- Velden hernoemen
- Gegevens verminderen
- Gegevens transformeren
- Aaneenschakelen
- Cirkelreferenties
- Synthetische sleutels
- Gegevens in een app gebruiken

1.5 Meer informatie en bronnen

- [Qlik](#) biedt een groot aantal bronnen, voor het geval u nog meer wilt leren.
- [Qlik online help](#) is beschikbaar.
- Training, inclusief gratis online cursussen, is beschikbaar in de [Qlik Continuous Classroom](#).
- Discussieforums, blogs, en meer kunt u vinden in de [Qlik Community](#).

2 Scriptgebruik in de editor voor laden van gegevens

Qlik Sense voert een script voor het laden van gegevens uit dat wordt beheerd in de Editor voor laden van gegevens, zodat er voor het ophalen van gegevens verbinding kan worden gelegd met diverse gegevensbronnen. Een gegevensbron kan een gegevensbestand zijn, bijvoorbeeld een Excel-bestand of een .csv-bestand. Een gegevensbron kan ook een database zijn, bijvoorbeeld een Google BigQuery- of Salesforce-database.

Het is ook mogelijk om gegevens te laden met behulp van Gegevensbeheer, maar wanneer u een load-script voor gegevens wilt maken, bewerken en uitvoeren gebruikt u de editor voor het laden van gegevens.

De velden en tabellen die moeten worden geladen, worden aangegeven in het script. Het maken van scripts wordt hoofdzakelijk gebruikt om op te geven welke gegevens moeten worden geladen vanuit uw gegevensbronnen. Het is ook mogelijk de gegevensstructuur te manipuleren met behulp van scriptopdrachten.

Terwijl de gegevens worden geladen, identificeert Qlik Sense gemeenschappelijke velden uit verschillende tabellen (sleutelvelden) om de gegevens met elkaar te associëren. De resulterende gegevensstructuur van de gegevens in de app kan worden bewaakt in de gegevensmodelviewer. U kunt de gegevensstructuur wijzigen door velden een nieuwe naam te geven. Hierdoor worden de associaties tussen tabellen veranderd.

Nadat de gegevens in Qlik Sense zijn geladen, worden ze opgeslagen in de app.

2.1 Editor voor laden van gegevens

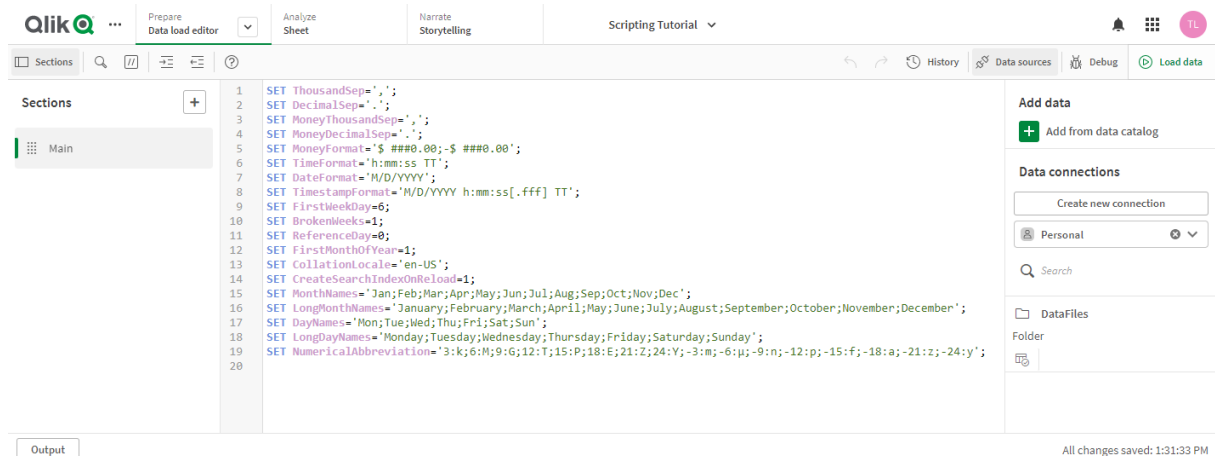
U kunt scripts maken om gegevens te laden in de editor voor laden van gegevens. De editor is beschikbaar in de vervolgkeuzelijst in Qlik Sense.

Als u de editor voor laden van gegevens opent, verschijnt de scripteditor in het midden van het venster. Scriptsecties worden als tabbladen weergegeven in het linkermenu. Met Qlik Sense wordt automatisch de **hoofdsectie** aangemaakt. Gegevensverbindingen staan in het rechtermenu.

Het script moet worden geschreven met de scriptsyntaxis van Qlik Sense. De sleutelwoorden van Qlik Sense worden blauw weergegeven.

2 Scriptgebruik in de editor voor laden van gegevens

Editor voor laden van gegevens



2.2 Scripteditor


De editor heeft diverse functies voor het opstellen van het load-script.

Help bij syntaxis openen voor opdrachten en functies

U kunt op verschillende manieren toegang krijgen tot help bij een sleutelwoord van de Qlik Sense-syntaxis.

De helpportal openen

U kunt op twee verschillende manieren toegang krijgen tot gedetailleerde help op de Qlik Sense-helpportal.

- Klik op  op de werkbalk om de helpmodus voor syntaxis te activeren. Als de helpmodus actief is, verschijnt na een klik op een sleutelwoord (blauw en onderstreept) helpinformatie over de syntaxis.
- De cursor op of aan het einde van een sleutelwoord plaatsen en drukken op Ctrl+H.



U kunt het script niet bewerken in de helpmodus voor syntaxis.

Via de functie voor automatisch aanvullen


Als u begint met het typen van een Qlik Sense-sleutelwoord, verschijnt een lijst met overeenkomende sleutelwoorden waarin u het sleutelwoord kunt selecteren. De lijst wordt steeds verder beperkt als u doorgaat met typen en u kunt kiezen uit sjablonen met voorgestelde syntaxis en parameters. De syntaxis van de functie, inclusief parameters en aanvullende instructies, evenals een koppeling naar de beschrijving van de instructie of functie in de helpportal, wordt via knopinfo weergegeven.





U kunt ook de toetscombinatie Ctrl+spatiebalk gebruiken om de lijst met trefwoorden weer te geven en Ctrl+Shift+spatiebalk om knopinfo weer te geven.

Opmerkingen toevoegen aan het script

U kunt opmerkingen invoegen in de scriptcode, of delen van het script deactiveren met markeringen voor opmerkingen. Alle tekst op een regel die volgt op `//` (twee schuine strepen) worden gezien als een opmerking en worden dus niet samen met de rest van het script uitgevoerd.

De werkbalk van de editor voor het laden van gegevens bevat een snelkoppeling om code in het script aan te duiden als opmerking of juist niet. Klik op  of druk op Ctrl + K om code aan te duiden als opmerking of juist niet.

Code laten inspringen

U kunt de code laten inspringen om de leesbaarheid te verbeteren. Klik op  om de tekst te laten inspringen (inspringing vergroten) of klik op  om de inspringing van de tekst te verkleinen.

Een voorbereid testscript invoegen

U kunt een voorbereid testscript invoegen dat een set inline gegevensvelden laadt. Hiermee kunt u snel een gegevensverzameling maken voor testdoeleinden. Druk op Ctrl + 00 om het testscript in te voegen.

3 De opdrachten LOAD en SELECT

U kunt gegevens in Qlik Sense laden via de LOAD- en SELECT-opdrachten. Elk van deze opdrachten genereert een interne tabel. LOAD wordt gebruikt voor het laden van gegevens uit bestanden, terwijl SELECT wordt gebruikt voor het laden van gegevens uit databases.

In deze zelfstudie gebruikt u gegevens uit bestanden en dus LOAD-opdrachten.

U kunt ook een voorafgaande LOAD gebruiken om de inhoud van de geladen gegevens te kunnen bewerken. Zo moet het hernoemen van velden gebeuren aan de hand van een LOAD-opdracht; SELECT-opdrachten staan geen wijzigingen aan veldnamen toe.

De volgende regels zijn van toepassing bij het laden van gegevens in Qlik Sense:

- Qlik Sense maakt geen onderscheid tussen tabellen die zijn gegenereerd via een LOAD- of SELECT-opdracht. Dit betekent dat als er meerdere tabellen worden geladen, het niet uitmaakt of dit met LOAD- of SELECT-instructies of beide is gebeurd.
- De volgorde van de velden in de opdracht of in de oorspronkelijke tabel in de database is niet belangrijk in de Qlik Sense-logica.
- Veldnamen zijn hoofdlettergevoelig en worden gebruikt om koppelingen tussen gegevenstabellen te creëren. Hierdoor is het soms nodig om velden in het load-script een nieuwe naam te geven, om op die manier het gewenste gegevensmodel te bewerkstelligen.

4 Gegevens selecteren en laden

Het laden van gegevens vanuit bestanden, zoals Microsoft Excel of andere ondersteunde bestandsindelingen, kan op eenvoudige wijze worden uitgevoerd via het dialoogvenster voor gegevensselectie in de editor voor laden van gegevens.

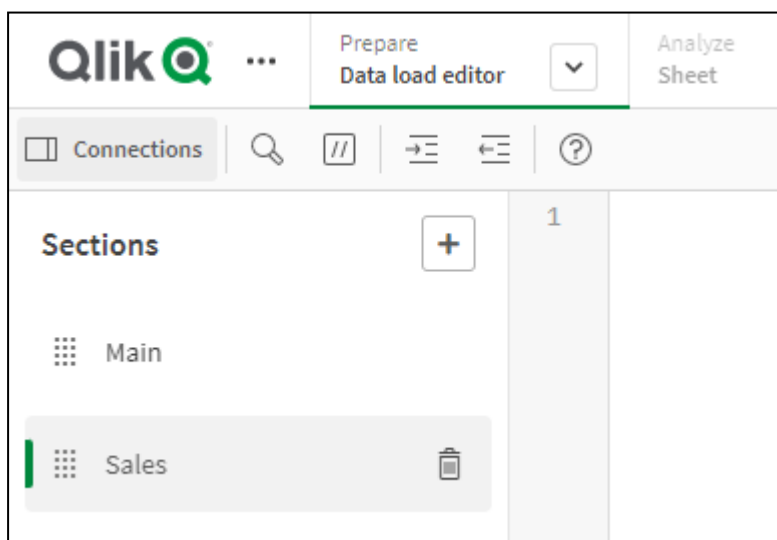
Doe het volgende:

1. Open Qlik Sense.
2. Maak een nieuwe app.
3. Noem uw app *Scripting Tutorial* en klik dan op **Maken**.
4. Open de app.



*Voordat u voor het eerst gegevens in uw app gaat laden, is er een optie beschikbaar om **Gegevens toevoegen** te gebruiken om op eenvoudige wijze gegevens vanuit bestanden te laden. In deze zelfstudie willen we echter het script bekijken, dus gebruiken we de editor voor laden van gegevens.*

5. Open de editor voor laden van gegevens uit het vervolgkeuzemenu in de bovenste werkbalk.
6. Klik op **+** in het linkermenu om een nieuwe scriptsectie toe te voegen onder de sectie met de naam *Main*.
U kunt uw script keurig op orde houden door meer dan één sectie te gebruiken. De scriptsectie wordt uitgevoerd in de volgorde van hoe u gegevens laadt.
7. Geef de sectie een naam door *Sales* te typen.
Nieuw tabblad Sales in de Scripting Tutorial -app



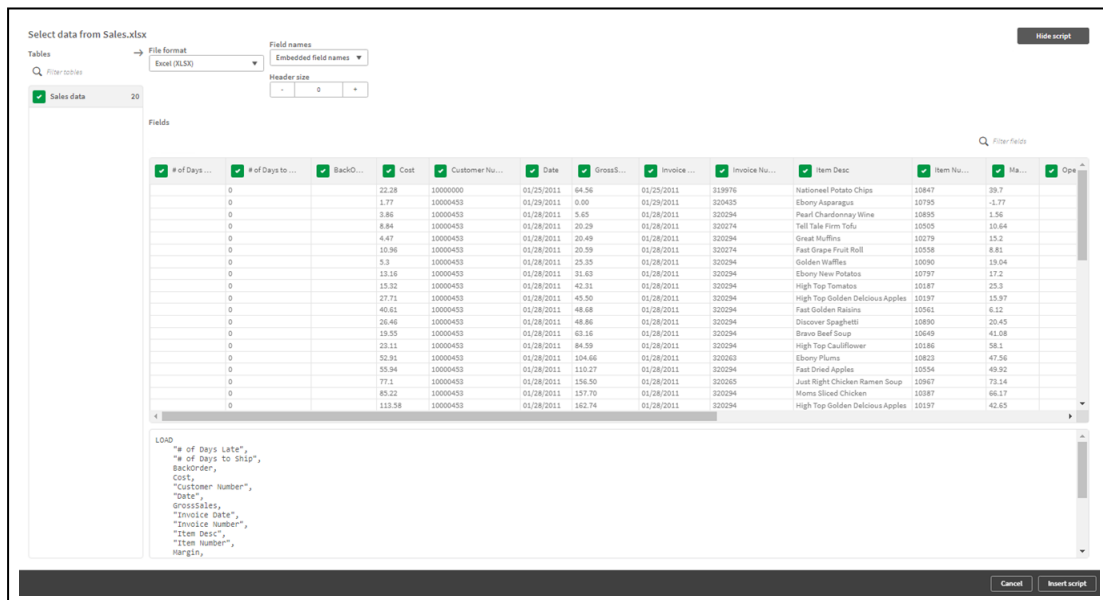
8. Onder **AttachedFiles** in het rechtermenu klikt u op **Gegevens selecteren**.

Gegevensvenster selecteren



9. Upload en selecteer *Sales.xlsx*. Het venster Voorbeeld gegevens wordt geopend.

Venster met gegevensvoorbeeld voor gegevensbestand Sales

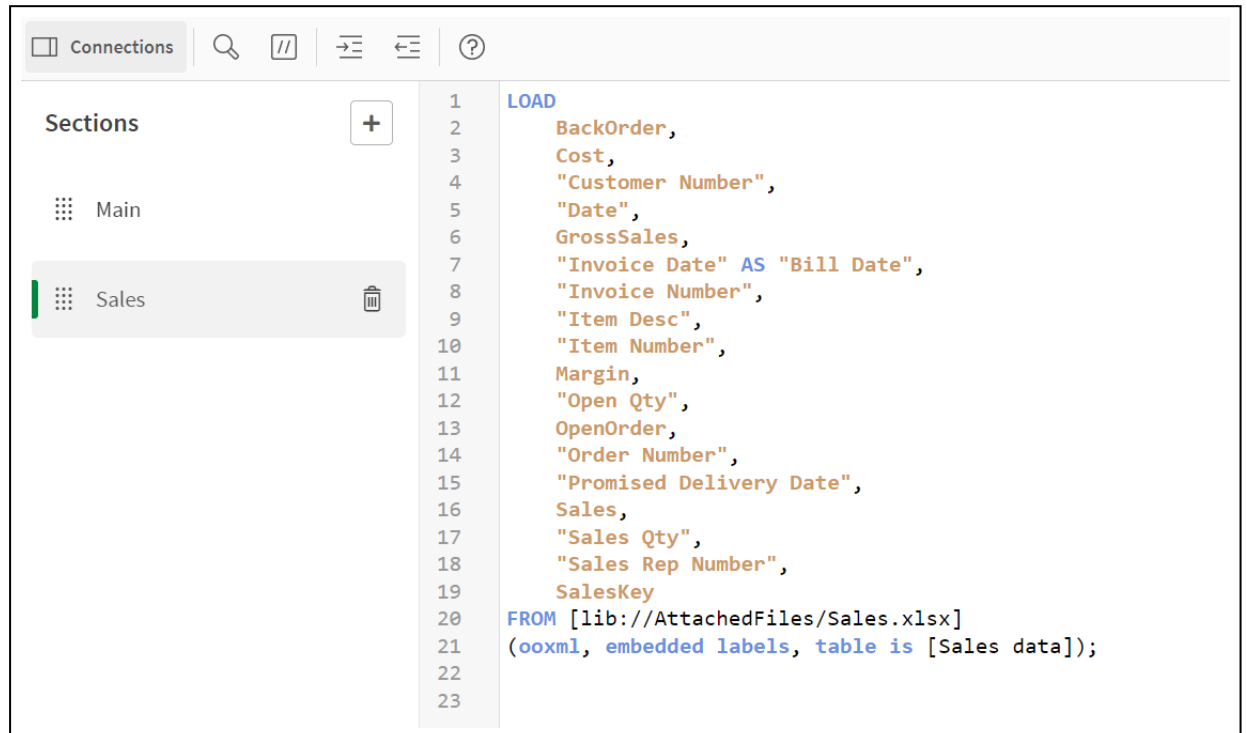


10. Hef de selectie van de velden *# of Days Late* en *# of Days to Ship* op. Mogelijk moet u op de veldnamen klikken om de volledige veldnamen te zien.
11. Zoek naar *date* in het zoekveld **Filtervelden**.
12. Klik op de koptekst *Invoice Date* en typ *Bill Date* om het veld te hernoemen.

13. Klik op **Script invoegen**. Het load-script wordt ingevoerd in de sectie *Sales* van de scripteditor. Let op: Qlik Sense plaatst dubbele aanhalingstekens bij veldnamen met een spatie.

Uw script zou er als volgt moeten uitzien:

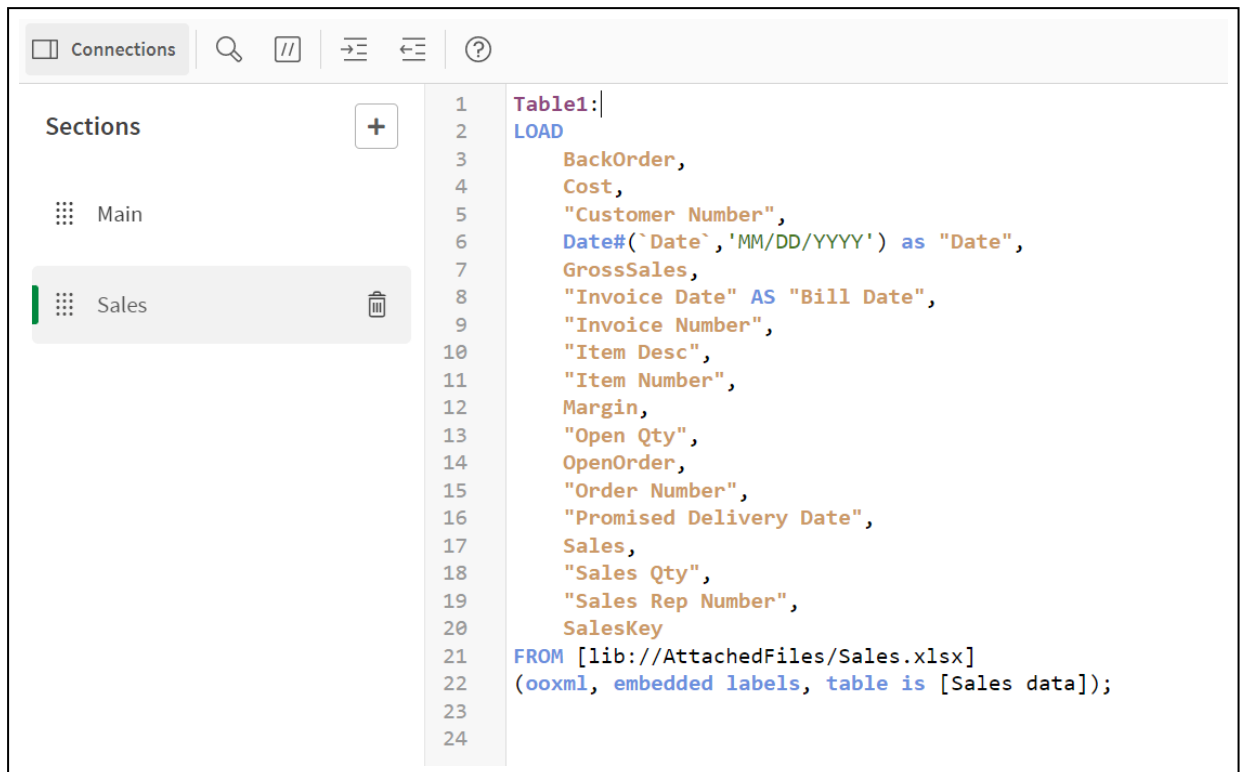
Load-script op tabblad Sales






14. Voeg de volgende rij toe boven de *LOAD*-opdracht om de tabel *Table1* een naam te geven:
Table1:
15. Nu passen we het script aan om er zeker van te zijn dat de datums correct worden geïnterpreteerd. Verander het veld *Date* naar het volgende:
Date#(`Date`, 'MM/DD/YYYY') as "Date",

Uw script zou er als volgt moeten uitzien:

Bijgewerkt load-script op tabblad Sales





16. Klik in de rechterbovenhoek op **Gegevens laden**.
Hiermee worden de gegevens in de app geladen. Er wordt een voortgangsvenster voor de scriptuitvoering weergegeven. Als dit is voltooid, ziet u een overzicht van mogelijke fouten en synthetische sleutels (zelfs als deze niet beschikbaar zijn).
17. Klik op **Sluiten**.
18. Open de gegevensmodelviewer uit het vervolgkeuzemenu in de bovenste werkbalk. Als u op  klikt, wordt de gegevensmodelviewer in een nieuw tabblad geopend.
19. Selecteer  en  in het bovenste menu om de tabelweergave op te halen die wordt gebruikt in deze zelfstudie. Als uw tabel niet correct wordt weergegeven, kunt u het bestaande load-script verwijderen en het vervolgens opnieuw opbouwen.

Tabelweergave in gegevensmodelviewer van Sales-gegevens

Table1
BackOrder
Cost
Customer Number
Date
GrossSales
Bill Date
Invoice Number
Item Desc
Item Number
Margin
Open Qty
OpenOrder
Order Number
Promised Delivery Date
Sales
Sales Qty
Sales Rep Number
SalesKey

Nu laden we een andere tabel met de naam *Datum*. Nadat we deze tabel hebben geladen, verbindt Qlik Sense deze met de tabel *Verkoop* op het veld *Datum*.

20. Open de **Editor voor laden van gegevens**.
21. Klik **+** om een nieuwe scriptsectie toe te voegen.
22. Geef de sectie *Dates* een naam. Als de nieuwe sectie *Dates* niet al onder *Sales* is geplaatst, zet u de cursor op  en sleept u de sectie naar beneden onder de sectie *Sales* om de volgorde aan te passen.
23. Klik op de bovenste rij van het script en klik op .
Zorg dat `//` aan het script wordt toegevoegd.
24. Voeg de volgende tekst toe achter `//`:
`Loading data from Dates.xlsx`

De bovenste regel van uw script zou er nu als volgt moeten uitzien:

```
// Loading data from Dates.xlsx
```

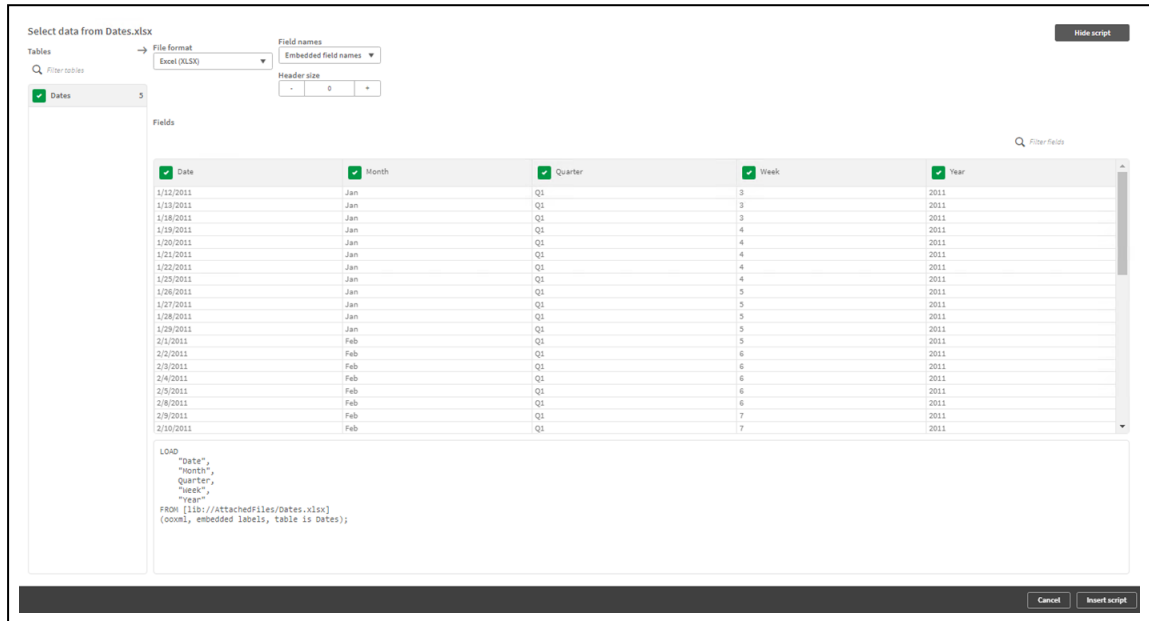
25. Onder **AttachedFiles** in het rechtermenu klikt u op **Gegevens selecteren**.



Selecteer alle velden en zorg ervoor dat **Ingesloten veldnamen** onder **Veldnamen** is geselecteerd om de namen van de tabelvelden mee te nemen bij het laden van de gegevens.

26. Upload en selecteer *Dates.xlsx*. Het venster Voorbeeld gegevens wordt geopend.

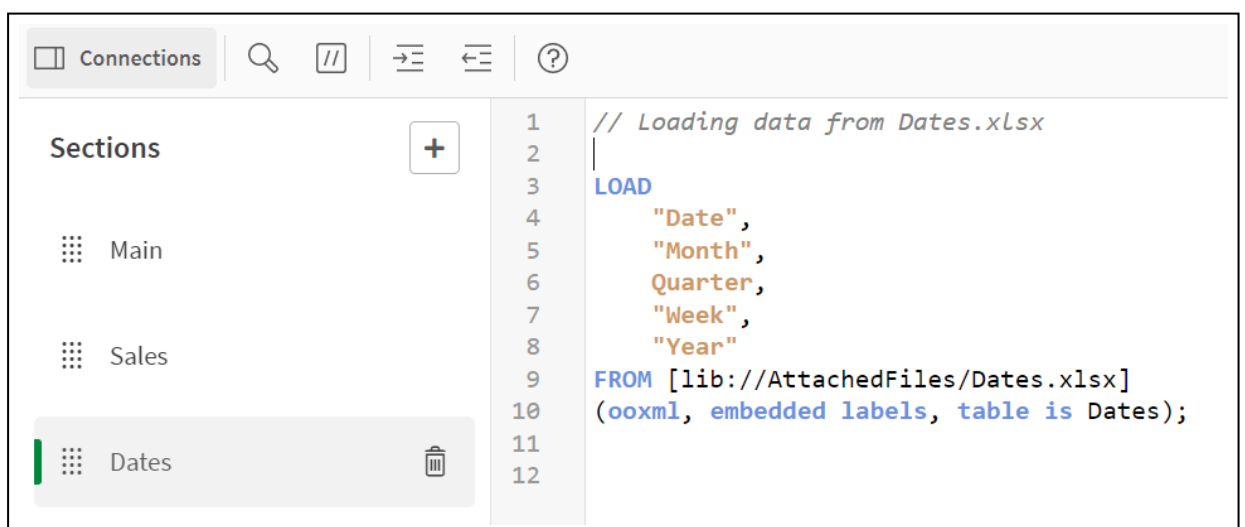
Venster met gegevensvoorbeeld voor gegevensbestand Dates



27. Klik op **Script invoegen**.

Uw script zou er als volgt moeten uitzien:

Load-script op tabblad Dates



28. Voeg het volgende toe aan de rij boven de `LOAD`-instructie om de tabel `Table2` een naam te geven:
`Table2:`

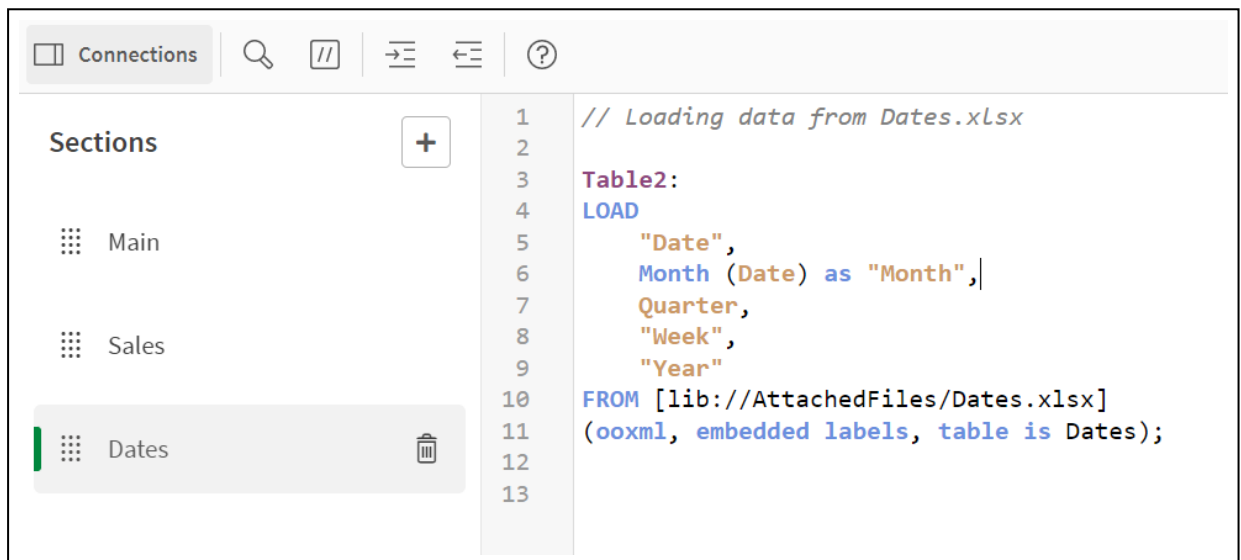
29. Om ervoor te zorgen dat de kolom `Month` in het bestand `Dates.xlsx` correct wordt geïnterpreteerd in Qlik Sense, moeten we de functie `Month` toepassen op het veld `Date`.

Verander het veld `Maand` naar het volgende:

`Month (Date) as "Month",`

Uw script zou er als volgt moeten uitzien:

Bijgewerkt load-script op tabblad Dates



Nu hebt u een script gemaakt voor het laden van de geselecteerde gegevens vanuit het bestand `Dates.xlsx`. Het wordt tijd om de gegevens in de app te laden.

30. Klik in de rechterbovenhoek op **Gegevens laden**.

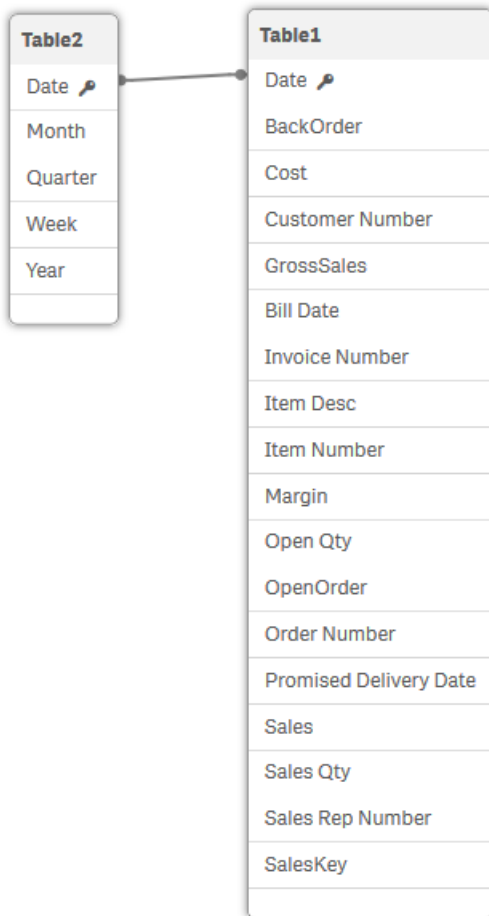
Als u op **Gegevens laden** klikt, worden de gegevens in de app geladen en wordt het script opgeslagen.

31. Als de uitvoering van het script is voltooid, klikt u op **Sluiten**.

32. Open de **gegevensmodelviewer**.

Nu kunt u zien dat een verbinding tot stand is gebracht tussen de twee velden genaamd `Date` in de twee tabellen.

Tabelweergave in gegevensmodelviewer

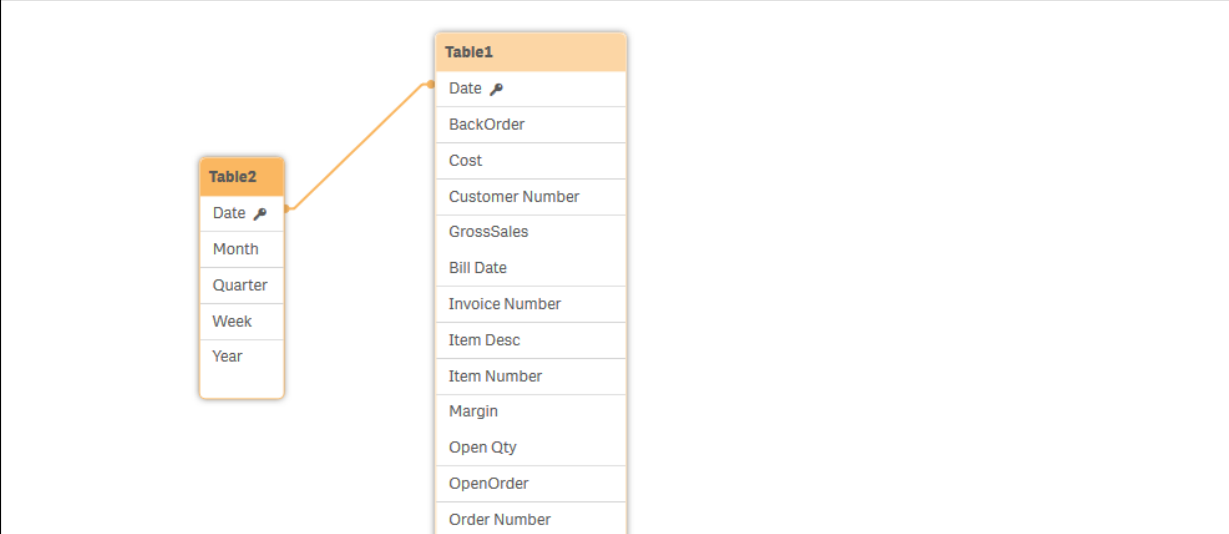


33. Klik op **Voorbeeld** in de linkerbenedenhoek. Klik op de naam van de tabel *Table2*.

Hiermee wordt informatie over de tabel weergegeven. In het veld **Voorbeeld** ziet u dat 628 rijen met gegevens in de interne tabel *Table2* zijn geladen. Als u daarentegen op een veld in de tabel klikt, ziet u informatie over het desbetreffende veld.

4 Gegevens selecteren en laden

Tabelvoorbeeld in gegevensmodelviewer



▼ Preview

Table2		Preview of data				
		Date	Month	Quarter	Week	Year
Rows	628	01/12/2011	Jan	Q1	3	2011
Fields	5	01/13/2011	Jan	Q1	3	2011
Keys	1	01/18/2011	Jan	Q1	3	2011
Tags	\$key \$numeric \$integer \$timestamp \$date \$ascii \$text	01/19/2011	Jan	Q1	4	2011
		01/20/2011	Jan	Q1	4	2011
		01/21/2011	Jan	Q1	4	2011
		01/22/2011	Jan	Q1	4	2011

De gegevens zijn nu beschikbaar om te worden gebruikt in de visualisaties in een app. We zullen u dit later in deze zelfstudie laten zien.

5 Velden hernoemen

In het vorige onderwerp lieten we u zien hoe u velden moet hernoemen in het venster Voorbeeld gegevens. In de koptekst van het venster Voorbeeld gegevens hebt u de naam van *Invoice Date* veranderd naar *Bill Date*. Wanneer u het load-script hebt ingevoegd, kunt u zien dat de naam van het veld wordt veranderd met het trefwoord *AS*.

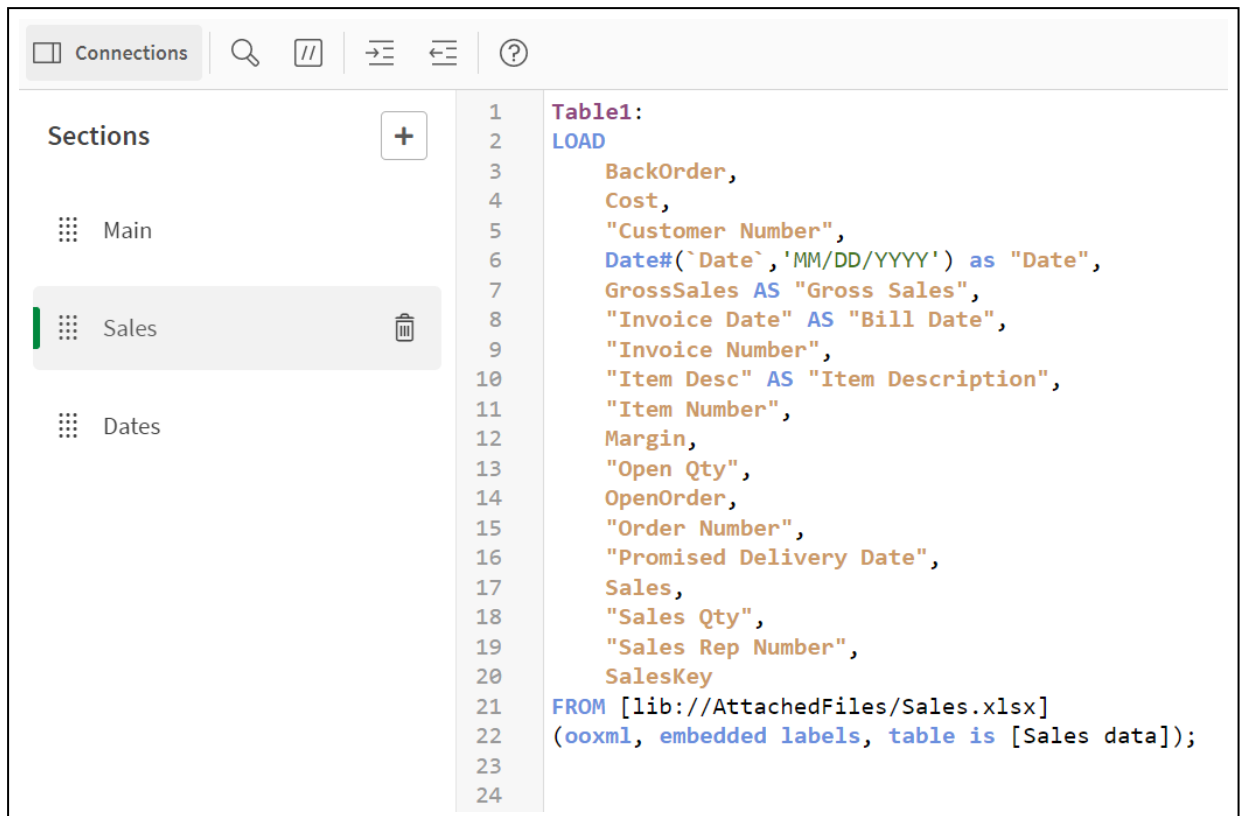
We kunnen deze actie ook direct in het script uitvoeren.

Doe het volgende:

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik op het tabblad *Sales*.
3. In het load-script brengen we de volgende wijzigingen aan. Let op: u moet de haakjes gebruiken bij velden met een spatie.
 - i. Wijzig *GrossSales*, in:
`GrossSales AS "Gross Sales",`
 - ii. Wijzig *"Item Desc"*, in:
`"Item Desc" AS "Item Description",`

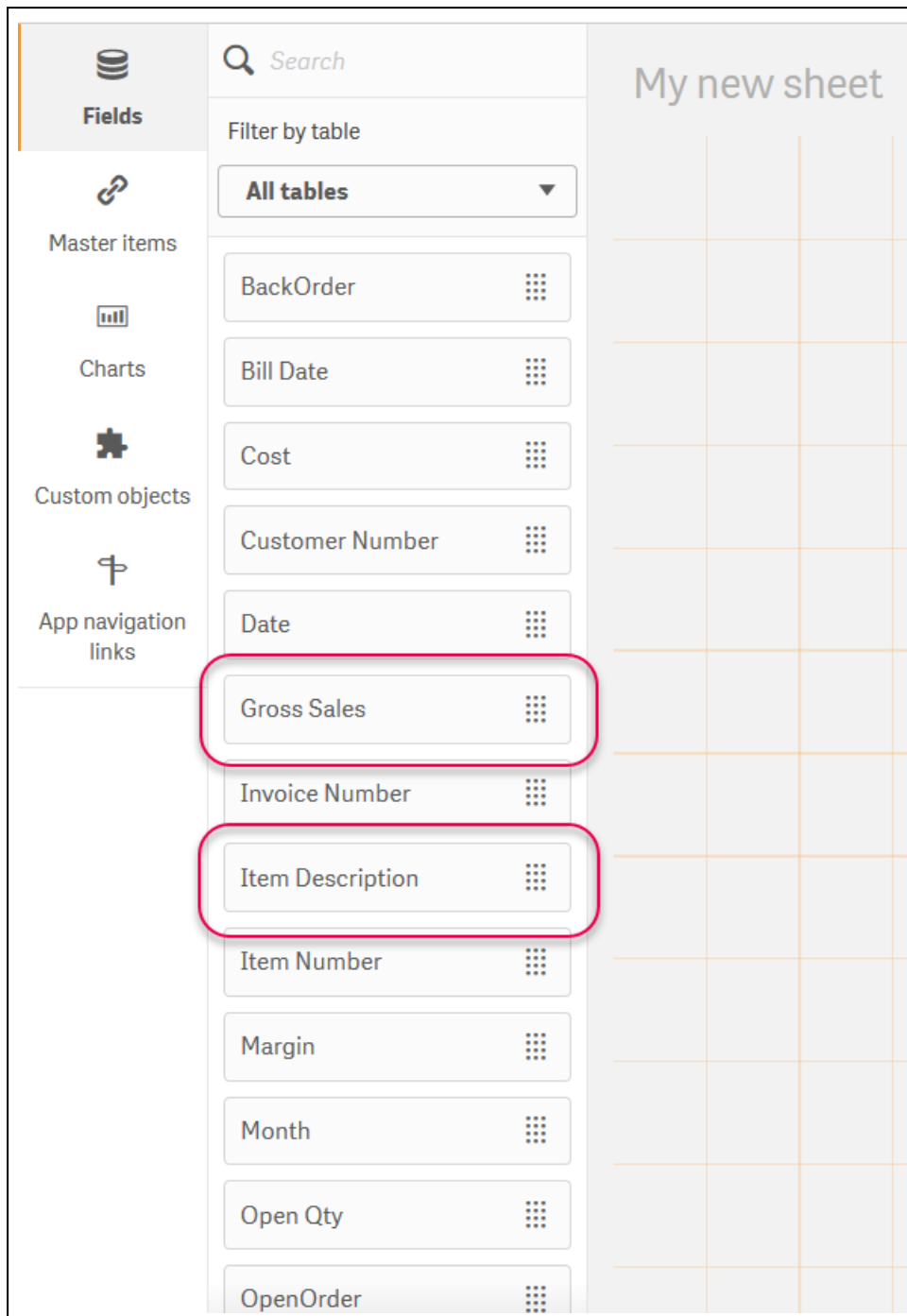
Uw script zou er als volgt moeten uitzien:

Het venster Load-script met script om velden te hernoemen



4. Klik op **Gegevens laden**. De gegevens worden geladen.
5. Open de **gegevensmodelviewer**. Bevestig dat de namen van de velden zijn hernoemd.
6. U kunt ook al uw velden in een app bekijken. Klik op het tabblad **Analyse/Werkblad** in de bovenste werkbalk. De app wordt in de werkbladweergave geopend.
7. Klik op **Werkblad bewerken** en klik dan op **Velden** in het bedrijfsmiddelenvenster. U kunt zien welke veldnamen zijn veranderd. U kunt elk van deze velden gebruiken in de visualisaties die u maakt in uw app.

Hernoemde velden in de analyseweergave



6 Gegevens verminderen

Qlik Sense geeft u verschillende manieren om de hoeveelheid gegevens die u in de app laadt te verminderen. U kunt bijvoorbeeld gegevens filteren van bestanden of van gegevensconnectoren.

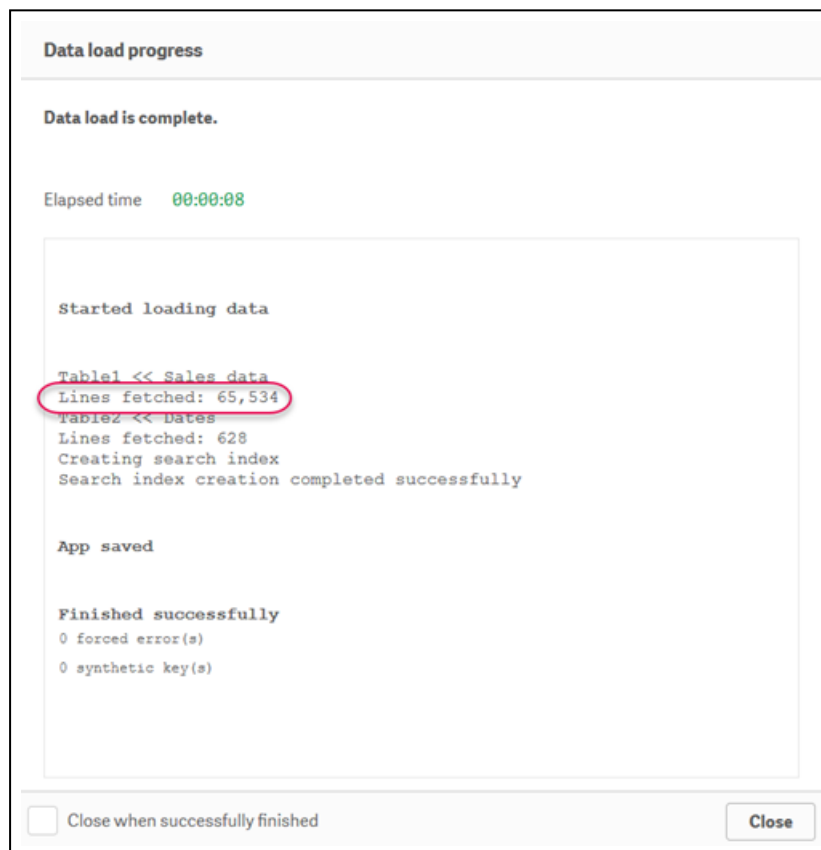
U kunt de gegevens ook direct in het load-script verminderen.

Doe het volgende:

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik op **Gegevens laden**.

Gebaseerd op het load-script dat u tot nu toe hebt geschreven, laadt Qlik Sense 65.534 regels uit het gegevensbestand *Sales.xlsx* naar *Tabel1*. Let op: *Verkoopgegevens* is de naam van het tabblad dat uw tabel bevat in het originele *Sales.xlsx*-bestand.

Venster met voortgang laden van gegevens

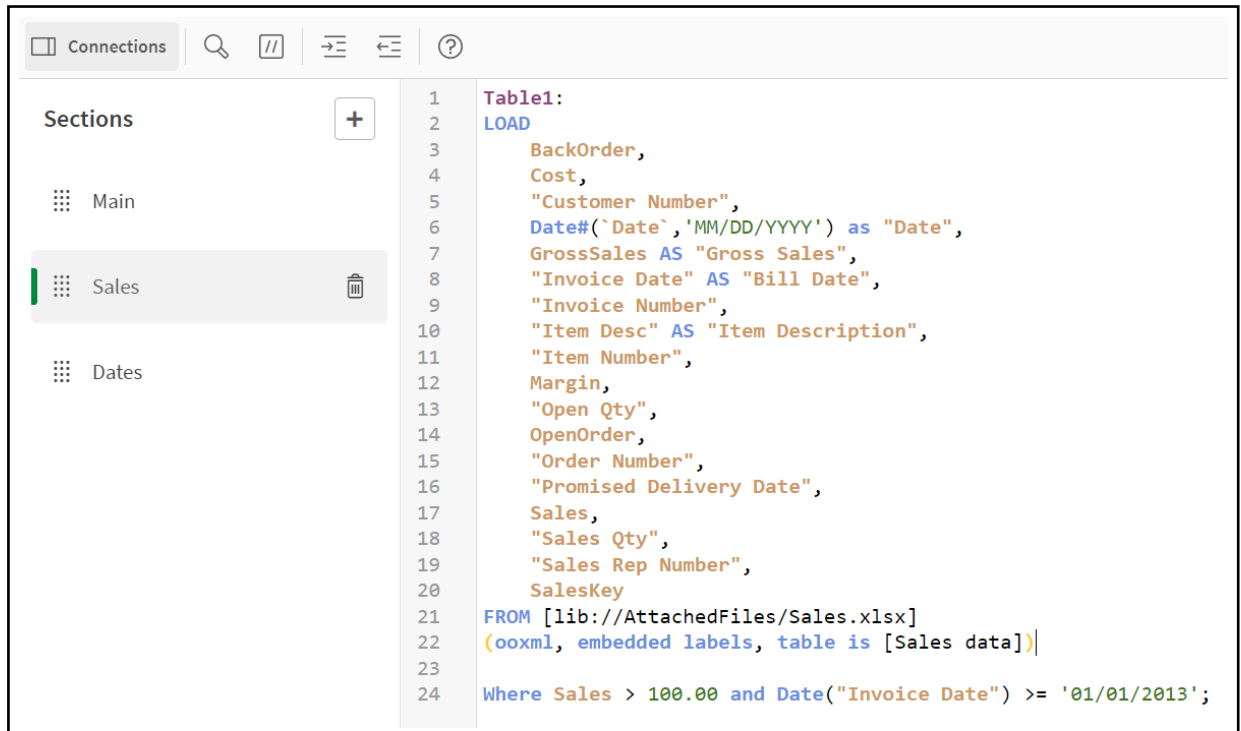


3. Klik in de editor voor laden van gegevens op het tabblad *Sales*.
4. Verwijder de puntkomma aan het eind van deze regel:
`(ooxml, embedded labels, table is [Sales data]);`
5. Voeg de volgende regel toe aan het eind van uw load-script:
`where Sales > 100.00 and Date("Invoice Date") >= '01/01/2013';`

Dit laat Qlik Sense weten om alleen gegevens te laden waar de verkoop hoger is dan 100,00 dollar. Het gebruikt ook de datumfunctie om gegevens te laden waarbij de datum gelijk is aan of later is dan 1 januari 2013.

Uw script zou er als volgt moeten uitzien:

Het venster Load-script met script om de hoeveelheid gegevens die zijn geladen te verminderen



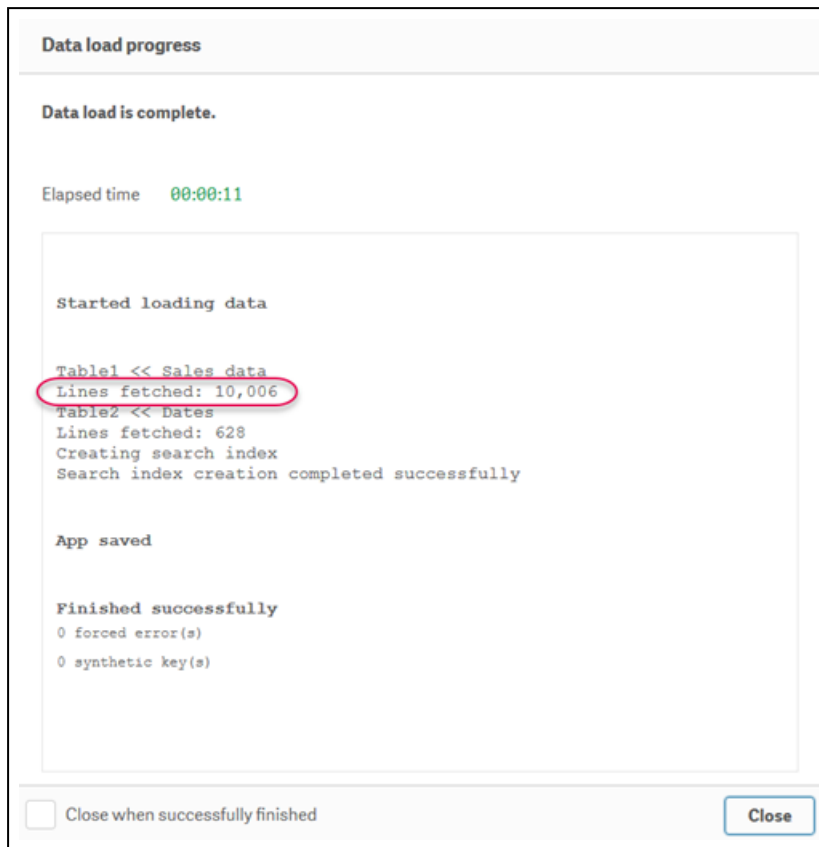
The screenshot shows the Qlik Sense Load Script editor. On the left, the 'Sections' pane lists 'Main', 'Sales' (selected), and 'Dates'. The main editor area displays a script for 'Table1' with the following code:

```
1 Table1:
2 LOAD
3     BackOrder,
4     Cost,
5     "Customer Number",
6     Date#(`Date`, 'MM/DD/YYYY') as "Date",
7     GrossSales AS "Gross Sales",
8     "Invoice Date" AS "Bill Date",
9     "Invoice Number",
10    "Item Desc" AS "Item Description",
11    "Item Number",
12    Margin,
13    "Open Qty",
14    OpenOrder,
15    "Order Number",
16    "Promised Delivery Date",
17    Sales,
18    "Sales Qty",
19    "Sales Rep Number",
20    SalesKey
21 FROM [lib://AttachedFiles/Sales.xlsx]
22 (ooxml, embedded labels, table is [Sales data])
23
24 Where Sales > 100.00 and Date("Invoice Date") >= '01/01/2013';
```

6. Klik op **Gegevens laden**.

Op basis van uw bijgewerkte load-script laadt Qlik Sense nu minder regels uit het gegevensbestand *Sales.xlsx*.

Venster met voortgang laden van gegevens met verminderde gegevensload



- Als u de gegevens van een tabel in uw app laadt, ziet u dat alleen de gegevens die voldoen aan de voorwaarden die u hebt opgesteld, worden geladen.

De tabel bevat velden *Bill Date* en *Sales*

My new sheet

Click to add title

Bill Date	Sales
01/01/2015	100.47
01/01/2015	106.93
01/01/2015	107.72
01/01/2015	108.88
01/01/2015	113.94
01/01/2015	118.89
01/01/2015	122.13
01/01/2015	123.6
01/01/2015	123.79
01/01/2015	124.32

Let erop dat we het *Sales* veld als een dimensie hebben toegevoegd. Dit hebben we gedaan zodat de *Sales*-waarden afzonderlijk worden getoond. Als we *Sales* hadden toegevoegd als een meting, zouden de waarden in plaats daarvan worden geaggregeerd per datum.

Normaal gesproken zou u *Sales* toevoegen als een meting. Met metingen hebt u de optie om waarden weer te geven als valutabedragen (bijvoorbeeld dollars) door getalnotatie te gebruiken voor de kolom.

Getalnotatie is toegepast op meting Sales

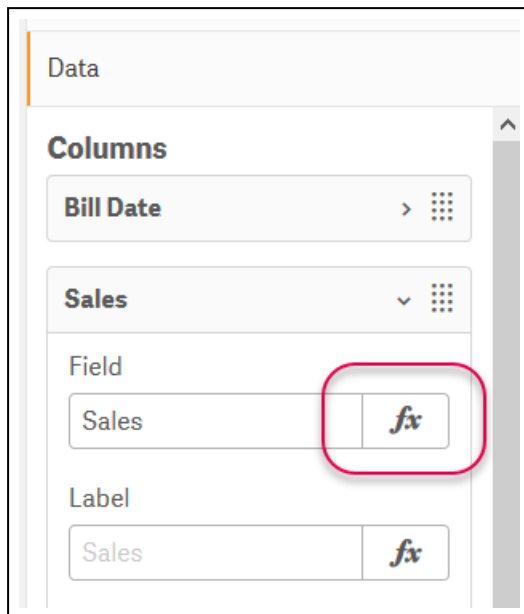
The screenshot shows the 'Data' panel in Qlik Sense. At the top, there are tabs for 'Sum' and 'Sales'. Below these, the 'Expression' field contains 'Sum(Sales)' and an 'fx' icon. The 'Label' field also contains 'Sum(Sales)' and an 'fx' icon. The 'Number formatting' dropdown is set to 'Money'. The 'Format pattern' field shows '\$ ###0.00;- \$ ###0.00'. The 'Example' field displays '\$ 1000.12'.

We hebben echter een andere benadering nodig, omdat we *Sales* gebruiken als dimensie. In dit geval gebruiken we een diagramuitdrukking. Ook al gaan we niet gedetailleerd in op diagramfuncties tijdens deze zelfstudie, het is wel een goede kans voor een klein voorbeeld.

Wanneer we *Sales* als dimensie gebruiken, worden de waarden numeriek weergegeven, zoals u in de tabel hieronder kunt zien.

Om dit te repareren, kunt u het veld *Sales* diagramuitdrukkingseeditor openen door op **fx** te klikken, zodat u vervolgens de functie *Money* gebruikt.

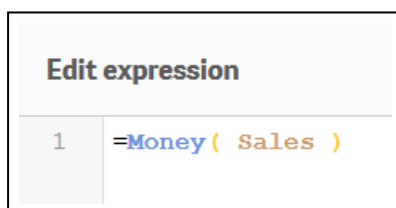
Uitdrukingseditor openen



Voer het volgende in uitdrukingseditor in.

`=Money(Sales)`

Uitdrukingseditor



De waarden in het veld worden nu weergegeven als monetaire eenheden. Deze eenheden (in dit geval dollars) worden gespecificeerd in de **Hoofdsectie** van het load-script.

De tabel bevat velden Bill Date en Sales. Sales wordt nu weergegeven in dollars

Bill Date	Money(Sales)
01/01/2015	\$100.47
01/01/2015	\$108.89
01/01/2015	\$107.72
01/01/2015	\$108.88
01/01/2015	\$115.94
01/01/2015	\$118.89
01/01/2015	\$122.13
01/01/2015	\$123.60
01/01/2015	\$123.73
01/01/2015	\$124.32

8. Nu we dit voorbeeld hebben afgerond, schakelen we de Where-opdracht uit. Vergeet niet een puntkomma toe te voegen aan het eind van uw eerste LOAD-opdracht.
Uw script zou er als volgt moeten uitzien:

Venster Load-script met Where-opdracht uitgeschakeld

The screenshot shows the Qlik Sense script editor interface. On the left, the 'Sections' pane lists 'Main', 'Sales' (selected), and 'Dates'. The main editor area displays a load script for 'Table1'. The script includes a 'LOAD' statement with various fields: 'BackOrder', 'Cost', 'Customer Number', 'Date' (formatted from 'Date' field), 'GrossSales AS "Gross Sales"', 'Invoice Date AS "Bill Date"', 'Invoice Number', 'Item Desc AS "Item Description"', 'Item Number', 'Margin', 'Open Qty', 'OpenOrder', 'Order Number', 'Promised Delivery Date', 'Sales', 'Sales Qty', 'Sales Rep Number', and 'SalesKey'. The data is loaded from '[lib://AttachedFiles/Sales.xlsx]' using the '(ooxml, embedded labels, table is [Sales data])' format. A comment at the bottom indicates that the 'Where' clause has been disabled: '// Where Sales > 100.00 and Date("Invoice Date") >= '01/01/2013';'. The script is numbered from 1 to 24.

```
1 Table1:
2 LOAD
3     BackOrder,
4     Cost,
5     "Customer Number",
6     Date#(`Date`, 'MM/DD/YYYY') as "Date",
7     GrossSales AS "Gross Sales",
8     "Invoice Date" AS "Bill Date",
9     "Invoice Number",
10    "Item Desc" AS "Item Description",
11    "Item Number",
12    Margin,
13    "Open Qty",|
14    OpenOrder,
15    "Order Number",
16    "Promised Delivery Date",
17    Sales,
18    "Sales Qty",
19    "Sales Rep Number",
20    SalesKey
21 FROM [lib://AttachedFiles/Sales.xlsx]
22 (ooxml, embedded labels, table is [Sales data]);
23
24 // Where Sales > 100.00 and Date("Invoice Date") >= '01/01/2013';
```

7 Gegevens transformeren

U kunt gegevens veranderen en manipuleren door verschillende technieken te gebruiken in Editor voor laden van gegevens.

Een van de voordelen van het bewerken van gegevens is dat u ervoor kunt kiezen om slechts een subset van de gegevens uit een bestand te laden, bijvoorbeeld enkele specifieke kolommen van een tabel, om de gegevens op efficiëntere wijze te kunnen hanteren. U kunt de gegevens ook meer dan één keer laden om de onbewerkte gegevens op te splitsen in verschillende nieuwe logische tabellen. Het is tevens mogelijk om gegevens uit meerdere bronnen te laden en samen te voegen tot één tabel in Qlik Sense.

In dit onderwerp voert u een aantal standaard gegevenstransformaties uit met behulp van een Resident-load en daarna een Preceding-load.

7.1 Resident LOAD

U kunt de bronkwalificatie Resident gebruiken in een LOAD-opdracht om gegevens van een eerder geladen tabel te laden. Dit is ook handig als u berekeningen wilt uitvoeren met gegevens die geladen zijn met een SELECT-opdracht waarbij u geen Qlik Sense-functies kunt gebruiken, zoals datum of het afhandelen van numerieke waarden.

In dit voorbeeld maakt u een nieuwe tabel met de naam *Sales_Buckets* en laadt u daarna de gegevens vanuit *Table1* met een resident load. In de tabel *Sales_Buckets* maakt u een variabele met de naam *quantity_threshold* en gebruikt u een Where-opdracht om alleen gegevens te laden die aan die drempelwaarde voldoen.

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik op het tabblad *Sales*.
3. Voeg het volgende toe aan het eind van uw script:

```
SET quantity_threshold = 12000;
```

```
Sales_Buckets:  
LOAD  
"Sales Qty" as "High_Quantity",  
"Item Description" as "Item",  
"Customer Number" as "Customer"  
Resident Table1  
where ("Sales Qty" > $(quantity_threshold));
```

Uw script zou er als volgt moeten uitzien:

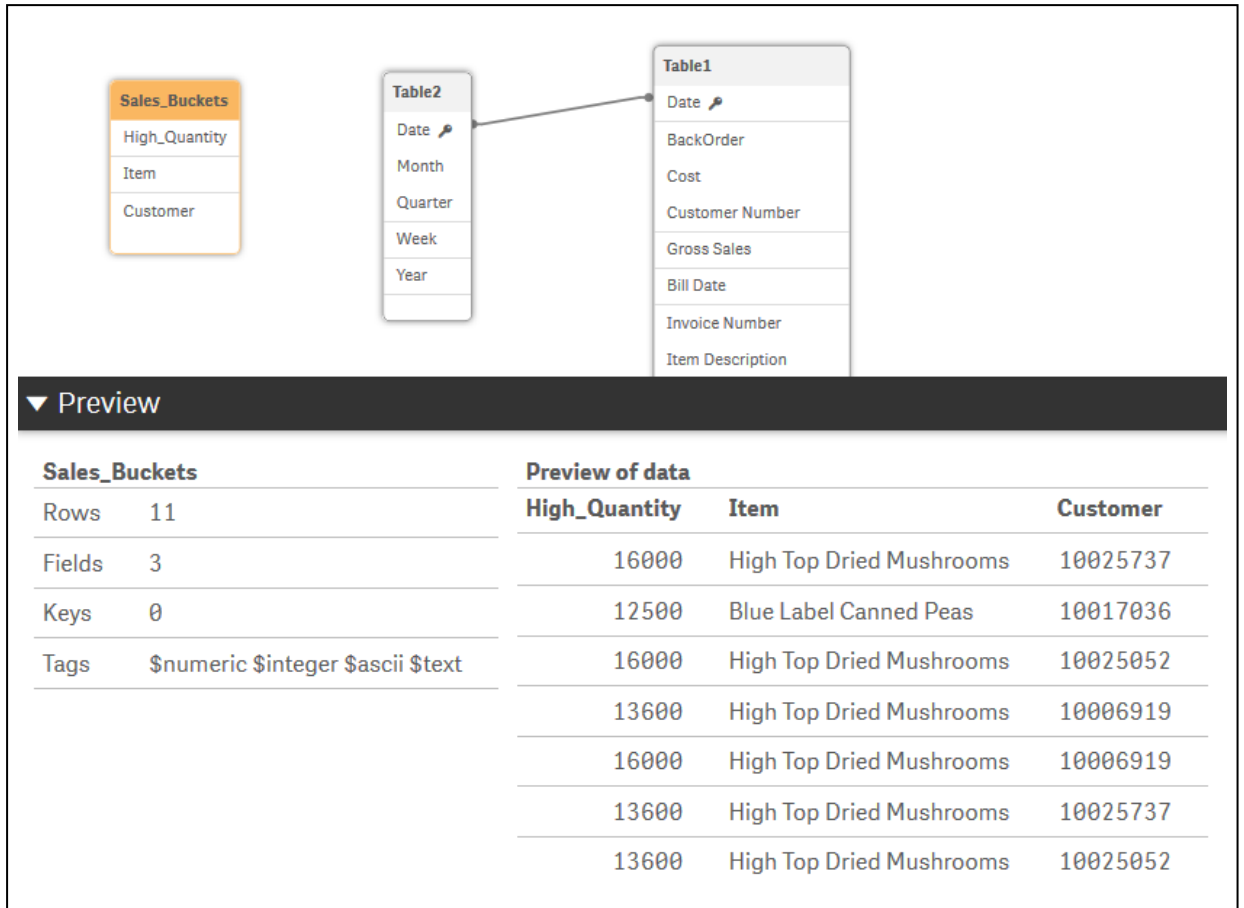
Venster load-script met script om een nieuwe tabel te maken met de naam *Sales_Buckets*

The screenshot shows the Qlik Sense script editor interface. On the left, the 'Sections' pane lists 'Main', 'Sales' (selected), and 'Dates'. The main editor area displays a script with line numbers 1 through 32. The script defines a table named 'Table1' and a new table named 'Sales_Buckets' based on 'Table1' with a filter on 'Sales Qty'.

```
1 Table1:
2 LOAD
3     BackOrder,
4     Cost,
5     "Customer Number",
6     Date#(`Date`, 'MM/DD/YYYY') as "Date",
7     GrossSales AS "Gross Sales",
8     "Invoice Date" AS "Bill Date",
9     "Invoice Number",
10    "Item Desc" AS "Item Description",
11    "Item Number",
12    Margin,
13    "Open Qty",
14    OpenOrder,
15    "Order Number",
16    "Promised Delivery Date",
17    Sales,
18    "Sales Qty",|
19    "Sales Rep Number",
20    SalesKey
21 FROM [lib://AttachedFiles/Sales.xlsx]
22 (ooxml, embedded labels, table is [Sales data]);
23
24 SET quantity_threshold = 12000;
25
26 Sales_Buckets:
27 LOAD
28     "Sales Qty" as "High_Quantity",
29     "Item Description" as "Item",
30     "Customer Number" as "Customer"
31 Resident Table1
32 Where ("Sales Qty" > $(quantity_threshold));
```

4. Klik op **Gegevens laden**.
5. Open de **gegevensmodelviewer**. U kunt zien dat u een nieuwe tabel hebt gemaakt met de naam *Sales_Buckets* met de geladen gegevens volgens de velden die u hebt gespecificeerd en de drempelwaarde die u hebt ingesteld.

Sales_Buckets-tabel in gegevensmodelviewer



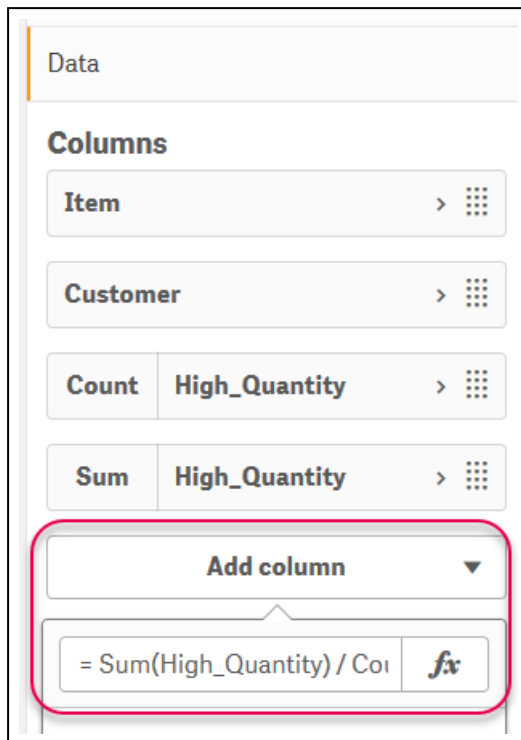
The screenshot shows the Qlik Sense data model viewer. At the top, three tables are displayed: **Sales_Buckets** (orange), **Table2** (grey), and **Table1** (grey). **Sales_Buckets** has fields: High_Quantity, Item, and Customer. **Table2** has fields: Date, Month, Quarter, Week, and Year. **Table1** has fields: Date, BackOrder, Cost, Customer Number, Gross Sales, Bill Date, Invoice Number, and Item Description. A line connects the **Date** field in **Table2** to the **Date** field in **Table1**. Below the tables is a 'Preview' section with a dropdown arrow. The preview shows the **Sales_Buckets** table with 11 rows and 3 fields. It also displays a 'Preview of data' table with columns: High_Quantity, Item, and Customer. The data rows show various quantities and item names, with some items appearing multiple times.

Sales_Buckets		Preview of data		
Rows	11	High_Quantity	Item	Customer
Fields	3	16000	High Top Dried Mushrooms	10025737
Keys	0	12500	Blue Label Canned Peas	10017036
Tags	\$numeric \$integer \$ascii \$text	16000	High Top Dried Mushrooms	10025052
		13600	High Top Dried Mushrooms	10006919
		16000	High Top Dried Mushrooms	10006919
		13600	High Top Dried Mushrooms	10025737
		13600	High Top Dried Mushrooms	10025052

- Voeg de gegevens aan een tabel toe in uw app. Voeg *Item* en *Customer* toe als dimensies. Voeg *High-Quantity* toe als een meting die is geaggregeerd op *Count* en daarna weer geaggregeerd op *Sum*. Voeg daarna een nieuwe kolom toe als een meting met de volgende formule:

$$= \text{Sum}(\text{High_Quantity}) / \text{Count}(\text{High_Quantity})$$

Nieuwe meting met formule



Uw tabel laat bijvoorbeeld zien dat *Customer* 10025737 vier grote bestellingen heeft geplaatst van *Gedroogde paddenstoelen* met een gemiddelde hoeveelheid van 14.800. Om de gegevens in de velden te sorteren, sluit u de modus **Bewerken** door op **Gereed** te klikken.

Tabel laat klanten zien die grote bestellingen hebben geplaatst

Item	Customer	Count(High_Quantity)	Sum(High_Quantity)	= Sum(High_Quantity) / Count(High_Quantity)
Totals		11	158100	14372.727272727
Blue Label Canned Peas	10017036	1	12500	12500
High Top Dried Mushrooms	10006919	3	43200	14400
High Top Dried Mushrooms	10025052	3	43200	14400
High Top Dried Mushrooms	10025737	4	59200	14800

- Nu we dit voorbeeld hebben afgerond, schakelen we het script uit voor de variabele *quantity_threshold* en de tabel *Sales_Buckets*.

Het einde van uw script zou er nu als volgt moeten uitzien:

Uitgeschakeld script

```

21 FROM [lib://AttachedFiles/Sales.xlsx]
22 (ooxml, embedded labels, table is [Sales data]);
23
24 // SET quantity_threshold = 12000;
25
26 // Sales_Buckets:
27 // LOAD
28 // "Sales Qty" as "High_Quantity",
29 // "Item Description" as "Item",
30 // "Customer Number" as "Customer"
31 // Resident Tables
32 // Where ("Sales Qty" > $(quantity_threshold));

```

7.2 Voorafgaande LOAD

Met een voorafgaande load kunt u transformaties uitvoeren en filters toepassen, zodat u de gegevens in een keer kunt laden. In wezen is het een LOAD-opdracht die moet worden geladen vanuit de onderstaande LOAD- of SELECT-opdracht, zonder dat een bronkwalificatie zoals From of Resident wordt opgegeven, zoals u normaal gesproken zou doen. Op deze manier kunt u elk gewenste aantal LOAD-opdrachten stapelen. De opdracht onderaan wordt eerst geëvalueerd, dan de opdracht erboven, enzovoort, totdat de bovenste opdracht is geëvalueerd.

Zoals u al eerder hebt gezien in deze zelfstudie, kunt u gegevens laden naar Qlik Sense met gebruik van de opdrachten LOAD en SELECT. Met elk van deze opdrachten wordt een interne tabel gegenereerd. LOAD wordt gebruikt voor het laden van gegevens vanuit bestanden of een inline-tabel, terwijl SELECT wordt gebruikt voor het laden van gegevens uit databases. In deze zelfstudie hebt u gegevens gebruikt uit bestanden. In dit voorbeeld gebruikt u een inline-tabel. Het is echter het vermelden waard dat een voorafgaande load kan worden gebruikt boven een SELECT-instructie om uw gegevens te manipuleren. De basisprincipes zijn hetzelfde, zoals u hier ziet bij het gebruik van LOAD.

Dit voorbeeld houdt geen verband met de gegevens die wij laden in deze zelfstudie. Het wordt uitsluitend gebruikt om te laten zien hoe een voorafgaande load eruit kan zien. U maakt een inline-tabel in de editor voor laden van gegevens met de naam *Transacties*. De datuminterpretatie wordt uitgevoerd in de voorafgaande LOAD, waar een nieuw bestand wordt gemaakt met de naam *transaction_date*. Dit veld wordt gemaakt vanuit het veld *sale_date*.

1. Maak een nieuwe app en noem deze *ReformatDate*.
2. Open de editor voor laden van gegevens en maak vervolgens een nieuw tabblad met de naam *TransactionData*.
3. Voeg het volgende script toe:

```

Transactions:
Load *,
Date(Date#(sale_date,'YYYYMMDD'),'DD/MM/YYYY') as transaction_date;
Load * Inline [ transaction_id, sale_date, transaction_amount, transaction_quantity,
customer_id, size, color_code
3750, 20180830, 23.56, 2, 2038593, L, Red
3751, 20180907, 556.31, 6, 203521, m, orange
3752, 20180916, 5.75, 1, 5646471, S, blue
3753, 20180922, 125.00, 7, 3036491, l, Black
3754, 20180922, 484.21, 13, 049681, xs, Red
3756, 20180922, 59.18, 2, 2038593, M, Blue
3757, 20180923, 177.42, 21, 203521, XL, Black ];

```


Uw script zou er als volgt moeten uitzien:

Load-script met voorafgaande load

Sections

Main

TransactionData

```

1 Transactions:
2 Load *,
3 Date(Date#(sale_date,'YYYYMMDD'),'DD/MM/YYYY') as transaction_date;
4 Load * Inline [ transaction_id, sale_date, transaction_amount, transaction_quantity, customer_id, size, color_code
5 3750, 20180830, 23.56, 2, 2038593, L, Red
6 3751, 20180907, 556.31, 6, 203521, m, orange
7 3752, 20180916, 5.75, 1, 5646471, S, blue
8 3753, 20180922, 125.00, 7, 3036491, l, Black
9 3754, 20180922, 484.21, 13, 049681, xs, Red
10 3756, 20180922, 59.18, 2, 2038593, M, Blue |
11 3757, 20180923, 177.42, 21, 203521, XL, Black ];

```

- Klik op **Gegevens laden**.
- Open de **gegevensmodelviewer**. Selecteer de tabel *Transacties* en vouw deze uit. U kunt zien dat alle velden zijn geladen zoals aangegeven in de * in de voorafgaande-loadinstructie. Een nieuw veld met de naam *transaction_date* is gemaakt. Het veld heeft de gewijzigde datumnotatie.

Nieuw veld met de naam transaction_date in gegevensmodelviewer

Transactions

transaction_id

sale_date

transaction_amount

transaction_quantity

customer_id

size

color_code

transaction_date

▼ Preview

Transactions		Preview of data							
Rows	7	transaction_id	sale_date	transaction_amount	transaction_quantity	customer_id	size	color_code	transaction_date
Fields	8	3750	20180830	23.56	2	2038593	L	Red	30/08/2018
Keys	0	3751	20180907	556.31	6	203521	m	orange	07/09/2018
Tags	\$numeric \$integer \$ascii \$text \$timestamp \$date	3752	20180916	5.75	1	5646471	S	blue	16/09/2018
		3753	20180922	125.00	7	3036491	l	Black	22/09/2018
		3754	20180922	484.21	13	049681	xs	Red	22/09/2018
		3756	20180922	59.18	2	2038593	M	Blue	22/09/2018
		3757	20180923	177.42	21	203521	XL	Black	23/09/2018

8 Aaneenschakelen

Een aaneenschakeling is een bewerking waarmee twee tabellen worden samengevoegd tot één tabel.

De twee tabellen worden aan elkaar toegevoegd door de één boven op de andere te stapelen, met een kolom voor elke distinctieve kolomnaam. De gegevens worden niet gewijzigd en de resulterende tabel bevat hetzelfde aantal records als de twee originele tabellen samen. Verschillende aaneenschakelingsbewerkingen kunnen na elkaar worden uitgevoerd, zodat de resulterende tabel de aaneenschakeling is van verschillende tabellen.

8.1 Automatisch aaneenschakelen

Als de veldnamen en het aantal velden van twee of meer geladen tabellen exact gelijk zijn, zal Qlik Sense de inhoud van de verschillende opdrachten automatisch aaneenschakelen in één tabel.

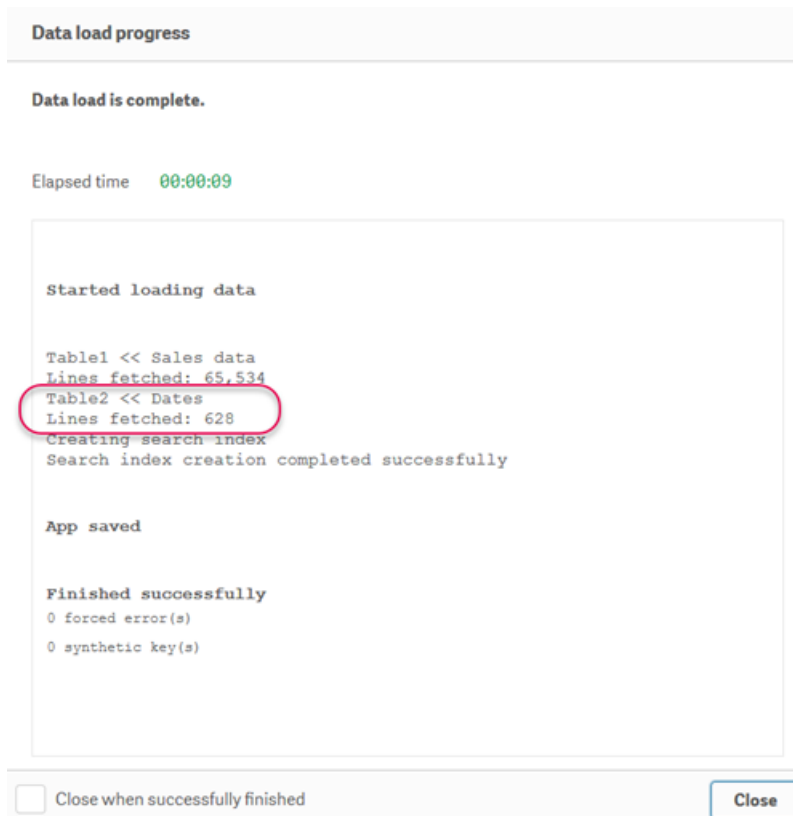
Automatisch aaneenschakelen is alleen mogelijk als het aantal en de namen van de velden exact gelijk zijn. De volgorde van de twee opdrachten LOAD is willekeurig, maar de tabel krijgt de naam van de tabel die als eerste is geladen.

Doe het volgende:

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik op het tabblad **Dates**.
3. Klik op **Gegevens laden**.

Op basis van het load-script dat u tot nu toe hebt geschreven laadt Qlik Sense 628 regels uit het gegevensbestand *Dates.xlsx* in *Table2*.

Venster met voortgang laden van gegevens



4. Kopieer de opdracht LOAD voor *Table2* naar een nieuwe regel in de sectie *Dates* van het script. Hierdoor worden de gegevens tweemaal geladen. Geef de tweede tabel de naam *Table2a*. U kunt ook het bestaande script verwijderen en het volgende kopiëren en plakken:

Table2:

```

LOAD
    "Date",
    Month (Date) as "Month",
    Quarter,
    "Week",
    "Year"
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);

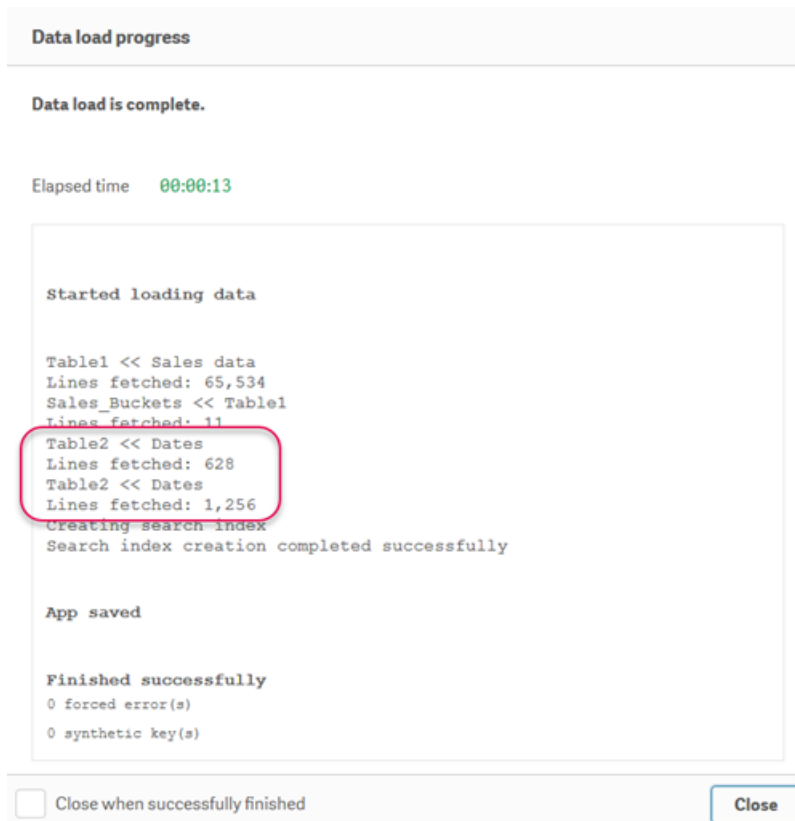
Table2a:
LOAD
    "Date",
    Month (Date) as "Month",
    Quarter,
    "Week",
    "Year"
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);

```

5. Klik op **Gegevens laden**.

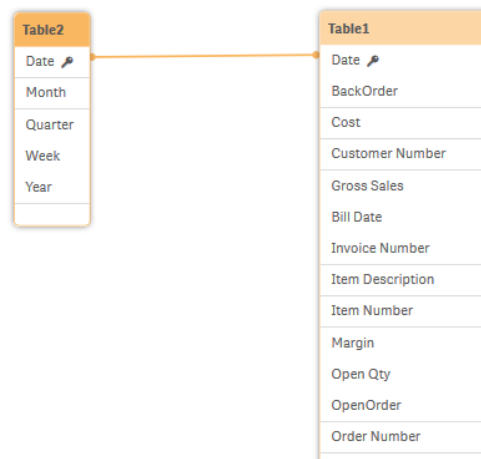
Qlik Sense laadt *Table2* niet, gevolgd door *Table2a*. In plaats daarvan wordt gedetecteerd dat *Table2a* dezelfde veldnamen en hetzelfde aantal velden heeft als *Table2*. Daarna worden de gegevens van *Table2a* aan *Table2* toegevoegd, waarna de tabel *Table2a* wordt verwijderd. Het resultaat is dat *Table2* nu 1256 regels heeft.

Aaneenschakeling in venster met voortgang laden van gegevens

6. Open de **gegevensmodelviewer**.7. Klik op **Voorbeeld weergeven**.

Alleen *Table2* is gemaakt. Selecteer *Table2*. De tabel heeft 256 rijen.

Gegevensmodelviewer toont Table2



▼ Preview

Table2		Preview of data				
Rows	1256	Date	Month	Quarter	Week	Year
Fields	5	01/12/2011	Jan	Q1	3	2011
Keys	1	01/13/2011	Jan	Q1	3	2011
Tags	\$key \$numeric \$integer \$timestamp \$date \$ascii \$text	01/18/2011	Jan	Q1	3	2011
		01/19/2011	Jan	Q1	4	2011
		01/20/2011	Jan	Q1	4	2011
		01/21/2011	Jan	Q1	4	2011
		01/22/2011	Jan	Q1	4	2011

8.2 Geforceerd aaneenschakelen

Zelfs als twee of meer tabellen niet exact dezelfde set velden hebben, kunt u de tabellen in Qlik Sense toch geforceerd aaneenschakelen. Daarvoor gebruikt u de prefix Concatenate in het script, waarmee een tabel met een andere benoemde tabel of met de als laatste gemaakte tabel wordt aaneengeschaakeld.

Doe het volgende:

1. Bewerk de opdracht LOAD voor *Table2a* door Concatenate toe te voegen en *Week* als opmerking aan te merken.

Uw script moet er nu als volgt uitzien:

Table2a:

```
Concatenate LOAD
"Date",
Month (Date) as "Month",
Quarter,
// "Week",
"Year"
```

```
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);
```

Door *Week* uit te commentariëren zorgen wij ervoor dat de tabellen niet identiek zijn.

2. Klik op **Gegevens laden**.
3. Open de **gegevensmodelviewer**.
Nu kunt u zien dat tabel *Table2a* niet is gemaakt.
4. Klik op *Table2* in de gegevensmodelviewer en klik op **Voorbeeld**.
De tabel bevat de velden *Date*, *Month*, *Quarter*, *Week* en *Year*. Het veld *Week* wordt nog steeds weergegeven, omdat het is geladen vanuit *Table2*.
5. Klik op *Week* in *Table2*. Het voorbeeld geeft aan dat het aantal niet-null-waarden voor het veld 628 is. Als u echter op een van de overige velden klikt, ziet u dat het aantal niet-null-waarden 1256 is. *Week* is slechts eenmaal geladen, vanuit *Table2*. Het aantal waarden of records is de som van het aantal records in *Table2* en *Table2a*.

8.3 Aaneenschakelen voorkomen

Als de veldnamen en het aantal velden van twee of meer geladen tabellen exact gelijk zijn, zal Qlik Sense de inhoud van de verschillende opdrachten automatisch aaneenschakelen in één tabel. Dit kan worden voorkomen met een NoConcatenate-opdracht. De tabel die wordt geladen met de toegewezen LOAD- of SELECT-instructie, wordt vervolgens niet aaneengeschakeld met de bestaande tabel.

Doe het volgende:

1. Om de inhoud van de twee tabellen volledig te kunnen scheiden voegt u NoConcatenate toe aan de opdracht LOAD in *Table2a* en geeft u de velden nieuwe namen om te voorkomen dat Qlik Sense een synthetische sleutel op basis van de overeenkomstige velden creëert. Merk *Week* in *Table2* aan als opmerking, zodat de twee tabellen dezelfde waarden bevatten.

Uw script moet er nu als volgt uitzien:

Table2:

```
LOAD
  "Date",
  Month (Date) as "Month",
  Quarter,
  "Week",
  "Year"
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);
```

Table2a:

```
NoConcatenate LOAD
  "Date" as "Date2",
  Month (Date) as "Month2",
  Quarter as "Quarter2",
  "Week" as "week2",
```

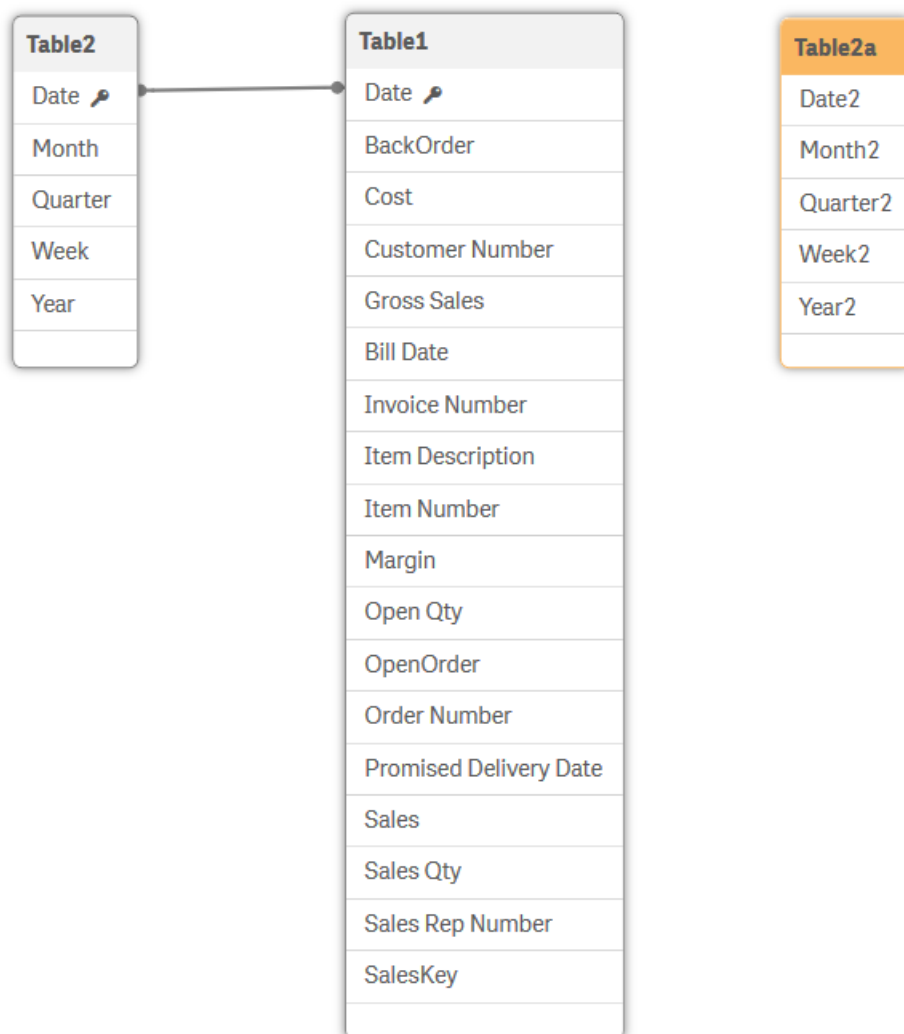
```
"Year" as "Year2"
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);
```

2. Klik op **Gegevens laden**.

3. Open de **gegevensmodelviewer**.

Nu kunt u zien dat de twee tabellen volledig van elkaar zijn gescheiden.

De gegevensmodelviewer met Table2 en Table 2a



4. Nu we onze demonstratie van aaneenschakeling hebben voltooid, hebben we *Table2a* niet langer nodig. Verwijder alle rijen in de opdracht LOAD voor *Table2a* en klik op **Gegevens laden**.

9 Cirkelreferenties

Bij cirkelreferenties ("lussen") in een gegevensstructuur, worden de tabellen zodanig geassocieerd dat er sprake is van meer dan één pad met associaties tussen velden.

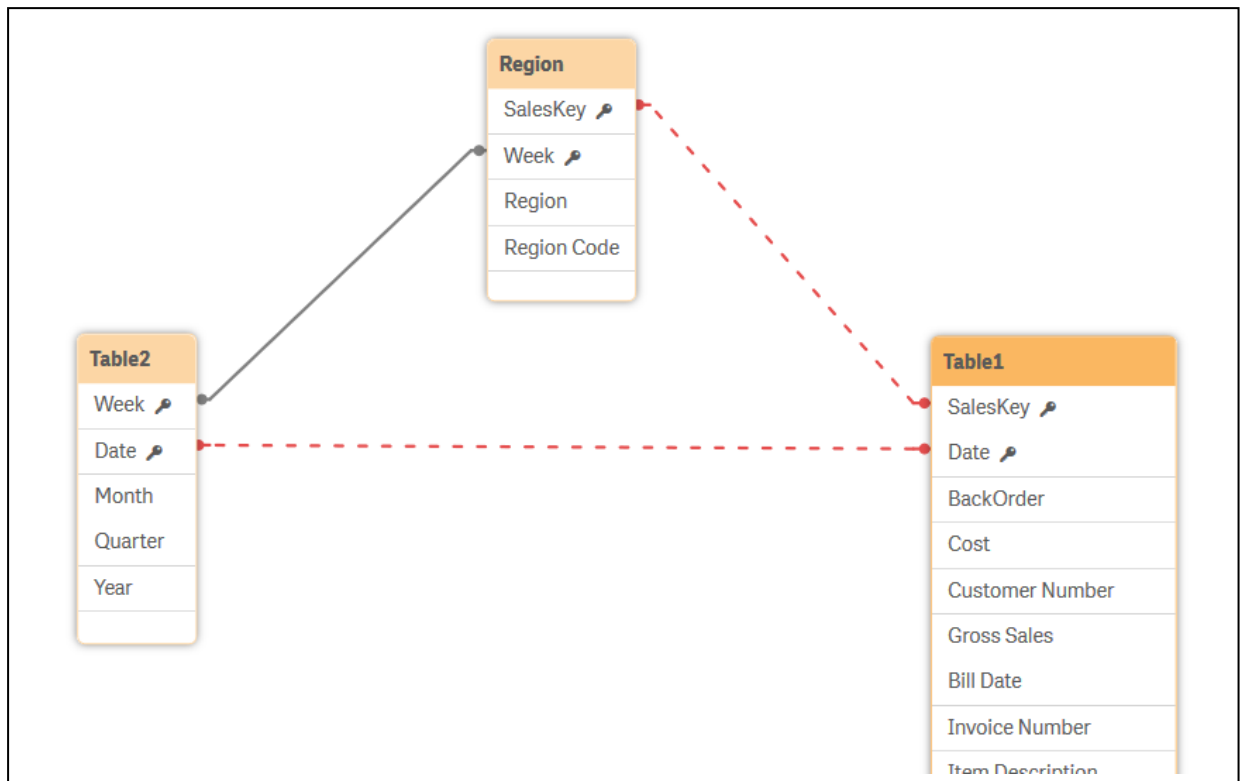
Doe het volgende:

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik **+** om een nieuwe scriptsectie toe te voegen.
3. Geef de sectie *Region* een naam.
4. Onder **AttachedFiles** in het rechtermenu klikt u op **Gegevens selecteren**.
5. Upload en selecteer *Region.txt*. Het venster Voorbeeld gegevens wordt geopend.
6. Selecteer alle velden en zorg ervoor dat **Ingesloten veldnamen** onder **Veldnamen** is geselecteerd om de namen van de tabelvelden mee te nemen bij het laden van de gegevens.
7. Klik op **Script invoegen**.
8. Klik op **Gegevens laden**.

Het lijkt er in dit geval op dat er iets is misgegaan met het laden van uw gegevens. Er is een cirkelreferentie gemaakt. In het **voortgangsvenster voor het laden van gegevens** wordt een foutbericht weergegeven waarin wordt aangegeven dat een cirkelreferentie is gevonden tijdens het laden. Het laden wordt echter voltooid en de app wordt opgeslagen.

9. Open de **gegevensmodelviewer**.
U kunt de tabellen rondslepen om deze op zodanige wijze te ordenen dat de verbindingen tussen de tabellen gemakkelijk te zien zijn.

Gegevensmodelviewer met cirkelreferentie



De rood gestippelde lijnen geven aan dat een cirkelreferentie is gemaakt. Dit is iets dat u moet vermijden omdat het tot onduidelijkheden bij de gegevensinterpretatie kan leiden.

9.1 Cirkelreferenties oplossen

Als we willen begrijpen waardoor de cirkelreferenties werden veroorzaakt, moeten we uw tabellen nader in detail gaan bekijken in de **gegevensmodelviewer**.

Als u kijkt naar *Table1* en de *Table2* in de schermafbeelding hierboven, ziet u dat ze het veld *Date* gemeen hebben. U ziet ook dat *Table1* en *Region* het veld *SalesKey* gemeen hebben. Tot slot is te zien dat *Table2* en *Region* het veld *Week* gemeen hebben. Dit betekent dat er een lus, een cirkelreferentie, is ontstaan. Aangezien dit later tot problemen bij de gegevensanalyse kan leiden, gaan wij deze cirkelreferentie verwijderen.

De gemakkelijkste manier om dit te doen is door een van de velden te hernoemen of te verwijderen. In ons geval hebben wij een aantal gegevens geladen die wij niet nodig hebben voor onze app, zodat wij deze kunnen verwijderen.

Doe het volgende:

1. Open de **Editor voor laden van gegevens**.
2. Klik op de sectie *Region* en verwijder de volgende twee rijen uit de opdracht *LOAD*:
 "week",
 SalesKey

3. Verwijder ook de komma na "Region Code".

Uw script moet er nu als volgt uitzien:

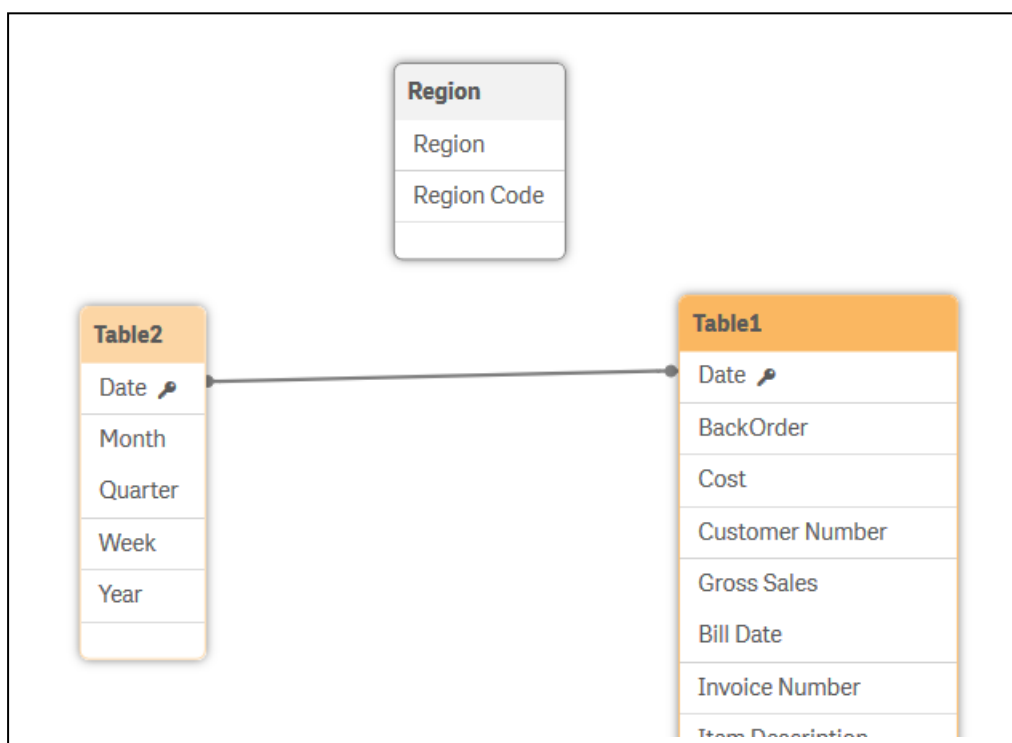
```
LOAD
    Region,
    "Region Code"
FROM [lib://AttachedFiles/Region.txt]
(txt, codepage is 28591, embedded labels, delimiter is '\t', msq);
```

4. Klik op **Gegevens laden**.

5. Open de **gegevensmodelviewer**.

De ongewenste verwijzingen naar *Region* zijn verwijderd.

Gegevensmodelviewer waarin te zien is dat de cirkelreferentie verwijderd is



10 Synthetische sleutels

Wanneer twee of meer interne tabellen twee of meer velden gemeenschappelijk hebben, duidt dit op een samengestelde-sleutelrelatie. Qlik Sense handelt dit af met behulp van synthetische sleutels. Deze sleutels zijn anonieme velden die alle voorkomende combinaties van de samengestelde sleutel vertegenwoordigen.

Een hoger aantal samengestelde sleutels kan resulteren in een hoger geheugengebruik en kan van invloed zijn op de prestaties. Dit is ook afhankelijk van de hoeveelheid gegevens, tabelstructuur en andere factoren.

Wanneer er meerdere synthetische sleutels zijn die van elkaar afhankelijk zijn, is het goed gebruik om deze te verwijderen.

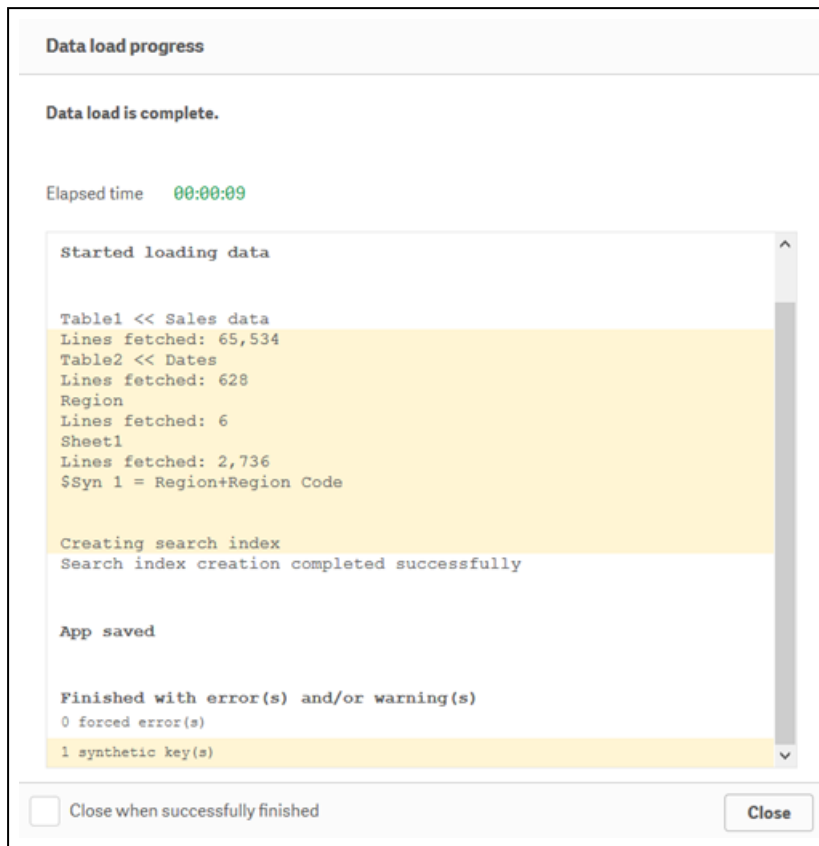
Nu wordt het tijd om onze definitieve gegevensverzameling te gaan laden.

Doe het volgende:

1. Open de **Editor voor laden van gegevens** in de *Scripting Tutorial* app.
2. Klik **+** om een nieuwe scriptsectie toe te voegen.
3. Geef de sectie *Customers* een naam.
4. Onder **AttachedFiles** in het rechtermenu klikt u op **Gegevens selecteren**.
5. Upload en selecteer *Customers.xlsx*. Het venster Voorbeeld gegevens wordt geopend.
6. Select *Sheet1*
7. Klik op **Script invoegen**.
8. Klik op **Gegevens laden**.

Nu kunt u in het voortgangsvenster voor het laden van gegevens zien dat een synthetische sleutel is gemaakt.

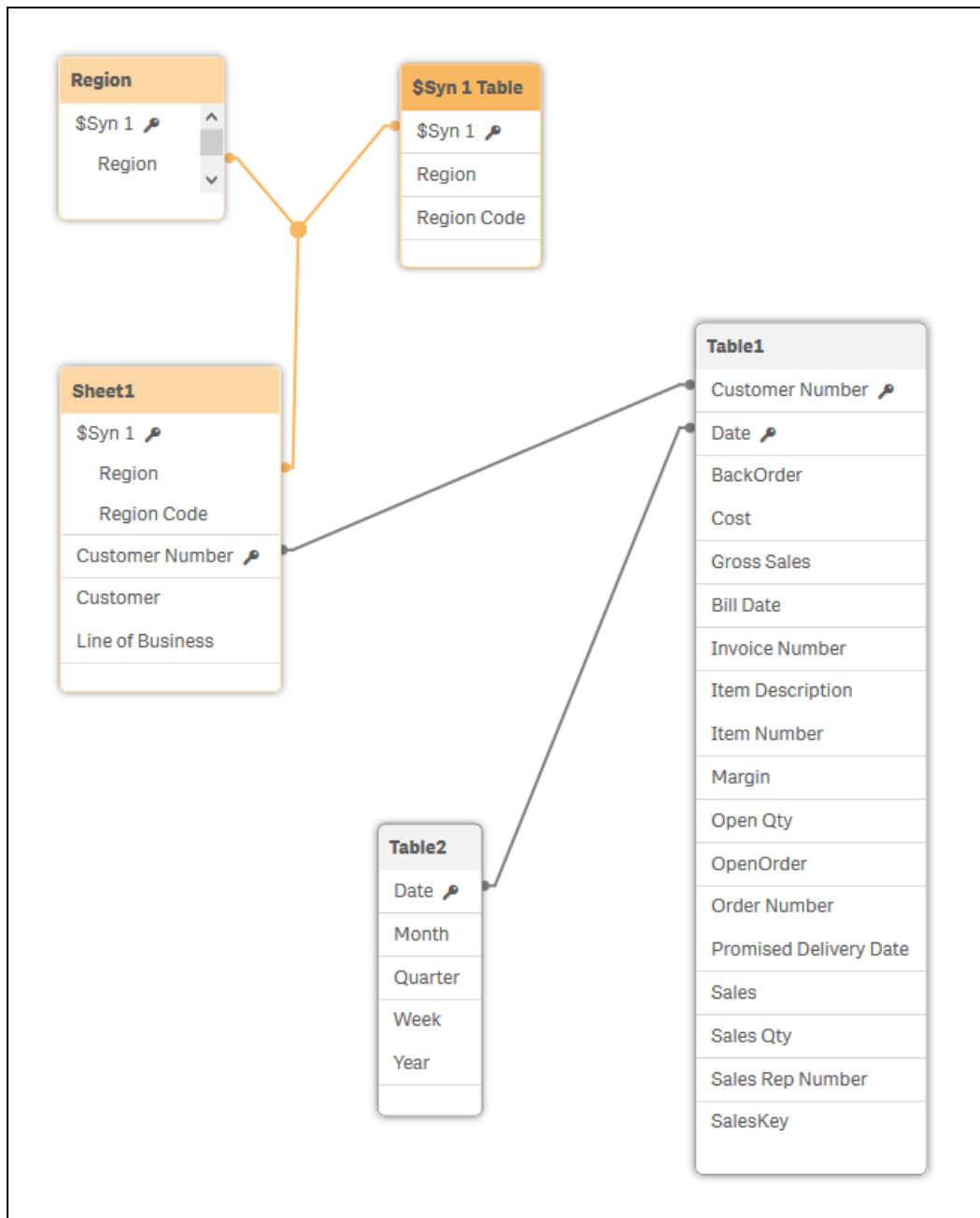
Venster met voortgang gegevensload met melding van synthetische sleutel



9. Open de **gegevensmodelviewer**.

We kunnen zien dat een synthetische sleutel is gemaakt doordat er een nieuwe tabel, *\$Syn 1 Table*, is gemaakt. Deze bevat alle velden, *Region* en *Region code*, die de verbonden tabellen *Sheet1* en *Region* gemeenschappelijk hebben. In dit geval maakt deze tabel de verbindingen een beetje verwarrend en misleidend, dus kan deze beter niet worden bewaard.

Gegevensmodelviewer Synthetische sleutel



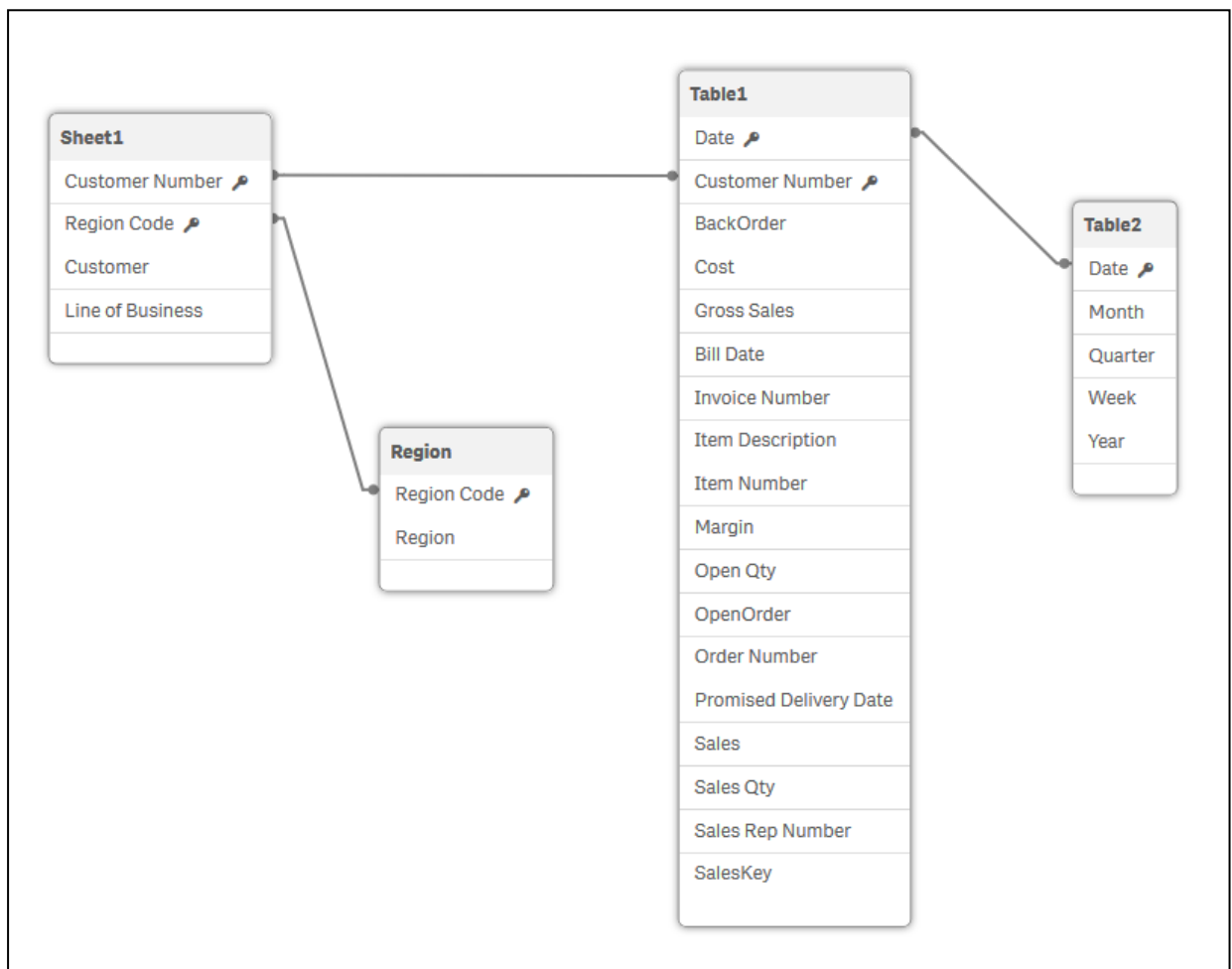
10.1 Synthetische sleutels oplossen

De gemakkelijkste manier om synthetische sleutels te verwijderen is door de naam van een of meer velden in de tabellen te wijzigen. Dit kan worden gedaan bij het laden van de gegevens. Nu doorlopen we de stappen voor het verwijderen van een synthetische sleutel.

Doe het volgende:

1. Open de **Editor voor laden van gegevens**.
2. Klik op de sectie *Klanten* en verwijder de rij uit de **LOAD**-opdracht met de tekst:
Region,
3. Klik op **Gegevens laden**.
4. Open de **gegevensmodelviewer**.
De synthetische sleutel is verwijderd.

Gegevensmodelviewer toont dat synthetische sleutel is verwijderd



11 Gegevens in een app gebruiken

Als afronding van deze zelfstudie gaat u nu uw geladen gegevens in een visualisatie in uw app opnemen.

11.1 Een diagram toevoegen

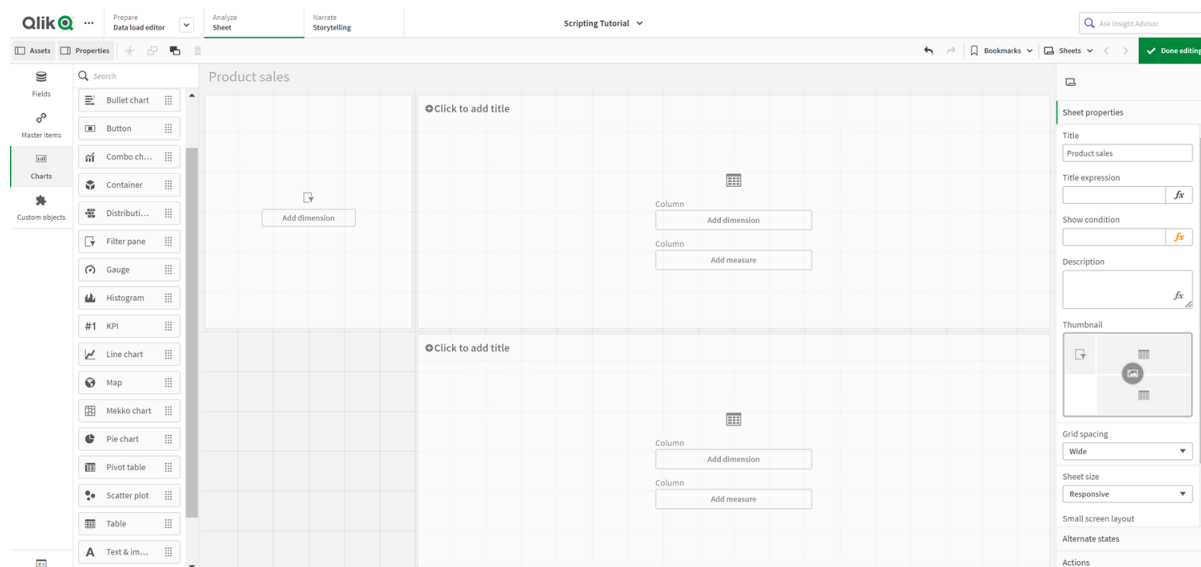
Zodra uw gegevens zijn geladen, kunt u een aantal diagrammen maken met de gegevens. Diagrammen worden vaak ook visualisaties genoemd. Pas nadat u ook de vereiste dimensies en metingen hebt toegevoegd is een diagram voltooid. U begint door alle diagrammen toe te voegen en vervolgens dimensies en metingen toe te voegen.

Doe het volgende:

1. Maak een nieuw werkblad in uw *Scripting Tutorial*-app.
2. Sleep een filtervak vanaf het tabblad **Diagrammen** naar het werkblad en gebruik de grepen om het formaat aan te passen, zodat het 3 vakjes breed en 4 vakjes hoog is. Plaats het filtervak in de linkerbovenhoek van het werkblad.
3. Sleep een staafdiagram naar de rechterbovenhoek. Maak dit 5 cellen hoog en breed genoeg om tot aan de rand van het werkblad te reiken.
4. Sleep een lijndiagram naar de resterende ruimte.

De pictogrammen op het werkblad geven aan wat voor soort diagram u hebt toegevoegd. Nu kunt u dimensies en metingen toevoegen aan uw diagrammen om deze te voltooien als visualisaties.

Qlik Sense-werkblad met lege diagrammen




11.2 Dimensies en metingen toevoegen

De volgende stap is het toevoegen van dimensies en metingen. U begint met het toevoegen van tijdsdimensies aan het filtervak linksboven. Het voordeel van een filtervak is dat u er ruimte mee bespaart. In plaats van een filtervak voor elk *Year*, *Quarter*, *Month* en *Week*, gebruikt u maar één filtervak voor hetzelfde doel.

Dimensies maken en toevoegen

Doe het volgende:


1. Klik in het bedrijfsmiddelenvenster aan de linkerkant op  om **Velden** te openen. Hier vindt u alle velden in alle tabellen die u in de editor voor het laden van gegevens hebt geladen.
2. Blader naar de onderste vermelding in de lijst en klik op het veld *Year*. Sleep het naar het midden van het filtervak linksboven.
3. Voeg op dezelfde wijze *Quarter*, *Month* en *Week* toe aan het filtervak.
U hebt nu een filtervak met vier dimensies gemaakt: *Year*, *Quarter*, *Month* en *Week*.

Metingen maken en toevoegen

De meeste visualisaties hebben zowel dimensies als metingen nodig. Een meting is het resultaat van een aggregatie-uitdrukking, wat meestal een algemene functie is, zoals **Sum**, **Max**, **Min**, **Avg** (gemiddelde) of **Count**.


In het staafdiagram toont u de verkoop per regio.

Doe het volgende:

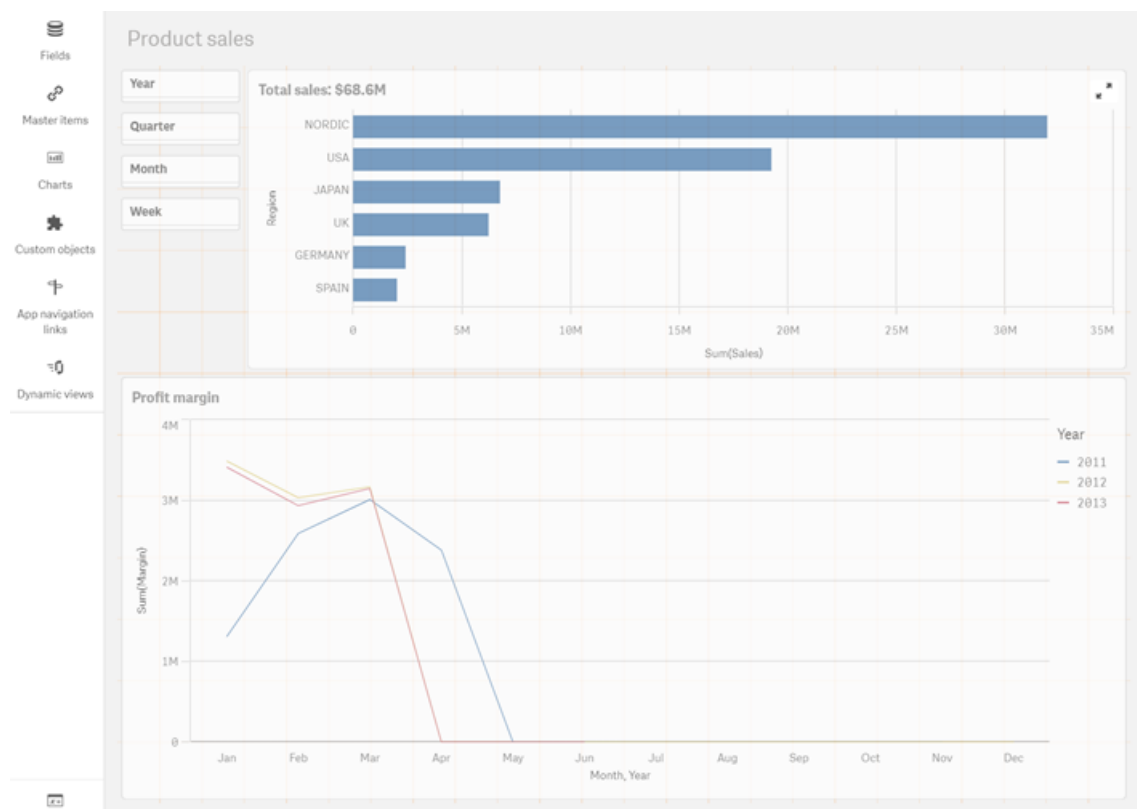
1. Klik op **Velden** .
2. Klik op het veld *Region* en sleep het naar het midden van het staafdiagramgebied.
3. Klik op **'Region' toevoegen**.
4. Klik op het veld *Sales* en sleep het naar het midden van het staafdiagramgebied.
5. Klik op **Toevoegen als meting > Sum(Sales)**.
6. Klik in het eigenschappenvenster aan de rechterzijde op **Uiterlijk** en vervolgens op **Presentatie**.
Selecteer **Horizontaal**.
De staven worden nu horizontaal weergegeven.
7. Klik in het eigenschappenvenster aan de rechterkant op **Sorteren**.
De sorteervolgorde wordt weergegeven.
8. Sleep *Sum([Sales])* boven *Region*, zodat de dimensies worden gesorteerd op *Sum([Sales])* (waarde van de meting) in plaats van op *Region* (waarde van de dimensie, alfabetisch).
Het staafdiagram is voltooid, met de verkoopresultaten voor de verschillende regio's. Dit is een basis-staafdiagram. In de eigenschappenvenster (aan de rechterkant) zijn heel veel opties om het te verbeteren. U kunt bijvoorbeeld het titelgebied gebruiken voor iets anders dan alleen de titel.
9. Plak het volgende in het titelveld van het staafdiagram:
`= 'Total sales: $' & Round(Sum(Sales)/1000000, 0.1) & 'M'`
10. Druk op Enter.

De laatste visualisatie op dit werkblad is een lijndiagram.

Doe het volgende:

1. Klik op **Velden** .
2. Klik op het veld *Month* en sleep het naar het midden van het lijndiagramgebied.
3. Klik op **'Month' toevoegen**.
4. Klik op het veld *Year* en sleep het naar het midden van het lijndiagramgebied.
5. Klik op **'Year' toevoegen**.
6. Klik op het veld *Margin* en sleep het naar het midden van het lijndiagramgebied.
7. Klik op **Toevoegen als meting > Sum(Margin)**.
8. Voeg de titel *Profit margin* bovenaan het lijndiagram toe.

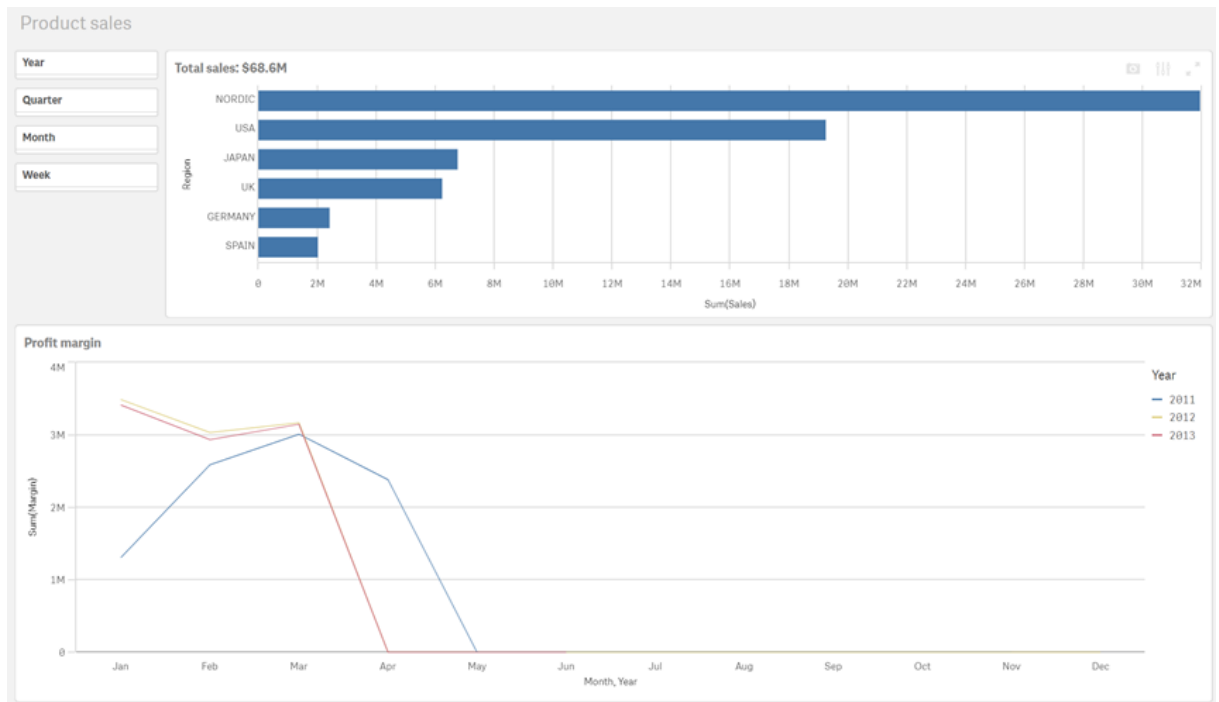
Diagrammen met gegevens



9. Stop met de bewerking van het werkblad.
Het werkblad is nu voltooid en u kunt gaan rondklikken en de inhoud van het werkblad bewerken.

11 Gegevens in een app gebruiken

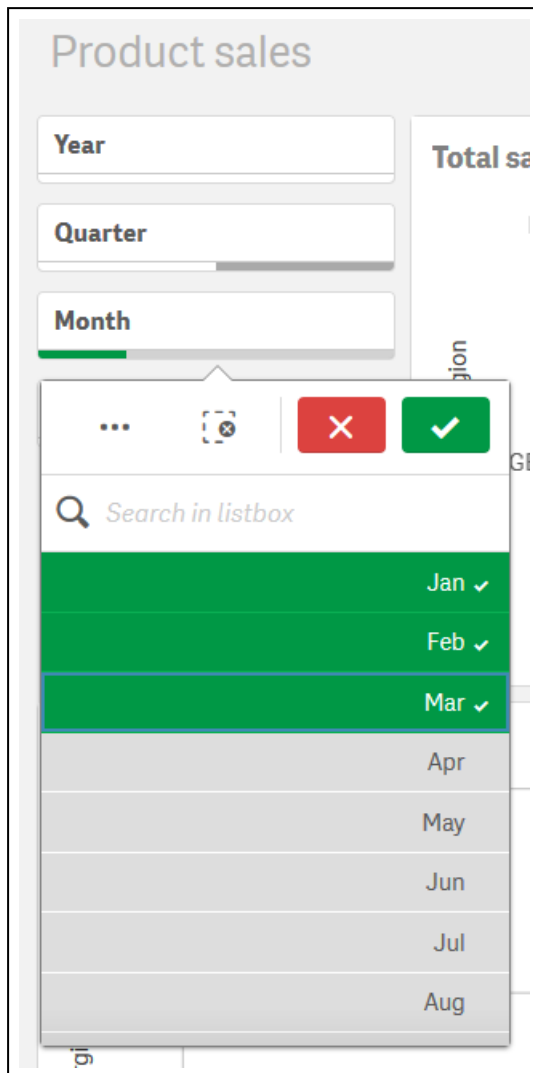
Voltooid werkblad



Er zijn weinig gegevens beschikbaar voor ons diagram na het eind van de maand maart voor elk jaar, omdat we de hoeveelheid van gegevens voor maandelijkse verkopen in ons originele bestand *Sales.xlsx* hebben beperkt. U kunt selecties maken in uw filtervak, zodat u alleen de eerste drie maanden van elk jaar vergelijkt.

10. Klik op het veld *Month* in het filtervak en selecteer daarna *Jan*, *Feb* en *Mar*.

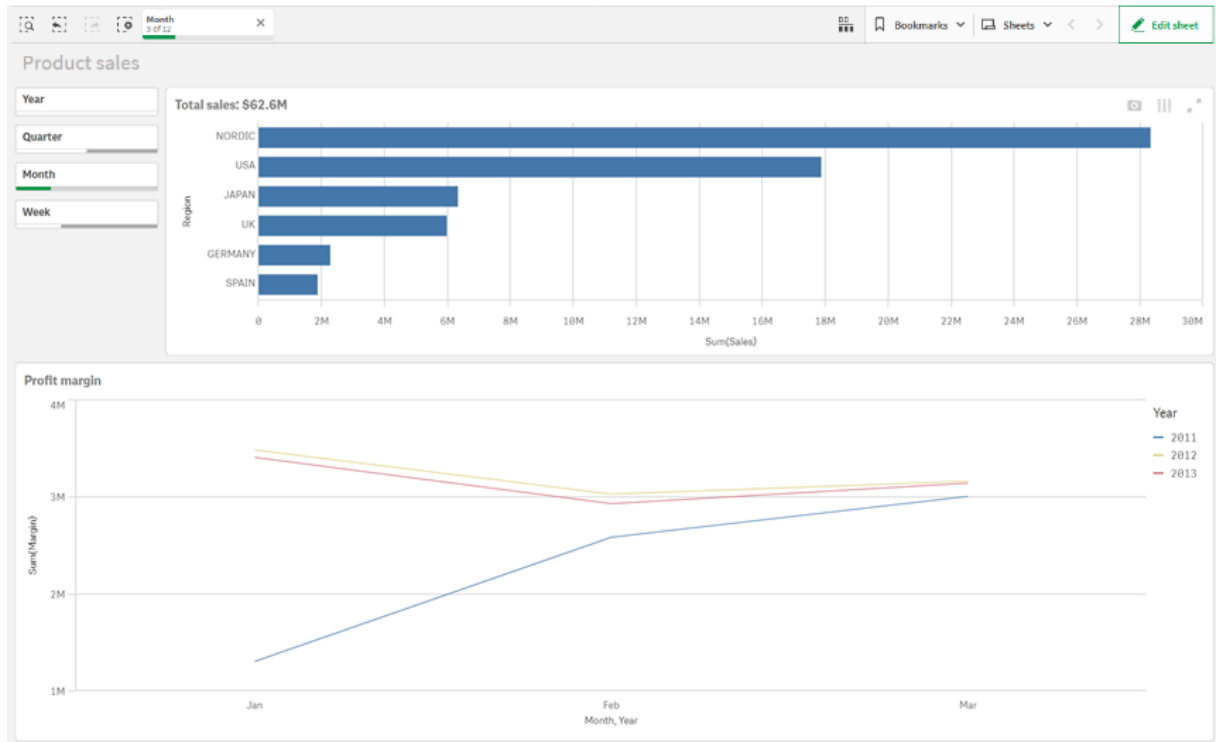
Filtervak



11. Sluit het filtervak. Het *Profit margin*-diagram toont de gegevens van de eerste drie maanden van elk jaar.

11 Gegevens in een app gebruiken

Het Profit margin-diagram wordt bijgewerkt op basis van selecties



11.3 Hartelijk dank!

U hebt deze zelfstudie nu voltooid en hopelijk beschikt u nu over basiskennis van het gebruiken van scripts in Qlik Sense. Breng een bezoek aan onze website voor meer inspiratie voor uw apps.