



Tutorial - Skripterstellung für Anfänger

Qlik Sense®

May 2024

Copyright © 1993-jjjj QlikTech International AB. Alle Rechte vorbehalten.

1 Herzlich willkommen!	4
1.1 Lerninhalte	4
1.2 Zielgruppe für diesen Kurs	4
1.3 Paketinhalt	4
1.4 Lektionen in diesem Tutorial	5
1.5 Weitere Informationsquellen und Ressourcen	5
2 Skripterstellung im Dateneditor	6
2.1 Dateneditor	6
2.2 Skript-Editor	7
Zugriff auf die Syntaxhilfe für Befehle und Funktionen	7
Kommentieren im Skript	8
Einrücken von Code	8
Einfügen eines vorbereiteten Test-Skripts	8
3 LOAD- und SELECT-Anweisungen	9
4 Auswählen und Laden von Daten	10
5 Umbenennen von Feldern	18
6 Reduzieren von Daten	20
7 Umwandeln von Daten	25
7.1 Resident LOAD	25
7.2 Vorangehendes LOAD	29
8 Zusammenfassung von Tabellen	31
8.1 Automatische Zusammenfassung	31
8.2 Manuelle Zusammenfassung	34
8.3 Zusammenfassung von Tabellen verhindern	35
9 Zirkelbezüge	37
9.1 Auflösen von Zirkelbezügen	38
10 Synthetische Schlüssel	40
10.1 Auflösen synthetischer Schlüssel	42
11 Verwenden der Daten in einer App	44
11.1 Hinzufügen eines Diagramms	44
11.2 Hinzufügen von Dimensionen und Kennzahlen	45
Erstellen und Hinzufügen von Dimensionen	45
Erstellen und Hinzufügen von Kennzahlen	46
11.3 Vielen Dank!	50

1 Herzlich willkommen!

Willkommen bei diesem Tutorial, das die grundlegenden Schritte zur Skripterstellung in Qlik Sense erläutert.

Bevor Sie Visualisierungen in Ihrer App in Qlik Sense erstellen können, müssen Sie Ihre Daten laden. Wenn Sie sich mit der Verwendung von Ladeskripts auskennen, können Sie Ihre Daten vor dem Laden in die App vorbereiten und bearbeiten.

Daten können mit dem Datenmanager oder dem Dateneditor geladen werden. Den Dateneditor verwenden Sie, um ein Datenladeskript zu erstellen, zu bearbeiten und auszuführen.

1.1 Lerninhalte

Nachdem Sie dieses Tutorial abgeschlossen haben, sollten Sie mit Datenladevorgängen mithilfe von Skripten, der Bearbeitung von Skripten und der Umwandlung von Daten vertraut sein.

1.2 Zielgruppe für diesen Kurs

Sie sollten mit den Grundlagen von Qlik Sense vertraut sein. Dazu gehört die Erstellung von Apps und Visualisierungen.

Sie benötigen Zugriff auf den Dateneditor und müssen berechtigt sein, Daten in Qlik Sense Enterprise on Windows zu laden.

1.3 Paketinhalt

Das heruntergeladene Paket enthält die folgenden Datendateien, die Sie zum Bearbeiten des Tutorials benötigen:

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

Zudem enthält das Paket eine Kopie der App *Skript-Tutorial*. Sie können die App in Ihren Hub hochladen.

Sie sollten die App wie im Tutorial beschrieben selbst erstellen, um den Lernerfolg zu maximieren. Zudem müssen Sie die Datendateien wie im Tutorial beschrieben hochladen und verbinden, damit die Datenladevorgänge funktionieren.

Wenn Sie Probleme haben, kann Sie die App aber bei der Problembehebung unterstützen. Wir haben angegeben, welche Skriptsegmente zu den einzelnen Lektionen gehören.

1.4 Lektionen in diesem Tutorial

Je nach Ihrer Erfahrung mit Qlik Sense dauert die Bearbeitung dieses Tutorials 3-4 Stunden. Die Themen sollten ihrer Reihenfolge nach bearbeitet werden. Sie können die Arbeit daran jedoch jederzeit unterbrechen und später fortsetzen. Glücklicherweise umfasst das Tutorial keine Tests.

- Einführung in das Laden von Daten
- LOAD- und SELECT-Befehle
- Auswählen und Laden von Daten
- Umbenennen von Feldern
- Reduzieren von Daten
- Umwandeln von Daten
- Zusammenfassung von Tabellen
- Zirkelbezüge
- Synthetische Schlüssel
- Verwenden der Daten in einer App

1.5 Weitere Informationsquellen und Ressourcen

-  [Qlik](#) bietet eine Vielzahl von Ressourcen, wenn Sie noch mehr erfahren möchten.
- [Qlik Online-Hilfe](#) ist verfügbar.
- Schulungen, einschließlich kostenloser Online-Kurse, stehen im  [Qlik Continuous Classroom](#) zur Verfügung.
- Diskussionsforen, Blogs und mehr finden Sie in der  [Qlik Community](#).

2 Skripterstellung im Dateneditor

Qlik Sense benutzt ein Datenladeskript, das im Dateneditor verwaltet wird, mit dem Sie eine Verbindung zur Datenbank herstellen können, um verschiedene Datenquellen abzurufen. Eine Datenquelle kann eine Datendatei sein, beispielsweise eine Excel-Datei oder eine .csv-Datei. Eine Datenquelle kann auch eine Datenbank sein, beispielsweise eine Google BigQuery- oder Salesforce-Datenbank.

Sie können Daten auch mithilfe des Datenmanagers laden. Wenn Sie jedoch ein Datenladeskript erstellen, bearbeiten und ausführen möchten, verwenden Sie den Dateneditor.

Im Skript werden die zu ladenden Felder und Tabellen festgelegt. Die Skripterstellung wird häufig verwendet, um anzugeben, welche Daten aus Ihren Datenquellen geladen werden sollen. Sie können auch die Datenstruktur ändern, indem Sie Skriptanweisungen verwenden.

Während der Ausführung des Skripts legt Qlik Sense allgemeine Felder aus verschiedenen Tabellen (Schlüsselfelder) fest, um die Daten zu verknüpfen. Die Datenstruktur der Dateien in der App können Sie im Datenmodell überprüfen. Änderungen an der Datenstruktur können durch Umbenennen der Felder vorgenommen werden, um unterschiedliche Verknüpfungen zwischen den Tabellen zu erhalten.

Nachdem die Daten in Qlik Sense geladen sind, werden sie in der App gespeichert.

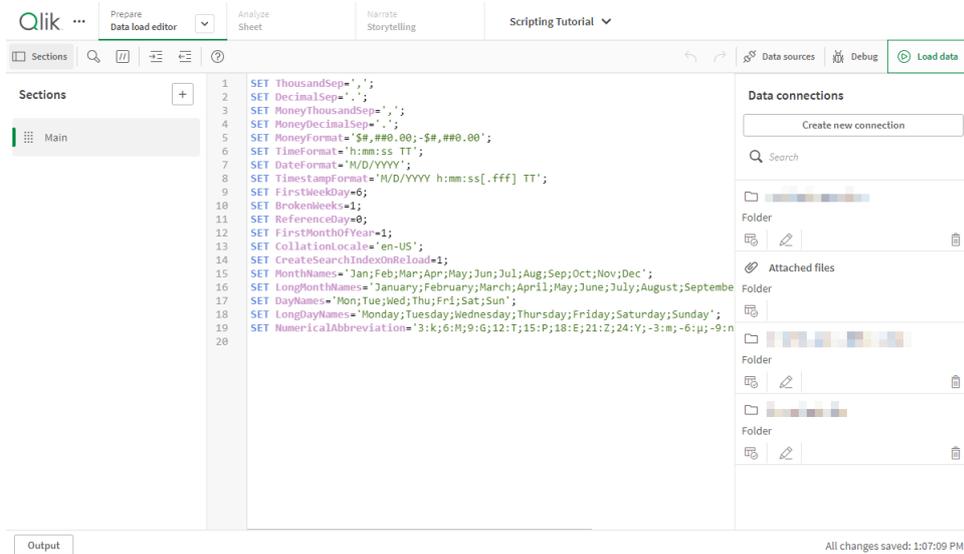
2.1 Dateneditor

Sie können Skripte zum Laden von Daten im Dateneditor erstellen. Den Editor öffnen Sie über das Dropdown-Menü in Qlik Sense.

Wenn Sie den Dateneditor öffnen, wird der Skripteditor in der Mitte des Bildschirms angezeigt. Skriptabschnitte werden als Registerkarten im linken Menü angezeigt. Qlik Sense erstellt den **Hauptabschnitt** automatisch. Datenverbindungen werden im rechten Menü angezeigt.

Das Skript muss unter Verwendung der Qlik Sense-Skriptsyntax geschrieben werden. Qlik Sense-Syntax-Schlüsselwörter sind blau markiert.

Dateneditor



2.2 Skript-Editor

Es gibt im Editor eine Reihe von Funktionen, die Sie beim Entwickeln des Ladeskripts unterstützen.

Zugriff auf die Syntaxhilfe für Befehle und Funktionen

Die detaillierte Syntaxhilfe für ein Qlik Sense-Syntax-Schlüsselwort kann auf mehrere Arten aufgerufen werden:

Zugriff auf das Hilfe-Portal

Sie haben zwei Möglichkeiten, um die detaillierte Hilfe im Qlik Sense-Hilfe-Portal aufzurufen.

- Klicken Sie in der Symbolleiste auf , um den Syntaxhilfe-Modus aufzurufen. Im Syntaxhilfe-Modus können Sie auf ein Syntax-Schlüsselwort (blau markiert und unterstrichen) klicken, um die Syntaxhilfe aufzurufen.
- Platzieren Sie den Mauszeiger auf oder am Ende des Schlüsselworts und drücken Sie Strg+H.



Das Skript kann im Syntaxhilfe-Modus nicht bearbeitet werden.

Verwenden der Auto-Vervollständigungsfunktion

Wenn Sie anfangen, ein Qlik Sense-Skript-Schlüsselwort einzugeben, wird eine Auto-Vervollständigungsliste passender Schlüsselwörter angezeigt, aus denen Sie wählen können. Die Auswahl wird immer weiter eingegrenzt, während Sie weitertippen. Außerdem können Sie aus Vorlagen mit empfohlener Syntax und Parametervorschlägen wählen. Eine Popup-Hilfe enthält die Syntax der Funktion mit ihren Parametern und zusätzlichen Anweisungen sowie einen Link zur Beschreibung der betreffenden Anweisung oder Funktion im Hilfe-Portal.



Sie können über STRG+Leertaste die Schlüsselwortliste und über STRG+Umschalt+Leertaste eine Popup-Hilfe aufrufen.

Kommentieren im Skript

Sie können Kommentare in den Skriptcode einfügen oder Teile des Skriptcodes durch Auskommentieren deaktivieren. Sämtlicher Text, der in einer Zeile rechts von // (zwei Schrägstrichen) steht, wird als Kommentar betrachtet und beim Ausführen des Skripts nicht berücksichtigt.

In der Symbolleiste des Dateneditors befindet sich eine Verknüpfung für das Ein- und Auskommentieren von Code. Klicken Sie auf oder drücken Sie Ctrl + K, um den Code zu kommentieren oder auszukommentieren.

Einrücken von Code

Sie können den Code zur Verbesserung der Lesbarkeit einrücken. Klicken Sie auf , um den Text nach rechts zu rücken (Einzug vergrößern), oder klicken Sie auf , um den Text nach links zu rücken (Einzug verkleinern).

Einfügen eines vorbereiteten Test-Skripts

Sie können ein vorbereitetes Testskript einfügen, das einen Satz von Inline-Datenfeldern lädt. Auf diese Weise können Sie schnell einen Datensatz zu Testzwecken erstellen. Drücken Sie Ctrl + 00, um das Testskript einzufügen.

3 LOAD- und SELECT-Anweisungen

Sie können mithilfe von LOAD- und SELECT-Befehlen Daten in Qlik Sense laden. Durch jeden dieser Anweisungen wird eine interne Tabelle erstellt. LOAD wird zum Laden von Daten aus Dateien verwendet, während mit SELECT Daten aus Datenbanken geladen werden.

In diesem Tutorial verwenden Sie Daten aus Dateien. Daher werden LOAD-Anweisungen verwendet.

Sie können auch ein vorangestellte LOAD verwenden, um den Inhalt der geladenen Daten beeinflussen zu können. Beispielsweise muss das Umbenennen von Feldern in einer LOAD-Anweisung erfolgen, während die SELECT-Anweisung keine Änderungen an Feldnamen zulässt.

Beim Laden von Daten in Qlik Sense gelten die folgenden Regeln:

- Qlik Sense unterscheidet nicht zwischen Tabellen, die durch eine LOAD- oder eine SELECT-Anweisung generiert werden. Wenn mehrere Tabellen geladen werden, ist es daher unerheblich, ob die Tabellen durch LOAD- oder SELECT-Anweisungen oder eine Kombination beider Anweisungen geladen wurden.
- Die Reihenfolge der Felder in der zugrunde liegenden Datenbank ist für Qlik Sense nicht relevant.
- Feldnamen unterscheiden zwischen Groß- und Kleinschreibung und werden verwendet, um Zuordnungen zwischen Datentabellen einzurichten. Aus diesem Grund müssen Felder im Ladeskript manchmal umbenannt werden, um das gewünschte Datenmodell zu erhalten.

4 Auswählen und Laden von Daten

Das Laden von Daten aus Dateien, wie beispielsweise aus Microsoft Excel oder anderen unterstützten Dateiformaten, erfolgt einfach über das Datenauswahl-Dialogfeld im Dateneditor.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie Qlik Sense.
2. Erstellen Sie eine neue App.
3. Geben Sie der App den Namen *Scripting Tutorial* und klicken Sie dann auf **Erstellen**.
4. Öffnen Sie die App.

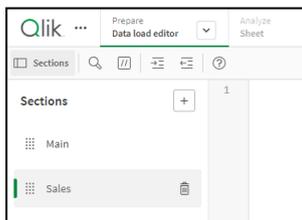


*Bevor Sie zum ersten Mal Daten in Ihre App laden, gibt es die Option **Daten hinzufügen**, mit der sie Daten aus Dateien einfach laden. In diesem Tutorial wollen wir jedoch das Skript sehen, daher verwenden wir den Dateneditor.*

5. Klicken Sie im linken Menü auf **+**, um einen neuen Skriptabschnitt unter dem Abschnitt mit dem Namen *Main* hinzuzufügen.

Durch die Verwendung mehrerer Abschnitte lässt sich Ihr Skript besser organisieren. Der Skriptabschnitt wird der Reihe nach ausgeführt, wenn Sie Daten laden.

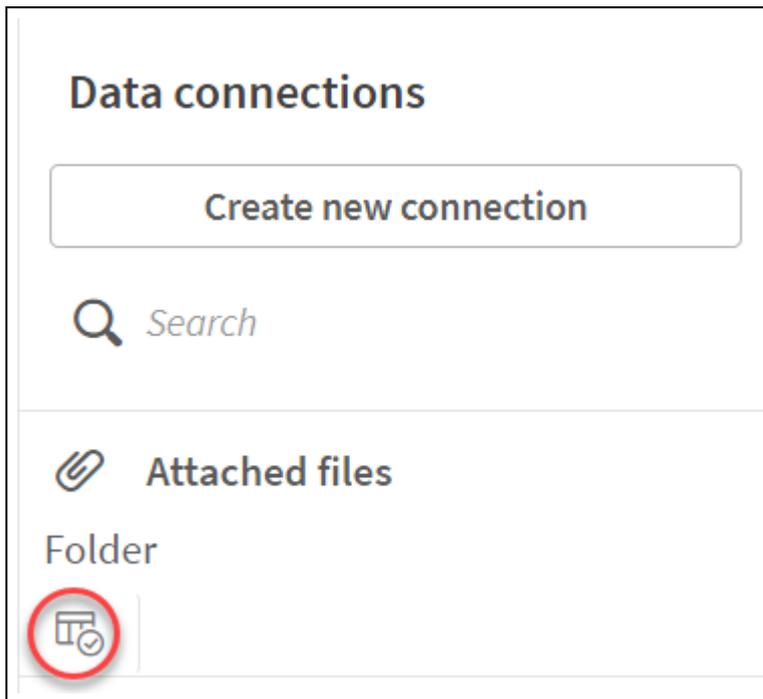
6. Geben Sie dem Abschnitt einen Namen, indem Sie *Sales* eingeben.
Neue Registerkarte Sales in der Scripting Tutorial -App



7. Klicken Sie im rechten Menü unter **AttachedFiles** auf **Daten auswählen**.

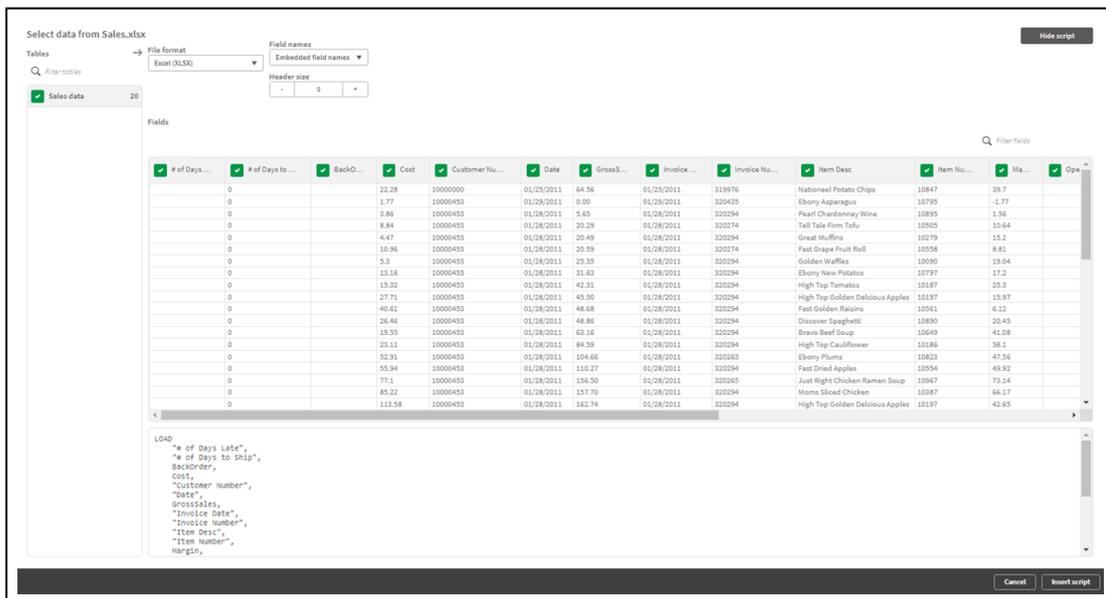
4 Auswählen und Laden von Daten

Fenster „Daten auswählen“



8. Laden Sie *Sales.xlsx* hoch und wählen Sie die Datei aus. Ein Datenvorschauenfenster wird geöffnet.

Datenvorschauenfenster für die Sales-Datendatei



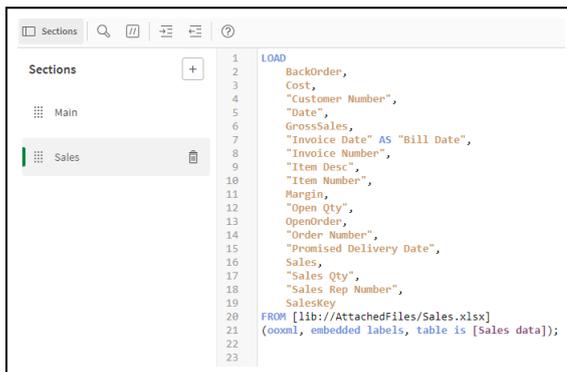
9. Heben Sie die Auswahl der Felder *# of Days Late* und *# of Days to Ship* auf. Sie müssen ggf. auf die Feldüberschriften klicken, damit die vollständigen Feldnamen angezeigt werden.
10. Suchen Sie nach *date* im Suchfeld **Filterfelder**.

4 Auswählen und Laden von Daten

11. Klicken Sie auf die Überschrift *Invoice Date* und geben Sie *Bill Date* ein, um das Feld umzubenennen.
12. Klicken Sie auf **Skript einfügen**. Das Ladeskript wird in den Abschnitt *Sales* des Skript-Editors eingefügt. Beachten Sie, dass Qlik Sense doppelte Anführungszeichen um Feldnamen setzt, die ein Leerzeichen enthalten.

Ihr Skript sollte folgendermaßen aussehen:

Ladeskript auf der Registerkarte Sales

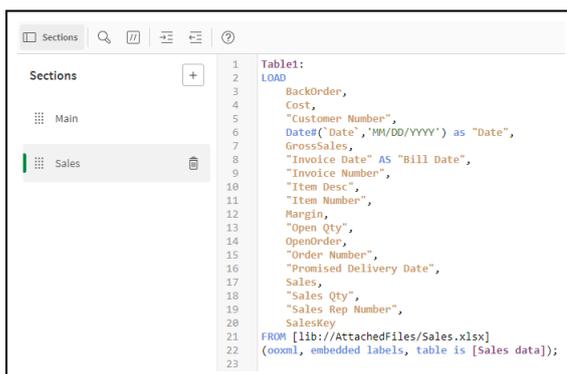


```
1 LOAD
2   BackOrder,
3   Cost,
4   "Customer Number",
5   "Date",
6   GrossSales,
7   "Invoice Date" AS "Bill Date",
8   "Invoice Number",
9   "Item Desc",
10  "Item Number",
11  Margin,
12  "Open Qty",
13  OpenOrder,
14  "Order Number",
15  "Promised Delivery Date",
16  Sales,
17  "Sales Qty",
18  "Sales Rep Number",
19  SalesKey
20 FROM [lib://AttachedFiles/Sales.xlsx]
21 (ooxml, embedded labels, table is [Sales data]);
22
23
```

13. Fügen die folgende Zeile oberhalb der *LOAD*-Anweisung hinzu, um die Tabelle *Table1* zu benennen:
Table1:
14. Nehmen Sie nun eine Anpassung am Skript vor, um sicherzustellen, dass die Datumsangaben richtig interpretiert werden. Ändern Sie das Feld *Date* wie folgt:
date#(`Date`, 'MM/DD/YYYY') as "Date",

Ihr Skript sollte folgendermaßen aussehen:

Aktualisiertes Ladeskript auf der Registerkarte Sales



```
1 Table1:
2 LOAD
3   BackOrder,
4   Cost,
5   "Customer Number",
6   Date#("Date", 'MM/DD/YYYY') as "Date",
7   GrossSales,
8   "Invoice Date" AS "Bill Date",
9   "Invoice Number",
10  "Item Desc",
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
```

15. Klicken Sie oben rechts auf **Daten laden**.

4 Auswählen und Laden von Daten

Dadurch werden die Daten in die App geladen. Ein Fortschrittsfenster für die Skriptausführung wird angezeigt. Nach Abschluss des Vorgangs wird eine Zusammenfassung möglicher Fehler und synthetischer Schlüssel angezeigt, selbst wenn keine vorhanden sind.

16. Klicken Sie auf **Schließen**.
17. Öffnen Sie die Datenmodellansicht über das Dropdown-Menü in der oberen Symbolleiste. Durch die Auswahl von  wird das Datenmodell auf einer neuen Dialogseite geöffnet.
18. Wählen Sie im oberen Menü   und  aus, um die Tabellenansicht anzuzeigen, die in diesem Tutorial verwendet wird. Falls Ihre Tabelle nicht richtig angezeigt wird, können Sie das vorhandene Ladeskript entfernen und das Skript erneut erstellen.

Tabellenansicht von Sales-Daten in der Datenmodellansicht

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

Jetzt laden wir eine weitere Tabelle mit dem Namen *Dates*. Nach dem Laden der Tabelle verbindet Qlik Sense sie mit der Tabelle *Sales* im Feld *Date*.

19. Öffnen Sie den **Dateneditor**.
20. Klicken Sie auf **+**, um einen neuen Skriptabschnitt hinzuzufügen.

4 Auswählen und Laden von Daten

21. Nennen Sie den Abschnitt *Dates*. Wenn der neue Abschnitt *Dates* noch nicht unter *Sales* eingefügt wurde, platzieren Sie den Cursor auf  und ziehen Sie den Abschnitt unter den Abschnitt *Sales*, um die Reihenfolge zu ändern.
22. Klicken Sie auf die oberste Zeile des Skripts und klicken Sie auf .
Stellen Sie sicher, dass dem Skript zwei Schrägstriche (//) hinzugefügt wurden.
23. Fügen Sie den folgenden Text nach // hinzu:
Loading data from Dates.xlsx

Die oberste Zeile Ihres Skripts sieht jetzt folgendermaßen aus:

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

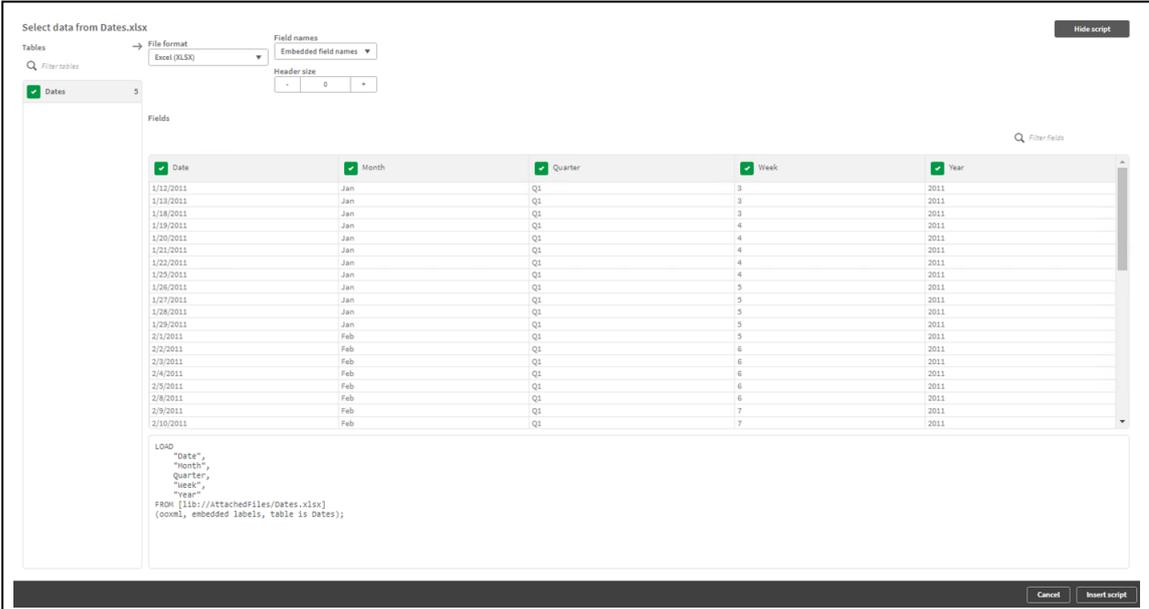
24. Klicken Sie im rechten Menü unter **AttachedFiles** auf **Daten auswählen**.



Achten Sie unter **Feldnamen** darauf, dass der Eintrag **Eingebettete Feldnamen** ausgewählt ist, damit die Namen der Tabellenfelder beim Laden der Daten eingeschlossen sind.

25. Laden Sie *Dates.xlsx* hoch und wählen Sie die Datei aus. Ein Datenvorschauenfenster wird geöffnet.

Datenvorschauenfenster für die Dates-Datendatei

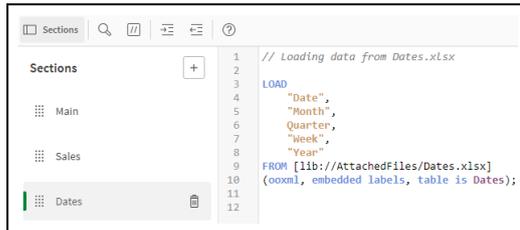


Date	Month	Quarter	Week	Year
1/12/2011	Jan	Q1	3	2011
1/13/2011	Jan	Q1	3	2011
1/18/2011	Jan	Q1	3	2011
1/19/2011	Jan	Q1	4	2011
1/20/2011	Jan	Q1	4	2011
1/21/2011	Jan	Q1	4	2011
1/22/2011	Jan	Q1	4	2011
1/25/2011	Jan	Q1	4	2011
1/26/2011	Jan	Q1	5	2011
1/27/2011	Jan	Q1	5	2011
1/28/2011	Jan	Q1	5	2011
1/29/2011	Jan	Q1	5	2011
2/1/2011	Feb	Q1	5	2011
2/2/2011	Feb	Q1	6	2011
2/3/2011	Feb	Q1	6	2011
2/4/2011	Feb	Q1	6	2011
2/5/2011	Feb	Q1	6	2011
2/6/2011	Feb	Q1	6	2011
2/9/2011	Feb	Q1	7	2011
2/10/2011	Feb	Q1	7	2011

26. Klicken Sie auf **Skript einfügen**.

Ihr Skript sollte folgendermaßen aussehen:

Ladeskript auf der Registerkarte Dates



```
1 // Loading data from Dates.xlsx
2
3 LOAD
4     "Date",
5     "Month",
6     "Quarter",
7     "Week",
8     "Year"
9 FROM [lib://AttachedFiles/Dates.xlsx]
10 (ooxml, embedded labels, table is Dates);
11
12
```

27. Fügen Sie Folgendes in der Zeile über der *LOAD*-Anweisung hinzu und benennen Sie die Tabelle *Table2*:

Table2:

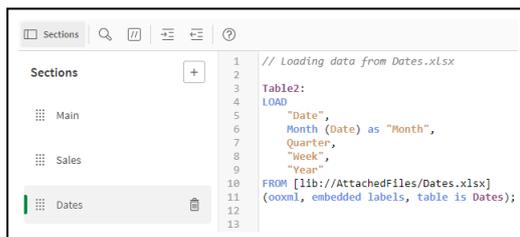
28. Um sicherzustellen, dass die Spalte *Month* in der Datei *Dates.xlsx* in Qlik Sense richtig interpretiert wird, muss die Funktion *Month* auf das Feld *Date* angewendet werden.

Ändern Sie das Feld *Monat* wie folgt:

Month (Date) as "Month",

Ihr Skript sollte folgendermaßen aussehen:

Aktualisiertes Ladeskript auf der Registerkarte Dates

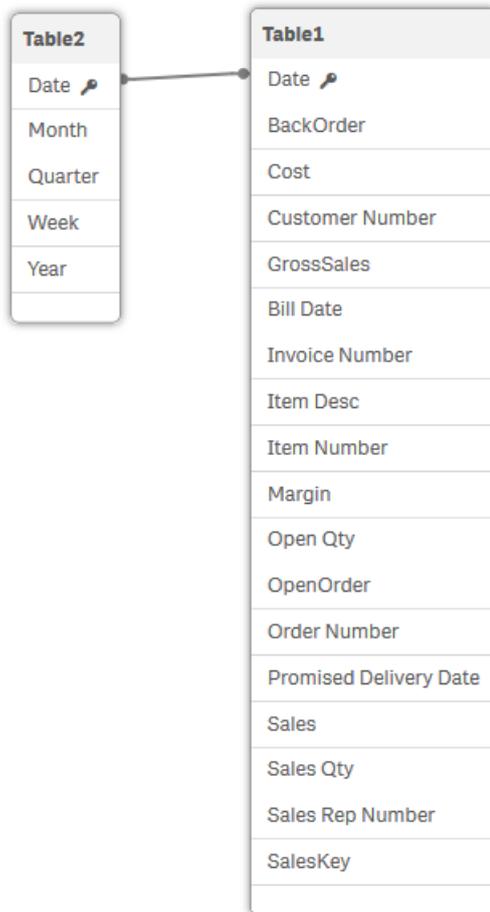


```
1 // Loading data from Dates.xlsx
2
3 Table2:
4 LOAD
5     "Date",
6     Month (Date) as "Month",
7     "Quarter",
8     "Week",
9     "Year"
10 FROM [lib://AttachedFiles/Dates.xlsx]
11 (ooxml, embedded labels, table is Dates);
12
13
```

Sie haben nun ein Skript erstellt, mit dem die ausgewählten Daten aus der Datei *Dates.xlsx* geladen werden. Jetzt ist es an der Zeit, die Daten in die App zu laden.

29. Klicken Sie oben rechts auf **Daten laden**.
Durch Klicken auf **Daten laden** werden die Daten in die App geladen und das Skript wird gespeichert.
30. Klicken Sie nach Abschluss der Skriptausführung auf **Schließen**.
31. Öffnen Sie die **Datenmodellansicht**.
Sie können jetzt sehen, dass eine Verbindung zwischen den beiden Feldern *Date* in den beiden Tabellen erstellt wurde.

Tabellenansicht in der Datenmodellansicht

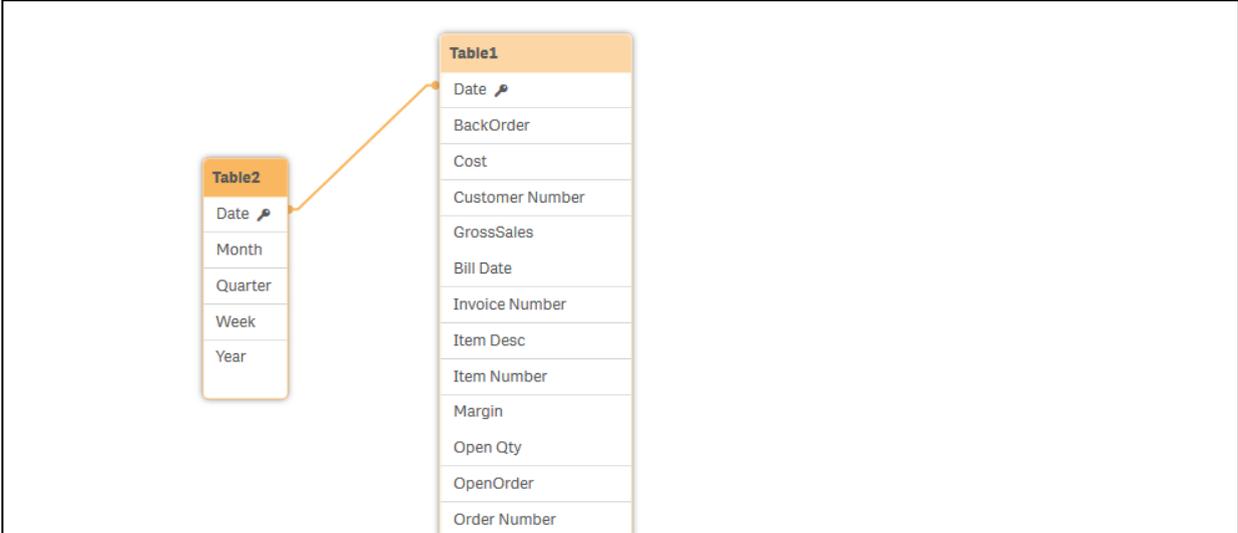


32. Klicken Sie in der linken unteren Ecke auf **Vorschau**. Klicken Sie auf den Namen der Tabelle *Table2*.

Dadurch werden Informationen über die Tabelle angezeigt. Im Feld **Vorschau** können Sie sehen, dass 628 Zeilen mit Daten in die interne Tabelle *Table2* geladen wurden. Wenn Sie stattdessen auf ein Feld in der Tabelle klicken, werden Informationen über das Feld angezeigt.

4 Auswählen und Laden von Daten

Tabellenvorschau in der Datenmodellansicht



▼ Preview

Table2		Preview of data				
Rows	628	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

Die Daten stehen jetzt zur Verwendung in Visualisierungen in einer App zur Verfügung. Die Vorgehensweise wird später in diesem Tutorial gezeigt.

5 Umbenennen von Feldern

In einem früheren Thema wurde gezeigt, wie Sie Felder im Datenvorschaufenster umbenennen. In der Überschrift des Datenvorschaufensters benannten Sie *Invoice Date* in *Bill Date* um. Beim Einfügen des Ladeskripts konnten Sie sehen, dass das Feld mit dem Schlüsselwort *AS* umbenannt werden würde.

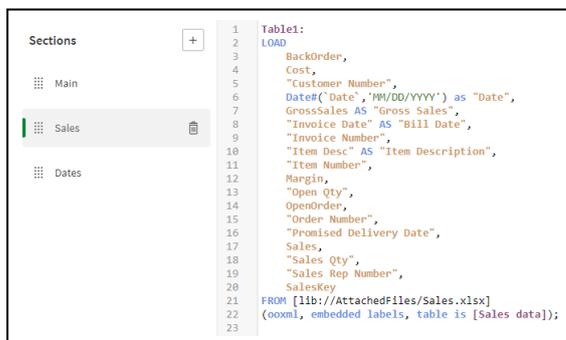
Diese Aktion kann auch direkt im Skript durchgeführt werden.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial* App.
2. Klicken Sie auf die Registerkarte *Sales*.
3. Nehmen Sie im Ladeskript die folgenden Änderungen vor. Beachten Sie, dass Sie Felder, die ein Leerzeichen enthalten, in Klammern einschließen müssen.
 - i. Ändern Sie *GrossSales*, in:
GrossSales AS "Gross Sales",
 - ii. Ändern Sie *"Item Desc"*, in:
"Item Desc" AS "Item Description",

Ihr Skript sollte folgendermaßen aussehen:

Ladeskriptfenster mit Skript zum Umbenennen von Feldern

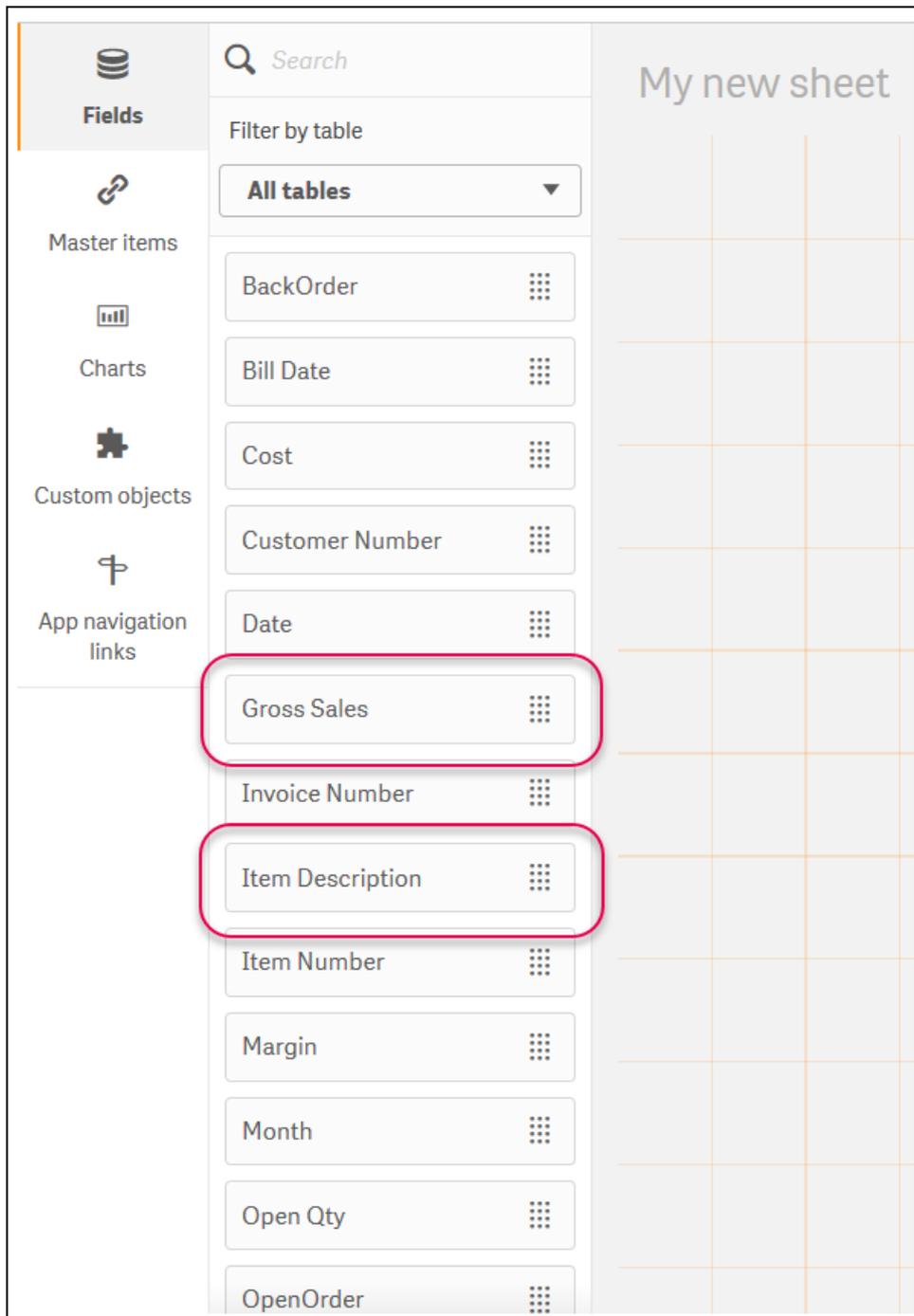


```

1 Table:
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
  
```

4. Klicken Sie auf **Daten laden**. Die Daten werden geladen.
5. Öffnen Sie die **Datenmodellansicht**. Bestätigen Sie, dass die Felder umbenannt wurden.
6. Sie können auch alle Felder in einer App anzeigen. Klicken Sie auf der oberen Symbolleiste auf die Registerkarte **Analyse/Arbeitsblatt**. Die App wird in der Arbeitsblatt-Ansicht geöffnet.
7. Klicken Sie auf **Arbeitsblatt bearbeiten** und dann im Extras-Fenster auf **Felder**. Sie können die Feldnamen sehen, die Sie geändert haben. Sie können beliebige dieser Felder in den Visualisierungen verwenden, die Sie in Ihrer App erstellen.

Umbenannte Felder in der Analyseansicht



6 Reduzieren von Daten

Qlik Sense stellt verschiedene Möglichkeiten zum Reduzieren der Datenmenge bereit, die Sie in Ihre App laden. Sie können beispielsweise Daten aus Dateien oder von Datenkonnektoren filtern.

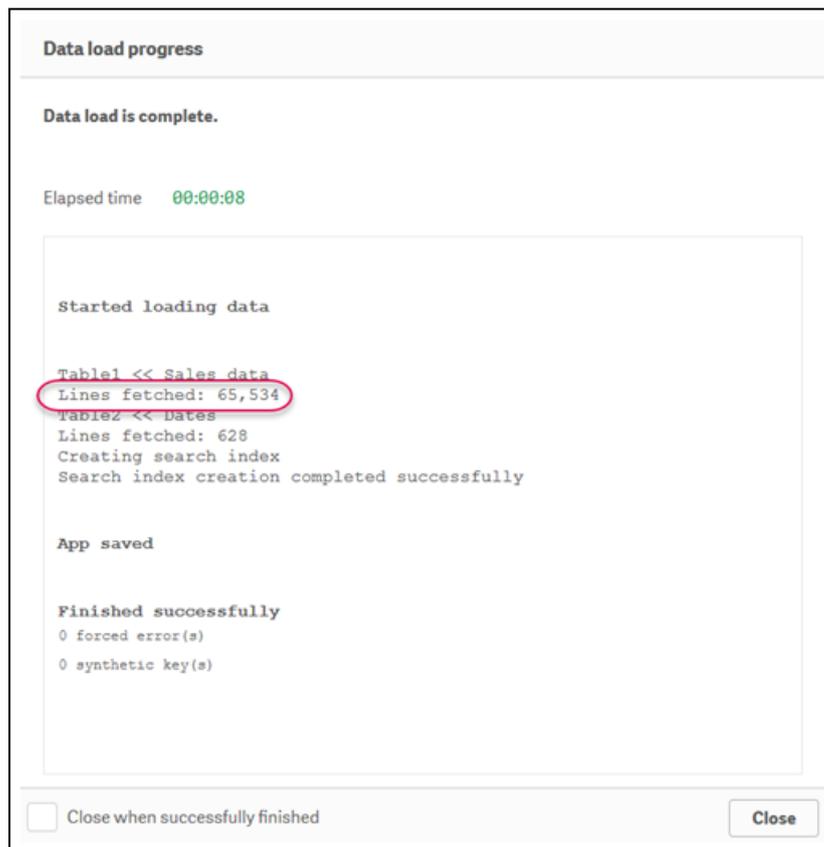
Sie können Daten auch direkt im Ladeskript reduzieren.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial* App.
2. Klicken Sie auf **Daten laden**.

Gestützt auf das bisher von Ihnen geschriebene Ladeskript lädt Qlik Sense 65.534 Zeilen aus der Datendatei *Sales.xlsx* in *Table1*. Beachten Sie, dass *Sales data* der Name der Registerkarte ist, die Ihre Daten in der Originaldatei *Sales.xlsx* enthält.

Fenster für Datenladefortschritt



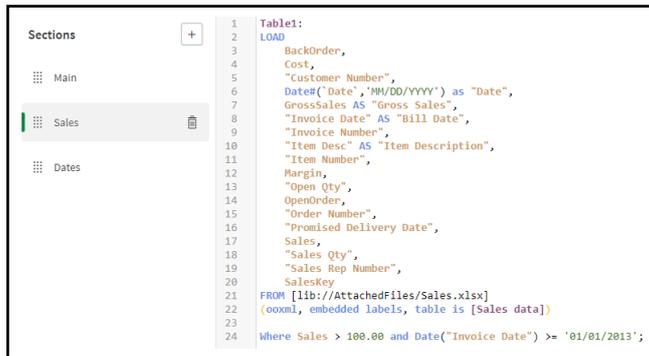
3. Klicken Sie im Dateneditor auf die Registerkarte *Sales*.
4. Löschen Sie den Strichpunkt am Ende dieser Zeile:
(`ooxm1, embedded labels, table is [Sales data]);`)
5. Fügen Sie die folgende Zeile am Ende des Ladeskripts hinzu:

```
where Sales > 100.00 and Date("Invoice Date") >= '01/01/2013';
```

Damit wird Qlik Sense angewiesen, nur Daten zu laden, bei denen der Umsatz höher als 100,00 USD war. Es wird auch die Date-Funktion genutzt, um Daten zu laden, deren Datum größer oder gleich dem 1. Januar 2013 ist.

Ihr Skript sollte folgendermaßen aussehen:

Ladeskriptfenster mit Skript zum Reduzieren der geladenen Datenmenge

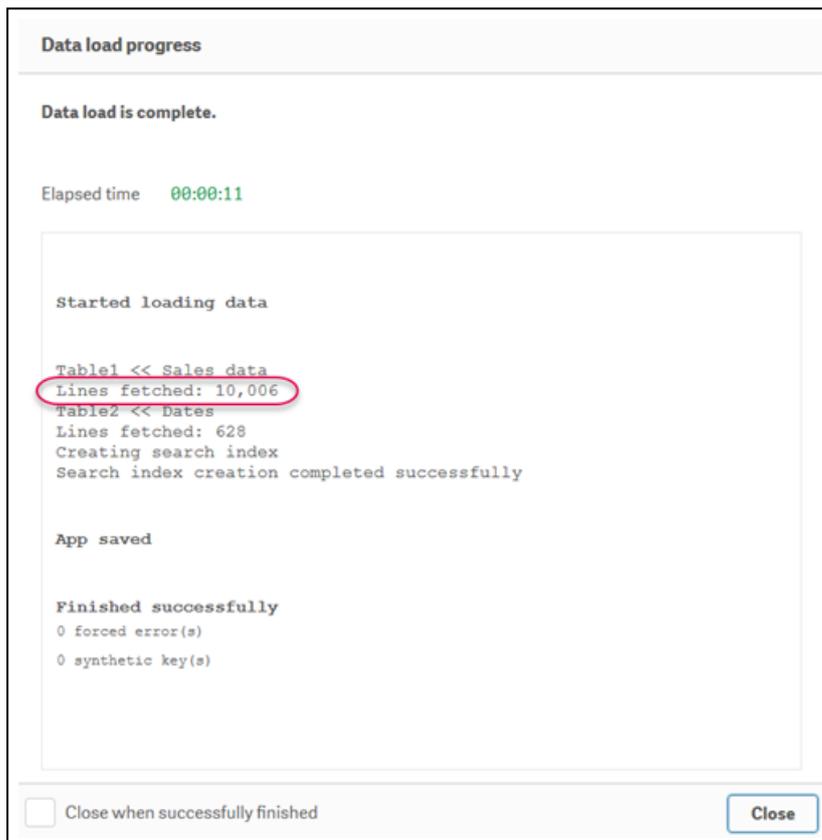


```
1 Table1:
2 LOAD
3   BackOrder,
4   Cost,
5   "Customer Number",
6   Date("Date", "MM/DD/YYYY") as "Date",
7   GrossSales AS "Gross Sales",
8   "Invoice Date" AS "bill date",
9   "Invoice Number",
10  "Item Desc" AS "Item Description",
11  "Item Number",
12  Margin,
13  "Open Qty",
14  OpenOrder,
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. Klicken Sie auf **Daten laden**.

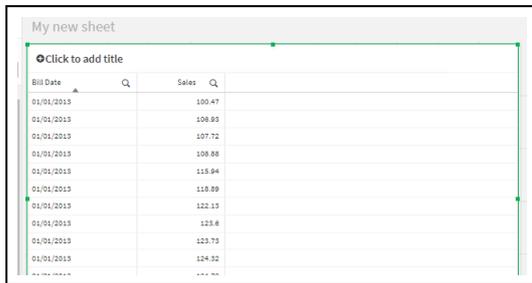
Gestützt auf Ihr aktualisiertes Ladeskript lädt Qlik Sense jetzt weniger Zeilen aus der Datendatei *Sales.xlsx*.

Fenster für Datenladefortschritt mit reduzierter Datenladung



- Wenn Sie die Daten zu einer Tabelle in Ihrer App hinzufügen, sehen Sie nur die Daten, die den beim Laden erstellten Bedingungen entsprechen.

Tabelle mit den Feldern Bill Date und Sales

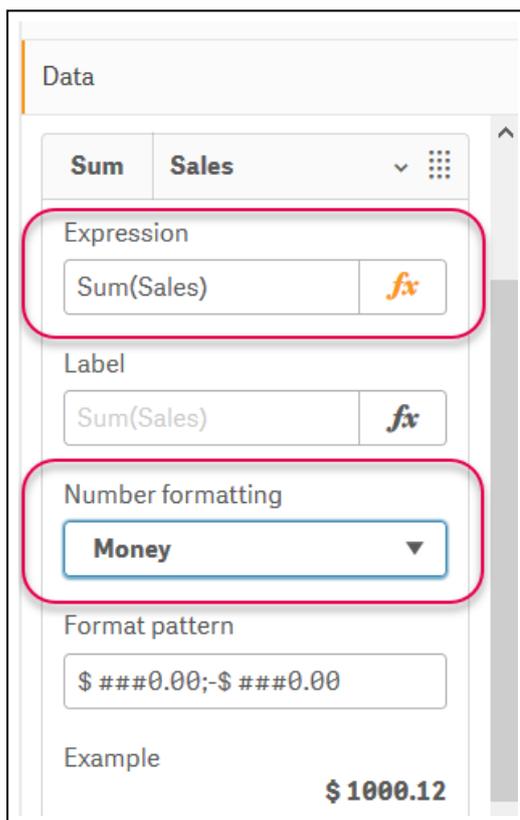


Bill Date	Sales
01/01/2015	100.47
01/01/2015	106.99
01/01/2015	107.72
01/01/2015	108.88
01/01/2015	115.94
01/01/2015	118.89
01/01/2015	122.13
01/01/2015	123.8
01/01/2015	123.73
01/01/2015	124.32

Beachten Sie, dass das *Sales* Feld als Dimension hinzugefügt wurde. Der Grund dafür ist, dass die *Sales*-Werte einzeln angezeigt werden. Wenn wir *Sales* als Kennzahl hinzugefügt hätten, wären die Daten stattdessen nach Datum aggregiert worden.

In der Regel wird *Sales* als Kennzahl hinzugefügt. Für Kennzahlen haben Sie die Option, Werte als Währungsbeträge (z. B. Dollar) anzuzeigen, indem Sie Zahlenformatierung auf die Spalte anwenden.

Auf die Kennzahl Sales angewendete Zahlenformatierung



Data

Sum **Sales**

Expression
Sum(Sales) *fx*

Label
Sum(Sales) *fx*

Number formatting
Money

Format pattern
\$ ###0.00;-\$ ###0.00

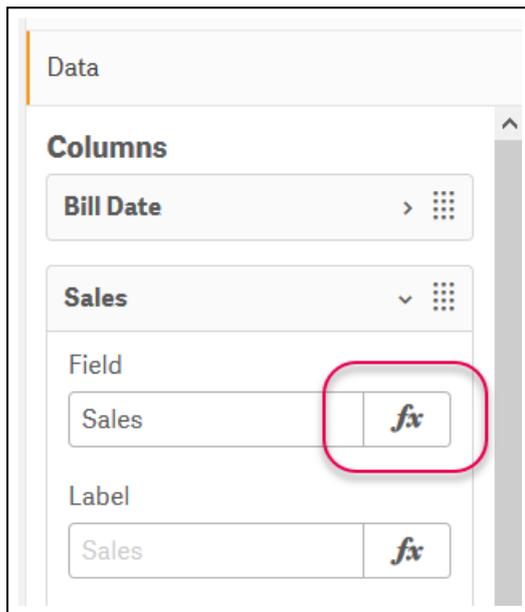
Example
\$ 1000.12

Da wir aber *Sales* als Dimension verwenden, benötigen wir einen anderen Ansatz. In diesem Fall verwenden wir eine Diagrammformel. In diesem Tutorial wird nicht detailliert auf Diagrammfunktionen eingegangen; dies ist aber eine gute Gelegenheit für ein Beispiel.

Wenn *Sales* als Dimension verwendet wird, werden die Werte als numerisch angezeigt, wie in der Tabelle oben gezeigt.

Um dies zu beheben, können Sie das Feld *Sales* im Diagrammformel-Editor öffnen, indem Sie auf **fx** klicken, und dann die Funktion *Money* verwenden.

Öffnen des Formel-Editors



Geben Sie Folgendes im Formel-Editor ein.

```
=Money( Sales )
```

Formel-Editor



Die Werte im Feld werden jetzt als Geldeinheiten angezeigt. Diese Einheiten (im vorliegenden Fall Dollar) werden im **Hauptabschnitt** des Ladeskripts angegeben.

Tabelle mit den Feldern Bill Date und Sales. Sales wird jetzt in Dollar angezeigt

Bill Date	Money(Sales)
01/01/2013	\$100.47
01/01/2013	\$108.93
01/01/2013	\$107.72
01/01/2013	\$108.88
01/01/2013	\$115.94
01/01/2013	\$118.89
01/01/2013	\$122.13
01/01/2013	\$123.60
01/01/2013	\$123.73
01/01/2013	\$124.32

8. Nachdem dieses Beispiel jetzt abgeschlossen ist, kommentieren Sie die Where-Anweisung aus. Vergessen Sie nicht, einen Strichpunkt am Ende der ersten LOAD-Anweisung hinzuzufügen.

Ihr Skript sollte folgendermaßen aussehen:

Ladeskriptfenster mit auskommentierter Where-Anweisung

```

1 Table:
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 Umwandeln von Daten

Sie können Daten anhand vieler verschiedener Techniken im Dateneditor umwandeln und bearbeiten.

Einer der Vorteile der Datenbearbeitung ist, dass Sie beschließen können, nur eine Teilmenge der Daten aus einer Datei zu laden, z. B. einige wenige ausgewählte Spalten einer Tabelle. So können Sie die Datenverarbeitung effizienter gestalten. Sie können die Daten auch mehrmals laden, um die Rohdaten in mehrere neue interne Tabellen aufzuspalten. Außerdem können Sie auch Daten aus mehreren Quellen laden und sie zu einer Tabelle in Qlik Sense zusammenführen.

In diesem Thema werden einige grundlegende Datenumformungen durchgeführt. Dabei verwenden Sie zuerst einen Resident load, dann einen Preceding load.

7.1 Resident LOAD

Sie können den Quellzusatz Resident in einer LOAD-Anweisung verwenden, um Daten aus einer bereits geladenen Tabelle zu laden. Das ist auch hilfreich, wenn Sie Berechnungen für Daten durchführen möchten, die mit einer SELECT-Anweisung geladen wurden, und Sie die Qlik Sense Funktionen wie die Verarbeitung von Datum und Zahlenwerten nicht nutzen können.

In diesem Beispiel erstellen Sie eine neue Tabelle mit dem Namen *Sales_Buckets* und laden dann die Daten aus *Table1* anhand eines resident load. In der Tabelle *Sales_Buckets* erstellen Sie eine Variable mit dem Namen *quantity_threshold* und verwenden dann eine Where-Anweisung, um nur die Daten zu laden, die diesem Schwellenwert entsprechen.

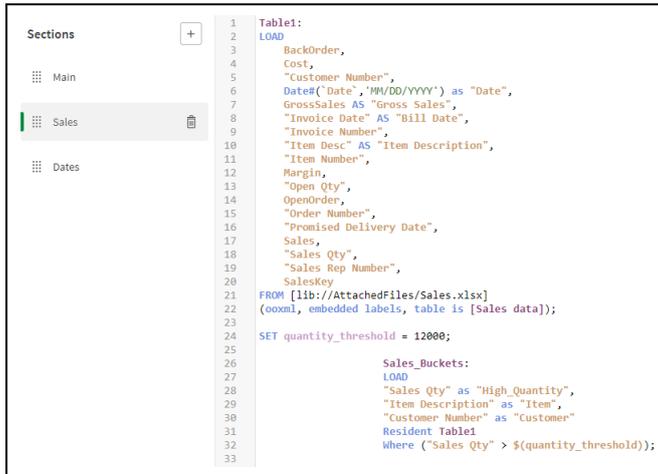
1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial App*.
2. Klicken Sie auf die Registerkarte *Sales*.
3. Fügen Sie Folgendes zum Ende des Skripts hinzu:

```
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));
```

Ihr Skript sollte folgendermaßen aussehen:

Ladeskriptfenster mit Skript zum Erstellen einer neuen Tabelle namens *Sales_Buckets*



```
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
27     Sales_Buckets:
28     LOAD
29       "Sales Qty" as "High Quantity",
30       "Item Description" as "Item",
31       "Customer Number" as "Customer"
32     Resident Table1
33     Where ("Sales Qty" > $(quantity_threshold));
```

4. Klicken Sie auf **Daten laden**.
5. Öffnen Sie die **Datenmodellansicht**. Sie können sehen, dass Sie eine neue Tabelle namens *Sales_Buckets* erstellt haben, in die Daten entsprechend den angegebenen Feldern und dem festgelegten Schwellenwert geladen wurden.

Tabelle Sales_Buckets in der Datenmodellansicht

The screenshot shows a data model view with three tables: Sales_Buckets, Table2, and Table1. 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 relationship line connects the Date field in Table2 to the Date field in Table1.

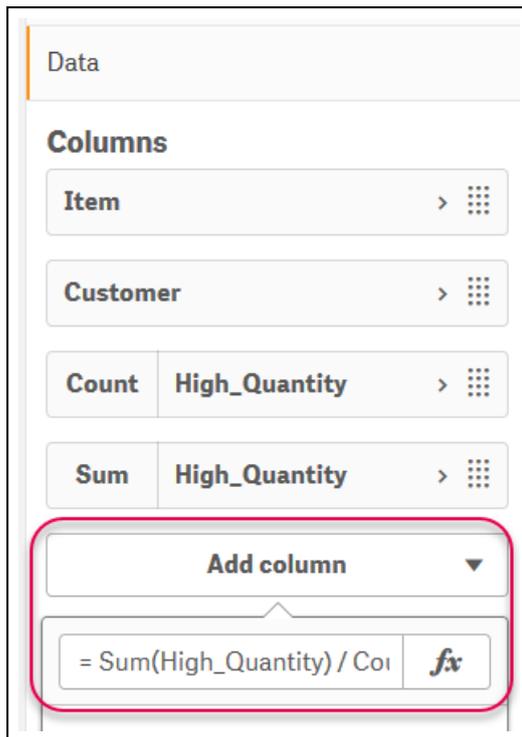
Preview

Sales_Buckets		Preview of data		
		High_Quantity	Item	Customer
Rows	11			
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

6. Fügen Sie die Daten zu einer Tabelle in Ihrer App hinzu. Fügen Sie *Item* und *Customer* als Dimensionen hinzu. Fügen Sie *High-Quantity* als über *Count* und dann erneut über *Sum* aggregierte Kennzahl hinzu. Fügen Sie dann eine neue Spalte als Kennzahl mit der folgenden Formel hinzu:

= Sum(High_Quantity) / Count(High_Quantity)

Neue Kennzahl mit Formel



Ihre Tabelle zeigt beispielsweise, dass *Customer* 10025737 vier große Bestellungen von *High Top Dried Mushrooms* mit einer durchschnittlichen Menge von 14.800 aufgegeben hat. Um Sortierungen für die Daten in den Feldern vorzunehmen, schließen Sie den **Bearbeitungsmodus**, indem Sie auf **Fertig** klicken.

Tabelle mit Kunden, die große Bestellungen aufgegeben haben

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

- Nachdem dieses Beispiel jetzt abgeschlossen ist, kommentieren Sie das Skript für die Variable *quantity_threshold* und die Tabelle *Sales_Buckets* aus.
Das Ende Ihres Skripts sollte nun wie folgt aussehen:

Auskommentiertes Skript

```

(ooxml, embedded labels, table is [Sales data]);
// 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});

```

7.2 Vorangehendes LOAD

Mit einem vorangestellten load-Befehl können Sie Umwandlungen durchführen und Filter anwenden, um Daten in einem Durchgang laden zu können. Im Grunde ist es ein LOAD-Befehl, der über den Befehl LOAD oder SELECT unten lädt, ohne einen Qualifizierer für die Quelle wie From oder Resident anzugeben, was normalerweise der Fall ist. Sie können auf diese Art und Weise beliebig viele LOAD-Befehle stapeln. Der Befehl unten wird zuerst ausgewertet, dann der Befehl darüber usw., bis der oberste Befehl ausgewertet wurde.

Wie weiter oben in diesem Tutorial erwähnt, können Sie Daten in Qlik Sense anhand der Anweisungen LOAD und SELECT laden. Jeder dieser Anweisungen erstellt eine interne Tabelle. LOAD wird zum Laden von Daten aus Dateien oder aus einer Inline-Tabelle verwendet, während mit SELECT Daten aus Datenbanken geladen werden können. In diesem Tutorial haben Sie Daten aus Dateien verwendet. In diesem Beispiel verwenden Sie eine Inline-Tabelle. Beachten Sie aber, dass ein preceding load oberhalb einer SELECT-Anweisung zur Bearbeitung der Daten verwendet werden kann. Die Grundlagen sind die Gleichen wie hier bei Verwendung von LOAD.

Dieses Beispiel bezieht sich nicht auf die Daten, die wir in diesem Tutorial laden. Es dient nur dazu, zu zeigen, wie ein preceding load aussehen kann. Sie erstellen im Dateneditor eine Inline-Tabelle mit dem Namen *Transactions*. Die Datumsinterpretation erfolgt im preceding LOAD, wobei ein neues Feld mit dem Namen *transaction_date* erstellt wird. Dieses Feld wird anhand des Felds *sale_date* erstellt.

1. Erstellen Sie eine neue App und nennen Sie sie *ReformatDate*.
2. Öffnen Sie den Dateneditor und erstellen Sie eine neue Registerkarte mit dem Namen *TransactionData*.
3. Fügen Sie das folgende Skript hinzu:

```

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 ];

```

Ihr Skript sollte folgendermaßen aussehen:

Ladeskript mit preceding load

```

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 ];
    
```

4. Klicken Sie auf **Daten laden**.
5. Öffnen Sie die **Datenmodellansicht**. Wählen Sie die Tabelle *Transactions* aus und erweitern Sie sie. Sie sehen, dass alle Felder wie vom * in der preceding load-Anweisung festgelegt geladen wurden. Ein neues Feld mit dem Namen *transaction_date* wurde erstellt. Das Feld hat das neu formatierte Datum.

Neues Feld namens transaction_date in der Datenmodellansicht

Transactions
 transaction_id
 sale_date
 transaction_amount
 transaction_quantity
 customer_id
 size
 color_code
 transaction_date

▼ Preview

Transactions	Preview of data								
Rows	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 Zusammenfassung von Tabellen

Bei der Zusammenfassung von Tabellen werden zwei Tabellen zu einer kombiniert.

Die beiden Tabellen werden mit einer Spalte für jeden distinkten Spaltennamen zusammengefügt. Die Daten werden nicht geändert und die entstehende Tabelle enthält genauso viele Datensätze wie die beiden ursprünglichen Tabellen zusammen. Es können auch mehr als zwei Tabellen zusammengefasst werden, indem der Zusammenfassungsvorgang mehrmals nacheinander ausgeführt wird.

8.1 Automatische Zusammenfassung

Stimmen bei zwei oder mehr geladenen Tabellen die Zahl der Felder und die Feldnamen genau überein, fasst Qlik Sense den Inhalt der verschiedenen Befehle automatisch zu einer Tabelle zusammen.

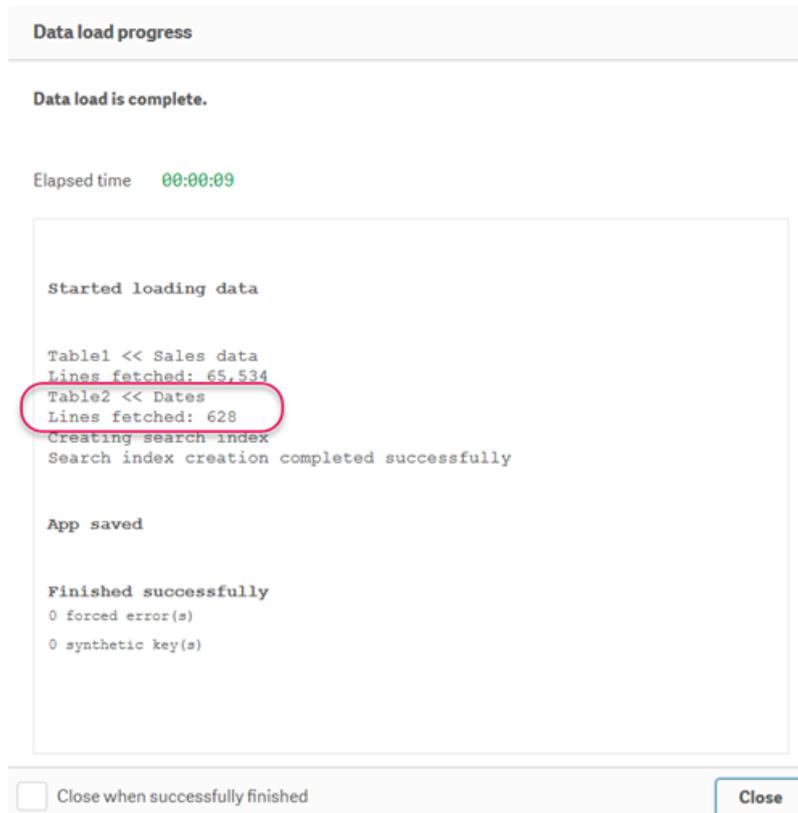
Die Anzahl und die Namen der Felder müssen exakt übereinstimmen, damit eine automatische Zusammenfassung stattfinden kann. Die Reihenfolge der zwei LOAD-Anweisungen ist beliebig, aber die Tabelle erhält den Namen der zuerst geladenen Tabelle.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial* App.
2. Klicken Sie auf die Registerkarte **Dates**.
3. Klicken Sie auf **Daten laden**.

Gestützt auf das bisher von Ihnen geschriebene Ladeskript lädt Qlik Sense 628 Zeilen aus der Datendatei *Dates.xlsx* in *Table2*.

Fenster für Datenladefortschritt



4. In einer neuen Zeile im Skript im Abschnitt *Dates* kopieren Sie die LOAD-Anweisung für *Table2* und fügen Sie sie ein. Damit werden die Daten zweimal geladen. Nennen Sie die zweite Tabelle *Table2a*.

Sie können das vorhandene Skript auch löschen und Folgendes kopieren und einfügen:

Table2:

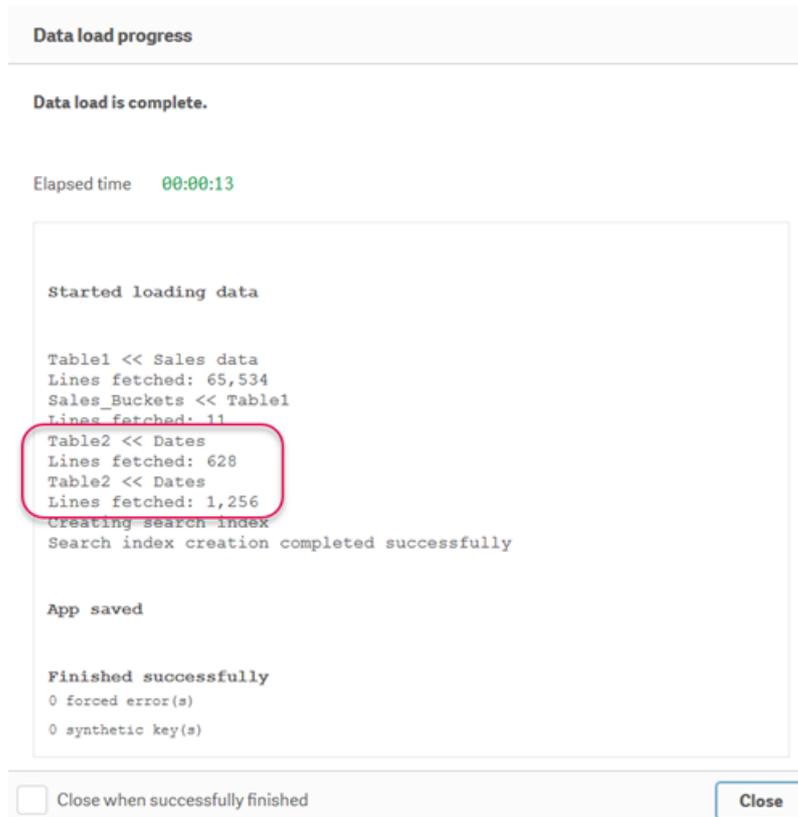
```
LOAD
  "Date",
  Month (Date) as "Month",
  Quarter,
  "Week",
  "Year"
FROM [lib://AttachedFiles/Dates.xlsx]
(ooxml, embedded labels, table is Dates);
```

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

5. Klicken Sie auf **Daten laden**.

Qlik Sense lädt nicht *Table2* und dann *Table2a*. Stattdessen wird erkannt, dass *Table2a* die gleichen Feldnamen und die gleiche Anzahl Felder wie *Table2* hat. Dann werden die Daten von *Table2a* zu *Table2* hinzugefügt und Tabelle *Table2a* gelöscht. Das Ergebnis ist, dass *Table2* jetzt 1256 Zeilen umfasst.

Zusammenfassung im Fenster für Datenladefortschritt



6. Öffnen Sie die **Datenmodellansicht**.

7. Klicken Sie auf **Vorschau anzeigen**.

Nur *Table2* wurde erstellt. Wählen Sie *Table2* aus. Die Tabelle hat 256 Zeilen.

Die Datenmodellansicht zeigt Table2

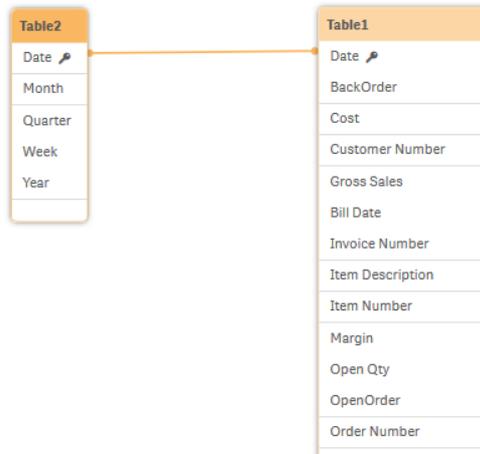


Table2		Preview of data				
Rows	Fields	Date	Month	Quarter	Week	Year
1256	5	01/12/2011	Jan	Q1	3	2011
1		01/13/2011	Jan	Q1	3	2011
		01/18/2011	Jan	Q1	3	2011
		01/19/2011	Jan	Q1	4	2011
		01/20/2011	Jan	Q1	4	2011
		01/21/2011	Jan	Q1	4	2011
		01/22/2011	Jan	Q1	4	2011

8.2 Manuelle Zusammenfassung

Falls zwei oder mehr Tabellen nicht die gleichen Felder besitzen, kann eine Zusammenfassung der beiden Tabellen in Qlik Sense erzwungen werden. Dieser Vorgang erfolgt mithilfe des Zusatzes Concatenate im Skript. Er fasst eine Tabelle mit einer anderen benannten Tabelle oder der zuletzt erstellten Tabelle zusammen.

Gehen Sie folgendermaßen vor:

1. Bearbeiten Sie die LOAD-Anweisung für *Table2a*, indem Sie Concatenate hinzufügen und *Week* auskommentieren.

Ihr Skript sieht nun wie folgt aus:

Table2a:

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

8 Zusammenfassung von Tabellen

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

Durch das Auskommentieren von *Week* wird sichergestellt, dass die Tabellen nicht identisch sind.

2. Klicken Sie auf **Daten laden**.
3. Öffnen Sie die **Datenmodellansicht**.
Sie können nun sehen, dass *Table2a* nicht erstellt wurde.
4. Klicken Sie in der Datenmodellansicht auf *Table2* und dann auf **Vorschau**.
Die Tabelle enthält die Felder *Date*, *Month*, *Quarter*, *Week* und *Year*. Das Feld *Week* wird weiterhin angezeigt, weil es aus *Table2* geladen wurde.
5. Klicken Sie auf *Week* in *Table2*. Die Vorschau zeigt jetzt, dass die Anzahl der Nicht-Null-Werte für das Feld 628 ist. Wenn Sie aber auf eines der anderen Felder klicken, sehen Sie, dass die Anzahl der Nicht-Null-Felder 1256 ist. *Week* wurde nur einmal geladen, und zwar aus *Table2*. Die Anzahl an Werten oder Datensätzen ist die Summe der Anzahl an Datensätzen in *Table2* und *Table2a*.

8.3 Zusammenfassung von Tabellen verhindern

Stimmen bei zwei oder mehr Tabellen die Zahl der Felder und die Feldnamen genau überein, fasst Qlik Sense diese Tabellen automatisch zu einer Tabelle zusammen. Dies kann durch den Befehl `NoConcatenate` verhindert werden. Die mit der zugehörigen `LOAD`- oder `SELECT`-Anweisung geladene Tabelle wird dadurch nicht mit der bestehenden Tabelle zusammengefasst.

Gehen Sie folgendermaßen vor:

1. Um den Inhalt der beiden Tabellen vollständig trennen zu können, fügen Sie `NoConcatenate` zur `LOAD`-Anweisung in *Table2a* hinzu und benennen Sie die Felder um, damit Qlik Sense keinen synthetischen Schlüssel basierend auf den übereinstimmenden Feldern erstellt. Kommentieren Sie *Week* in *Table2* aus, damit die beiden Tabellen die gleichen Felder haben. Ihr Skript sieht nun wie folgt aus:

```
Table2:  
  
LOAD  
"Date",  
Month (Date) as "Month",  
Quarter,  
"week",  
"Year"  
FROM [lib://AttachedFiles/Dates.xlsx]  
(ooxml, embedded labels, table is Dates);  
  
Table2a:  
NoConcatenate LOAD
```

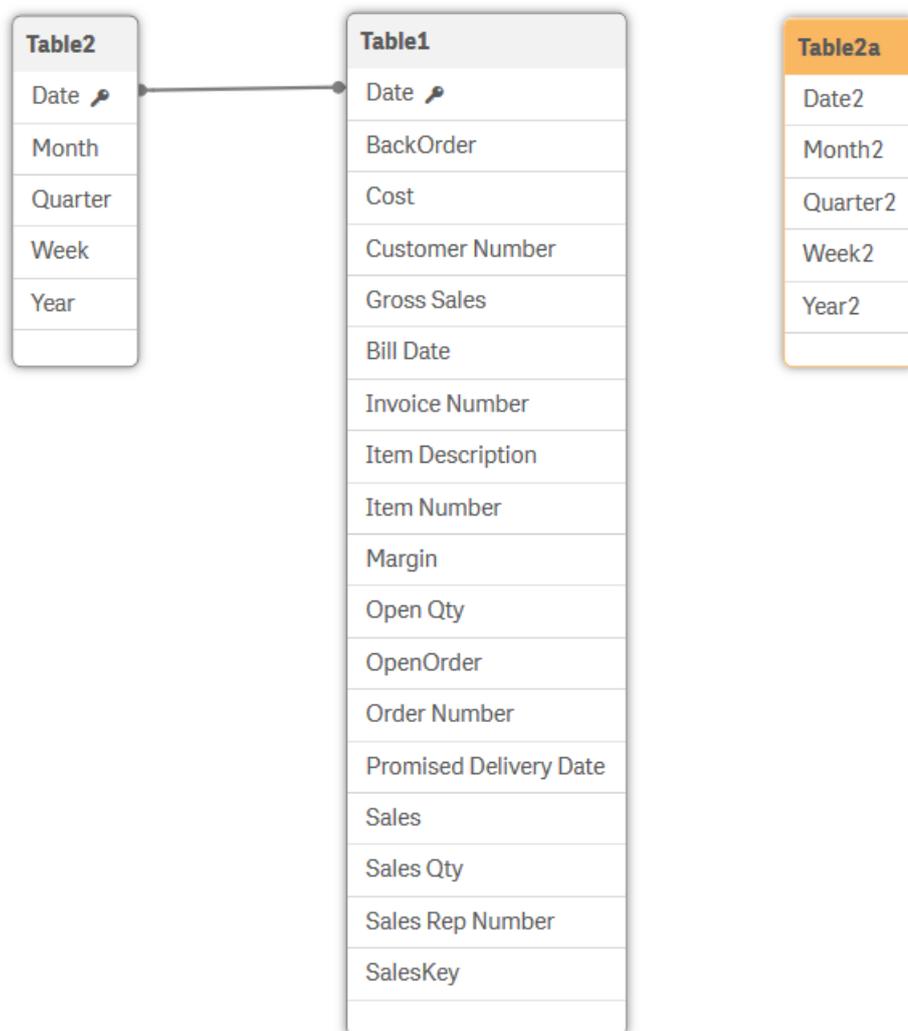
8 Zusammenfassung von Tabellen

```
"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. Klicken Sie auf **Daten laden**.
3. Öffnen Sie die **Datenmodellansicht**.

Jetzt sehen Sie, dass die beiden Tabellen vollständig getrennt sind.

Die Datenmodellansicht zeigt *Table2* und *Table 2a*



4. Jetzt, da die Demonstration der Zusammenfassung abgeschlossen ist, wird *Table2a* nicht mehr benötigt. Löschen Sie alle Zeilen in der LOAD-Anweisung für *Table2a* und klicken Sie dann auf **Daten laden**.

9 Zirkelbezüge

Sind Tabellen so miteinander verknüpft, dass Werte auf mehr als einem Weg miteinander verknüpft sind, spricht man von Zirkelbezügen oder auch zirkulären Verknüpfungen ("Loops") in der Datenstruktur.

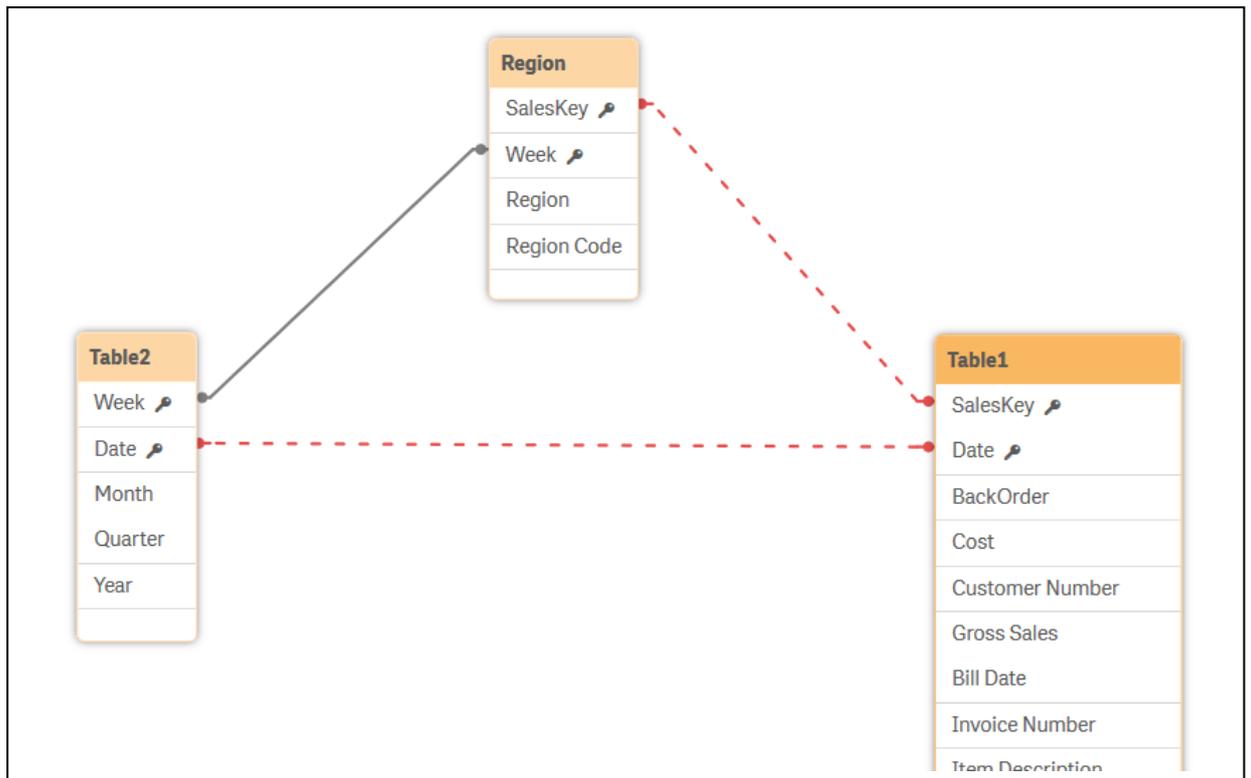
Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial* App.
2. Klicken Sie auf **+**, um einen neuen Skriptabschnitt hinzuzufügen.
3. Nennen Sie den Abschnitt *Region*.
4. Klicken Sie im rechten Menü unter **AttachedFiles** auf **Daten auswählen**.
5. Laden Sie *Region.txt* hoch und wählen Sie die Datei aus. Ein Datenvorschaufenster wird geöffnet.
6. Wählen Sie alle Felder aus und achten Sie darauf, dass unter **Feldnamen** der Eintrag **Eingebettete Feldnamen** ausgewählt ist, damit die Namen der Tabellenfelder beim Laden der Daten eingeschlossen sind.
7. Klicken Sie auf **Skript einfügen**.
8. Klicken Sie auf **Daten laden**.

Dieses Mal sieht es so aus, als ob beim Laden der Daten etwas schief gelaufen ist. Ein Zirkelbezug wurde erstellt. Im Fenster **Datenladefortschritt** wird eine Fehlermeldung angezeigt, die darauf hinweist, dass während des Ladevorgangs ein Zirkelbezug gefunden wurde. Der Ladevorgang wird jedoch abgeschlossen und die App wird gespeichert.

9. Öffnen Sie die **Datenmodellansicht**.
Sie können die Tabellen durch Ziehen so neu organisieren, dass die Verbindungen zwischen den Tabellen leicht sichtbar sind.

Datenmodellansicht mit Zirkelbezügen



Die rot gepunkteten Linien weisen darauf hin, dass ein Zirkelbezug erstellt wurde. Dies sollte vermieden werden, da dadurch Widersprüche bei der Interpretation der Daten entstehen können.

9.1 Auflösen von Zirkelbezügen

Um verstehen zu können, was den Zirkelbezug verursacht hat, sehen wir uns die Tabellen im **Datenmodell** genauer an.

Wenn Sie *Table1* und *Table2* im obigen Screenshot betrachten, sehen Sie, dass beiden das Feld *Date* gemein ist. Sie werden auch feststellen, dass *Table1* und *Region* das Feld *SalesKey* gemein ist. Außerdem haben *Table2* und *Region* das Feld *Week* gemeinsam. Das bedeutet, dass ein Zirkelbezug erstellt wurde. Da dieser später bei der Datenanalyse zu Problemen führen kann, werden wir ihn auflösen.

Am einfachsten lässt sich das Problem durch Umbenennen oder Entfernen eines Felds auflösen. In unserem konkreten Fall wurden Daten geladen, die für unsere App nicht erforderlich sind, weshalb wir sie entfernen können.

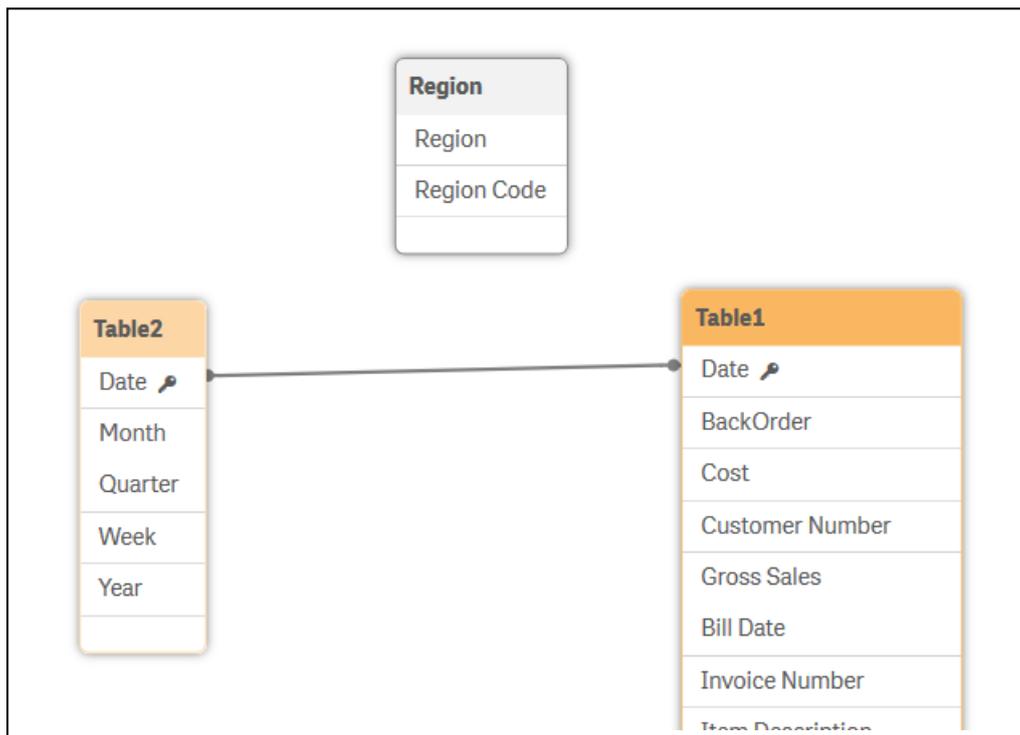
Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor**.
2. Klicken Sie auf den Abschnitt *Region* und löschen Sie die folgenden beiden Zeilen in der *LOAD*-Anweisung:
"week",
saleskey
3. Entfernen Sie unbedingt auch das Komma nach "*Region Code*".
Ihr Skript sieht nun wie folgt aus:

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

4. Klicken Sie auf **Daten laden**.
5. Öffnen Sie die **Datenmodellansicht**.
Die unerwünschten Bezüge zu *Region* wurden entfernt.

In der Datenmodellansicht ist zu sehen, dass die Zirkelbezüge entfernt wurden



10 Synthetische Schlüssel

Haben zwei interne Tabellen mehr als ein Feld gemeinsam, werden sie automatisch über beide Felder verknüpft, und es kann zu Fehlinterpretationen kommen. Um dieses Problem zu umgehen, generiert Qlik Sense synthetische Schlüssel. Synthetische Schlüssel enthalten alle möglichen Kombinationen der Werte zweier oder mehrerer Schlüsselfelder.

Eine höhere Anzahl zusammengesetzter Schlüssel kann den Arbeitsspeicherverbrauch erhöhen und die Leistung beeinträchtigen. Auch Datenmengen, Tabellenstruktur und andere Faktoren können eine Rolle spielen. Immer wenn es mehrere synthetische Schlüssel gibt, die voneinander abhängen, ist es eine gute Vorgehensweise, diese zu entfernen.

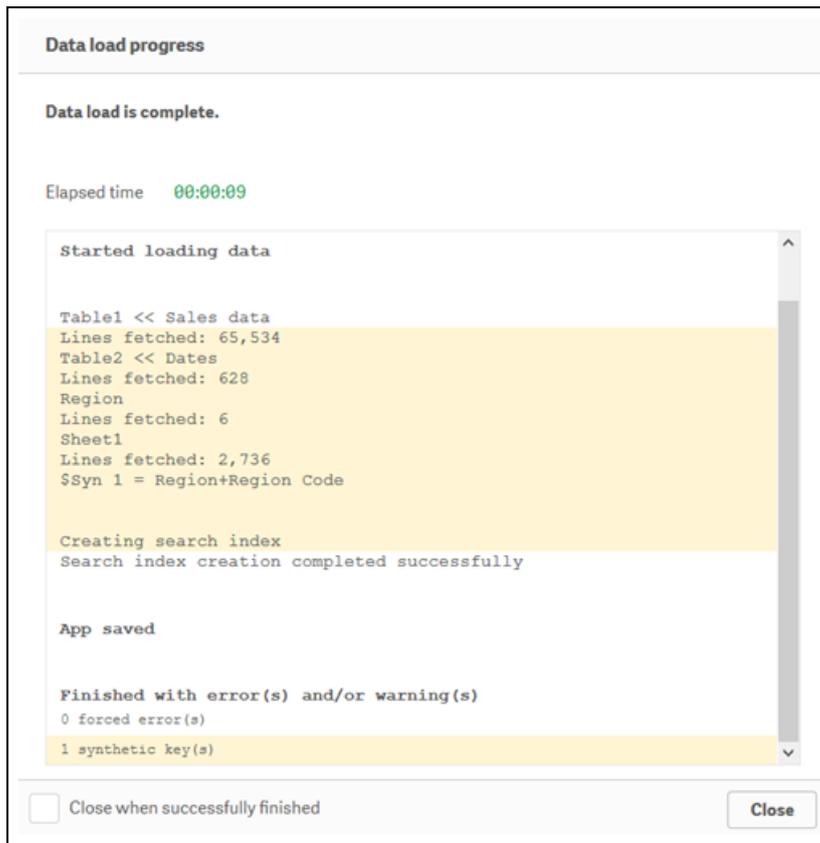
Jetzt ist es an der Zeit, den letzten Datensatz zu laden.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor** in der *Scripting Tutorial* App.
2. Klicken Sie auf **+**, um einen neuen Skriptabschnitt hinzuzufügen.
3. Nennen Sie den Abschnitt *Customers*.
4. Klicken Sie im rechten Menü unter **AttachedFiles** auf **Daten auswählen**.
5. Laden Sie *Customers.xlsx* hoch und wählen Sie die Datei aus. Das Datenvorschaufenster wird geöffnet.
6. Wählen Sie *Sheet1* aus.
7. Klicken Sie auf **Skript einfügen**.
8. Klicken Sie auf **Daten laden**.

Sie können jetzt im Fenster mit dem Datenladefortschritt sehen, dass ein synthetischer Schlüssel erstellt wurde.

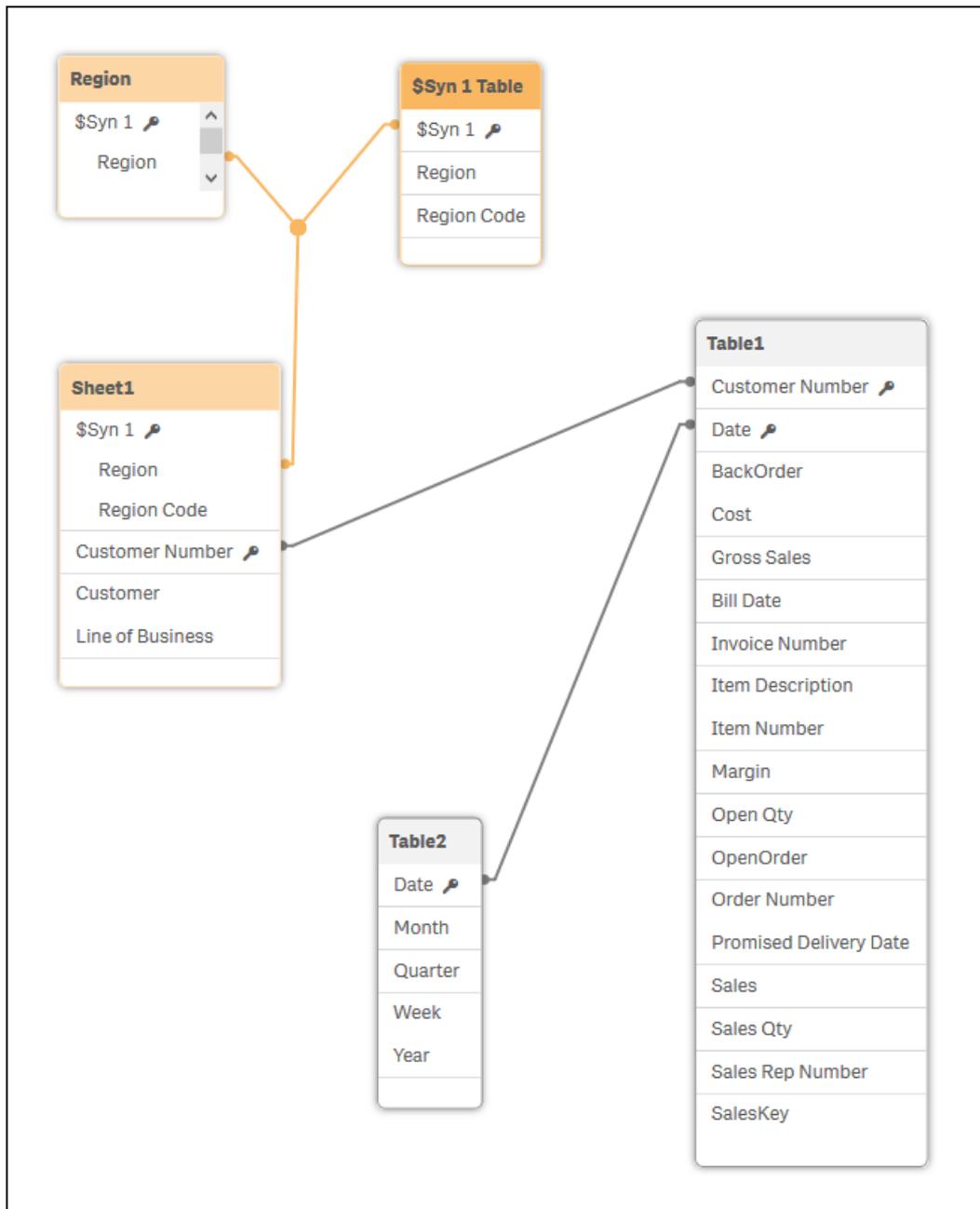
Fenster für Datenladefortschritt mit Warnung synthetischer Schlüssel



9. Öffnen Sie die **Datenmodellansicht**.

Aus der Erstellung einer neuen Tabelle *\$Syn 1 Table* ist ersichtlich, dass ein synthetischer Schlüssel erstellt wurde. Er enthält alle Felder, *Region* und *Region code*, welche die verbundenen Tabellen *Sheet1* und *Region* gemeinsam haben. In diesem Fall gestalten sich die Verbindungen dadurch etwas verwirrend und irreführend, was zu vermeiden ist.

Datenmodellansicht mit synthetischem Schlüssel



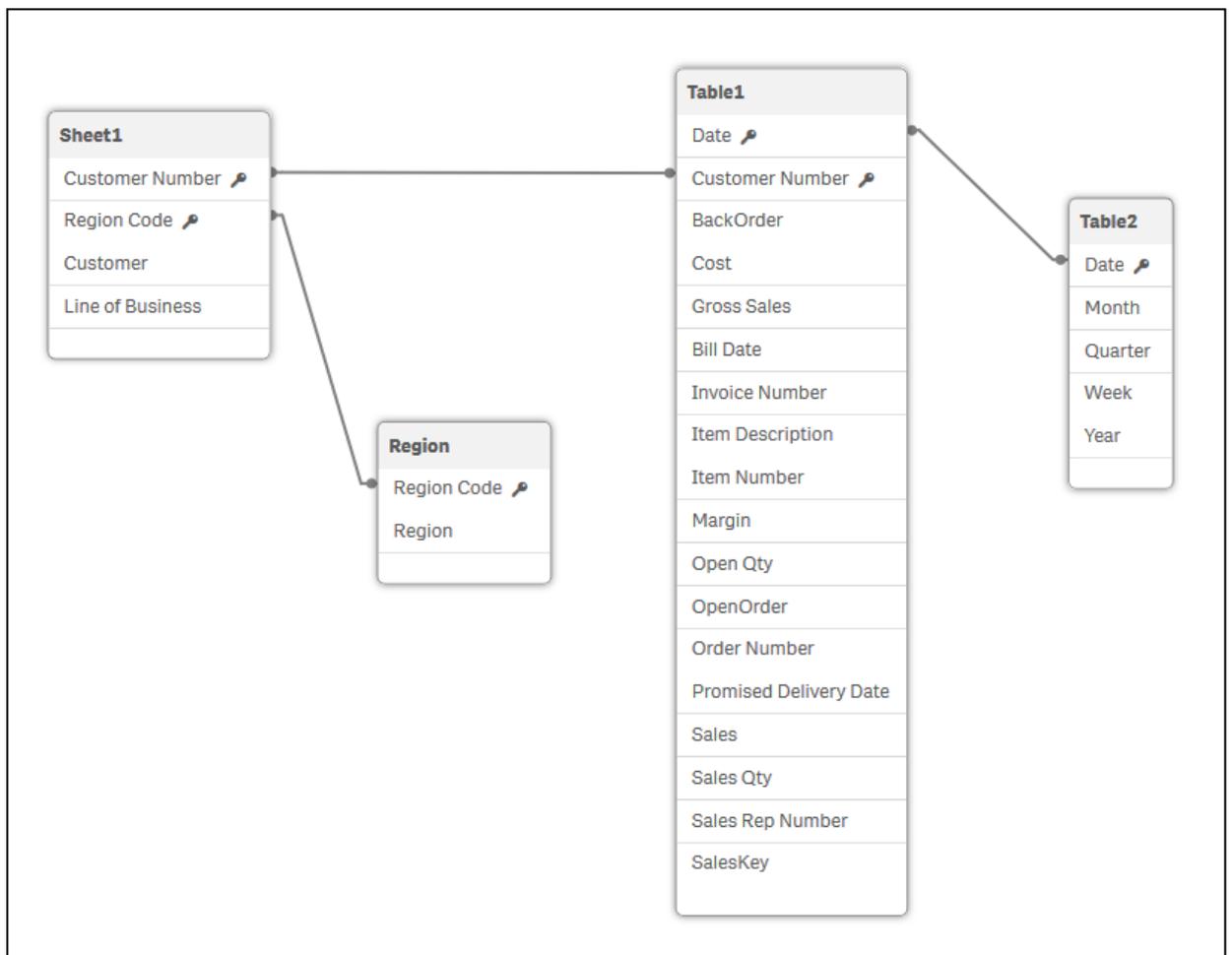
10.1 Auflösen synthetischer Schlüssel

Am einfachsten kann synthetischen Schlüsseln durch das Umbenennen von einem oder mehreren Feldern in der Tabelle entgegengewirkt werden. Dies kann beim Laden der Daten erfolgen. Jetzt werden die Schritte zur Entfernung eines synthetischen Schlüssels durchgegangen.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den **Dateneditor**.
2. Klicken Sie auf den Abschnitt *Customers* und löschen Sie die folgende Zeile in der **LOAD**-Anweisung:
Region,
3. Klicken Sie auf **Daten laden**.
4. Öffnen Sie die **Datenmodellansicht**.
Der synthetische Schlüssel wurde entfernt.

Die Datenmodellansicht zeigt, dass der synthetische Schlüssel entfernt wurde



11 Verwenden der Daten in einer App

Zum Abschluss dieses Tutorials verwenden Sie Ihre geladenen Daten in einer Visualisierung in Ihrer App.

11.1 Hinzufügen eines Diagramms

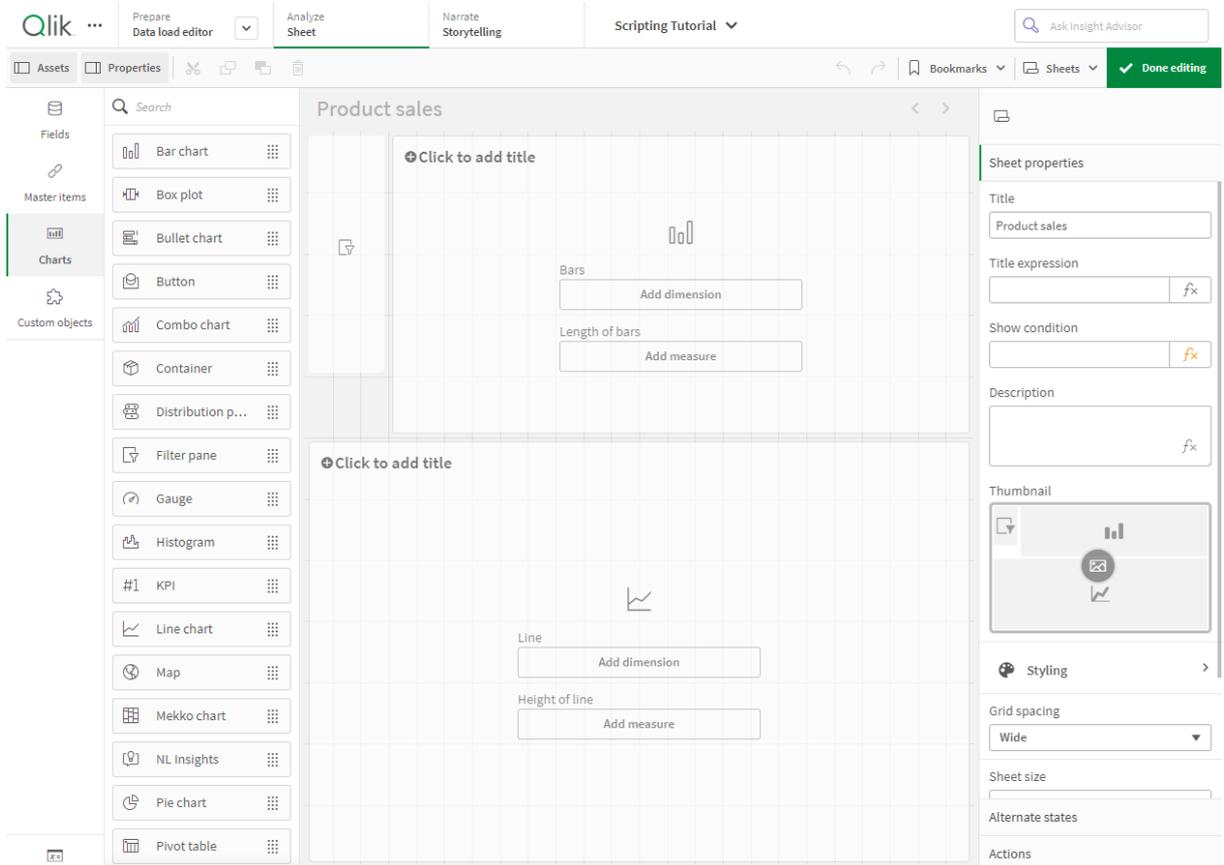
Nachdem Sie Ihre Daten geladen haben, können Sie Diagramme mit den Daten erstellen. Diagramme werden oft auch als Visualisierungen bezeichnet. Ein Diagramm ist jedoch erst vollständig, nachdem Sie die erforderlichen Dimensionen und Kennzahlen hinzugefügt haben. Fügen Sie zunächst die Diagramme hinzu und fahren Sie dann mit Dimensionen und Kennzahlen fort.

Gehen Sie folgendermaßen vor:

1. Erstellen Sie ein neues Arbeitsblatt in Ihrer *Scripting Tutorial* App.
2. Ziehen Sie aus der Registerkarte **Diagramme** ein Filterfenster auf das Arbeitsblatt und passen Sie die Größe mithilfe der Griffe so an, dass es 3 Zellen breit und 4 Zellen hoch ist. Platzieren Sie es links oben im Arbeitsblatt.
3. Ziehen Sie ein Balkendiagramm nach rechts oben. Machen Sie es 5 Zellen hoch und so breit, dass es bis zur Seite des Arbeitsblatts reicht.
4. Ziehen Sie ein Liniendiagramm auf den verbleibenden Platz.
Die Symbole auf dem Arbeitsblatt zeigen, welche Art von Diagramm Sie hinzugefügt haben. Jetzt können Sie zu Ihren Diagrammen Dimensionen und Kennzahlen hinzufügen und als Visualisierungen abschließen.

11 Verwenden der Daten in einer App

Qlik Sense Arbeitsblatt mit leeren Diagrammen



11.2 Hinzufügen von Dimensionen und Kennzahlen

Als nächsten Schritt fügen wir Dimensionen und Kennzahlen hinzu. Fügen Sie zunächst Zeitdimensionen zum Filterfenster oben links hinzu. Mit einem Filterfenster können Sie Platz sparen. Sie brauchen keine eigenen Filterfenster für *Year*, *Quarter*, *Month* und *Week*, sondern nur ein einziges Filterfenster.

Erstellen und Hinzufügen von Dimensionen

Gehen Sie folgendermaßen vor:

1. Klicken Sie links oben im Extras-Fenster auf , um **Felder** zu öffnen. Hier sind alle Felder in allen Tabellen aufgeführt, die Sie im Dateneditor geladen haben.
2. Scrollen Sie zum Ende der Liste und klicken Sie auf das Feld *Year*. Ziehen Sie es in die Mitte des Filterfensters oben links.
3. Fügen Sie auf dieselbe Art und Weise *Quarter*, *Month* und *Week* zum Filterfenster hinzu. Sie haben jetzt ein Filterfenster mit vier Dimensionen erstellt: *Year*, *Quarter*, *Month* und *Week*.

Erstellen und Hinzufügen von Kennzahlen

Für die meisten Visualisierungen sind sowohl Dimensionen als auch Kennzahlen erforderlich. Eine Kennzahl ist das Ergebnis einer Aggregierungsformel, die in vielen Fällen eine häufig verwendete Funktion wie **Sum**, **Max**, **Min**, **Avg** (Durchschnitt) oder **Count** ist.

Im Balkendiagramm stellen Sie die Umsätze nach Region dar.

Gehen Sie folgendermaßen vor:

1. Klicken Sie auf **Felder** .
2. Klicken Sie auf das Feld *Region* und ziehen Sie es in die Mitte des Bereichs des Balkendiagramms.
3. Klicken Sie auf „**Region**“ **hinzufügen**.
4. Klicken Sie auf das Feld *Sales* und ziehen Sie es in die Mitte des Bereichs des Balkendiagramms.
5. Klicken Sie auf **Als Kennzahl hinzufügen > Sum(Sales)**.
6. Klicken Sie im Eigenschaftsfenster auf der rechten Seite auf **Darstellung** und dann auf **Präsentation**. Wählen Sie **Horizontal** aus.
Die Balken werden jetzt horizontal angezeigt.
7. Klicken Sie im Eigenschaftsfenster auf der rechten Seite auf **Sortieren**.
Die Sortierreihenfolge wird angezeigt.
8. Ziehen Sie *Sum([Sales])* über *Region*, damit die Dimensionen nach *Sum([Sales])* (Kennzahlwert) und nicht nach *Region* (Dimensionswert, alphabetisch) sortiert werden.
Das Balkendiagramm ist vollständig und zeigt die Umsatzergebnisse für die verschiedenen Regionen. Dies ist ein einfaches Balkendiagramm. Im Eigenschaftsfenster (rechts) gibt es viele Optionen, um die Gestaltung aufzuwerten. Um nur eine der Möglichkeiten zu zeigen: Sie können den Titelbereich für mehr als nur einen Titel verwenden.
9. Kopieren Sie folgenden String in das Titelfeld des Balkendiagramms:
`= 'Total sales: $' & Round(Sum(Sales)/1000000, 0.1) & 'M'`
10. Drücken Sie die Eingabetaste.

Die letzte Visualisierung auf diesem Arbeitsblatt ist ein Liniendiagramm.

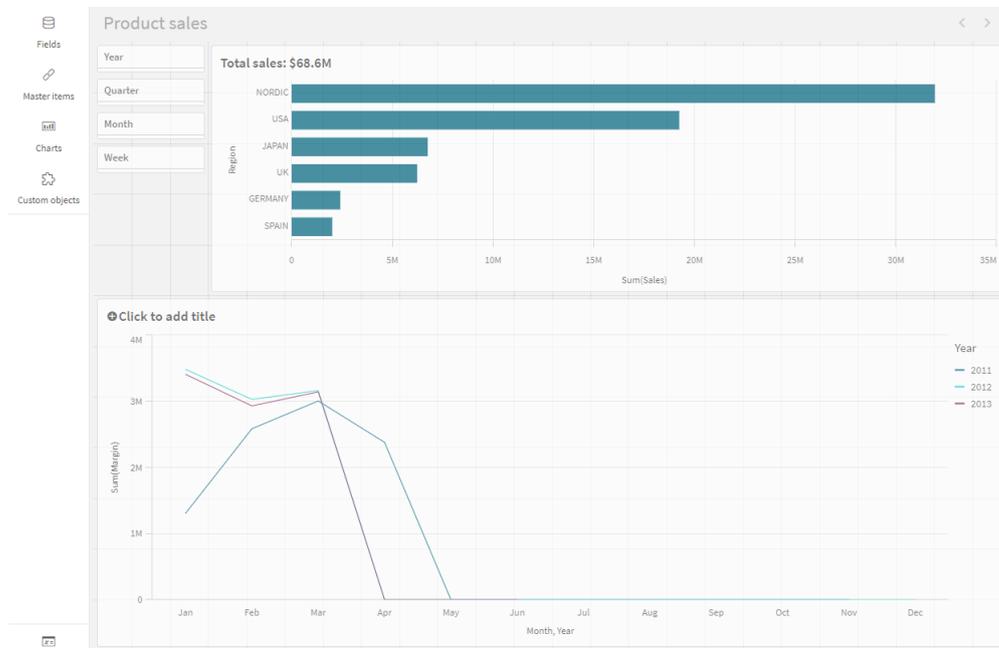
Gehen Sie folgendermaßen vor:

1. Klicken Sie auf **Felder** .
2. Klicken Sie auf das Feld *Month* und ziehen Sie es in die Mitte des Bereichs des Liniendiagramms.
3. Klicken Sie auf „**Monat**“ **hinzufügen**.
4. Klicken Sie auf das Feld *Year* und ziehen Sie es in die Mitte des Bereichs des Liniendiagramms.
5. Klicken Sie auf „**Jahr**“ **hinzufügen**.

11 Verwenden der Daten in einer App

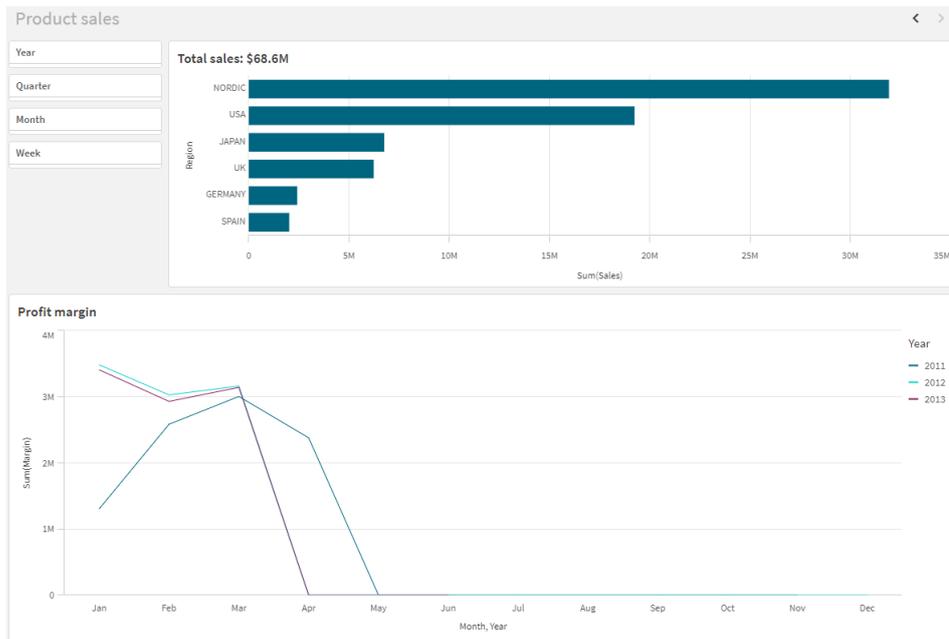
- Klicken Sie auf das Feld *Margin* und ziehen Sie es in die Mitte des Bereichs des Liniendiagramms.
- Klicken Sie auf **Als Kennzahl hinzufügen > Sum(Margin)**.
- Geben Sie den Titel *Profit margin* oben am Liniendiagramm ein.

Diagramme mit Daten



- Beenden Sie die Bearbeitung des Arbeitsblatts.
Das Arbeitsblatt ist jetzt vollständig und Sie können damit beginnen, sich durch die Daten zu klicken und mit dem Arbeitsblattinhalt zu interagieren.

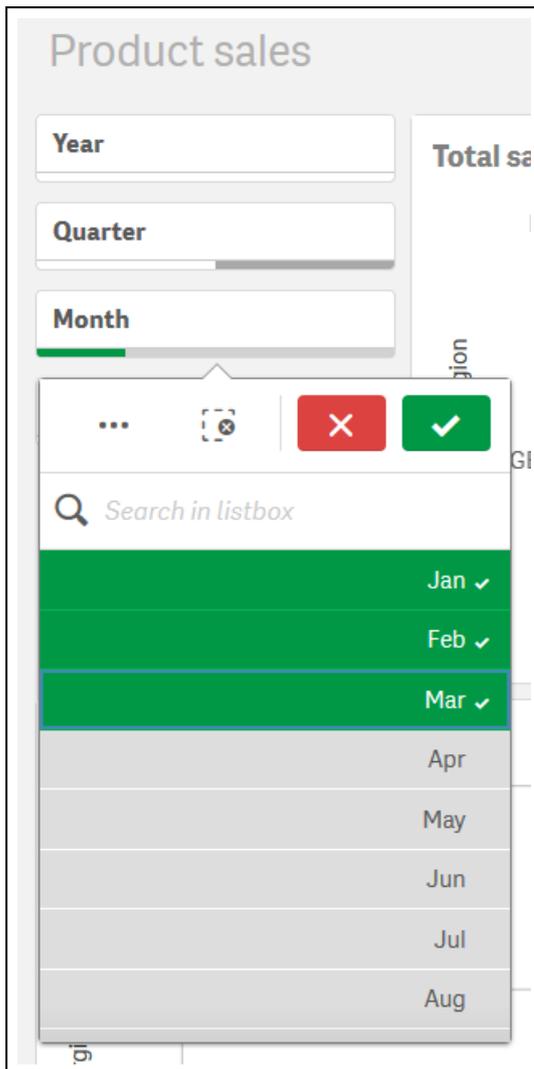
Fertiges Arbeitsblatt



Da in der ursprünglichen Datei *Sales.xlsx* nur wenige monatliche Umsatzdaten vorhanden sind, stehen im Diagramm nach Ende März der einzelnen Jahre nur wenige Daten zur Verfügung. Sie können Auswahlen im Filterfenster treffen, damit nur die ersten drei Monate der einzelnen Jahre verglichen werden.

10. Klicken Sie in der Filterbox auf das Feld *Month* und wählen Sie *Jan*, *Feb* und *Mar* aus.

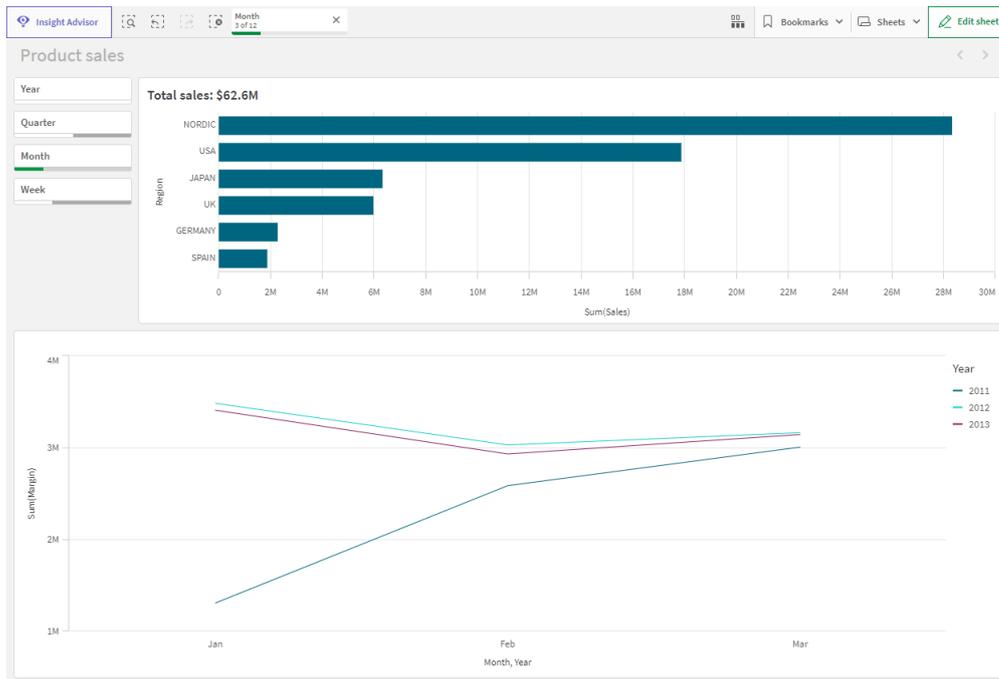
Filterbox



11. Schließen Sie das Filterfenster. Das Diagramm *Profit margin* zeigt die Daten für die ersten drei Monate der einzelnen Jahre.

11 Verwenden der Daten in einer App

Entsprechend den Auswahlen aktualisiertes Diagramm Profit margin



11.3 Vielen Dank!

Sie haben dieses Tutorial jetzt abgeschlossen und hoffentlich grundlegende Kenntnisse zur Verwendung von Skripts in Qlik Sense erworben. Besuchen Sie bitte unsere Website, um weitere Anregungen für Ihre Apps zu erhalten.