

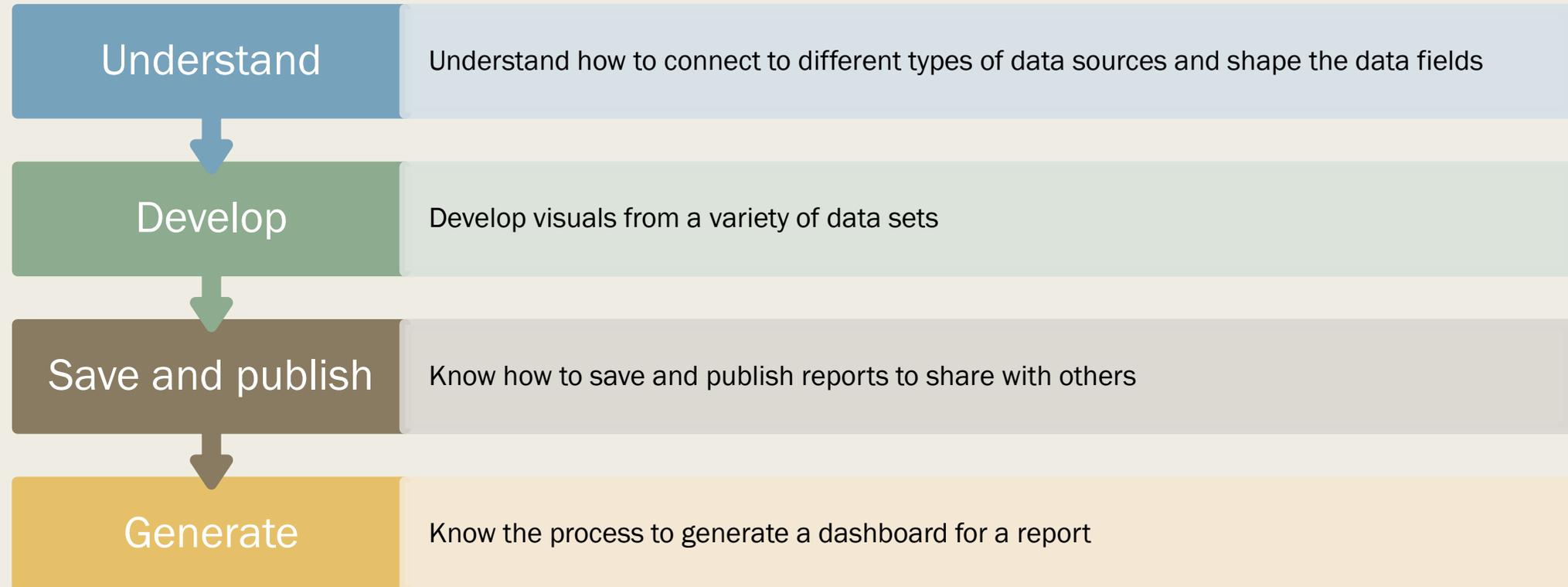
Bring your data to life with Microsoft Power BI

Instructor: Peggy Jones – Associate Director
Technology Support Services

Introduction to Microsoft Power BI



Course Takeaways





Microsoft Power BI is a business analytics collection of services, apps and visualizations that allows the users to visualize and share data for easy consumption.



Intuitive tool that is easy to learn that allows for users to easily connect to datasets, transform that data and tell a story..



Power BI consist of a Windows desktop application called Power BI Desktop and online SaaS service called Power BI Service



Workspaces are simply put, containers for your reports, datasets and dashboards in Power BI. Your workspaces can be “My Workspace” or the teams you are members of would also be listed.

What is Power BI?

What makes up a Power BI File

Visualizations

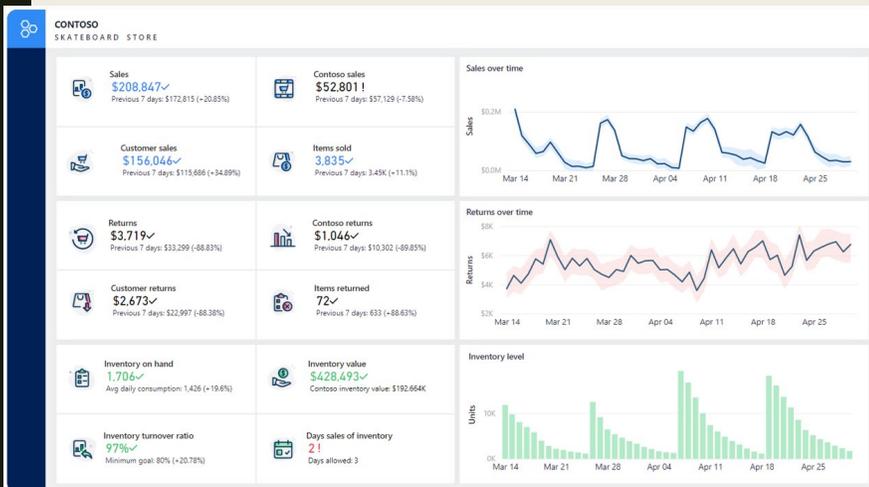
- A visual representation of data, such as a chart, a graph, a color-coded map, or other interesting things you can create to represent your data visually.

Report

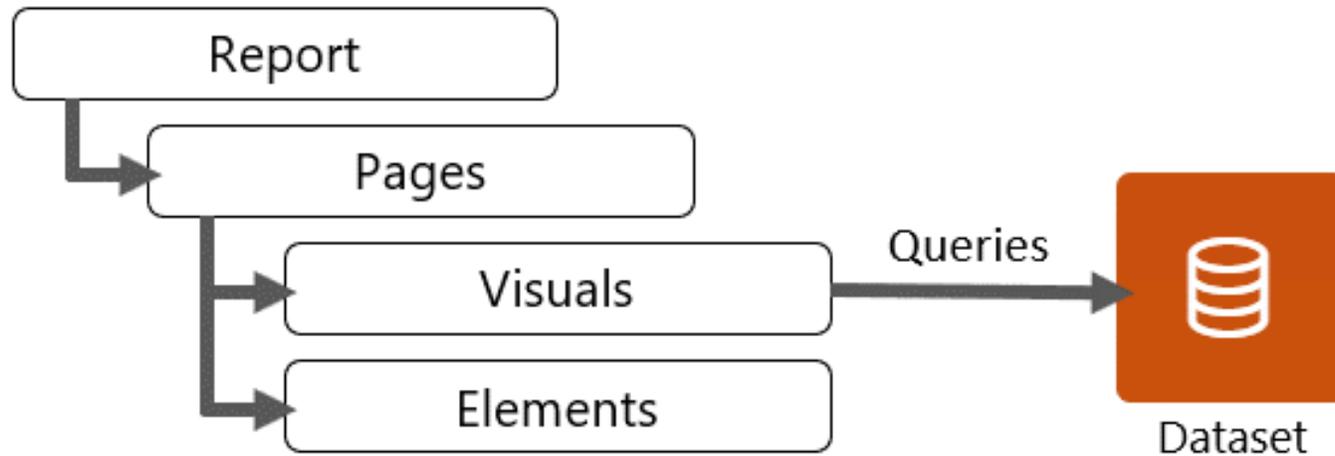
- A report is a collection of visualizations that appear together on one or more pages.
- A goal of a report is to help the viewer or user to find answers to a broad set of questions by utilizing the visuals.
 - Slicers are an important feature as they allow the user to filter the visuals quickly
- Operational reports can be used to see real-time data

Dashboards

- A dashboard is a collection of visuals from single or multiple pages that you can share with others. Often, it's a selected group of visuals that provide quick insight into the data or story you're trying to present. Think of a dashboard as a quick look at what is important, then you reopen the report to get more details.
 - The goal of creating a dashboard is so that the user can see the information as quickly as possible.
 - An example of a dashboard would be a departmental quick look or executive dashboard



See Your Data in New Ways



- Typically, a Power BI Report connects to one (or more) data sets and will have at least one report page.
- *It is very common to have multiple pages within your report as well*

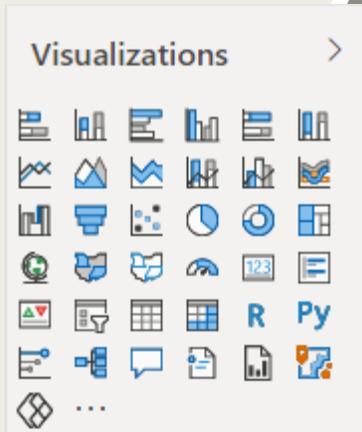
Parts of a Report

Visuals - a means to visualize a dataset

Elements - used to draw attention to visuals but do not interact with data

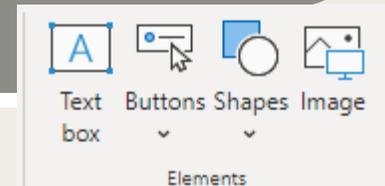
Visuals

- Microsoft Power BI includes over 30 standard visuals which are already available to the users
- You can create custom visuals, or they can be sourced from the AppSource

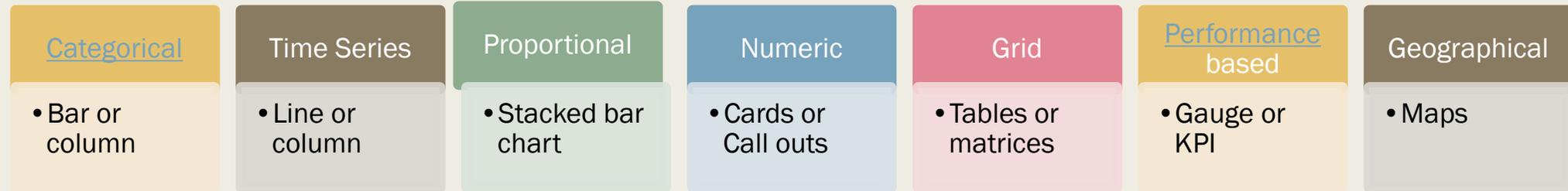


Elements

- Text Box - used to add titles or general instructions can be visually changed by font, color bold and alignment
- Buttons - you can assign a function to a button, so the user need only activate the button
- Shapes - can be used for decoration - information or to point to another element
- Images- can be used to add a logo to a report, or can even behave like a button



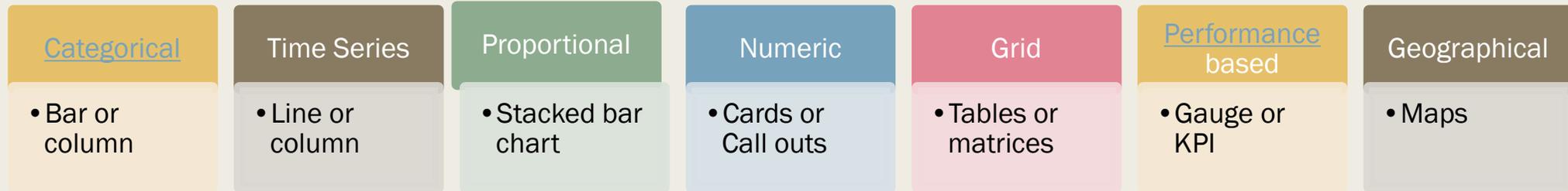
Types of Visuals

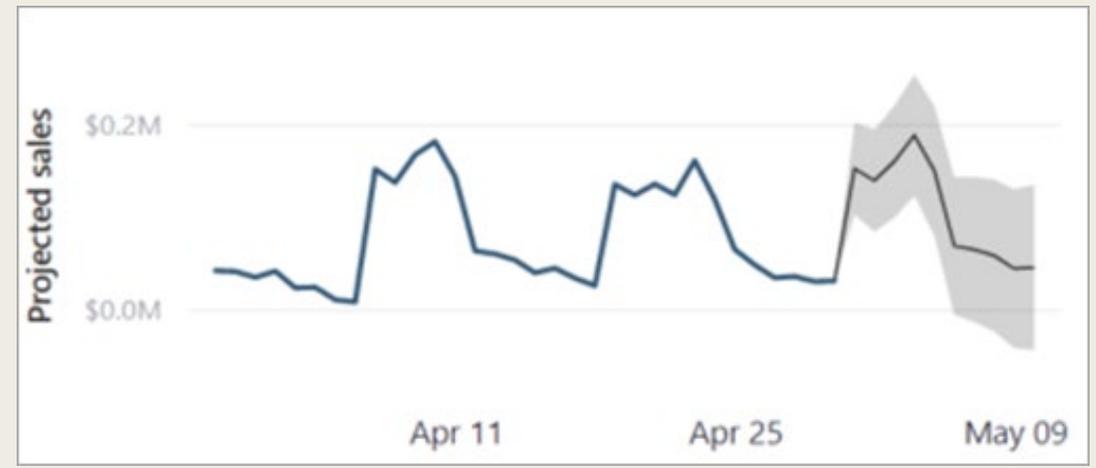




CATEGORICAL VISUAL

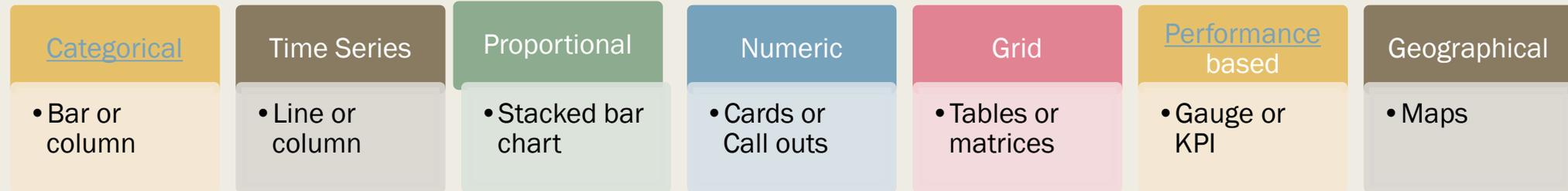
Types of Visuals

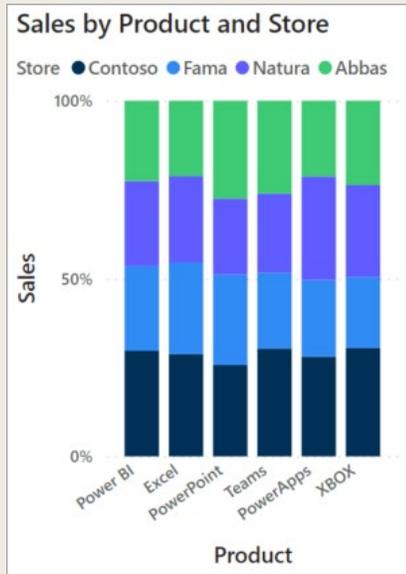




TIME SERIES VISUALS

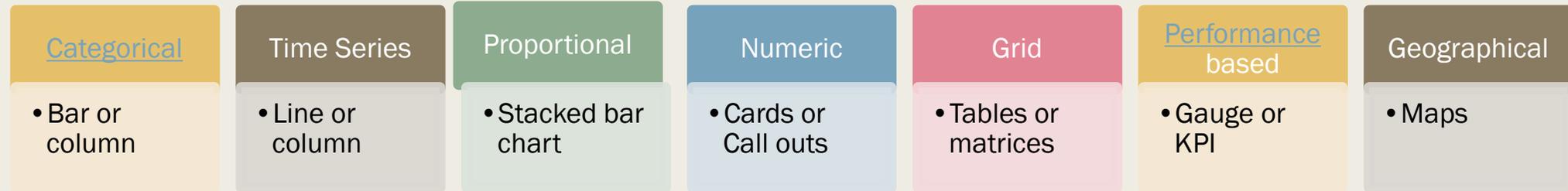
Types of Visuals

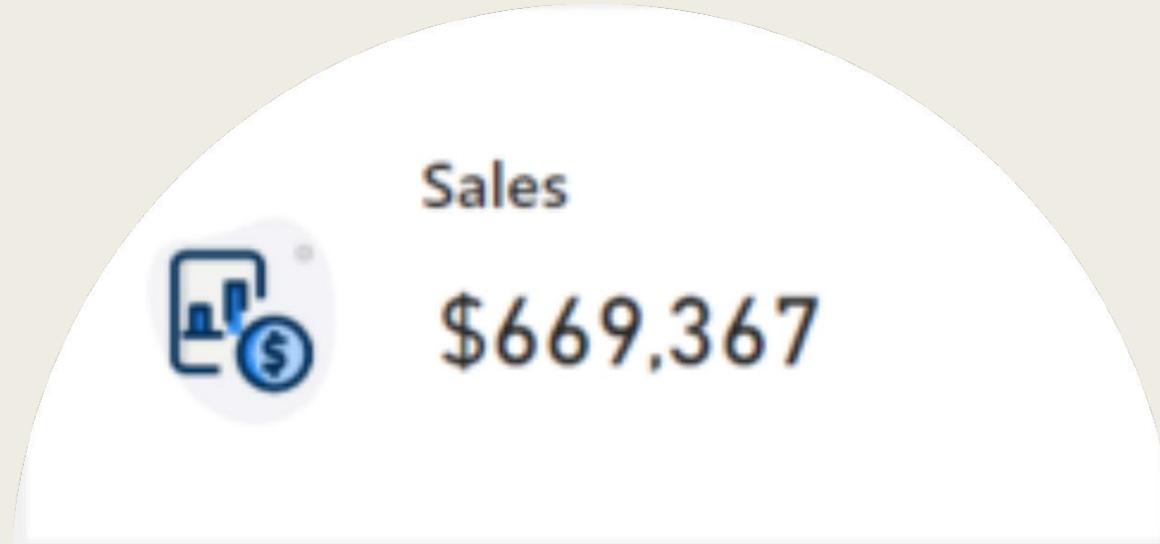




PROPORTIONAL VISUALS

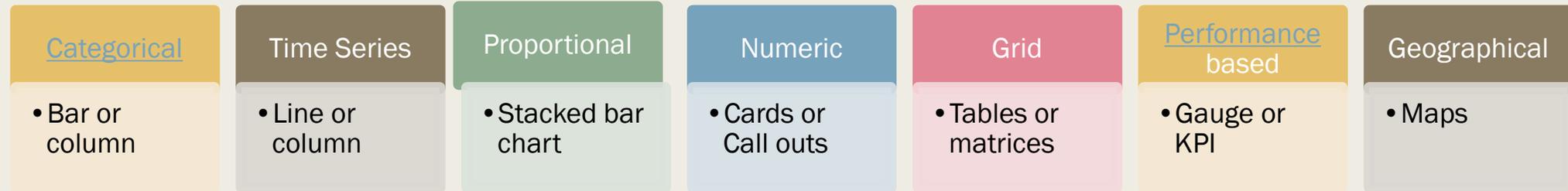
Types of Visuals





NUMERIC VISUALS

Types of Visuals



Inventory on hand breakdown

City	On hand	Inventory turnover ratio	Days sales of inventory
Los Angeles, CA, USA	32	100% ★	✓
Power BI	2 ○	100% ★	✓
Excel	0 ○	100% ★	⚠
PowerPoint	0 ○	100% ★	⚠
Teams	30 ○	99% ★	✓
PowerApps	0 ○	100% ★	⚠
Total	1,706	97% ★	✓

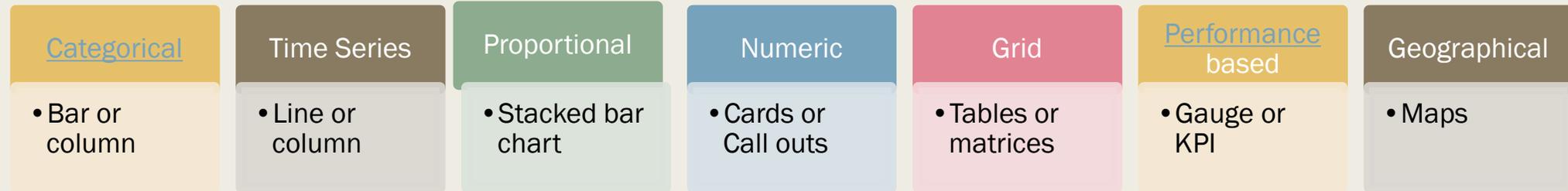
A matrix visual displays inventory by product and by store. It uses conditional formatting to show indicators, which provide visual cues to understanding the data.

Product	Sales	Units sold
Power BI	\$244,885	2,881
PowerPoint	\$172,320	2,154
Excel	\$102,250	2,045
Teams	\$88,144	3,148
PowerApps	\$80,496	2,236
XBOX	\$74,400	1,860
Total	\$762,495	14,324

Table Visual

GRID VISUALS

Types of Visuals





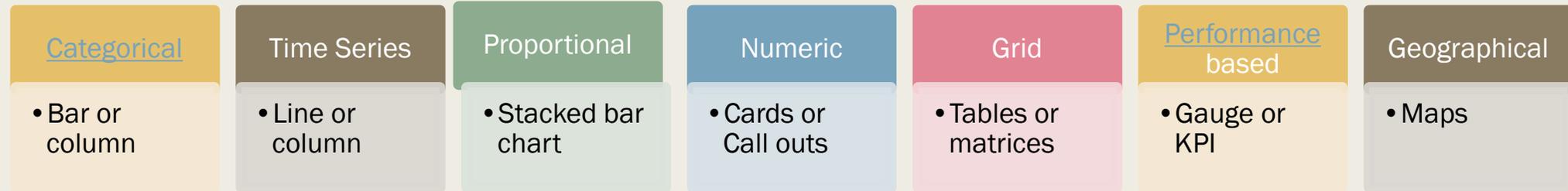
Contoso sales

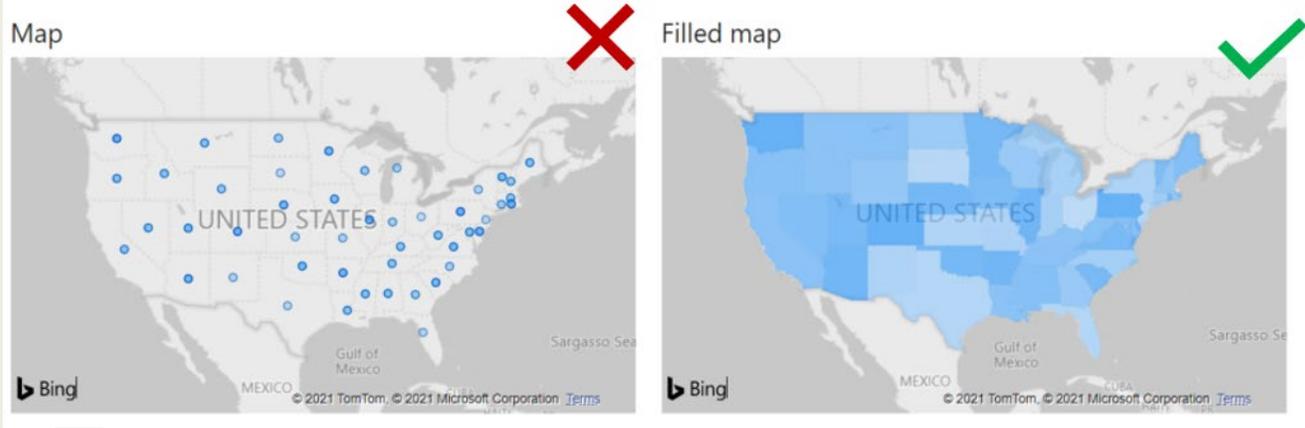
\$52,370✓

Target: \$50,000 (+4.74%)

PERFORMANCE VISUAL

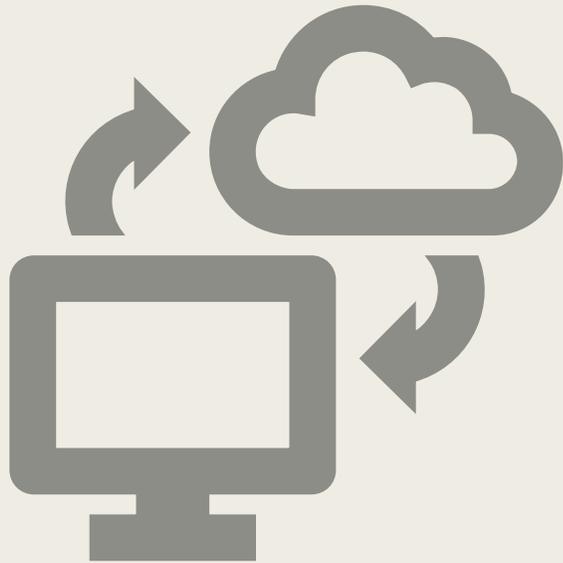
Types of Visuals





GEOSPATIAL VISUALS

Publish



Part of the Power BI Service

Publish directly to the cloud or on premise

Data refresh, so your reports are always up to date

Package your reports in mobile for easy consumption and control

Manage your analytics content with admin and governance tools



Power BI Service – Online Software as a Service

- We have been learning how to use Power BI Desktop to create reports
- Once you publish your report, you can create dashboards.
- Dashboards are used by the Power BI Service (online)



- *From here, you can pin the visuals from one or more reports that you have published to a single dashboard to allow easy consumption and quick answers to questions.*

Dashboards

A dashboard is a single page that is used to tell a story through visualizations

Pin Tiles from one or more reports to create a dashboard.

- Gives a quick view of information
- If more detail need a click of a visual will open the corresponding report



Resources

[Udemy TTU Self Paced Learning](#)

Power BI Documentation

- <https://docs.microsoft.com/en-us/power-bi/>

Power BI Guided Learning

- <https://learn.microsoft.com/en-us/training/paths/power-bi-effective/>
<https://docs.microsoft.com/en-us/power-bi/guided-learning/>

Power BI Dashboards

- <https://learn.microsoft.com/en-us/power-bi/create-reports/service-dashboard-create?source=recommendations>
- <https://learn.microsoft.com/en-us/training/modules/build-simple-dashboard/5-build-dashboard>

Power BI Webinars

- <https://docs.microsoft.com/en-us/power-bi/webinars>

Power BI Whitepapers

- <https://docs.microsoft.com/en-us/power-bi/whitepapers>

Power BI Blogs

- <https://powerbi.microsoft.com/en-us/blog/>

YouTube

- <https://youtu.be/IJKgWnvl6bQ>
- https://youtu.be/_3YAB6dNQ18?si=Kw3mw9HZyQAZk5li

Lab 1: Combine data from Excel and an OData feed

- It's common to have data spread across multiple data sources, such as product information in one database and sales information in another. With **Power BI Desktop**, you can combine data from different sources to create interesting, compelling data analyses and visualizations
- In this lab you learn how to combine data from two data sources: an Excel workbook that includes product information, and an OData feed that contains orders data. After you import each dataset and perform transformation and aggregation steps, you will use data from both sources to produce a sales analysis report with interactive visualizations. These techniques can also be applied to SQL Server queries, CSV files, and any other data sources in Power BI Desktop.
- For this lab you will need to open the link in the file explorer for “X” short courses and open the files we will be using for this lab exercise.

Import the product data from Excel

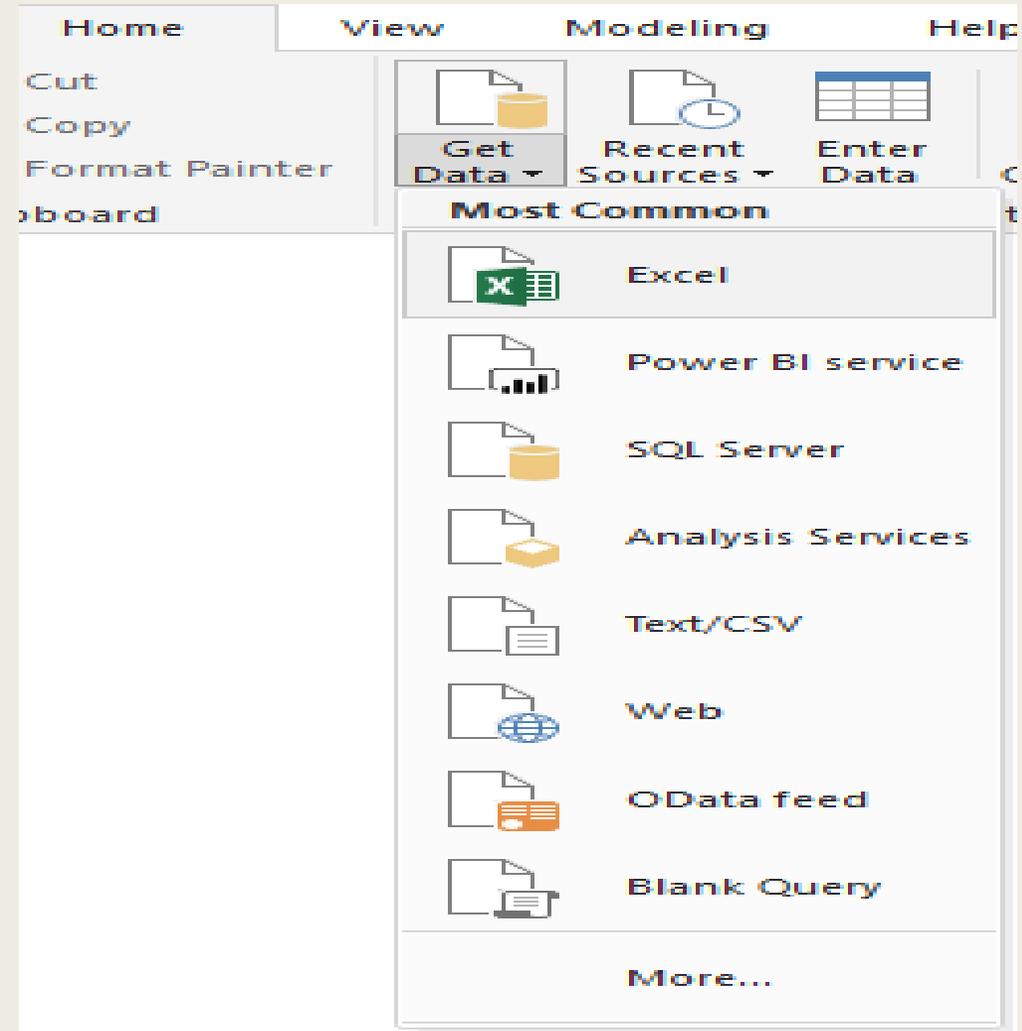
First, import the product data from the Excel Products.xlsx workbook into Power BI Desktop.

1. [Download the Products.xlsx Excel workbook](#), and save it as **Products.xlsx**. (This is saved in file location)
2. Select the dropdown arrow next to **Get Data** in the **Home** tab of the Power BI Desktop ribbon, and then select **Excel** from the **Most Common** dropdown.

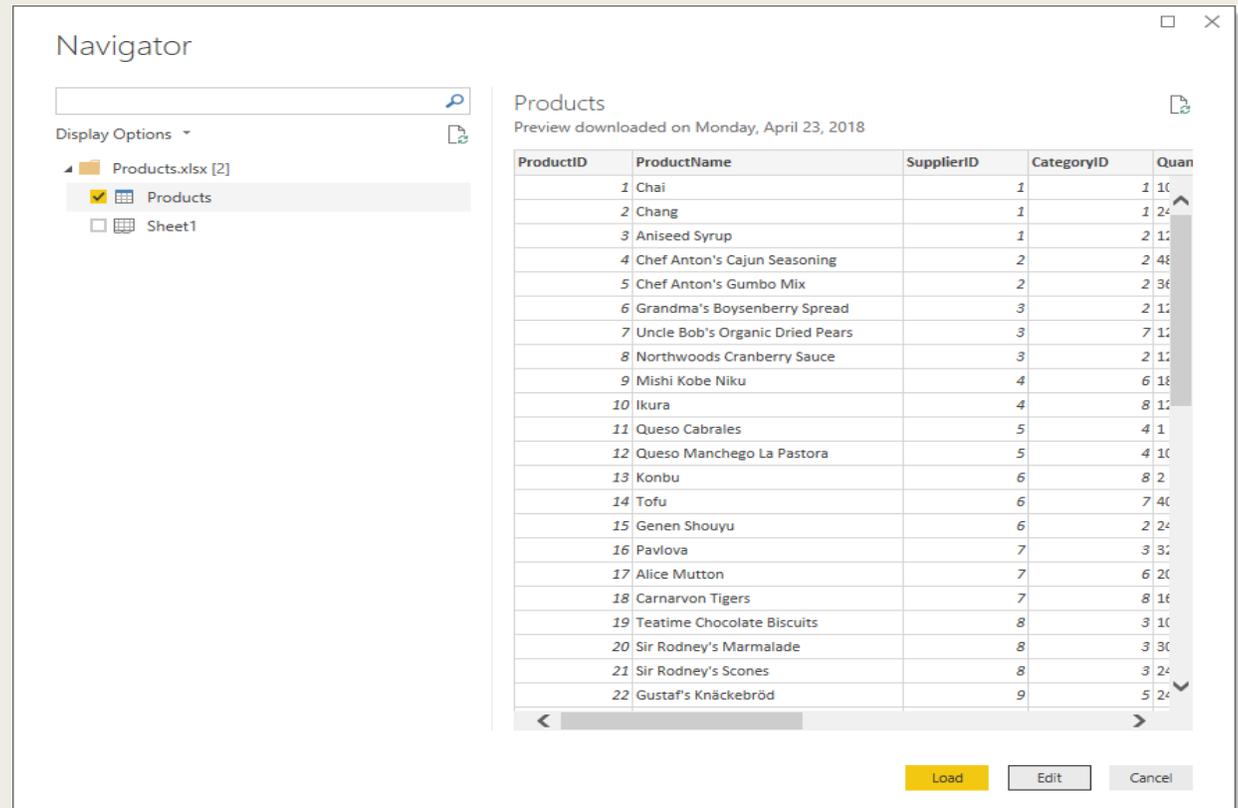
Note

You can also select the **Get Data** item itself, or select **Get Data** from the Power BI **Get started** dialog, then select **Excel** or **File > Excel** in the **Get Data** dialog box, and then select **Connect**.

3. In the **Open** dialog box, navigate to and select the **Products.xlsx** file, and then select **Open**.
4. In the **Navigator** pane, select the **Products** table and then select **Transform Data**.

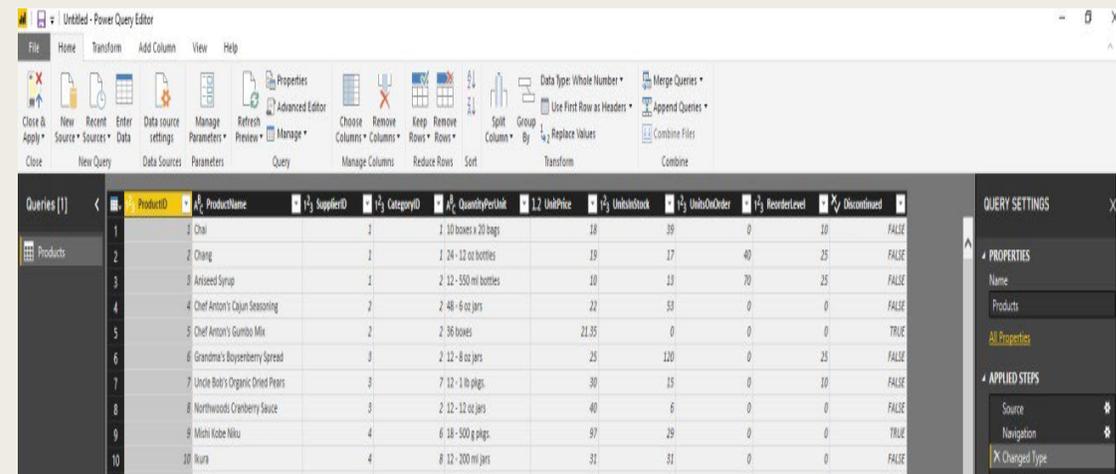


A preview of the table opens in the **Power Query Editor**, where you can apply transformations to clean up the data.



Note

You can also open the **Power Query Editor** by selecting **Edit Queries > Edit Queries** from the **Home** ribbon in Power BI Desktop, or by right-clicking or choosing **More options** next to any query in **Report View**, and selecting **Edit Query**.

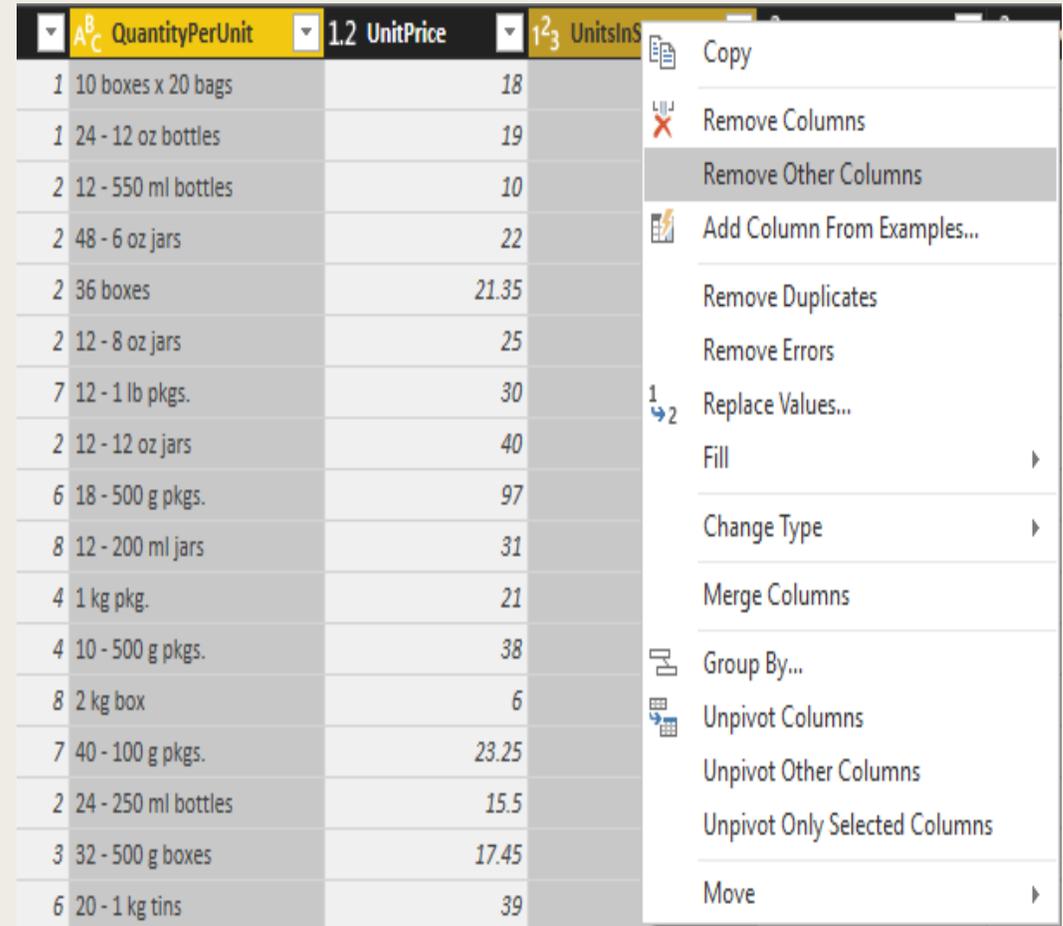


Clean up the products columns

Your combined report will only use the **ProductID**, **ProductName**, **QuantityPerUnit**, and **UnitsInStock** columns from the Excel workbook, so you can remove the other columns.

1. In **Power Query Editor**, select the **ProductID**, **ProductName**, **QuantityPerUnit**, and **UnitsInStock** columns (use **Ctrl+Click** to select more than one column, or **Shift+Click** to select columns that are next to each other).
2. Right-click any of the selected headers and select **Remove Other Columns** from the dropdown, to remove all except the selected columns from the table.

Note: You can also select **Remove Columns > Remove Other Columns** from the **Manage Columns** group in the **Home** ribbon tab.

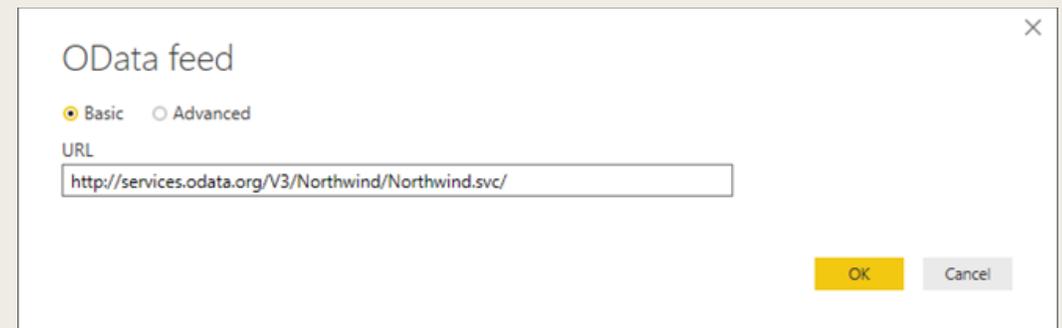
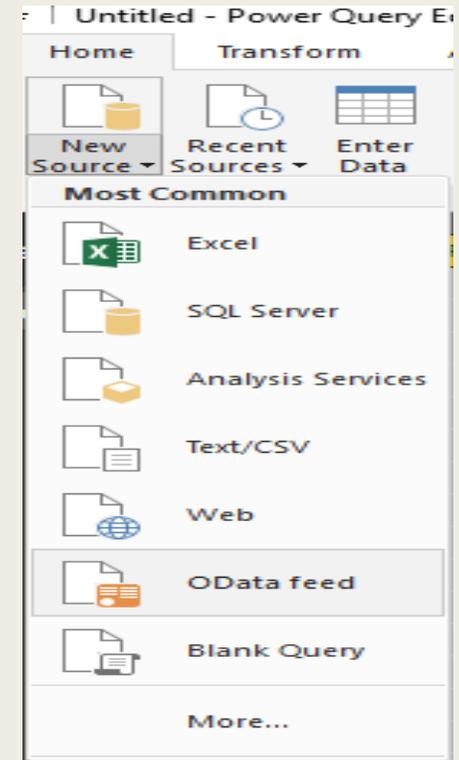


	QuantityPerUnit	UnitPrice	UnitsInStock
1	10 boxes x 20 bags	18	
1	24 - 12 oz bottles	19	
2	12 - 550 ml bottles	10	
2	48 - 6 oz jars	22	
2	36 boxes	21.35	
2	12 - 8 oz jars	25	
7	12 - 1 lb pkgs.	30	
2	12 - 12 oz jars	40	
6	18 - 500 g pkgs.	97	
8	12 - 200 ml jars	31	
4	1 kg pkg.	21	
4	10 - 500 g pkgs.	38	
8	2 kg box	6	
7	40 - 100 g pkgs.	23.25	
2	24 - 250 ml bottles	15.5	
3	32 - 500 g boxes	17.45	
6	20 - 1 kg tins	39	

Import the order data from an OData feed

Next, import the order data from the sample Northwind sales system OData feed.

1. In **Power Query Editor**, select **New Source** and then select **OData feed** from the **Most Common** dropdown.
2. In the **OData Feed** dialog box, paste the URL for the Northwind OData feed, <http://services.odata.org/V3/Northwind/Northwind.svc/>, and select **OK**.



3. In the **Navigator** pane, select the **Orders** table, and then select **OK** to load the data into **Power Query Editor**.

Note

In **Navigator**, you can select any table name, without selecting the checkbox, to see a preview.

The screenshot shows the 'Navigator' pane with a list of tables on the left and a preview of the 'Orders' table on the right. The 'Orders' table is selected, and its data is displayed in a table format. The data includes columns for OrderID, CustomerID, EmployeeID, OrderDate, and RequiredDate. The 'Orders' table is highlighted in the list, and the 'OK' button is visible at the bottom right.

OrderID	CustomerID	EmployeeID	OrderDate	RequiredDate
10248	VINET	5	7/4/1996 12:00:00 AM	8/1/1996
10249	TOMSP	6	7/5/1996 12:00:00 AM	8/16/1996
10250	HANAR	4	7/8/1996 12:00:00 AM	8/5/1996
10251	VICTE	3	7/8/1996 12:00:00 AM	8/5/1996
10252	SUPRD	4	7/9/1996 12:00:00 AM	8/6/1996
10253	HANAR	3	7/10/1996 12:00:00 AM	7/24/1996
10254	CHOPS	5	7/11/1996 12:00:00 AM	8/8/1996
10255	RICSU	9	7/12/1996 12:00:00 AM	8/9/1996
10256	WELLI	3	7/15/1996 12:00:00 AM	8/12/1996
10257	HILAA	4	7/16/1996 12:00:00 AM	8/13/1996
10258	ERNSH	1	7/17/1996 12:00:00 AM	8/14/1996
10259	CENTC	4	7/18/1996 12:00:00 AM	8/15/1996
10260	OTTIK	4	7/19/1996 12:00:00 AM	8/16/1996
10261	QUEDE	4	7/19/1996 12:00:00 AM	8/16/1996
10262	RATTC	8	7/22/1996 12:00:00 AM	8/19/1996
10263	ERNSH	9	7/23/1996 12:00:00 AM	8/20/1996
10264	FOLKO	6	7/24/1996 12:00:00 AM	8/21/1996
10265	BLONP	2	7/25/1996 12:00:00 AM	8/22/1996
10266	WARTH	3	7/26/1996 12:00:00 AM	9/6/1996
10267	FRANK	4	7/29/1996 12:00:00 AM	8/26/1996
10268	GROSR	8	7/30/1996 12:00:00 AM	8/27/1996
10269	WHITC	5	7/31/1996 12:00:00 AM	8/14/1996
10270	WARTH	1	8/1/1996 12:00:00 AM	8/29/1996

Expand the order data

When you connect to data sources that have multiple tables, such as relational databases or the Northwind OData feed, you can use references between tables to build up your queries.

The **Orders** table contains references to several related tables. You can add the **ProductID**, **UnitPrice**, and **Quantity** columns from the related **Order_Details** table into the subject (**Orders**) table by using the **Expand** operation.

1. Scroll to the right in the **Orders** table until you can see the **Order_Details** column. Note that instead of data, it contains references to another table.



Order_Details
Table

2. Select the **Expand** icon () in the **Order_Details** column header.
3. In the **Expand** drop-down:
 - a. Select **(Select All Columns)** to clear all columns.
 - b. Select **ProductID**, **UnitPrice**, and **Quantity**, and then select **OK**.

The dialog box shows the 'Expand' operation configuration. It includes a search bar, radio buttons for 'Expand' and 'Aggregate', a list of columns with checkboxes, and a checkbox for 'Use original column name as prefix'. The 'Quantity' checkbox is highlighted with a dashed box.

Customer	Employee	Order_Details
Search Columns to Expand		
<input checked="" type="radio"/> Expand <input type="radio"/> Aggregate		
<input checked="" type="checkbox"/> (Select All Columns)		
<input type="checkbox"/> OrderID		
<input checked="" type="checkbox"/> ProductID		
<input checked="" type="checkbox"/> UnitPrice		
<input checked="" type="checkbox"/> Quantity		
<input type="checkbox"/> Discount		
<input type="checkbox"/> Order		
<input type="checkbox"/> Product		
<input checked="" type="checkbox"/> Use original column name as prefix		
		OK
		Cancel

After you expand the **Order_Details** table, the **Order_Details** column is replaced by the three new columns from the nested table, and there are new rows in the table for the added data from each order.

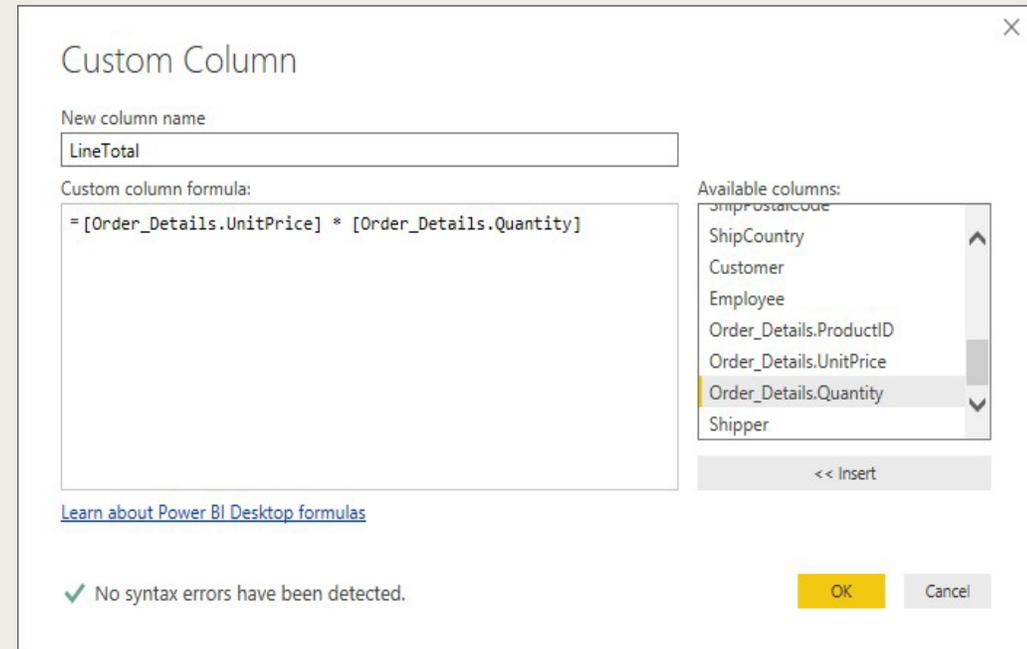
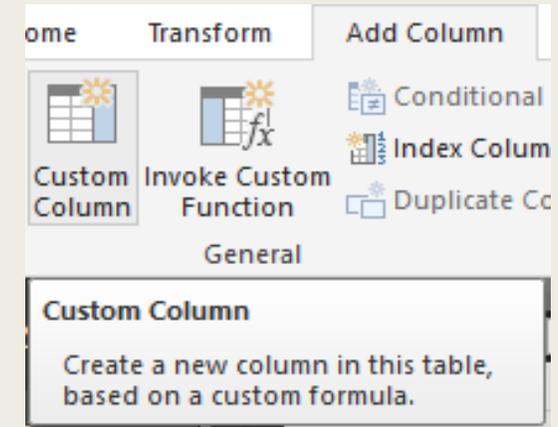
A ^B _C ShipCountry	Customer	Employee	1.2 ₃ Order_Details.ProductID	1.2 Order_Details.UnitPrice	1.2 Order_Details.Quantity
France	Record	Record	11	14	12
France	Record	Record	42	9.8	10
France	Record	Record	72	34.8	5
Germany	Record	Record	14	18.6	9
Germany	Record	Record	51	42.4	40
Brazil	Record	Record	41	7.7	10
Brazil	Record	Record	51	42.4	35
Brazil	Record	Record	65	16.8	15
France	Record	Record	22	16.8	6
France	Record	Record	57	15.6	15
France	Record	Record	65	16.8	20

Create a custom calculated column

Power Query Editor lets you create calculations and custom fields to enrich your data. You will create a custom column that calculates the total price for each line item in an order by multiplying the unit price by item quantity.

1. In the **Add Column** ribbon tab of Power Query Editor, select **Custom Column**.
2. In the **Custom Column** dialog box, type **LineTotal** in the **New column name** field.
3. In the **Custom column formula** field after the = , enter **[Order_Details.UnitPrice] * [Order_Details.Quantity]**. (You can also select the field names from the **Available columns** scroll box and select **<< Insert**, instead of typing them.)
4. Select **OK**.

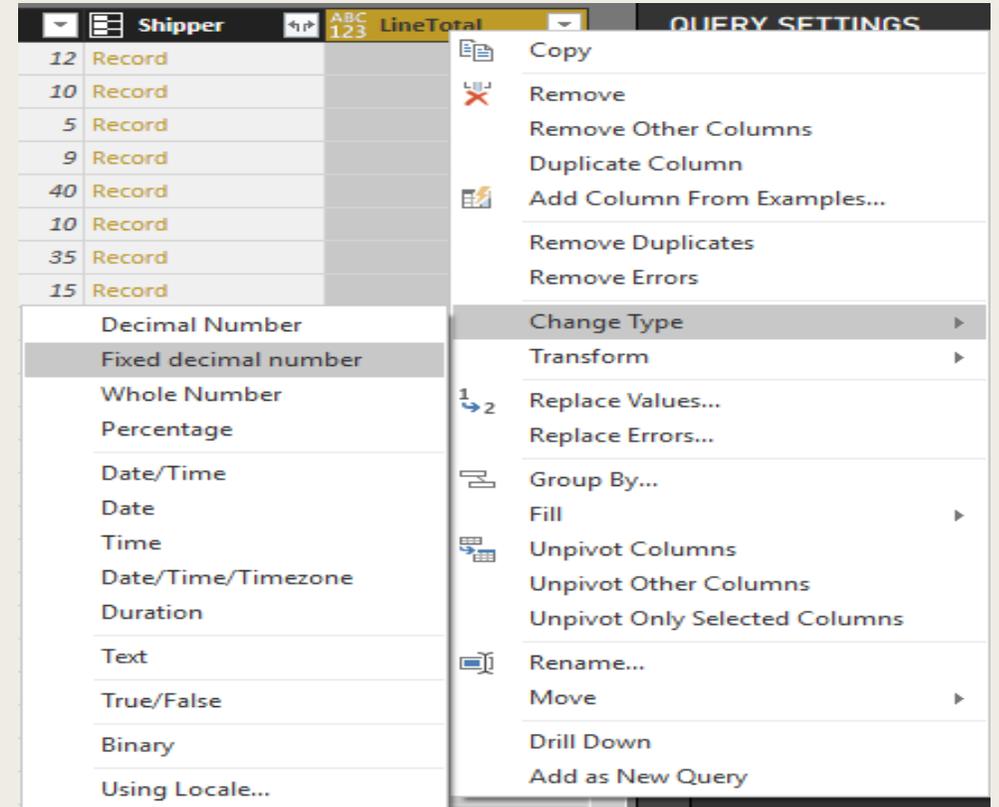
The new **LineTotal** field appears as the last column in the **Orders** table.



Set the data type for the new field

When Power Query Editor connects to data, it determines the best data type for each field and displays the data accordingly. You can see the data types assigned to fields by the icons in the headers, or under **Data Type** in the **Transform** group of the **Home** ribbon tab.

Your new **LineTotal** column has a data type of **Any**, but its values are currency. To assign a data type, right-click the **LineTotal** column header, select **Change Data Type** from the dropdown, and then select **Fixed decimal number**.



Note

You can also select the **LineTotal** column, then select the dropdown arrow next to **Data Type** in the **Transform** area of the **Home** ribbon tab, and then select **Fixed decimal number**.

Clean up the orders columns

To make your model easier to work with in reports, you can delete, rename, and reorder some of the columns.

Your report will only use the **OrderDate**, **ShipCity**, **ShipCountry**, **Order_Details.ProductID**, **Order_Details.UnitPrice**, and **Order_Details.Quantity** columns. You can select these columns and use **Remove Other Columns** as you did with the Excel data, or you can select all the columns except those listed, right-click on one of the selected columns, and select **Remove Columns** to remove them all.

You can make the **Order_Details.ProductID**, **Order_Details.UnitPrice** and **Order_Details.Quantity** columns easier to identify by removing the *Order_Details.* prefixes from the column names. To rename the columns to **ProductID**, **UnitPrice** and **Quantity**, respectively:

1. Double-click or tap and hold each column header, or right-click the column header and select **Rename** from the dropdown.
2. Delete the *Order_Details.* prefix from each name, and then press **Enter**.

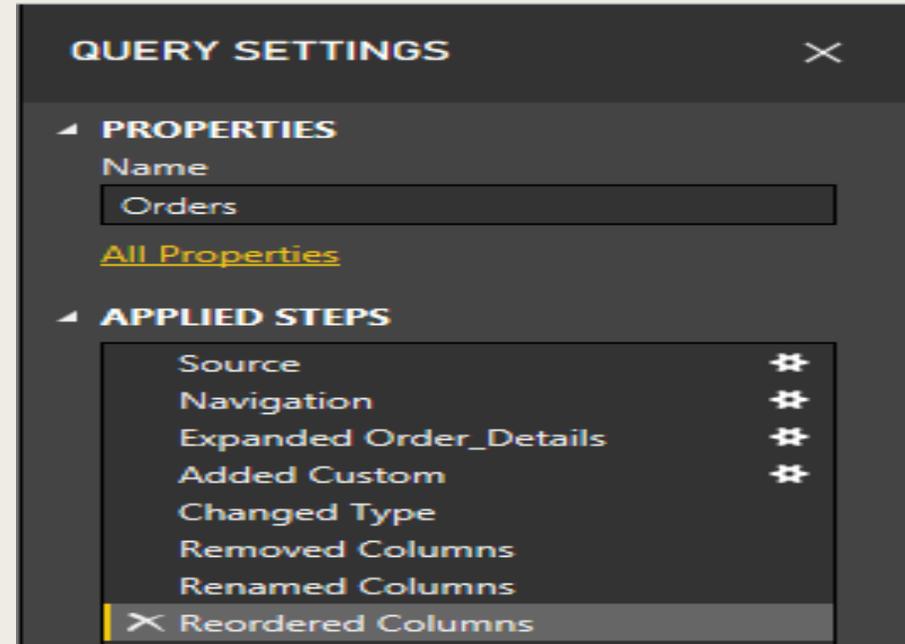
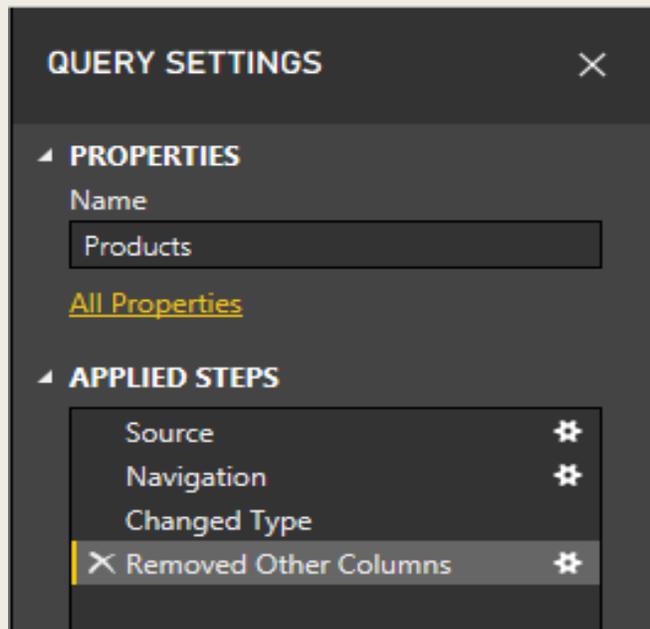
Finally, to make the **LineTotal** column easier to access, drag and drop it to the left, just to the right of the **ShipCountry** column.

	OrderDate	ShipCity	ShipCountry	LineTotal	ProductID	UnitPrice	Quantity
1	7/4/1996 12:00:00 AM	Reims	France	168	11	14	12
2	7/4/1996 12:00:00 AM	Reims	France	98	42	9.8	10
3	7/4/1996 12:00:00 AM	Reims	France	174	72	34.8	5
4	7/5/1996 12:00:00 AM	Münster	Germany	167.4	14	18.6	9
5	7/5/1996 12:00:00 AM	Münster	Germany	1696	51	42.4	40
6	7/8/1996 12:00:00 AM	Rio de Janeiro	Brazil	77	41	7.7	10
7	7/8/1996 12:00:00 AM	Rio de Janeiro	Brazil	1484	51	42.4	35
8	7/8/1996 12:00:00 AM	Rio de Janeiro	Brazil	252	65	16.8	15
9	7/8/1996 12:00:00 AM	Lyon	France	100.8	22	16.8	6
10	7/8/1996 12:00:00 AM	Lyon	France	234	57	15.6	15
11	7/8/1996 12:00:00 AM	Lyon	France	336	65	16.8	20

Review the query steps

As you shaped and transformed data in the Power Query Editor, each step was recorded in the **Applied Steps** area of the **Query Settings** pane on the right side of Power Query Editor. You can step back through the Applied Steps to review the changes you made, and edit, delete, or rearrange them if necessary (although this can be risky, because changing preceding steps can break later steps).

Select each of your queries in the **Queries** list on the left side of Power Query Editor, and review the **Applied Steps** in **Query Settings**. After applying the previous data transformations, the Applied Steps for your two queries should look like the following

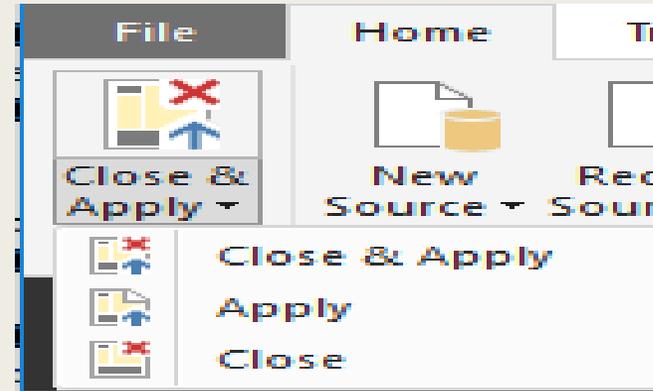


Tip

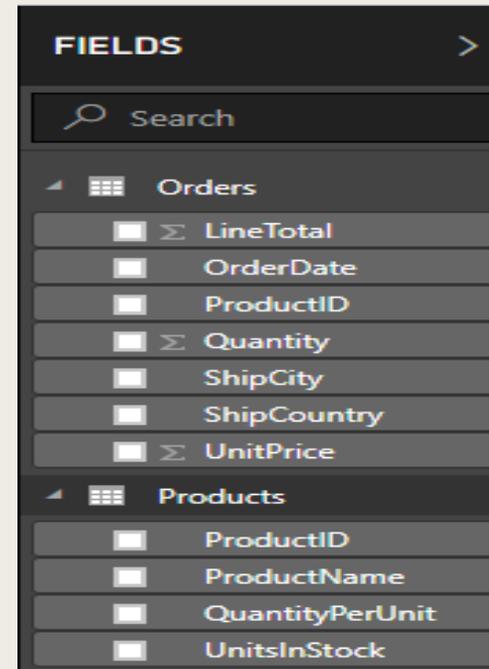
Underlying the Applied Steps are formulas written in the **Power Query Language**, also known as the **M**language. To see and edit the formulas, select **Advanced Editor** in the **Query** group of the Home tab of the ribbon.

Import the transformed queries

When you are satisfied with your transformed data, select **Close & Apply** > **Close & Apply** in the **Close** group of the **Home** ribbon tab, to import the data into Power BI Desktop Report View.



Once the data is loaded, the queries appear in the **Fields** list in the Power BI Desktop Report View .

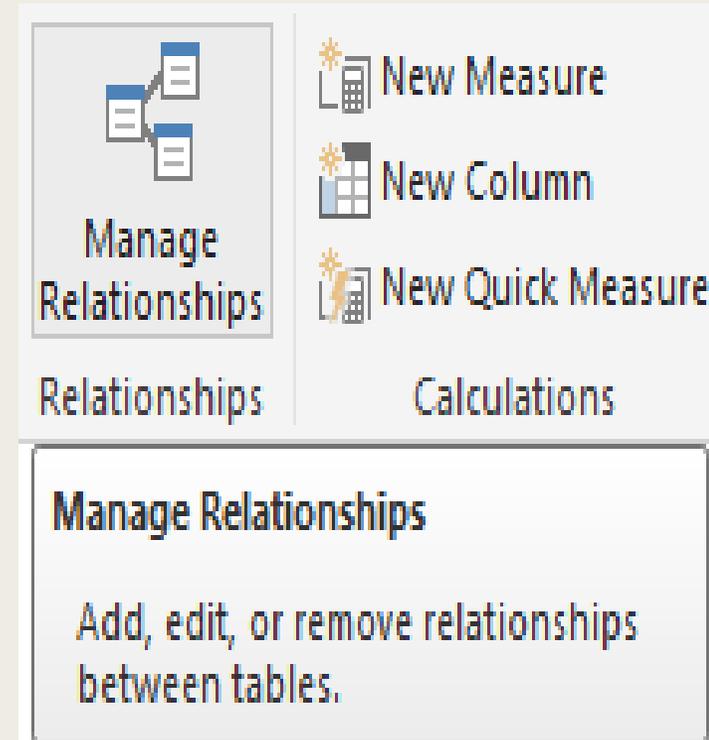


Manage the relationship between the datasets

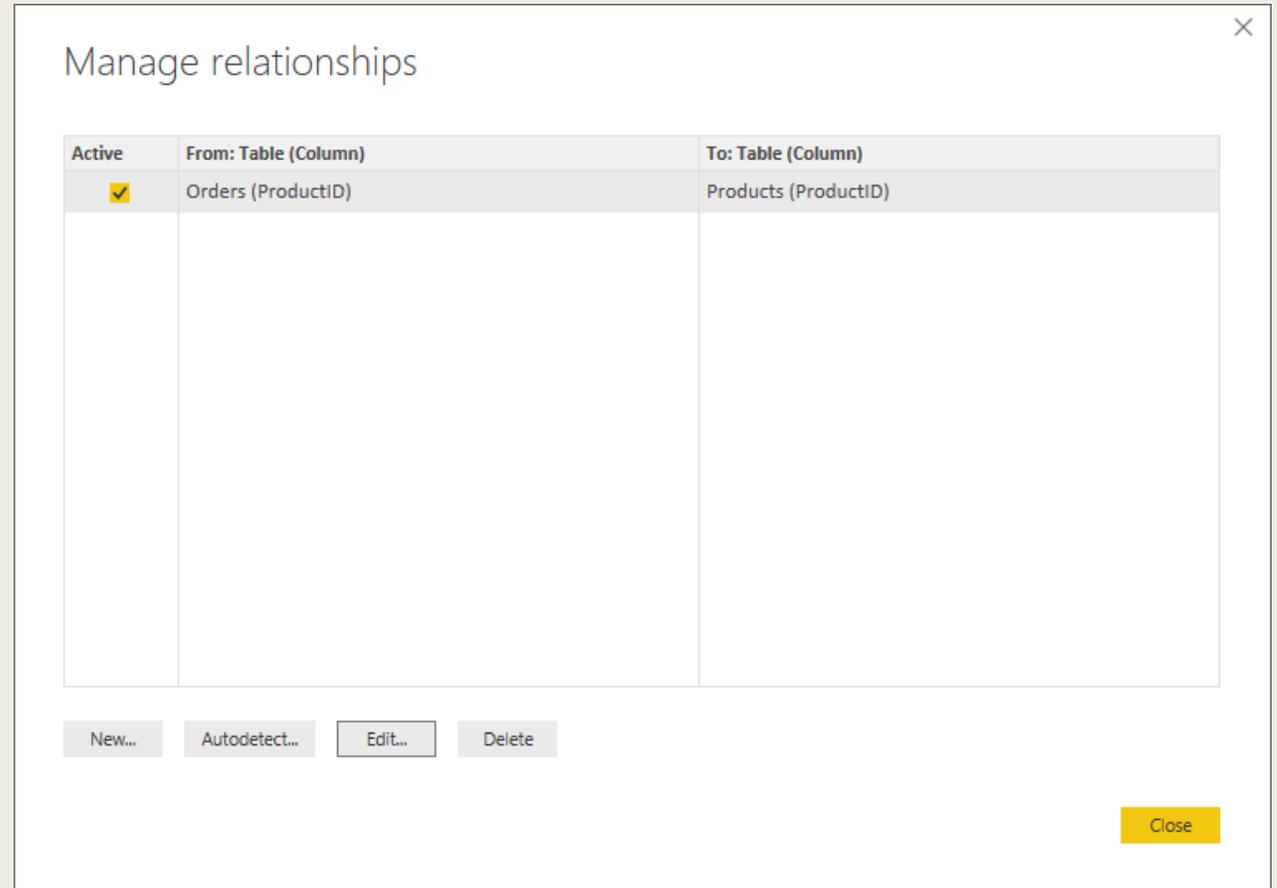
Power BI Desktop does not require you to combine queries to report on them. However, you can use the relationships between datasets, based on fields they have in common, to extend and enrich your reports. Power BI Desktop may detect relationships automatically, or you can create them in the Power BI Desktop **Manage Relationships** dialog. For more details about relationships in Power BI Desktop, see [Create and manage relationships](#).

The Orders and Products datasets in this tutorial share a common *ProductID* field, so there is a relationship between them based on that column.

1. In Power BI Desktop Report View, select **Manage Relationships** in the **Relationships** area of the **Home** ribbon tab.
2. In the **Manage Relationships** dialog, note that Power BI Desktop has already detected and listed an active relationship between the Products and Orders tables. To view the relationship, select **Edit**.



The **Edit Relationship** dialog opens, showing details about the relationship.



3. Power BI Desktop has autodetected the relationship correctly, so you can select **Cancel** and then **Close** to exit the relationship dialogs.

Edit relationship ✕

Select tables and columns that are related.

Orders

OrderDate	ShipCity	ShipCountry	LineTotal	ProductID	UnitPrice	Quantity
10/8/1996 12:00:00 AM	Boise	USA	\$291.9	16	13.9	21
10/8/1996 12:00:00 AM	Boise	USA	\$1,008	35	14.4	70
10/8/1996 12:00:00 AM	Boise	USA	\$288	46	9.6	30

Products

ProductID	ProductName	QuantityPerUnit	UnitsInStock
1	Chai	10 boxes x 20 bags	39
2	Chang	24 - 12 oz bottles	17
3	Aniseed Syrup	12 - 550 ml bottles	13

Cardinality: Many to one (*:1)

Cross filter direction: Single

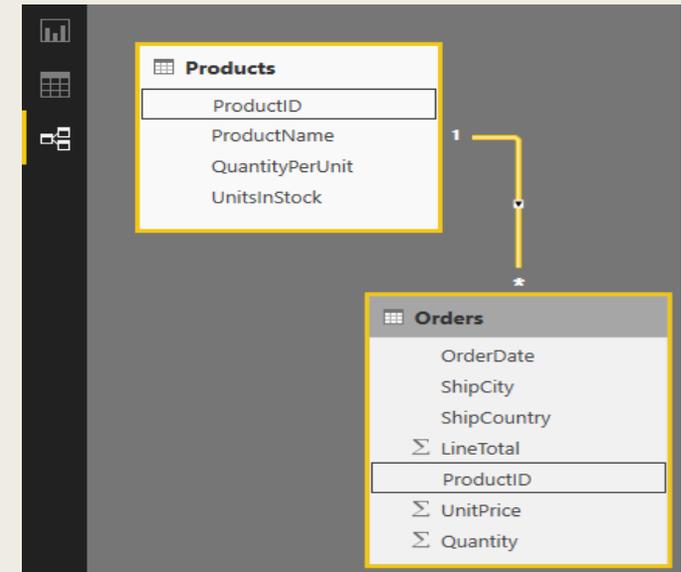
Make this relationship active

Assume referential integrity

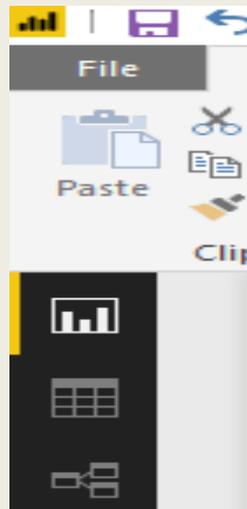
Apply security filter in both directions

OK Cancel

You can also view and manage the relationships between your queries by selecting the **Relationship** view at the left side of the Power BI Desktop window. Double-click the arrow on the line connecting the two queries to open the **Edit Relationship** dialog and view or change the relationship.



To get back to Report View from Relationships View, select the **Report View** icon.



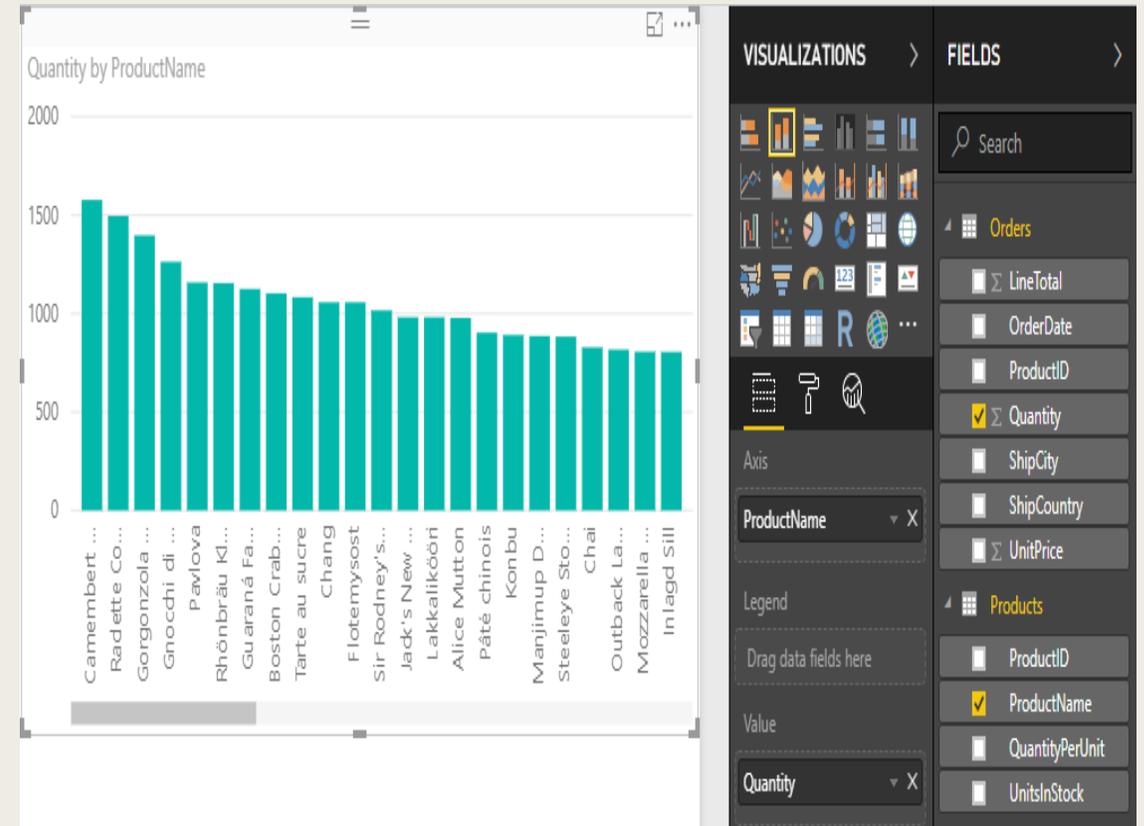
Create visualizations using your data

In Power BI Desktop Report View, you can create a variety of visualizations to gain insights from your data. You can build reports with multiple pages, and each page can have multiple visuals. You and others can interact with your visualizations to help analyze and understand your data. For more information about viewing and editing reports in Power BI Service (your site), see [Edit a Report](#).

You can use both of your data sets, and the relationship between them, to help visualize and analyze your sales data.

First, create a stacked column chart that uses fields from both queries to show the quantity of each product ordered.

1. Select the **Quantity** field from **Orders** in the **Fields** pane at the right, or drag it onto a blank space on the canvas. This creates a stacked column chart showing the total quantity of all products ordered.
2. Select **ProductName** from **Products** in the **Fields** pane, or drag it onto the chart, to show the quantity of each product ordered.
3. To sort the products by most to least ordered, select the **More options** ellipsis (...) at the upper right of the visualization, and then select **Sort By Quantity**.
4. Use the handles at the corners of the chart to enlarge it so more product names are visible.



Next, create a chart showing order dollar amounts (LineTotal) over time (OrderDate).

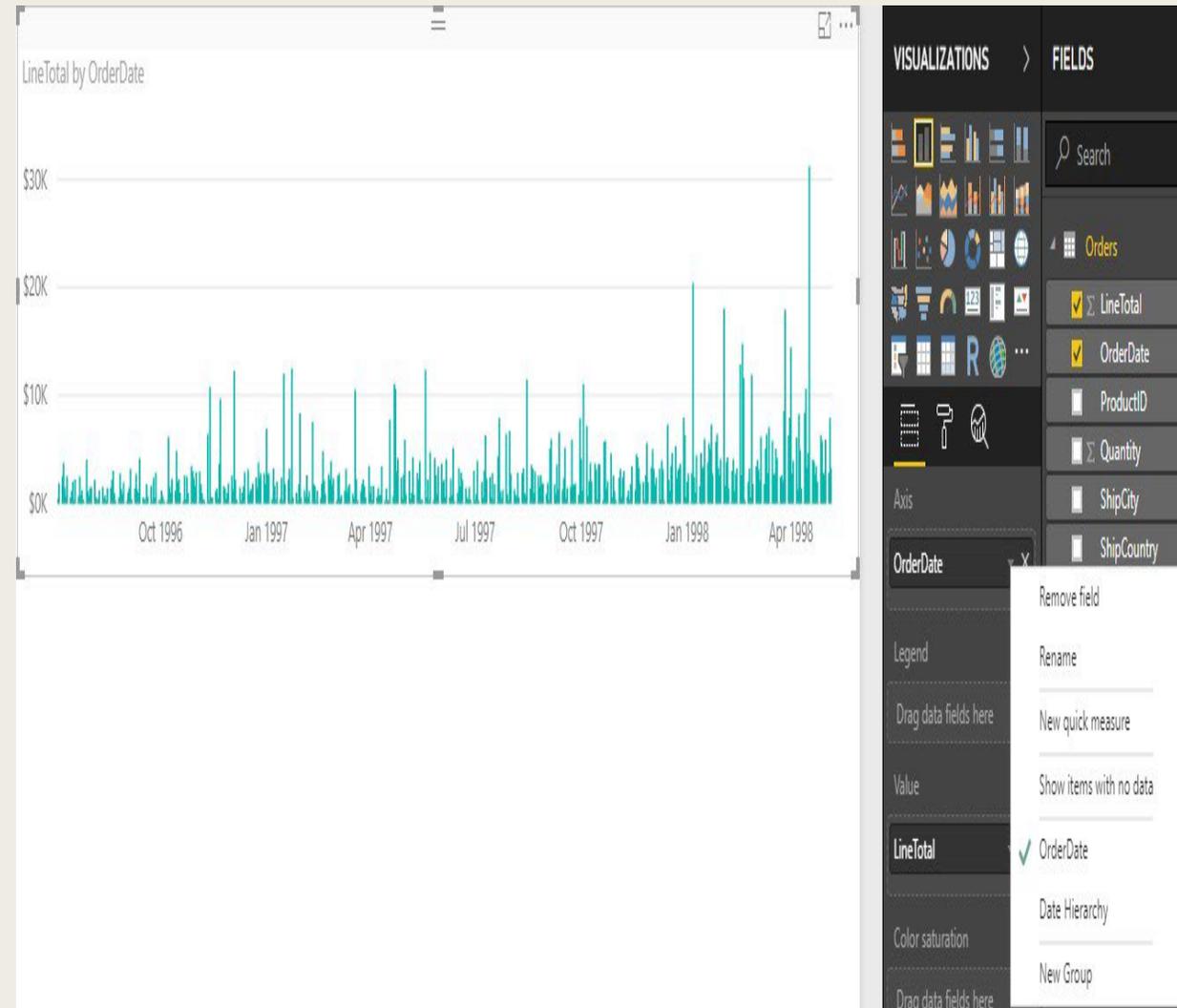
With nothing selected on the canvas, select LineTotal from Orders in the Fields pane, or drag it to a blank space on the canvas. The stacked column chart shows the total dollar amount of all orders.

With the chart selected, select OrderDate from Orders, or drag it onto the chart. The chart now shows line totals for each order date.

Resize the visualization by dragging the corners to be able to see more data.

Tip

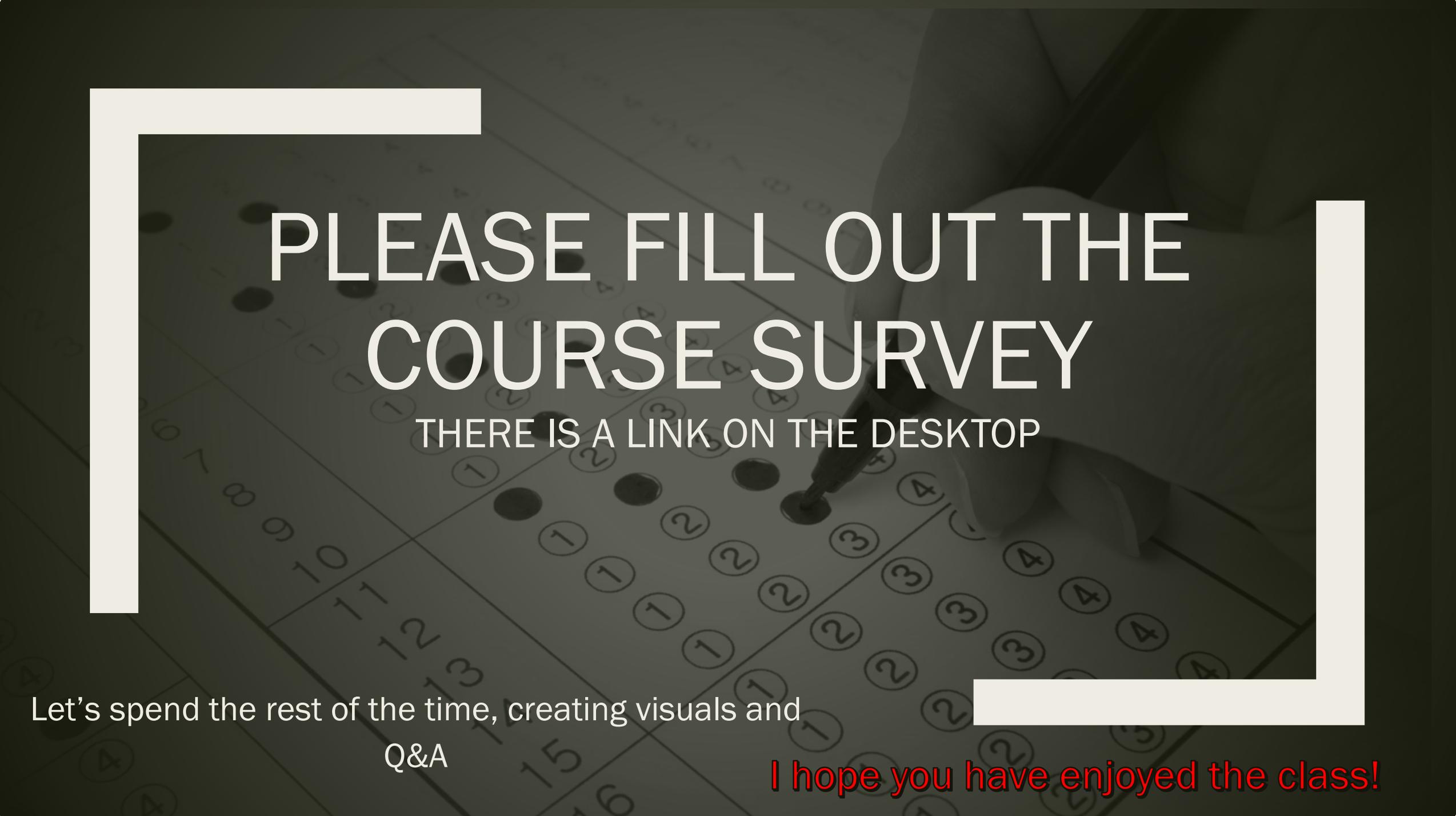
If you only see Years on the chart (only three data points), drop down the arrow next to **OrderDate** in the **Axis** field of the **Visualizations** pane, and select **OrderDate** instead of **Date Hierarchy**.



Finally, create a map visualization showing order amounts from each country.

With nothing selected on the canvas, select ShipCountry from Orders in the Fields pane, or drag it to a blank space on the canvas. Power BI Desktop detects that the data is country names, and automatically creates a map visualization, with a data point for each country that had orders. To make the sizes of the data points reflect the order amounts for each country, drag the LineTotal field onto the map (or drag it to Drag data fields here under Size, in the bottom half of the Visualizations pane). The sizes of the circles on the map now reflect the dollar amounts of the orders from each country.





PLEASE FILL OUT THE COURSE SURVEY

THERE IS A LINK ON THE DESKTOP

Let's spend the rest of the time, creating visuals and
Q&A

I hope you have enjoyed the class!