TechMentors

Helping you get the most out of technology

Microsoft Power BI Level 2

Student Manual

www.theTechMentors.com
info@theTechMentors.com
Revised August 2024

1. Review: Building Data Models and Basic Reports 1
   A. Get Data and Building a Report 2
   B. Explain the value and use of a Star Schema 5
2. Calculations, Measures and Hiearchies..... 9
   A. Insert and remove fields 10
   B. Add Measures 14
   D. Set up Hierarchies..... 20
   E. Create a Date Table 25
3. Visualizations 43
   A. Visualizations tell stories 44
   B. Which visualization should I use? 45
   C. Which rating is on top?..... 48
   D. Entering data to create custom sorts 52
   E. Using Small multiples 56
   F. Advanced Line Charts 58
   G. Combination Charts 62
   H. Matrix..... 65
   I. Gauge..... 68
   J. KPI 72
4. Additional Visualization Objects 79
   A. Inserting a text box..... 80
   B. Inserting a shape 85
   C. Inserting an image 86
   D. Adding bookmarks 87
   E. Inserting a button..... 90
5. Interactive Tools 94
   A. Spotlights..... 95
   B. Edit Interactions 96
   C. Filters and slicers affecting this visualization 98
   D. Focus Mode 99
   E. Show as a table 100
   F. Export Data 101
   G. Sorting 102
   H. Drill down 103

6. Creation Tools 107
A. Page Size 108
B. Page View 109
C. Themes 110
D. Aligning and grouping objects 112
E. Lock Objects 114
F. Selection pane 115
G. Mobile layout 117
7. DAX Calculations and Measures 118
A. What is DAX? 119
B. DAX Functions 121
C. DAX Operators 122
D. DAX Variables 123
E. Measures 124

1. Review: Building Data Models and Basic Reports

Complete this unit and you'll be able to:

A. Get Data and Building a Report

B. Explain the value and use of a Star Schema

C. Insert calculations and measures

D. Set up Hierarchies

E. Add a Date table to your data model

A. Get Data and Building a Report

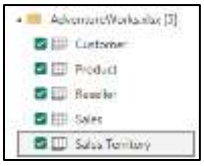
In the Power BI Level 1 course you learned how to use Microsoft Power BI Desktop to create Power BI reports with multiple visualizations. You also learned how to set up a data model, add calculated fields to that model, and more. In this course you will review many of these skills and begin to take them to the next level by creating the following report for the Adventure Works company.

Start a new Power BI and get your data

1. Start Power BI. From the Home screen click **Blank report**.

2. Import data from the **AdventureWorks.xlsx** Excel file.

3. Select all five tables: Customer, Product, Reseller, Sales and Sales Territory.



4. Click **Load**.

Analyze the data

1. Switch to the **Table** view.

2. Select the **Sales** table from the Data pane.

3. Notice that the status bar lists the name of the table and the number of rows (records) that are in the table.

4. Notice each column has a drop-down menu that lets you view, filter, sort and search the data in that column.

5. Using these tools analyze the data by answering the following questions:

Question		Your Answer
A.	Count the records in the Sales table.	
B.	List the distinct channels in the Sales table.	
C.	Using the filter drop down on the Sales Order field, can you quickly count the distinct Sales Orders in the Sales table?	

D.	Using the filter drop down on the Customer table, Customer Code column, which customers have a customer Code that is negative?	
E.	What is the name of the Customer with the Customer Code of -1?	
F.	The customer table has customers in how many countries?	
G.	What is the relationship between the Customer Code and the Customer ID?	
H.	Which fields can be used to link the Sales Territory Table and the Sales table?	
I.	In the Reseller table, are the Reseller Code and the Reseller ID related?	
J.	Based on the Product table, what industry would you say the Adventure Works company is in?	
K.	All of these tables seem to revolve around which central table?	

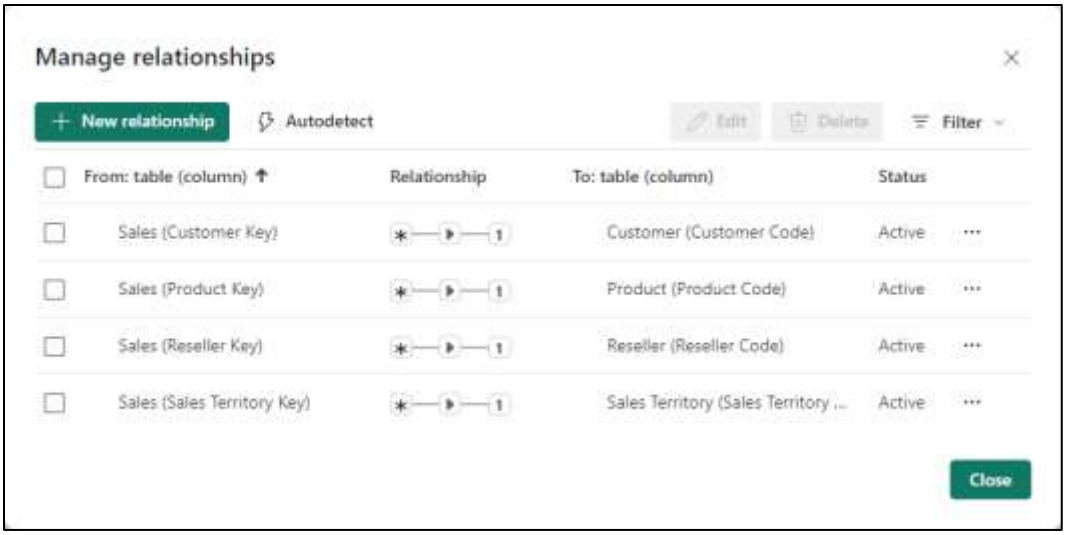
B. Explain the value and use of a Star Schema

In the previous exercise, you should have noticed that all the tables, one way or another, are related to the Sales table.

1. Switch to the Model view.

2. From the ribbon, choose **Home**, **Relationships**, **Manage relationships**.

3. Compare your screen with the one below. If needed, click the **New relationship** button and create any relationships that are missing.

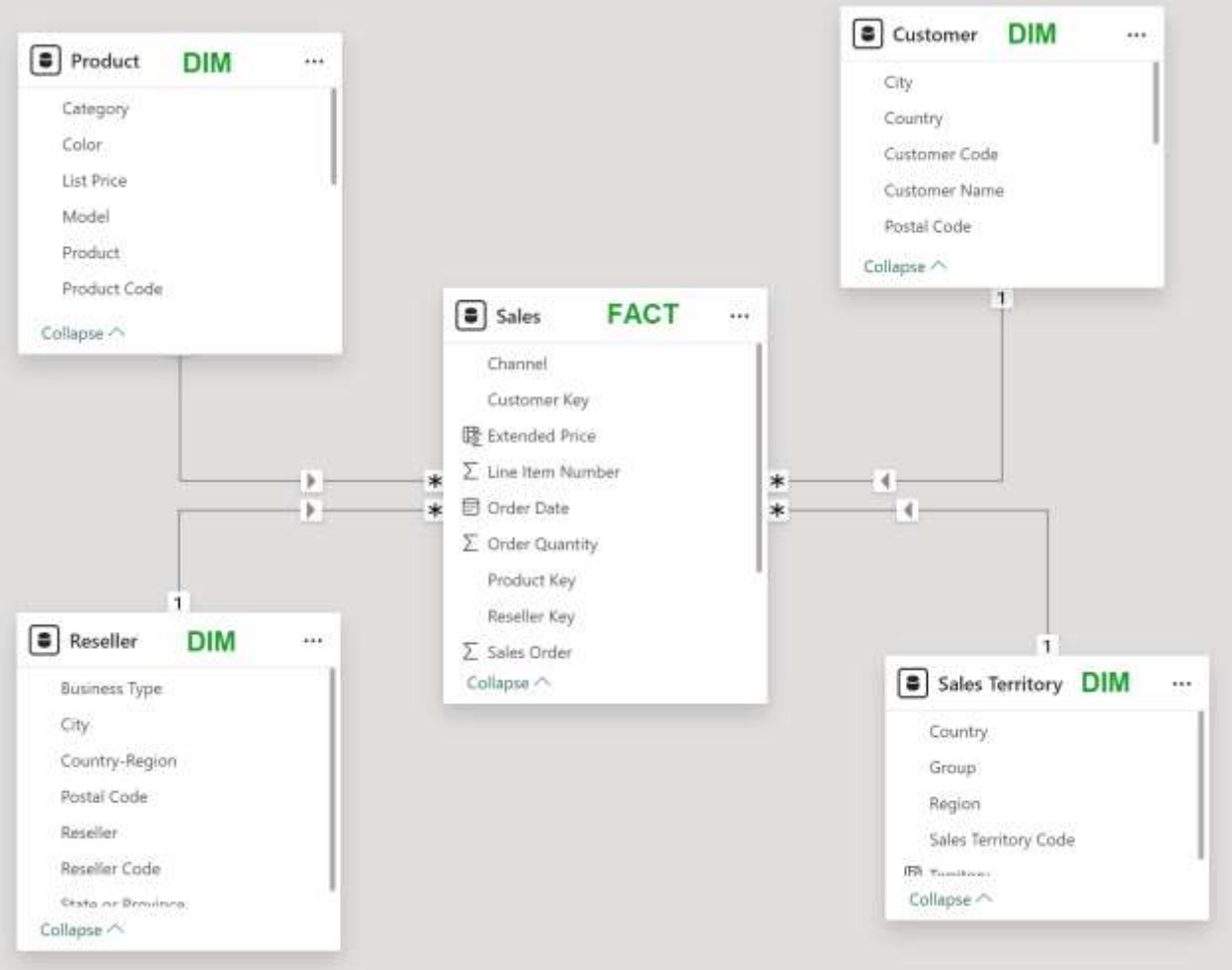


4. When finished, close the Manage relationships dialog box.

You might say they the Sales table is like a Sun or Star and the other tables are like planets that revolve around it.

What is star schema?

Star schema is a data modeling approach that is used by many relational data warehouses. This schema / plan places a "fact" table in the center of the data model, and then has various "dimension" tables surrounding it – thus making the data model look like a star with planets revolving around it.



What is a fact table?

Fact tables contain the transactions, events, enrollments, or sales. It is usually the table that contains the information you want to summarize with an aggregate (sum, average, maximum, minimum, etc.).

What is a dimension table?

Dimensions tables contain information about people, places, or things. For example, the customer table is a dimension table about people. The product table is a dimension table because it is about the things the organization sells, and the reseller table is another dimension table that contains information about the places / organizations that sell the wholesaler's products.

Each dimensions table contains one primary key field / column, and multiple attribute columns.

The **primary key** column is usually a number. In this dimension table, the primary key field is always unique; it is never duplicated in this table. The primary key field can

also never be blank. Many databases use an auto-number field as their primary key. For example, the primary key in an employee table will be the employee number field. When a new employee is hired, they are assigned the next available employee number.

The **attribute** columns share attributes about the person, place, or thing. For example, the customer table includes attribute columns for customer name, ID, address, city, state or province and postal code. Similarly, the product table includes attribute columns for product name, list price, category, color, etc.

In Power BI, the dimension tables are usually used for grouping and filtering data.

Why should I use a Star schema?

Setting up your data model with multiple tables in a Star schema will:

- Provide organization and usability:
  Having your fields in multiple categorical tables is like organizing your files with folders. In contrast, using only one large table is like putting all your files in the same folder. It becomes a big mess and makes it difficult to find things.

- Make formulas easier:
  Formulas become much more complex when all the fields are stuck in the same table. However, when the fields are organized in a logical Star schema the formulas are easier to write, read, and comprehend.

- Improve performance:
  The computer's performance in calculating formulas and apply filters is significantly improved when the data is organized into a logical Star schema.

- Yield faster refreshes:
  If you have tens or hundreds of millions of records, you will notice a report with a logical Star schema will refresh significantly fast than a similarly sized database that has all the data in one table.

Additional resources

If you would like to learn more about the benefits of a star schema, check out these resources:

- Understand star schema and the import for Power BI – Microsoft document https://docs.microsoft.com/en-us/power-bi/guidance/star-schema
- Why Power BI loves a Star schema – Guy in a Cube YouTube video https://www.youtube.com/watch?v=vZndrBBPiQc

2. Calculations, Measures and Hiearchies

A. Insert and remove fields

In this exercise we will continue creating our report and show the results of some aggregate calculations using Card visualizations. We will also review how to add a calculated field and remove some unnecessary fields.

Add Cards and a calculation to the report

1. Return to the **Report view**.

2. Change the background to light gray.

   - In the visualization pane click the **Format Page** icon.
   - Expand the **Canvas background** area.
   - Choose a **light gray** color.
   - Set the **transparency** to **0%**.

3. Add a card visualization to count total line items by doing the following:

   - Add a **Card** visualization .
   - Drag the Sales table's **Sales Order** field into the card.
   - Click the fields well's drop down and change the calculation to **count**.
   - Rename the field to **Total Line Items** by double clicking in the well and typing the new name.
   - Click the **Format Visual** icon, expand the **Callout value** area, and change the **display units** to **None**, and the **Value decimal places** to **0**.

4. Add another card visualization to show a distinct count of Sales Orders:

   - Add a **Card** visualization .
   - Drag the Sales table's **Sales Order** field into the card.
   - Click the fields well's drop down and change the calculation to **count (distinct)**.
   - Rename the field to **Total Orders** by double clicking in the well and typing the new name.
   - Click the **Format Visual** icon, expand the **Callout value** area, and change the **display units** to **None**, and the **Value decimal places** to **0**.

5. Create a calculated field for **Extended Price** by doing the following:

   - Right click on the **Sales** table and choose **New Column**.
   - Enter the following calculation:

```
Extended Price = Sales[Order Quantity] * Sales[Unit Price]
                 * (1 - Sales[Unit Price Discount Pct])
```

6. Add a third card visualization to show Total Sales:

   - Add a card visualization  .
   - Drag the Sales table's **Extended Price** field into the card.
   - Rename the field to **Total Sales** by double clicking in the well and typing the new name.
   - Click the **Format Visual** icon, expand the **Callout value** area, and change the **display units** to **None**, and the **Value decimal places** to **0**.

7. Standardize the height, and line up the three cards as follows:

   - Select all three cards.
   - In the **Visualization pane**, click the **Format Visual** icon.
   - Click on the **General** heading.
   - Expand the **Properties** section and set the **height** to **120**.
   - Expand the **Position** section and set the **vertical** to **75**.

8. Compare your screen with the screen shot below.

Add a calculated field for Territory name

In this exercise we want to create a new calculated field for the Sales Territory name that should include the country name and region if necessary. Many of the region names identify the country that they represent. However, those within the United States do not include the name of the country. As shown below, to overcome this we will create a calculated Territory name field that will include the country name and where needed region names also.

Sales Territory Code	Group	Country	Region	Territory
11	Corporate HQ	Corporate HQ	Corporate HQ	Corporate HQ
7	Europe	France	France	France
8	Europe	Germany	Germany	Germany
10	Europe	United Kingdom	United Kingdom	United Kingdom
6	North America	Canada	Canada	Canada
3	North America	United States	Central	United States - Central
2	North America	United States	Northeast	United States - Northeast
1	North America	United States	Northwest	United States - Northwest
5	North America	United States	Southeast	United States - Southeast
4	North America	United States	Southwest	United States - Southwest
9	Pacific	Australia	Australia	Australia

1. Switch to **Table view** and select the **Sales Territory** table.

2. Right click on the **Sales Territory** table, choose **New Column**, and enter the following calculation.

```
Territory = if([Country] = [Region], [Country], [Country] & " - " & [Region])
```

This calculation uses the IF function, which has three parts. First the function evaluates an expression, `[Country] = [Region]`, that will yield TRUE or FALSE. After the first comma the function shows the results that will occur if the answer is TRUE, `[Country]`. After the second comma the function shows the results that will occur if the answer is FALSE, `[Country] & " - " & [Region]`.

Therefore, if the Country and Region are equal, then only return the Country field. If not equal, include the Country, a dash, and the Region.

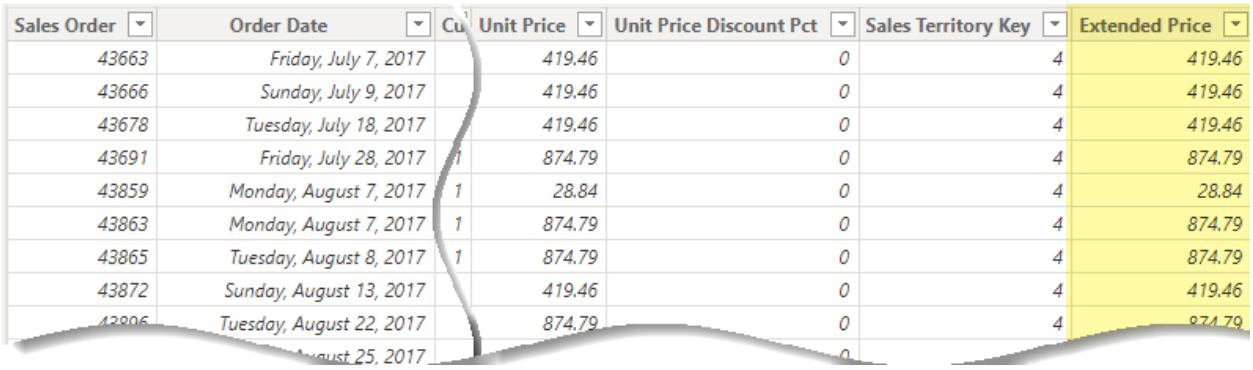
Remove unneeded fields

1. In the **Data pane** right click on the **Customer** table's **Customer ID** field, then choose **Delete from model**. When asked "Are you sure…" click **Yes**.

2. Repeat the process to **remove** the **Reseller**'s **Reseller ID** field.

B. Add Measures

In the previous exercises you added and removed calculations to and from your tables. As shown below, adding a calculation, such as Extended Price, creates a calculation for each row in the table.



Sales Order	Order Date	Cu	Unit Price	Unit Price Discount Pct	Sales Territory Key	Extended Price
43663	Friday, July 7, 2017		419.46	0	4	419.46
43666	Sunday, July 9, 2017		419.46	0	4	419.46
43678	Tuesday, July 18, 2017		419.46	0	4	419.46
43691	Friday, July 28, 2017	1	874.79	0	4	874.79
43859	Monday, August 7, 2017	1	28.84	0	4	28.84
43863	Monday, August 7, 2017	1	874.79	0	4	874.79
43865	Tuesday, August 8, 2017	1	874.79	0	4	874.79
43872	Sunday, August 13, 2017		419.46	0	4	419.46
43896	Tuesday, August 22, 2017		874.79	0	4	874.79
	August 25, 2017			0		

Measures on the other hand are added once per table, not once per row.

In the following exercises, you will create three measures. You will also learn how to create a new special type of table to hold all your measures.

Add a Total Line Items measure

Recall that in the report, the Total Line Items Card was created by placing the Sales table's [Sales Order] field in the field well, and then changing the calculation from SUM to COUNT. This is fine, except it cannot be reused in other visuals. We therefore will create a measure for the Total Line Items and use that measure in the card and in future exercises.

1. In the data pane, right click the **Sales table**, and choose **New measure**.

2. Enter the following formula:

```
Total Line Items = count(Sales[Sales Order])
```

3. Notice in the **table view** that the new measure appears in the **Sales table**, but it does not appear with the other columns. This is because **measures are not calculated per row but are instead calculated per the context in which they are used**.

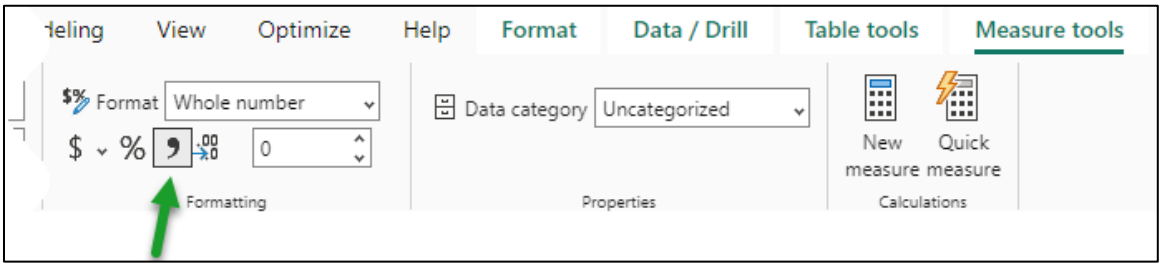
4. Return to the **Report view**.

5. Select the **Total Line Items Card**.

6. In the fields well, click the X to remove the existing "Total Line Items" calculation.

7. Drag the Sales table's **Total Line Items** measure into the card's well.

8. If needed, add commas by selecting the Sales table's **Total Line Items** measure and then on the ribbon's **Measure tools** click the **comma** button.



Add a measure to count Total Orders

In this exercise you will create a measure to count the total number of orders using the DistinctCount function.

1. In the data pane, right click the **Sales table**, and choose **New measure**.

2. Enter the following formula:

```
Total Orders = DISTINCTCOUNT(Sales[Sales Order])
```

3. On the ribbon's **Measure tools** tab, click the **comma** button, and set the **decimal places** to **0**.

4. In the **Total Orders card**, remove the existing field from the field well, and drag in the **Total Orders** measure.

Add a measure to sum Total Sales

In this exercise you will create a measure to sum the extended price to get total sales.

1. In the data pane, right click the **Sales table**, and choose **New measure**.

2. Enter the following formula:

```
Total Sales = sum(Sales[Extended Price])
```

3. On the ribbon's **Measure tools** tab, click the **$** button, and set the **decimal places** to **0**.

4. In the **Total Sales card**, remove the existing field from the field well, and drag in the **Total Sales** measure.

Add three measures for date related calculations

In this exercise you will create a measure to calculate the first sales date, the last sales date, and a measure named report time frame that lets the user know the report's time frame by concatenating the first date with the last date.

1. In the data pane, right click the **Sales table**, and choose **New measure**.

2. Enter the following formula for the first sales date.

```
First Date = min(Sales[Order Date])
```

   This measure uses the **Min** function to calculate the first Sales [Order Date].

3. In the data pane, right click the **Sales table**, and choose **New measure**.

4. Enter the following formula for the last sales date.

```
Last Date = max(Sales[Order Date])
```

   This measure calculates the **Max** function to calculate the last Sales [Order Date].

5. In the data pane, right click the **Sales table**, and choose **New measure**.

6. Enter the following formula for the report time frame.

```
Report Time Frame =
    format([First Date], "mm/dd/yyyy")
    & " - "
    & format([Last Date], "mm/dd/yyyy")
```

   This measure uses the **Format** function twice to specify that the two dates should be formatted with two-digit months, two-digit days, and four-digit years. It also includes a space and dash separator by putting the space and dash and another space between the quotes, and it also uses two ampersand characters, **&**, to concatenate or put together all three parts.

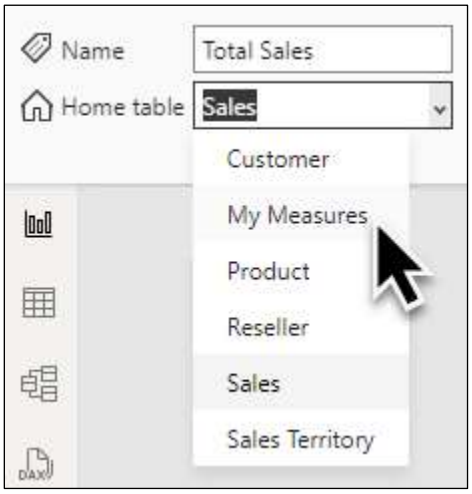
Congratulations, you have created five measures that can be used in multiple visuals that we will create in future exercises.

Create a My Measures table

When your report is finally finished you may discover that you have many measures. To help navigate the data model, many developers prefer to create a "measures" table. As its name implies, a measures table only holds measures.

Unfortunately, Power BI will not let us name our measures table "**Measures**". So, instead we will name it "**My Measures**" and then will move our measures into it.

1. On the ribbon's **Home tab**, click the **Enter data** button.
2. Change the Name to "**My Measures**". Notice we did not enter any data.
3. Click **Load**. Notice the My Measures table is listed in the Data pane underneath the Customer table.
4. **Expand** the **My Measures** table. Notice it has only one column.
5. In the **Sales table**, select the **Total Sales** field.
6. On the ribbon's **Measure tools** tab, change the **Home table** to **My Measures**.



7. Repeat the process to move the **Total Orders**, **Total Line Items**, **First Date**, **Last Date**, and **Report Time Frame** measures to the **My Measures** table.
8. In the **My Measures table**, right click **Column 1**, and choose **Delete from model**. When asked if you are sure, click **Yes**.

Notice because the My Measures table now only has measures, it is moved to the top of the Data pane.

C. Using measures with visualizations

To see how measures can affect your visualizations, in this exercise you will add a Stacked bar chart and some Slicers to your report. You will also be introduced to the concept of a date hierarchy and will see how it can be used to create a multi-level slicer.

1. Add a Stacked Bar Chart visualization to your report.

   - Add the **Sales Territory [Territory]** field to the **Y-Axis**.
   - Add the **Total Sales** measure to the **X-Axis**.
   - Add the **Sales [Channel]** field to the **Legend**.
   - Turn **off** the **Y-Axis title**.
   - Turn **off** the **X-Axis title**.
   - Add **Data labels**.
   - Change the **Data label**'s **Value**, **Display Units** to **Thousands** with **0** decimal places.
   - Set the visual's **Title**'s **font size** to **24**.
   - On the ribbon's **View** tab, change the **Theme** to **Accessible City Park**.

2. Add a slicer for the **Product [Category]** field.

3. Add a slicer for the **Product [Color]** field.

4. Add a card visualization for Report Time Frame measure.

   - Set the Card's Callout value to 28 pt.
   - Turn off the Card's Category label.
   - Turn off the Card's background by setting its transparency to 100%.
     (Hint navigate to the card's Format, General, Effects, Background area).

5. Recognize and use a date hierarchy:

   - In the Sales table, click the **Order Date** field's expand button.
     Notice it reveals a Date Hierarchy.
   - Click the **Date Hierarchy** expand button.
     Notice the date hierarchy is composed of four levels: year, quarter, month and day.
   - Create a new slicer.
   - Experiment with setting the slicer to the following and observe the results you get.
     - Order Date
     - Order Date, Date Hierarchy, Year
     - Order Date, Date Hierarchy, Quarter
     - Order Date, Date Hierarchy, Month
     - Order Date, Date Hierarchy

Notice when using just the Date Hierarchy the slicer gives you the option to drill down from year to quarter to month to day.

6. To demonstrate that our existing measures facilitate time-based analysis, using the various slicers, choose the values shown below, and see if you get the same values shown in gray.

Category	Color	Time Frame	Total Line Items	Total Sales Orders	Total Sales
All	All	2018	19,409	3,748	$ 30,701,248
All	All	2019 Q1	4,713	1,154	$6,772,105
Bikes	Black	2019 Q1	1,301	598	$ 2,983,688
Bikes	Black	2019 Q2	1,461	729	$ 3,526,470
Components	All	2018 Q1 January	40	11	$ 23,698

D. Set up Hierarchies

What is a hierarchy?

A hierarchy is an organizational construct that uses a top-down style. For example, as you saw with the Order Date hierarchy you had the Year at the top level, Quarter at level 2, Month at level 3, and Day at level 4.

As you will see in the next couple of exercises, you can create a hierarchy by following these steps:

1. Determine which table will have the hierarchy.
2. Determine which fields will be in the hierarchy, and their order.
3. Right click on the field that will be at the top level of the hierarchy and choose Create hierarchy.
4. Rename the newly created hierarchy.
5. Right click on the next field that will be in the hierarchy, and choose Add to hierarchy, and then choose the desired hierarchy.
6. Repeat step 5 for the other hierarchy fields.

Create a Location Hierarchy

In this exercise you will create a geographical hierarchy for the customer's Country, State Province, City and finally Postal code.

1. In the field list, expand the **Customer** table.

2. **Right click** on the **Country** field and choose **Create hierarchy**.

3. **Right click** on the new hierarchy and rename it **Location**.

4. Expand the Location hierarchy.
   Notice the Country field is in the hierarchy.

5. Add **State Province**, **City**, and **Postal Code** to the hierarchy by **right clicking** on the field, selecting **Add to hierarchy**, and choosing the **Location** hierarchy.

Practice Exercise

1. Create a hierarchy in the Product table with levels for the following fields: **Category**, **Subcategory**, **Model**, **Product**. Name the hierarchy **Products**.

2. Create a new **Sales by Product** page and on it create a **matrix** visualization that includes the **Products** hierarchy as the rows and the **Total Orders**, **Total Line Items**, and **Total Sales** measures as values.

   Experiment with the matrix by drilling down on the various levels of the hierarchy.

Category	Total Orders	Total Line Items	Total Sales
⊞ **Accessories**	**19,523**	**41,193**	**$1,278,714**
⊟ **Bikes**	**18,358**	**40,005**	**$95,115,011**
⊞ **Mountain Bikes**	**6,185**	**12,457**	**$36,622,231**
⊟ **Road Bikes**	**9,528**	**20,918**	**$43,947,709**
⊞ **Road-150**	**1,779**	**2,090**	**$7,913,700**
⊞ **Road-250**	**2,466**	**4,150**	**$13,832,755**
⊟ **Road-350-W**	**1,327**	**1,926**	**$5,267,084**
Road-350-W Yellow, 40	573	573	$1,663,512
Road-350-W Yellow, 42	455	455	$1,121,239
Road-350-W Yellow, 44	332	332	$694,003
Road-350-W Yellow, 48	566	566	$1,788,331
⊞ **Road-450**	**336**	**905**	**$1,874,821**
⊞ **Road-550-W**	**1,971**	**3,278**	**$5,184,890**
⊞ **Road-650**	**1,726**	**6,185**	**$8,127,182**
⊞ **Road-750**	**1,845**	**2,384**	**$1,747,277**
⊞ **Touring Bikes**	**2,645**	**6,630**	**$14,545,071**
⊞ **Clothing**	**9,871**	**21,368**	**$2,138,425**
⊞ **Components**	**2,646**	**18,687**	**$11,804,248**
Total	**31,455**	**121,253**	**$110,336,398**

3. Create another new page. Name it **Sales by Location**. On it create a **matrix** visual that includes the **Locations** hierarchy for the rows, and the **Total Orders**, **Total Line Items**, and **Total Sales** measures as the values.

   Experiment with the Locations matrix by drilling down on the various levels of the hierarchy.

Country	Total Orders	Total Line Items	Total Sales
⊞ **[Not Applicable]**	**3,796**	**60,855**	**$80,977,720**
⊞ **Australia**	**6,718**	**13,345**	**$9,061,001**
⊞ **Canada**	**3,375**	**7,620**	**$1,977,845**
⊟ **France**	**2,484**	**5,558**	**$2,644,018**
⊞ **Charente-Maritime**	**31**	**75**	**$34,442**
⊟ **Essonne**	**225**	**505**	**$279,297**
⊞ Les Ulis	140	323	$181,245
⊞ Morangis	42	85	$56,433
⊞ Verrieres Le Buisson	43	97	$41,620
⊞ **Garonne (Haute)**	**42**	**106**	**$54,642**
⊞ **Hauts de Seine**	**257**	**549**	**$263,416**
⊞ **Loir et Cher**	**23**	**60**	**$21,474**
⊞ **Loiret**	**83**	**193**	**$91,563**
⊞ **Moselle**	**82**	**196**	**$94,046**
⊞ **Nord**	**380**	**844**	**$391,400**
⊞ **Pas de Calais**	**18**	**45**	**$11,343**
⊞ **Seine (Paris)**	**522**	**1,174**	**$539,726**
⊞ **Seine et Marne**	**90**	**198**	**$109,735**
⊞ **Seine Saint Denis**	**379**	**847**	**$379,480**
⊞ **Somme**	**30**	**68**	**$29,555**
⊞ **Val de Marne**	**35**	**80**	**$28,478**
⊞ **Val d'Oise**	**63**	**134**	**$46,756**
⊞ **Yveline**	**224**	**484**	**$268,665**
⊞ **Germany**	**2,484**	**5,625**	**$2,894,312**
⊞ **United Kingdom**	**3,031**	**6,906**	**$3,391,712**
⊞ **United States**	**9,567**	**21,344**	**$9,389,790**
Total	**31,455**	**121,253**	**$110,336,398**

4. Create another new page for **Sales by Date** and on it create a **matrix** visual that includes the **Order Date field**, and the **Total Orders**, **Total Line Items**, and **Total Sales** measures.

   Experiment with the Dates matrix by drilling down on the various levels of the hierarchy.

Year	Total Orders	Total Line Items	Total Sales
⊞ **2017**	**1,567**	**5,339**	**$11,932,876**
⊟ **2018**	**3,748**	**19,409**	**$30,701,248**
⊞ **Qtr 1**	**772**	**2,485**	**$5,965,129**
⊞ **Qtr 2**	**859**	**3,094**	**$6,103,487**
⊞ **Qtr 3**	**1,030**	**7,580**	**$10,233,978**
⊞ **Qtr 4**	**1,087**	**6,250**	**$8,398,654**
⊞ **2019**	**14,046**	**55,542**	**$43,192,446**
⊟ **2020**	**12,094**	**40,963**	**$24,509,828**
⊞ **Qtr 1**	**6,358**	**20,266**	**$11,741,888**
⊟ **Qtr 2**	**5,736**	**20,697**	**$12,767,939**
⊞ **April**	**2,206**	**7,092**	**$4,009,980**
⊟ **May**	**2,411**	**8,471**	**$5,279,803**
1	53	140	$55,377
2	67	243	$125,571
3	72	265	$108,873
4	60	164	$63,087
5	92	359	$240,943
6	92	311	$191,753
7	88	352	$214,942
8	81	267	$170,767
9	77	309	$245,016
10	69	397	$301,116
11	89	285	$168,662
12	68	266	$186,555
13	66	220	$119,154
14	106	418	$213,664
15	95	410	$298,213
16	72	173	$91,798
17	90	236	$78,685
18	77	183	$66,211
19	89	292	$190,358
Total	**31,455**	**121,253**	**$110,336,398**

Expand the 2017 node and notice that the first date in the matrix is July 1st because the company made their first sale on July 1st, 2017.

Year	Total Orders	Total Line Items	Total Sales
2017	**1,567**	**5,339**	**$11,932,876**
Qtr 3	**822**	**2,838**	**$6,007,418**
July	**327**	**641**	**$1,423,357**
1	9	9	$28,409
2	8	19	$42,734
3	9	9	$29,325
4	6	6	$15,711
	11		$65,487

In the next section we will discuss how you can base your reports on a Fiscal Year instead of a Calendar Year by using a date table.

E. Create a Date Table

What is a date table?

A date table is a table that contains only dates and their related values.

Date	Year	YMD	Month Year	Qtr	Month Number	Month Name	Month Code	Week Number	Day	Weekday Number	Weekday Name	Weekday Code
12/31/2012 12:00:00 AM	2012	2012 12 31	Dec 12	Qtr 4	12	December	Dec	53	31	2	Monday	Mon
1/1/2013 12:00:00 AM	2013	2013 01 01	Jan 13	Qtr 1	1	January	Jan	1	1	3	Tuesday	Tue
1/2/2013 12:00:00 AM	2013	2013 01 02	Jan 13	Qtr 1	1	January	Jan	1	2	4	Wednesday	Wed
1/3/2013 12:00:00 AM	2013	2013 01 03	Jan 13	Qtr 1	1	January	Jan	1	3	5	Thursday	Thu
1/4/2013 12:00:00 AM	2013	2013 01 04	Jan 13	Qtr 1	1	January	Jan	1	4	6	Friday	Fri
1/5/2013 12:00:00 AM	2013	2013 01 05	Jan 13	Qtr 1	1	January	Jan	1	5	7	Saturday	Sat
	2013	2013 01 06	Jan 13		1	January		2	6			Sun

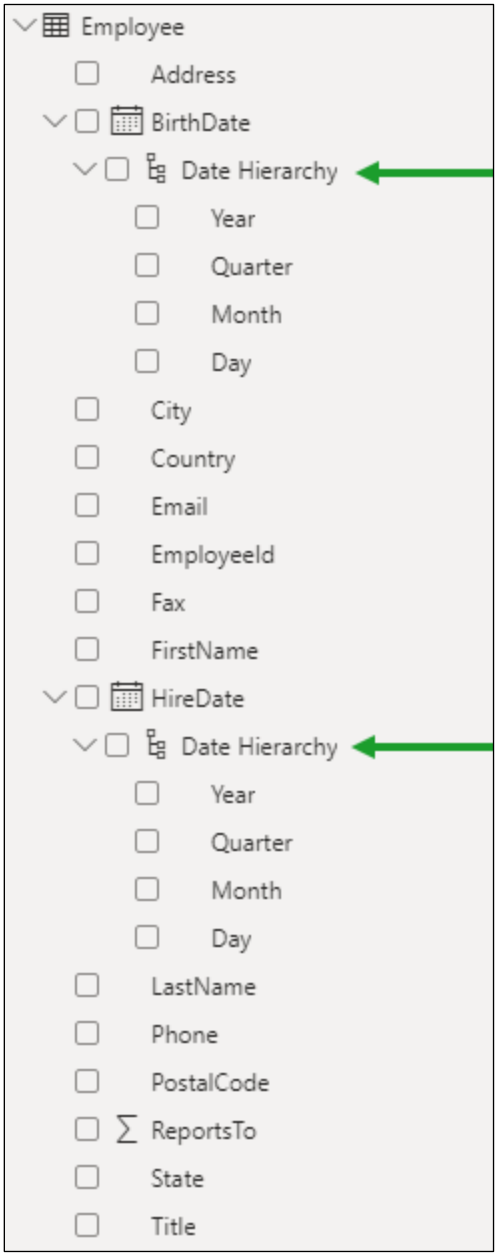
The date table should follow these rules:

- The table must have a date field.
- The date of the first record in the date table must be on or before the first date of any and all date fields in all the other tables in your data model.
- The date of the last record in the date table must be on or after the last date of any and all date fields in all the other tables in your data model.
- The date table must have a record for every calendar date that falls between the first and last date. No missing dates are allowed.
- The date table may have other fields that are related to the date field. For example, you can have calculated columns like:
  - The day of the week number (1 for Sunday, 2 for Monday, etc.)
  - The name of the day of the week
  - The day of the year
  - The week number relative to the start of the year
  - The month number (1 for January, 2 for February, etc.)
  - The name of the month
  - The month number relative to the when the fiscal year began
  - The quarter number (1, 2, 3, or 4)
  - The quarter number relative to when the fiscal year began
  - The year number

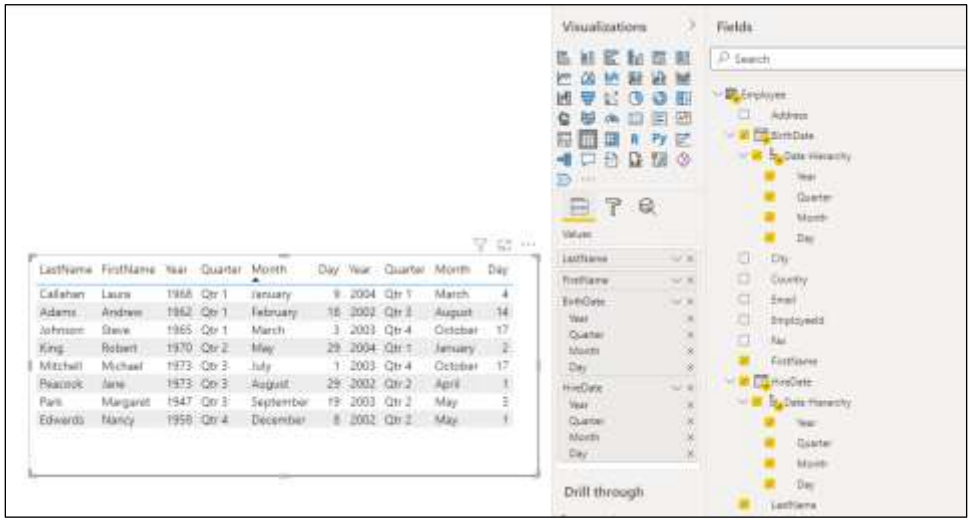
Why do I need a date table?

For every date field in your model Power BI creates a date table, in the background, to support date hierarchies. So, if you have two date fields in your model, Power BI creates two background date tables, and if you have twenty date fields, Power BI creates twenty background date fields.

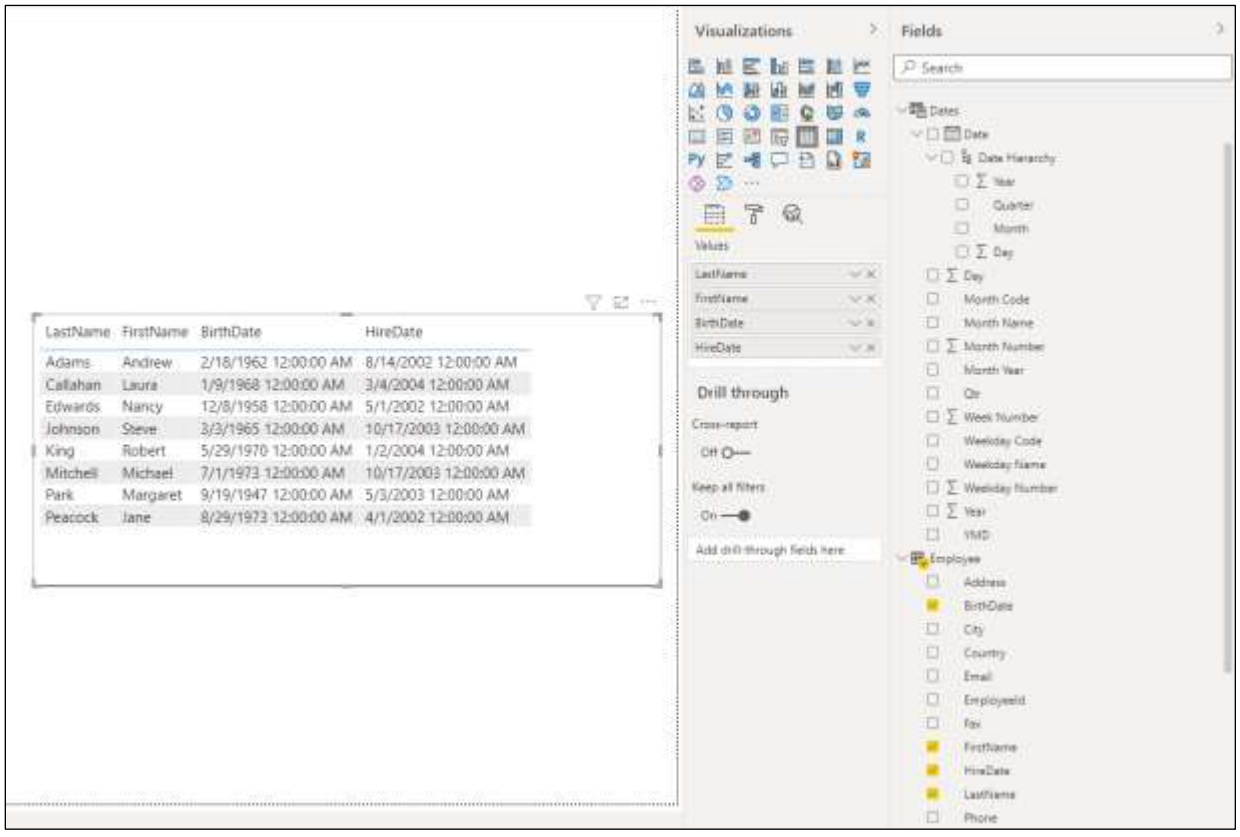
You can see the background date table in the field pane when you click the expand arrow next to a date. For example, in the screen shot below notice both the Employee BirthDate and the HireDate fields each have Date Hierarchy fields.



You can see the effect of this background date table when you add a date field to a visualization. For example, in the image below, notice that when the user added the BirthDate and HireDate fields to the table, the system automatically added the whole date hierarchy (Year, Quarter, Month and Day).

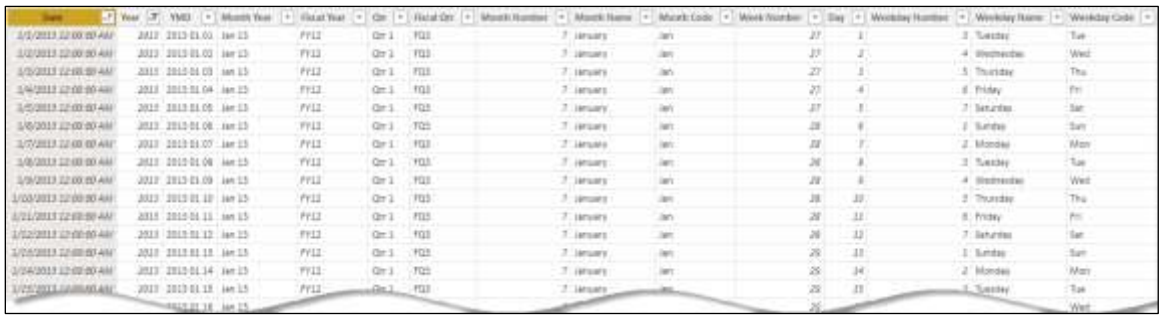


But, if you create a date table and link it to every date field in your model, then Power BI does not create the hidden date tables, as shown below, thus saving your file from a lot of date bloat.

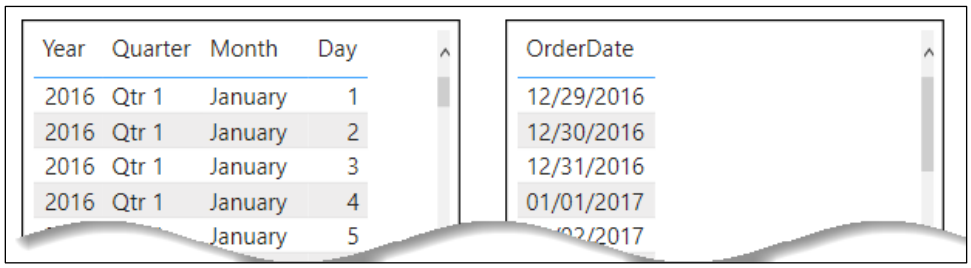


Fiscal Years

As stated earlier, another reason for creating a date table is to handle situations where the fiscal year does not begin on January 1st. Such a table will allow you to easily calculate when the fiscal year begins and ends, as well as which months / quarters are first, second, third, etc. relative to the fiscal year.



Exercise



Year	Quarter	Month	Day
2016	Qtr 1	January	1
2016	Qtr 1	January	2
2016	Qtr 1	January	3
2016	Qtr 1	January	4
		January	5

OrderDate
12/29/2016
12/30/2016
12/31/2016
01/01/2017
/02/2017

1. If necessary, open the AdventureWorks.pbix file.
2. Add a new page with a **Card** visualization.
3. Drag the My Measures **First Date** into the card.

   Notice the first date was on July 1, 2017.
4. Drag the My Measures **Last Date** into the card.

   Notice the last date was on June 15, 2020.
5. Delete the **Card** visualization.
6. Add a table visualization.
7. Place the **Sales** table's **OrderDate** field in the columns well.

Notice the following:

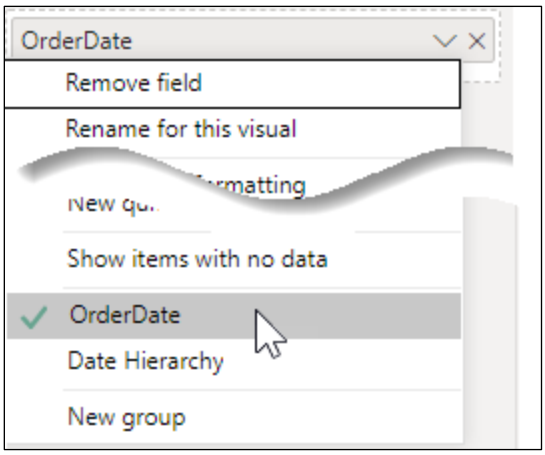
- The table displays four columns for **Year**, **Quarter**, **Month**, and **Day**.
- The first date is January 1, 2017 and the last date is December 31, 2020.

  This demonstrates that **there is a hidden date table that has records for four full years of data** (1/1/2017 through 12/31/2020) even though our order dates only cover about a three year time frame (7/1/2017 through 6/15/2020).
- The 1/1/2017 date is in Qtr 1, and the 12/31/2020 is in Qtr 4.

  The quarters are appropriate for a calendar year. But we want to change this report to be based on a fiscal year that begins on July 1st. Consequently, when we are finished the January dates should be in the third quarter, and the December dates should be in the second quarter. In a future exercise we will fix this by using a date table.

8. Duplicate the table visual and move the copy to the right.

9. In the values well, expand the **OrderDate** and choose **OrderDate** instead of Date Hierarchy.

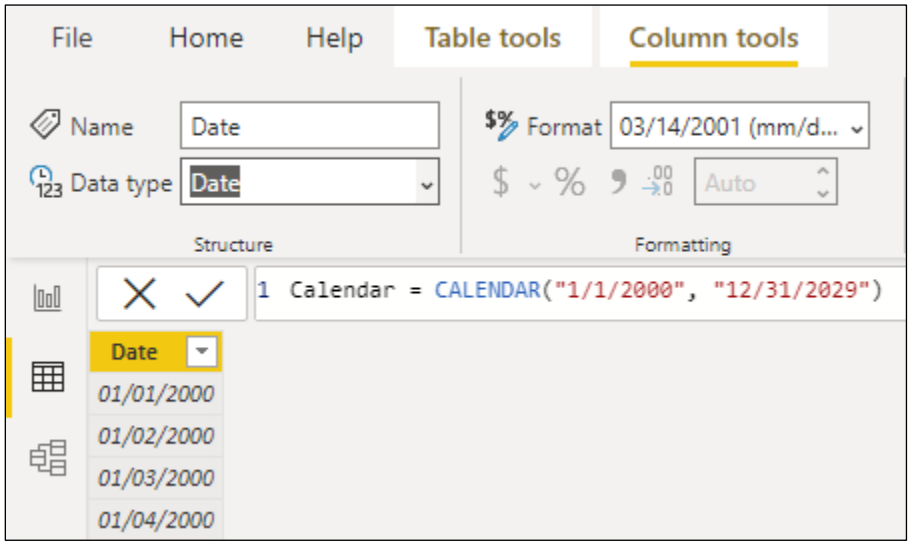


Now notice the following:

- Only one column, OrderDate, is displayed.
- The first date is now Saturday July 1, 2017 and the last date is Monday June 15, 2020.

Create a date table with the Calendar() function

You can create a data table in multiple ways. In the following exercises you will create multiple date tables, compare the results, and then finally choose one to keep and delete the others.



1. If necessary, open the **AdventureWorks.pbix** report.

2. Switch to the table view.

3. From the ribbon select **Home**, **Calculations**, **New table**.

4. Enter the following formula:

   Calendar = CALENDAR("1/1/2017", "12/31/2020")

   The CALENDAR() function creates a date table with one date field that starts with the first entered date and ends with the last entered date.

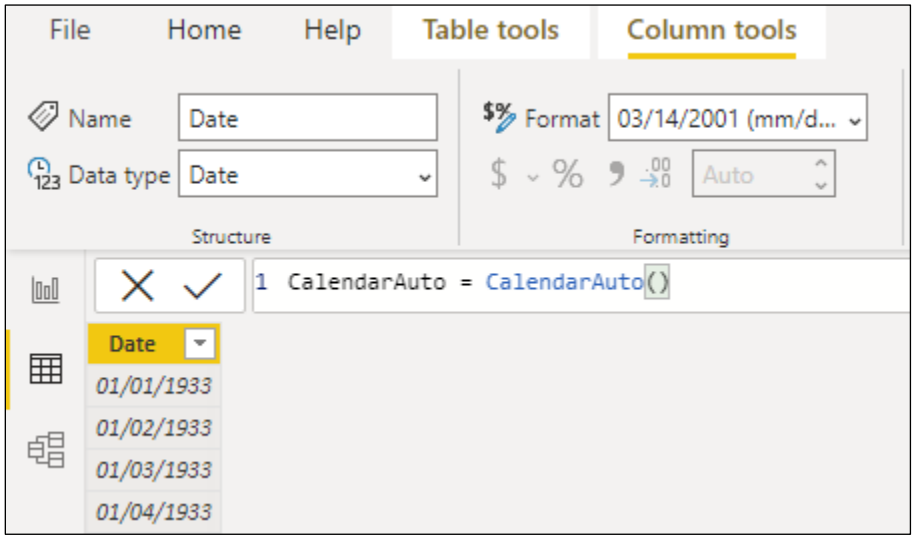
5. In the fields pane, **expand** the **Calendar table** and select the **Date** field.

6. Change the Date field's data type from Date/Time to Date by selecting from the ribbon **Column tools**, **Structure**, **Data type** and choosing **Date**.

7. From the ribbon select **Column tools**, **Formatting**, **Format** and choose the **mm/dd/yyyy** format.

Create a date table with the CalendarAuto() function

The CALENDARAUTO() function creates a calendar, but instead of asking you for the start and end date, it examines the dates in your data model and selects the first date found and ends with the last date found.



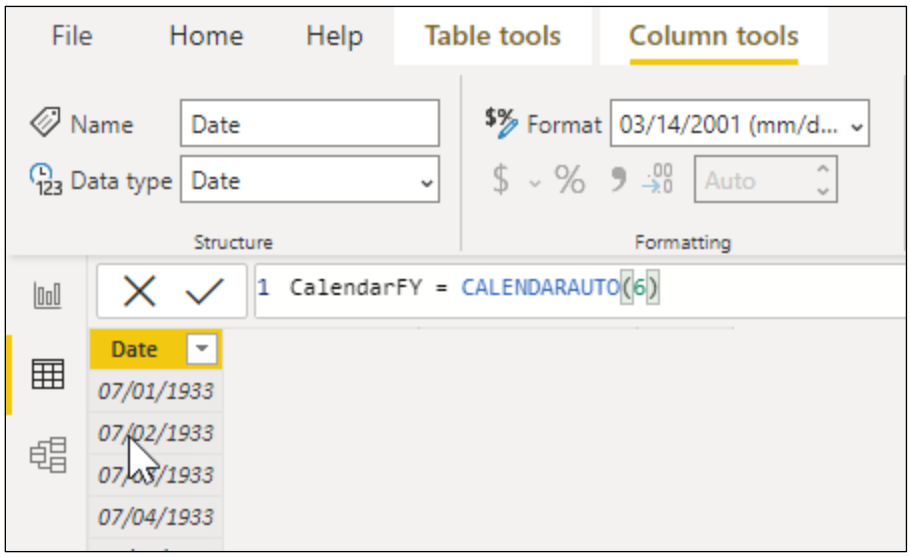
1. Switch to the data view.
2. From the ribbon select **Home**, **Calculations**, **New table**.
3. Enter the following formula:

   CalendarAuto = CalendarAuto()

4. Change the data type to Date and the format to **mm/dd/yyyy**.

Create a fiscal year date table

The CalendarAuto() function has an optional parameter that allows you to enter the end month of a fiscal year. When used, the function starts the calendar on the first date of the fiscal year based on the first date found in your model. It also ends with the last date of the final fiscal year's last month.



1. Switch to the table view if necessary.

2. From the ribbon select **Home**, **Calculations**, **New table**.

3. Enter the following formula:

   CalendarFY = CalendarAuto(6)

4. Change the data type to Date and the format to **mm/dd/yyyy**.

Mark the date table

After creating your table, you need to mark it as the "date table" for this data model. In this exercise you will mark the **CalendarFY** as your date table and then delete the other two calendar tables.

1. **Right click** on the **CalendarFY** table and choose **Mark as date table**, then turn on the **Mark as a date table** option.

2. In the Mark as date table dialog box, click the **Date Column** drop down and choose the **Date** field.

3. Click **OK**.

4. Right click on the **Calendar** table and choose **Delete from model**, and then click **OK**.

5. Repeat the process to delete the **CalendarAuto** table.

Add new date-oriented columns

Your date table does not need to be limited to one field. You may, and should, add additional date-oriented columns that you might use in your visualizations. In this exercise you will add many date-oriented columns to the date table.

1. Select the CalendarFY table and then add a new column in the CalendarFY table for each of calculations listed below:

Column = Calculation	Sample Results
Calendar Year = Year([Date]) This formula returns the year of the selected date.	2017
FY = "FY" & Right(If(Month([Date])<7, Year([DATE]), Year([Date])+1) ,2) This formula returns "FY" followed by a two-digit year. To calculate the year number this function uses the Year, Month, and Right functions. The **Year()** function returns the four-digit calendar year of the selected date. The **Month()** function returns the month number of the selected date. The **If()** function returns one value if a test returns TRUE and different value if the test returns FALSE. In this formula the test determines if the month of the selected date is less than 7. If TRUE, the current year is returned. If the month is 7 or later, the next current year is returned. The **Right()** function returns a specified number of characters from the right side of a value. In this formula the last 2 digits of the calculated fiscal year are returned.	FY18

Column = Calculation	Sample Results
Qtr = "Q" & Format(EDate([Date], -6), "Q") This formula returns a "Q" followed by the quarter number of the selected date, relative to the fiscal year. This formula uses the Format() and EDate(), which are explained below. The **Format()** function returns a value in a specified format. Here the "Q" represents the format for quarter number. The **EDate()** function returns the date that is the indicated number of months before or after the selected date. Here we used a negative 6 so it calculates the date that is 6 months before the date field. Putting it all together this formula displays a "Q" and then calculates the date 6 months prior to the [Date] column, and then formats that date as a quarter number.	Q1
Month Name = Format([Date],"mmmm") This formula uses the Format() function to return the full name of the month.	July
Month Name Short = Format([Date],"mmm") This formula uses the Format() function to return a three-character abbreviation for the name of the month. For example, for September it will return Sep.	Jul

Column = Calculation	Sample Results
`Month Number = Month(EDate([Date],-6))` This formula returns the month number of the selected date, relative to the fiscal year. Normally July 1st would be in the 7th month. But since fiscal year starts in July, July is the first month. So, to calculate a 1 we take the date, subtract six months, and then find the calendar month number of that date. This formula uses the EDate() and Month() functions, which are explained below. The **EDate()** function returns the date that is the indicated number of months before or after the selected date. Here we used a negative 6 so it calculated the date that is 6 months before the date field. The **Month()** function returns the month number of the calculated date. In our case it returns the month number of the date that is 6 months before the date field.	1
`Day Name = Format([Date],"dddd")` This formula uses the Format() function to return the full name of the day of the week.	Saturday
`Day Name Short= Format([Date],"ddd")` This formula uses the Format() function to return a three-character abbreviation for the name of the day of the week.	Sat
`Day Num = Day([Date])` This formula returns the day of the month	1

Column = Calculation	Sample Results
Week Num = WEEKNUM(EDate([Date],-6)) This formula returns the week number relative to the start of the fiscal year.	1
Week Day Num = WEEKDAY([Date]) This formula returns the number of the day of the week. For example, Sunday = 1, Monday = 2, etc.	1

Relating the date table to other tables

Now that you have created your date table, marked it as the official date table, and added useful date-oriented columns, you can begin creating relationships between the fact and dimension tables to the new date table. You will start with the Sales Order, Order Date field.

Sales Order - Order Date

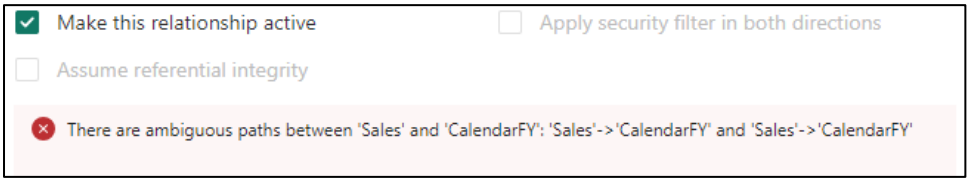
1. If necessary, switch to the Report view.
2. From the ribbon, select **Modeling**, **Relationships**, **Manage Relationships**.
3. Click the **New relationship** button.
4. For the first drop down choose **Sales**.
5. Scroll to the right and select the **OrderDate** field.
6. For the second drop down choose **CalendarFY**.
7. Click in the **Date** field.
8. Verify the **Make this relationship active** is checked. This option tells Power BI to use this as the default relationship for calculations and visualizations.
9. Click **Save**.

Sales - Due Date

Click the **New relationship** button.

10. For the first drop down choose **Sales**.
11. Scroll to the right and select the **DueDate** field.
12. For the second drop down choose **CalendarFY**.
13. Click in the **Date** field.
14. Check the **Make this relationship active** option.

    Notice an error occurs. Each pair of tables can only have one active or default relationship.



15. Uncheck the **Make this relationship active** option.
16. Click **Save**.
17. Repeat the process to establish an indirect relationship between the **Ship Date** and the date table.
18. Click **Close**.

Create a hierarchy for the date table

Now that you have created your date table, you will want to establish the following date hierarchy: Fiscal Year, Quarter, Month, and Date.

1. Switch to the **Table** view.

2. Expand the **CalendarFY** table.

3. Right click on the **FY** field and choose **Create hierarchy**.

   Notice a new **FY Hierarchy** field appears under the Date field.

4. **Right click** on the **FY Hierarchy** field, choose **rename,** and enter **Fiscal Year Hierarchy** as the new name.

5. For each of the following fields, **right click** on the field, select **Add to hierarchy**, and choose **Fiscal Year Hierarchy**.

   - Qtr
   - Month Name Short
   - Day Num

Effect on date fields

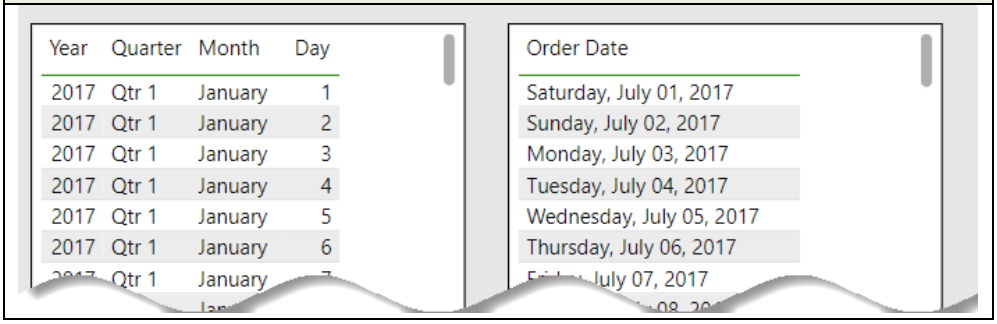
1. In the Fields pane expand the **Sales** table.

2. Notice the Due Date, Order Date, and Ship Date no longer have a calendar icon next to them. The computer no longer needs to maintain a separate hidden date table for each of them.

3. Switch to Report view.

   Notice the first table we created earlier no longer shows the quarter, month and day.

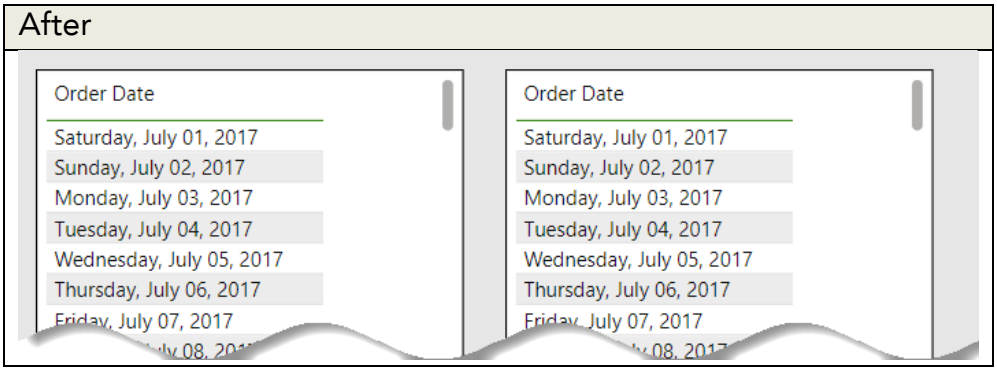
Before



Year	Quarter	Month	Day
2017	Qtr 1	January	1
2017	Qtr 1	January	2
2017	Qtr 1	January	3
2017	Qtr 1	January	4
2017	Qtr 1	January	5
2017	Qtr 1	January	6
2017	Qtr 1	January	

Order Date
Saturday, July 01, 2017
Sunday, July 02, 2017
Monday, July 03, 2017
Tuesday, July 04, 2017
Wednesday, July 05, 2017
Thursday, July 06, 2017
July 07, 2017

After



Order Date
Saturday, July 01, 2017
Sunday, July 02, 2017
Monday, July 03, 2017
Tuesday, July 04, 2017
Wednesday, July 05, 2017
Thursday, July 06, 2017
Friday, July 07, 2017

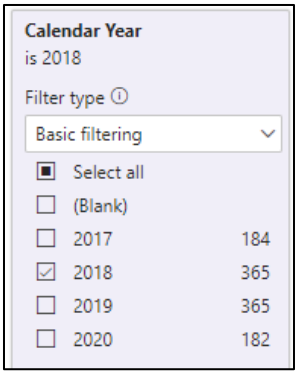
Order Date
Saturday, July 01, 2017
Sunday, July 02, 2017
Monday, July 03, 2017
Tuesday, July 04, 2017
Wednesday, July 05, 2017
Thursday, July 06, 2017
Friday, July 07, 2017

4. Select the **first table** and **remove** the **OrderDate** field from the values well.

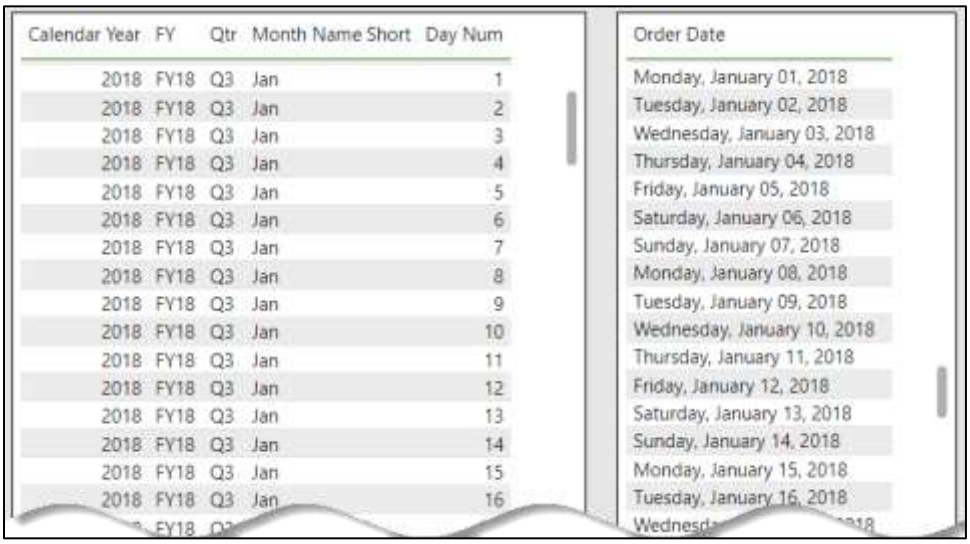
5. Add the CalendarFY's **Calendar Year** field to the values well.

6. Add the CalendarFY's **Fiscal Year Hierarchy** to the values well.

7. In the Filter's pane use, change the **Calendar Year** filter to only display dates from 2018.



8. Notice all the January dates are in Q3.



Calendar Year	FY	Qtr	Month Name Short	Day Num	Order Date
2018	FY18	Q3	Jan	1	Monday, January 01, 2018
2018	FY18	Q3	Jan	2	Tuesday, January 02, 2018
2018	FY18	Q3	Jan	3	Wednesday, January 03, 2018
2018	FY18	Q3	Jan	4	Thursday, January 04, 2018
2018	FY18	Q3	Jan	5	Friday, January 05, 2018
2018	FY18	Q3	Jan	6	Saturday, January 06, 2018
2018	FY18	Q3	Jan	7	Sunday, January 07, 2018
2018	FY18	Q3	Jan	8	Monday, January 08, 2018
2018	FY18	Q3	Jan	9	Tuesday, January 09, 2018
2018	FY18	Q3	Jan	10	Wednesday, January 10, 2018
2018	FY18	Q3	Jan	11	Thursday, January 11, 2018
2018	FY18	Q3	Jan	12	Friday, January 12, 2018
2018	FY18	Q3	Jan	13	Saturday, January 13, 2018
2018	FY18	Q3	Jan	14	Sunday, January 14, 2018
2018	FY18	Q3	Jan	15	Monday, January 15, 2018
2018	FY18	Q3	Jan	16	Tuesday, January 16, 2018

You have seen that creating a date table with a date hierarchy and relating the date table to the various date fields is an effective way to handle fiscal years. You have also seen that creating a date table reduces the overhead associated with each table having its own hidden date table.

Sorting a field by another field

You may have noticed as you scroll through the data that the months are sorted in the wrong order. For example, in Q3 they are listed as Feb, Jan, and Mar. To put them in calendar order we will use the **Sort by column** feature to sort of the **Month Name Short** field by the **Month Number** field.

1. In the Data pane, expand the CalendarFY table, and select the **Month Name Short** field.

2. On the ribbon's Column tools tab, click the **Sort by column** drop down, and then select the **Month Number** field.

3. Notice the months in the table visual are now in calendar order instead of alphabetical order.

Edit a Hierarchy

After a hierarchy has been created you can modify it. In this exercise you will fine tune the Fiscal Year Hierarchy.

1. Go to **Model** view.
2. In the fields pane, expand the **CalendarFY** table's list of fields.
3. Expand the **Fiscal Year Hierarchy**.
4. **Right click** on the **Month Name Short** field, choose **Rename**, then enter the **Month**.
5. Repeat the process to rename the following:

Old Name	New Name
FY	Fiscal Year
Qtr	Quarter
Day Num	Day

6. If necessary, **expand** the **properties pane**, and then click on the **Fiscal Year Hierarchy**.
7. Notice the following options:
   - You can enter a **description** for the hierarchy.
   - You can add more levels using the **Select a column to add level…** drop down.
   - You can hide a hierarchy.
   - You can **rearrange** the hierarchy levels in the area below the "Select column to add level…" drop down dragging and dropping the levels desired. When finished click **Apply Level Changes.**

   - Click the x next to a level to remove it from the hierarchy.
8. Click Apply Level Changes.
9. Return to the report view.
10. Save your report.

3. Visualizations

Complete this unit and you'll be able to:

A. Explain why visualizations tell stories

B. Explain when to use which visualizations

C. Create a visualization that shows the top rating

D. Create a custom sort

E. Use the Small Multiples feature

F. Create an advanced line chart

G. Create a combination chart

H. Create a matrix visualization

I. Create a gauge visualization

J. Create a KPI visualization

A. Visualizations tell stories

In the Power BI Level 1 course you learned how to use Microsoft Power BI Desktop to create Power BI reports with multiple visualizations. As its name implies, visualizations help viewers visualize information. In other words, a good visualization can help tell a data-driven story that…

- Helps provide context, meaning, relevance and clarity.
- Helps presents cold hard facts with attractive media that captures and keeps an audience's attention with an experience that is both analytical and emotional.
- Helps a decision maker see the big picture.
- Helps observers learn what has occurred in an organization.
- Helps viewers gain insights.
- Helps a marketer highlight the value of their products and services.
- Helps a presenter influence stakeholders towards their position.

Ling Wong defines data storytelling as "the practice of blending hard data with human communication to craft an engaging narrative that's anchored by facts."[1] Linda's article, "Data-Driven Storytelling: 9 Techniques for Effective Visualization" is worth reading and can be found at https://www.gokantaloupe.com/blog/best-techniques-for-data-driven-storytelling.

[1] https://www.gokantaloupe.com/blog/best-techniques-for-data-driven-storytelling

B. Which visualization should I use?

The following is a summary of various visualizations, and their best use in telling the data-driven story.

Table

Tables are useful for displaying multiple pieces of raw data (text, numbers, dates, etc.). They are like a spreadsheet because they show data in a series of rows and columns. Tables have column headers and total which can be turned on or off and formatted as desired.

Matrix

A matrix visualization is useful for showing summary data. Like a table, it too shows raw data in rows and columns, but it also shows aggregated calculations. In appearance and function it is similar to a PivotTable in Excel.

Bar and column charts

Bars and column charts are useful for displaying specific data across different categories in either a stacked or clustered format.

Line and area charts

Line and area charts are most useful for displaying trends over time. They both usually display time on an X (horizontal) axis and values on a Y (vertical) axis. The line chart of course shows a line, and the area chart shows the line with shaded color in the area underneath the line.

Ribbon charts

Ribbon charts are a special type of clustered column charts that are most useful for allowing viewers to easily spot trends in the data.

Treemaps, pie charts, and donut charts

Treemaps, pie charts, and donut charts are useful for displaying the quantitative relationship between various parts and their whole.

Pies and donut charts are best used when the number of pieces is 7 or less.

Combo charts

Power BI provides two types of combo charts: 1) a **Line and Stacked Column chart** and 2) a **Line and Clustered Column chart**. These are useful for…

- Comparing multiple measures with different value ranges
- Seeing the correlation between two different measures
- Determining if a measure meets a target

Card visualization

Card visualizations are useful for viewing a single data point (value) or a single total.

Multi-row card

Multi-row card visuals are useful for showing multiple data points with each data point listed on a separate row.

Funnel

Funnel visualizations are useful for displaying a linear sequential process or workflow with connected stages. For example, an email marketer can use a funnel chart to compare 1) the number of emails that were sent, 2) the number that were opened, 3) the number of readers who followed a hyperlink within the message, and 4) the number of readers who purchased a product or service from the website.

Funnels are best used when the process has at least four stages, and the number of items in the first stage is greater than the number of items in the final stage.

Gauge

A gauge looks like a car's speedometer. It is useful for seeing progress toward a goal, or an actual value compared to a maximum value. When creating a gauge a best practice is to include the actual value, as well as the minimum, maximum, and target values.

Waterfall

Waterfall charts are useful for…

- Showing a running total, and the impact each piece has on the grand total.
- Seeing changes over time or different divisions.
- Breaking down total revenue by the various products and services.
- Viewing an organization's beginning and ending headcount.
- Visualizing income and expenses and their impact on the total account balance

Scatter chart

A scatter chart is useful for showing patterns in large sets of data without regard to time, and for plotting groups of numbers on both an X and Y axis. It is also useful for displaying anomalies or outliers in your data.

Maps

Maps are useful for showing data that are specific to a geographical location (country, region, state, province, county, city, etc.)

Slicers

Slicers are useful for viewing and editing filters.

Q&A

Question and Answer visuals are useful for letting users ask questions, in natural language, and receive answers in the form of a visualization.

KPI

KPI is an acronym for Key Performance Indicator. KPI visuals are useful for tracking progress toward a specific goal.

Further reading

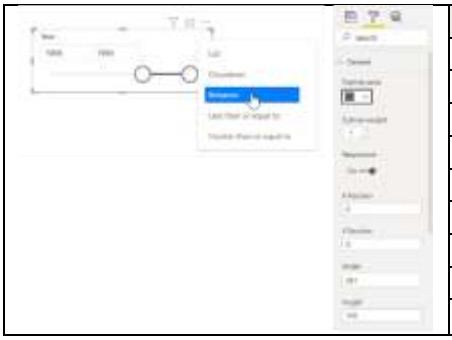
Microsoft provides an excellent overview of various visualizations at https://docs.microsoft.com/en-us/learn/modules/visuals-power-bi/3-effective-visualization.

C. Which rating is on top?

In this exercise you need to make a presentation regarding movies in America. You need to use the data to tell a story about Hollywood's preference for the various types of ratings. You want to know, and clearly demonstrate to others, if the percentage of movies per rating has changed in the last four decades of the 20th century.

1. Create a new Power BI report based on the **Movies.xlsx**, **Lots of Movies** sheet.

2. Create a slicer with the following properties:



Property	Setting
Field	Year
Type	Between
First	1968
Last	1990
Effects, Visual border	On
General, X Position	0
General, Y Position	0
General, Width	281
General, Height	100

3. Create a Stacked column chart with following parameters.

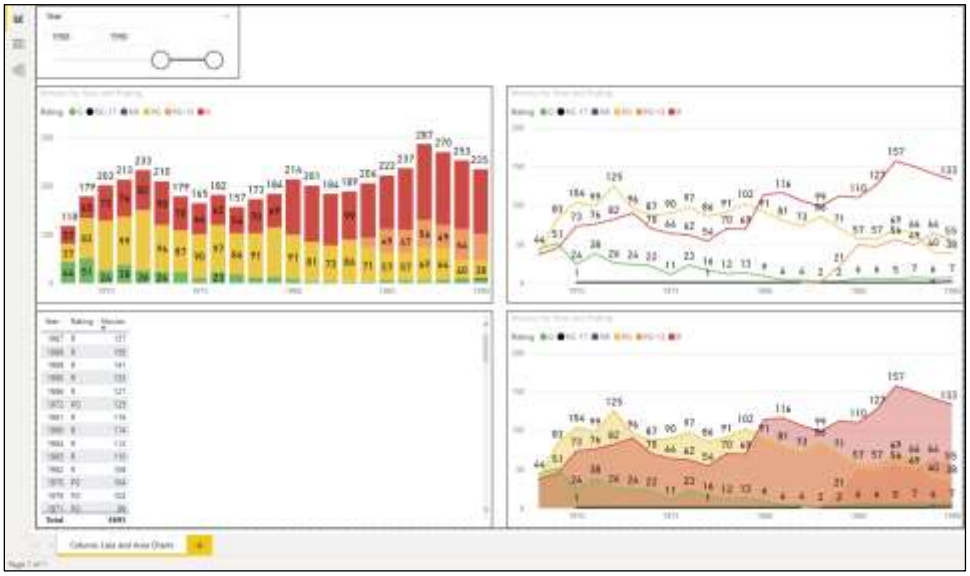
Property	Setting
X-Axis	Year
Y-Axis	Count of Titles
Y-Axis, Rename for this visual	Movies
Legend	Rating
Effects, Visual border	On
View, Theme	City Park
Columns, Colors G	Green
Columns, Colors PG	Yellow
Columns, Colors PG-13	Orange
Columns, Colors R	Red
Columns, Colors NC-17	Black
Columns, Colors NR	Blue
Data Labels	On
Data Labels, Values, Color	Black
Data Labels, Values, Font Size	12
Data Labels, Options, Label Density	100%
Total Labels	On
General, Properties, Position, Horizontal	0
General, Properties, Position, Vertical	110
General, Properties, Width	630
General, Properties, Height	300

4. Duplicate the chart, move it to the right, convert it to a line chart.

5. Duplicate the chart, move it below the column chart, convert it to a table.

6. Duplicate the chart, move it below the line chart, convert it to an area chart.

7. Name the page "**Column, Line and Area charts**".



8. Duplicate the page.

9. Delete the line chart, area chart, and table visualizations.

10. Stretch the column chart across to fit the width of the page.

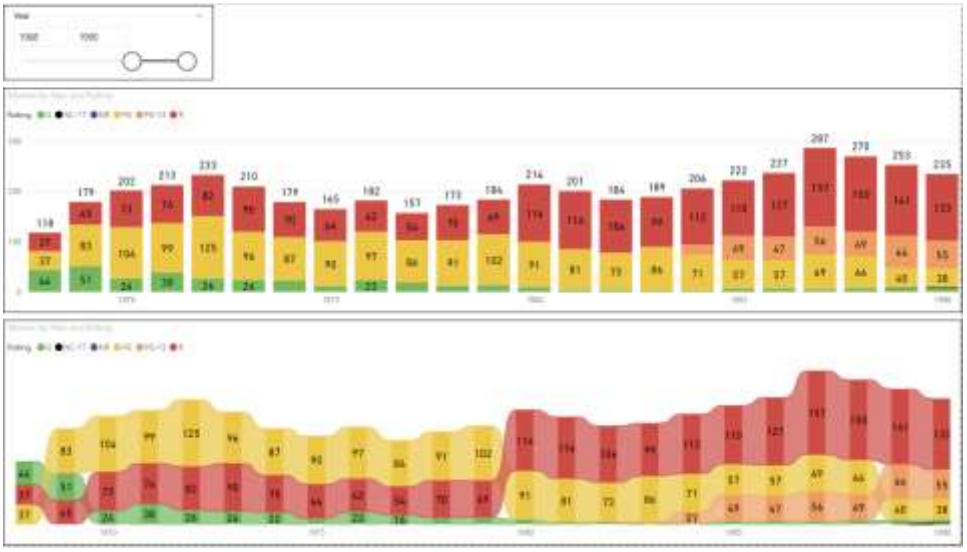
11. Duplicate the column chart and place it underneath the other one.

12. Change the new column chart into a ribbon chart.

13. Change the following properties of the ribbon chart.

Format Visual Property	Setting
Data Label, Values, Overflow text	On
Ribbons, Colors, Border	On

14. Rename the page "Column chart vs Ribbon chart".

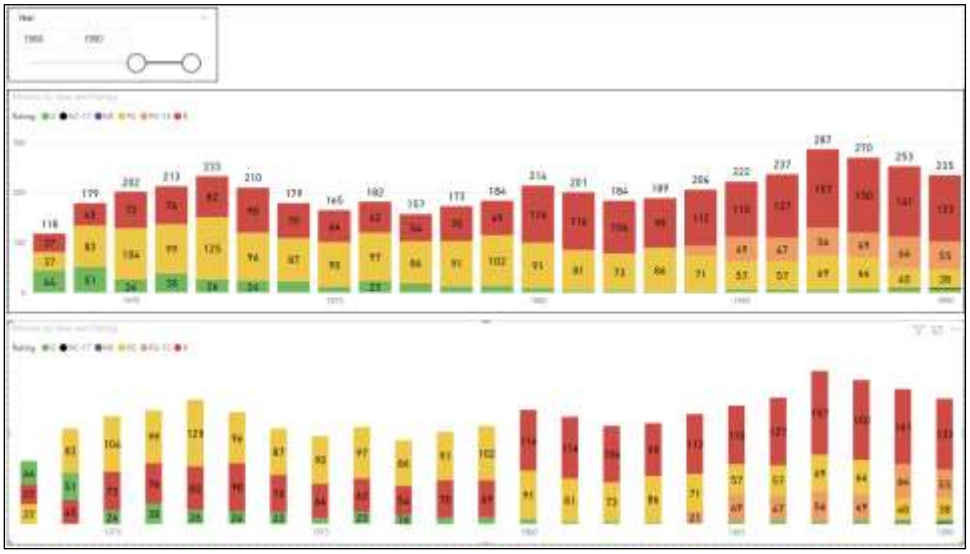


15. Duplicate the page.

16. Change the following settings for the ribbon chart.

Format Visual Property	Setting
Ribbons, Colors, Connector color	Off
Ribbons, Transparency	100
Ribbons, Border	Off

17. Rename the page as "Column chart vs Ribbon chart – no connector color".



18. Notice the following:
    - The ribbon chart now appears to be a clustered column chart.

- In the column chart, the sequence of boxes is the same for each year. R is on top, then if they exist PG-13, PG, NR, NC-17, and finally G on the bottom.
- In the ribbon chart the sequence of boxes is **not** the same for each year. The rating with the most movies for the year is always on top.

19. Return to the second page and notice the following:
    - The ribbon chart on this page has the same sequence of boxes as the one on the third page.
    - Because the connector is not 100% transparent, it is easier to see when a rating jumps into first place.
        - In 1968 G was on top, and PG was on the bottom.
        - In 1969 PG took first place, while G and R took second and third place.
        - In 1970 R moved to second place.
        - Ten years later, in 1980, R took first place, and PG took second. G was almost 0.
        - The first year with PG-13 was 1984.
        - In 1989 PG-13 took second place.
    - The tool tip for each year displays the year, rating, and count.
    - The tool tip between each year displays the year, rating and count for both the previous and the following years, and it displays the change from one year to another.

20. Save your report as **Movies.pbix**.

Of all the charts you have just created, which one best helps tell the story of the movement of PG to first place in the 1970s and R to first place in the 1980s?

D. Entering data to create custom sorts

For all the charts you just created, you will notice the legend of ratings is in alphabetical order: G, NC-17, NR, PG, PG-13, R

You have decided it will help tell your story if the legend was a more familiar sort where G is first and NC-17 and NR are last. To do this you will create a new Sort Order table that will define your desired sort.

Create a custom sort table

1. From the ribbon, choose **Home**, **Data**, **Enter data**.

2. Double click Column1 and enter "Rating Code".

3. Click the + in column 2 and rename the column "Sort Order".

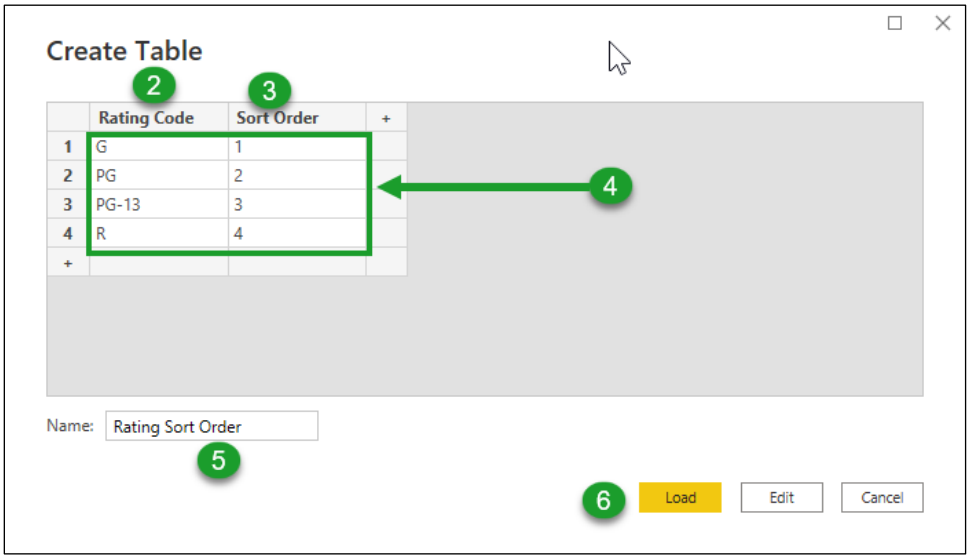
4. Enter the following values:

Rating Code	Sort Order
G	1
PG	2
PG-13	3
R	4

   Notice we did not include the NC-17 nor the NR rating. We will add them later.

5. Change the name of the table to "Rating Sort Order".

6. Click **Load**.



Edit data in a custom table

The Rating Sort Order table has been created, but we did not include the NC-17 and the NR ratings in it. In this exercise we will use the query editor to add the additional rating.

1. Right click on the **Rating Sort Order** table and choose **Edit query**.

2. In the Power Query Editor choose the **Rating Sort Order** table.

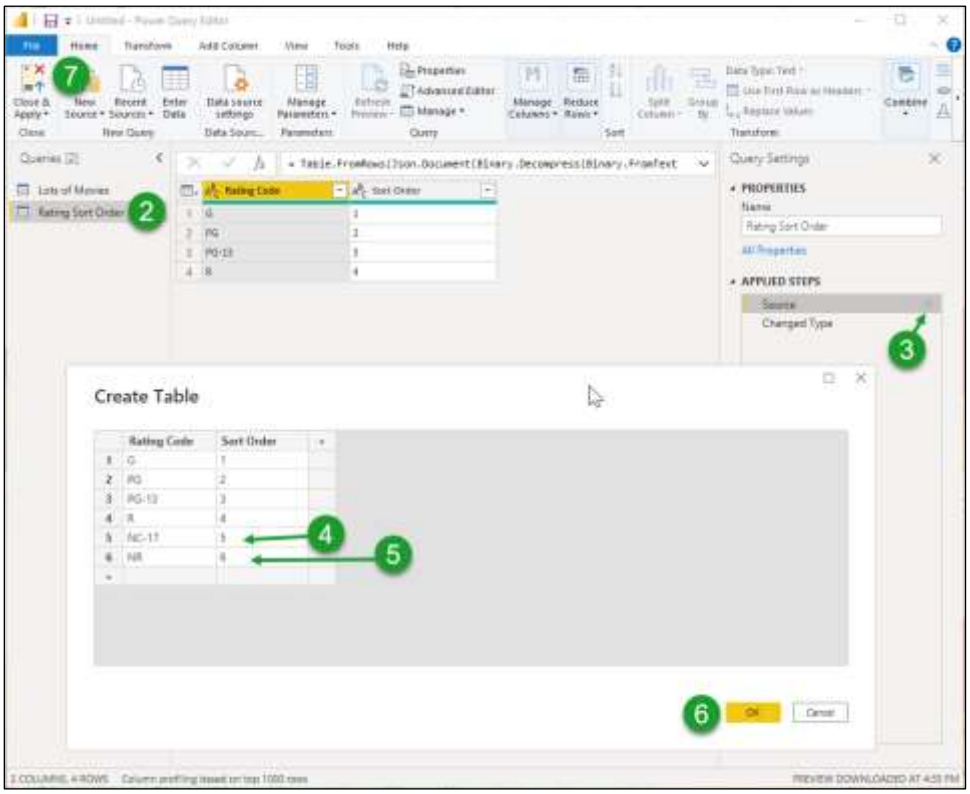
3. In the **Applied Steps** area, click the **gear icon** next to **source**.

4. In the rating column, click below R and enter **NC-17**.
   In the Sort Order column enter **5**.

5. In the rating column, click below NC-17 and enter **NR**.
   In the Sort Order column enter **6**.

6. Click the **OK** button.

7. On the ribbon select **Home**, **Close**, **Close & Apply**.



Update the data model with a new relationship

Now that the new **Rating Sort Order** table has been added, we need to update the data model to show there is a Many to One relationship between **Lots of Movies** and **Rating Sort Order**.

1. From the ribbon choose **Modeling**, **Relationships**, **Manage Relationships**.

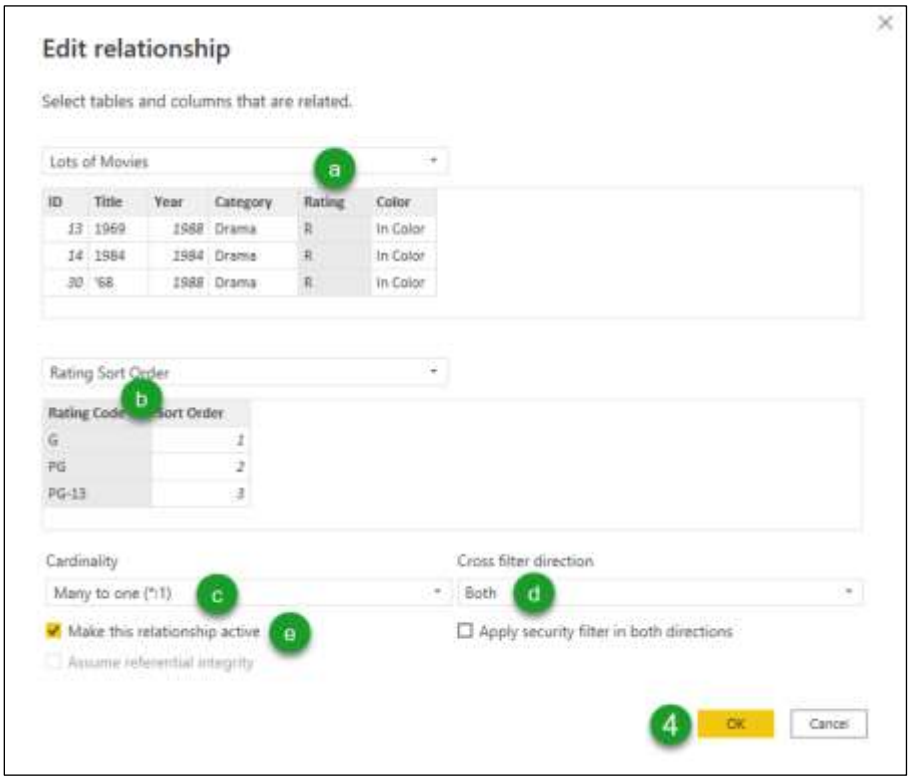
2. If the system is showing a relationship between Lots of Movies and Rating Sort Order, then click **Edit…**

   If the relationship for the two tables is not shown, click **New…**

3. As needed change the Edit relationship window to match the following:

   a) Select the **Lots of Movies**, **Rating** column.
   b) Select the **Rating Sort Order**, **Rating** column.
   c) Make sure the **Cardinality** is set to **Many to one (*:1)**.
   d) Make sure the **Cross filter direction** is set to **Both**.
   e) Make sure the Make this relationship active option.

4. Click **OK**.



5. Click **Close**.

Add a new Rating Custom Sort column to the Lots of Movie table

Power BI provides an option to override the default sort associated with a column. This feature is found in the ribbon's **Column tools** tab. When selected the designer can choose an alternate sort field. However, the alternate sort field must be from the same table. Therefore, in this exercise you will 1) create a new calculated column for the Lots of Movies table, and 2) use that calculated column to override the default sort.

1. Right click on the **Lots of Movie** table and select **New column**.

2. Enter the following formula:

```
Rating Custom Sort = Related('Rating Sort Order'[Sort Order])
```

   This is a DAX formula that works as follows:

   - The name of the field is prior to the equal sign.
   - The formula is after the equal sign.
   - The formula uses the **Related** function which can find information in another table. This is like Excel's VLookup function. However, the Related function is easier than VLookup because it can find related information via the data model.
   - The **Related** function takes one parameter, with two parts.
     - The first part is the name of the related table. Notice that the table is enclosed in single quotes because the table name includes spaces.
     - The second part refers to the desired field, enclosed between two square brackets. The square brackets are required, even if the field name has no spaces.
   - Notice Power BI uses color to help you write the formula.
     - The name of the new column is in black.
     - The function is in a light blue color.
     - The parameter, table name + field name, is in violet.

3. In the fields pane, select the **Lots of Movies**, **Rating** field.

4. From the ribbon, select **Column Tools**, **Sort**, **Sort by column**, and then choose **Rating Custom Sort**.

5. Notice that each chart's legend is now in the following order:
   G, PG, PG-13, R, NC-17, NR.

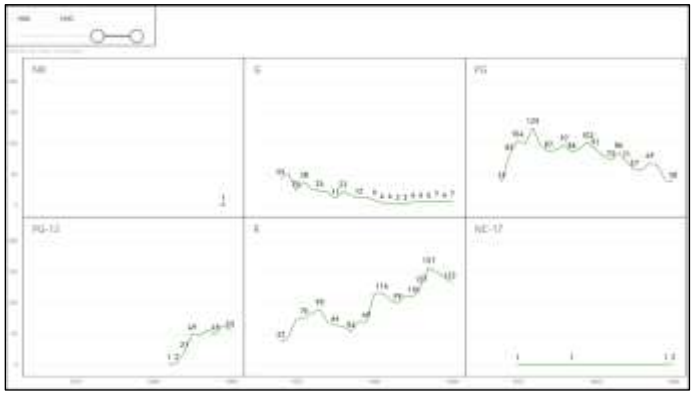
6. Save your report.

E. Using Small multiples

What are small multiples?

Small multiples, also known as trellising, is a feature that allows you to split a visual into multiple smaller versions of itself. Each version is then contained in a grid of multiple horizontal and vertical square.

Suppose for example that you wanted to show a separate line chart, per rating, of the number of movies made each year. You could do it the long way by creating one line chart for the G rated movies, and then duplicate it for each of the various ratings. Or you could save time by creating one line chart and then set the Small multiples well to Rating.



Create a line chart with small multiples

1. Duplicate the "**Column, Line and Area Charts**" page.
2. Rename the new page "**Line chart with small multiples**".
3. Delete all the visualizations except the slicer and the line chart.
4. Resize the line chart to fill most of the page.
5. Drag the **Rating** field from the Legend well to the Small multiples well.
6. Set the following formatting properties.

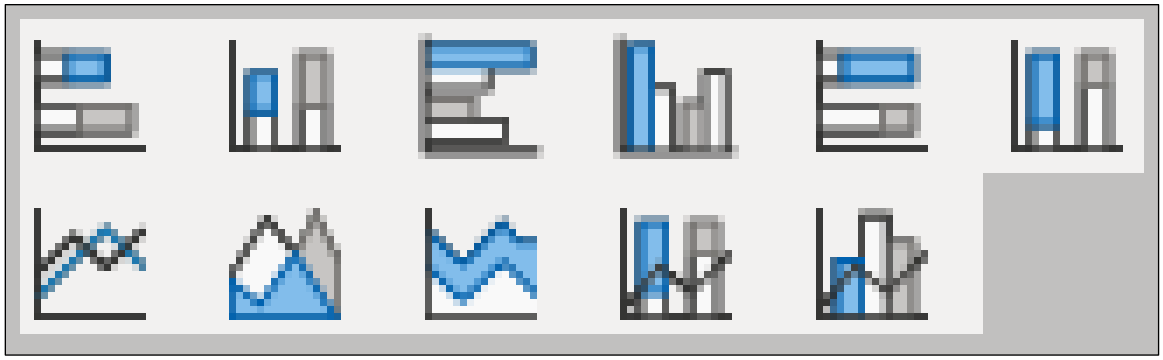
Property	Setting
Small multiple title, font	14 pt Bold
Small multiple grid, Layout, Rows	2
Small multiple grid, Layout, Columns	3
Small multiple grid, Border, Gridlines	Horizontal and vertical

7. Save your report.

Where visualizations support small multiples?

The following visualizations support small multiples:

Bar charts	Column charts	Line Charts	Area Charts
• Stacked bar chart • Clustered bar chart • 100% Stacked bar chart	• Stacked column chart • Clustered column chart • 100% Stacked column chart	• Line chart • Line and stacked column chart • Line and clustered column chart	• Area chart • Stacked Area chart

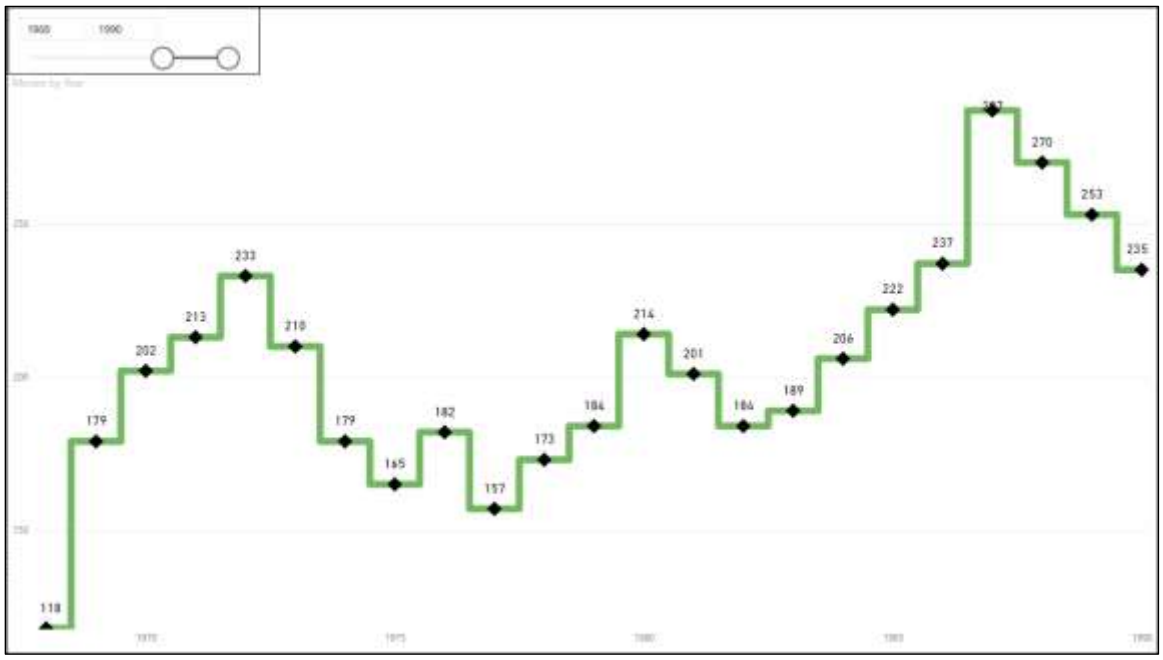


F. Advanced Line Charts

Line charts have many formatting options, including the following which we will cover in this exercise:

- Stroke width
- Line style
- Show marker: On or off
- Marker color
- Marker shape: bullet, square, diamond, triangle, and more
- Marker size
- Stepped: On or off

Formatting options



1. Duplicate the page from the previous exercise.

2. Rename the page as "**Formatted Line Charts**".

3. Remove the Rating from the Small multiples page.

4. Click the **Format** icon, and then expand the **Lines** area.

5. Change the **Shape**, **Line style** to **Dotted**, then **Dashed** and finally back to **Solid**.

   Notice the difference in the line.

6. Change the **Stroke width** to **8**.

   Notice the line is much thicker.

7. Change the **Stepped** option to **On**.

   Notice the diagonal lines have been replaced with stepped lines.

8. Collapse the **Lines** area and turn the **Markers** option to **On**.

9. Expand the **Markers** area and then expand the **Shape** area.

10. Change the **Marker color** to **black**.

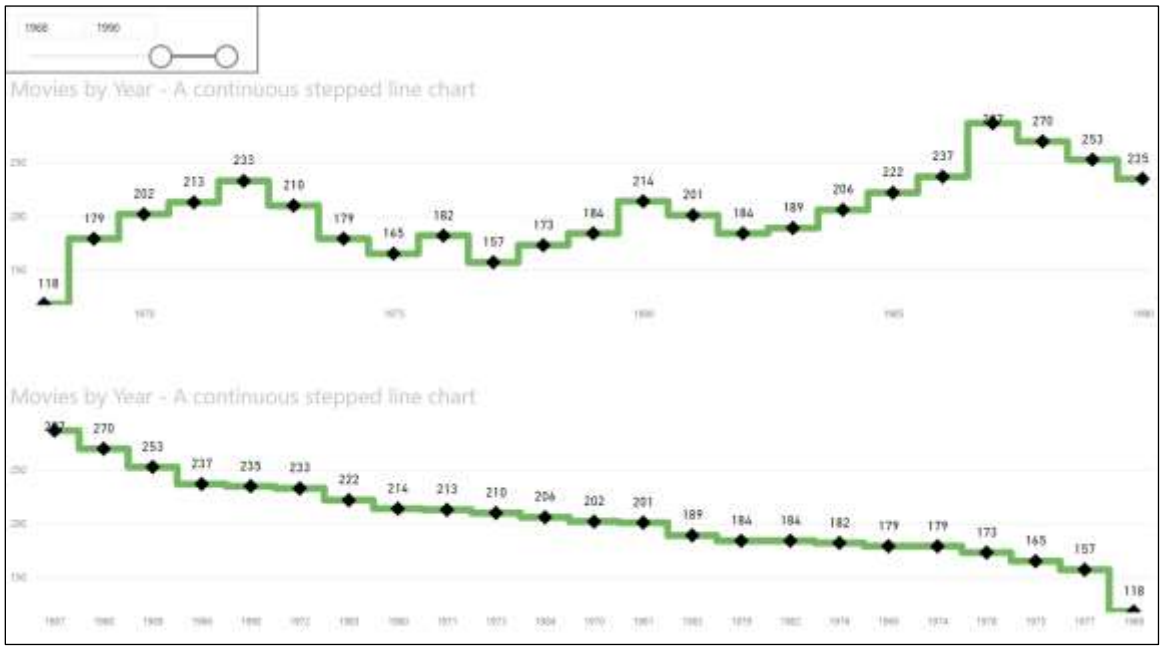
11. Change **Marker type** to a **square**, then **diamond**, and then to any shape of your choosing.

12. Change the **Marker size** to **10** and then to any size of your choosing.

13. Save your report.

Categorical Line Charts

Usually, a line chart has a continuous X axis that is based on time (years, months, dates, etc.). As you will see in this exercise a line chart can have a categorical X axis that is not continuous.



1. Change the Title of the chart to **Movies by Year – A continuous stepped line chart**.

2. Reduce the height of the line chart, so it only takes up half the page.

3. Duplicate the line chart and place the duplicate below the first.

4. Expand the **Format**, **Visual**, **X axis** section and change the **Type** option to **Categorical**.

   Notice the years on the X axis are no longer sequential from 1968 to 1990. Instead, the chart starts with the year that had the greatest number of movies, 1987. It then moves down to the year with the least number of movies, 1968.

5. Change the Title of the chart to **Movies by Year – A categorized stepped line chart**.

6. In the **X-Axis** well, **replace Year with Rating**.

7. Expand the **Format**, **Visual**, **X-axis** and notice the Type is categorical. Click the drop down and notice the continuous option cannot be selected.

   When the X axis is not numeric, the type cannot be continuous.

8. In the **Data**, **X-Axis** well, **replace Rating again with Year**.

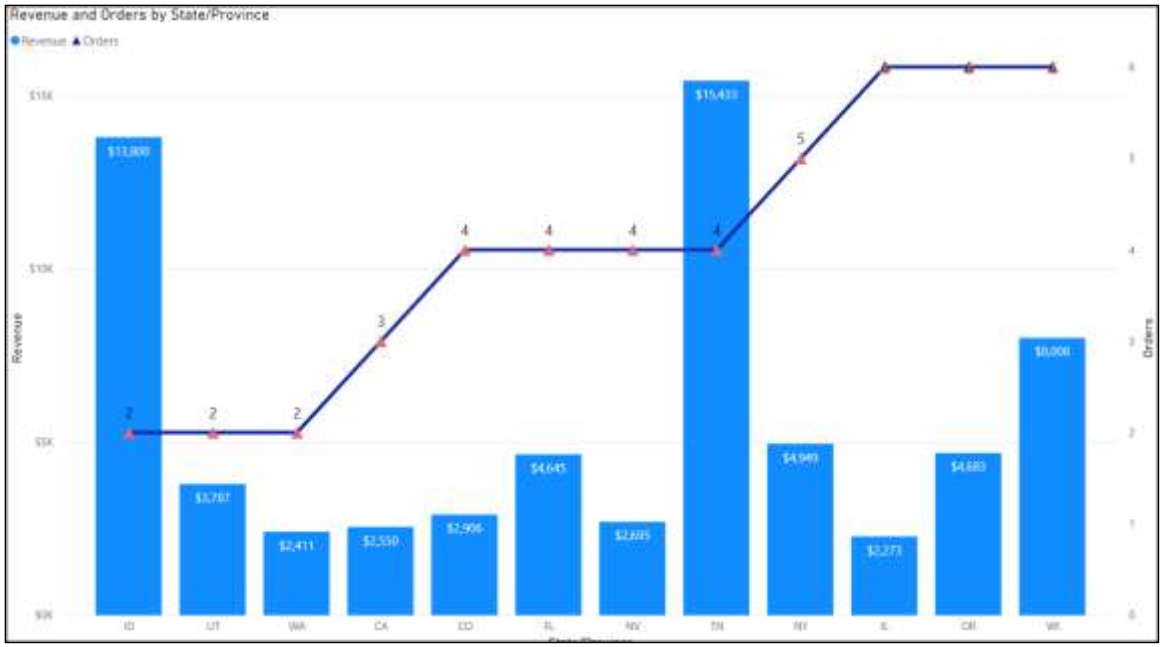
9. In the **Format**, **Visual**, **X-axis** area, change the **Type** back to **categorical**.

10. Save your report.

G. Combination Charts

Combination charts are useful when you need to compare two series that share the same X axis. The first series will be shown with a column chart, and the second series will show a line chart. The Y axis for both series is usually different.

For example, in this next exercise you want to determine if a correlation exists between total revenue and total number of orders for a food wholesaler named Northwind Traders.



1. Start a new Power BI report.

2. Get data from the **Northwind Traders.xlsx** file, and select the following tables:
   - Customers
   - Orders
   - Order Details
   - Products

3. In the Order Details table, add a new calculated field with the following calculation.

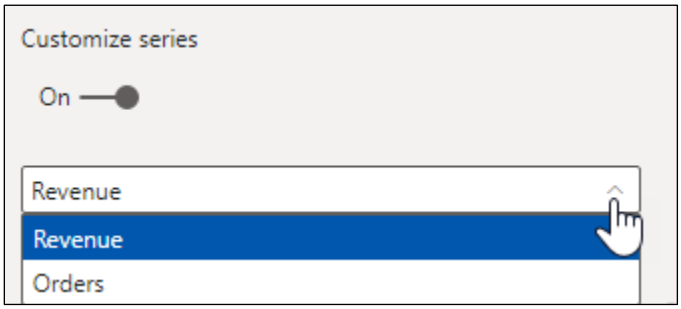
```
Extended Price = [Quantity] * [Unit Price]
```

4. Add the Line and Stacked column chart visualization.

5. Set the following properties

Property	Setting
Shared axis	State/Province
Column Values	Order Details, Extended Price
Column Values Rename for this visual	Revenue
Line Values	Order ID
Line Values Column Values	Orders
Data Labels	On
Data Labels, Display Units	None
Data Labels, Value decimal places	0
Data Labels, Orientation	Horizontal
Data Labels, Position	Inside End
Data Labels, Text size	10 pt
Customize Series	On

6. In the Data Labels area, below the Customize series, notice there is a drop down that lets you pick either the Revenue or Orders series.



When you pick a series from the drop down, all the properties below the drop down will be relative to that series.

7. Using the table below set the set the following properties for each series.

Property	Revenue Series Setting	Order Series Setting
Show	On	On
Color	White	Black
Display Units	None	None
Value decimal places	0	0
Position	Inside end	Above
Text size	10 pt	12 pt
Show background	Off	Off

8. Expand the shapes area, then set the following properties.

Property	Setting
Shade area	Off
Stroke width	4
Line style	Solid
Show marker	On
Marker shape	Triangle
Marker size	8
Marker color	Red
Stepped	Off

9. Expand the Y axis, then set the following properties.

Property	Setting
Show secondary	On
Align zeros	On
Position	Right
Start	0
End	Auto
Scale type	Linear
Title	On

10. Save your report as **Northwind.pbix**.

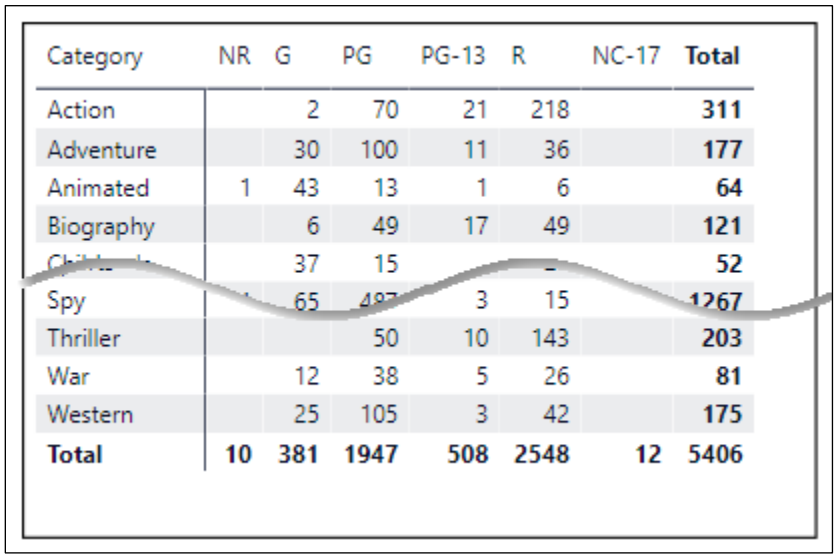
You should now have a combination chart that shows the sales revenue per state as a column chart and the number of orders per state as a line chart. Could you use this chart to convince management that there is no correlation between the number of orders and the earned revenue?

H. Matrix

As you know a table lets you view data in two dimensions (rows and columns). A matrix is similar to a table, but it supports a stepped layout, and it automatically summarizes or aggregates data.

Create a simple matrix

In this exercise you will create a matrix that answers the question "How many movies were made for each category and each rating.



Category	NR	G	PG	PG-13	R	NC-17	Total
Action		2	70	21	218		**311**
Adventure		30	100	11	36		**177**
Animated	1	43	13	1	6		**64**
Biography		6	49	17	49		**121**
[illegible]		37	15				**52**
Spy		65	[illegible]	3	15		**1267**
Thriller			50	10	143		**203**
War		12	38	5	26		**81**
Western		25	105	3	42		**175**
Total	**10**	**381**	**1947**	**508**	**2548**	**12**	**5406**

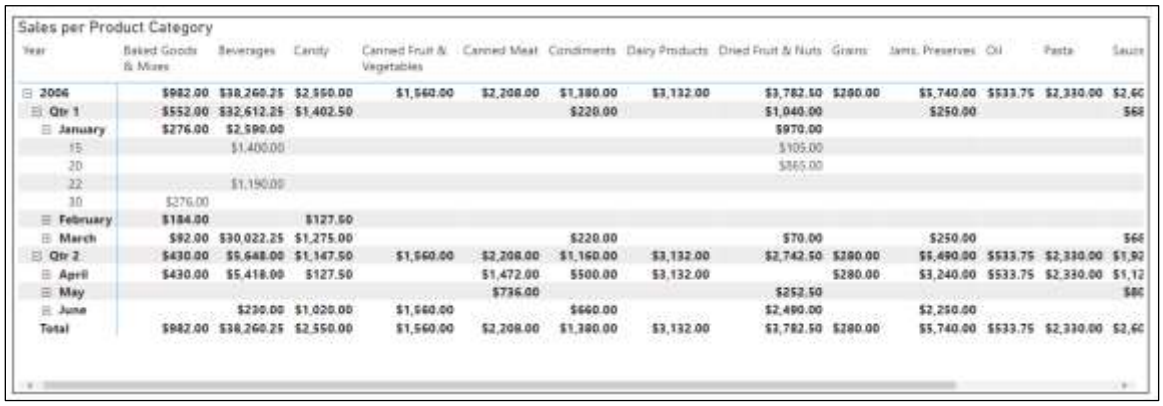
1. Open the **Movies.pbix** report from earlier.
2. Add a new page.
3. Rename the page as **Matrix**.
4. Set the following properties

Property	Setting
Rows	Category
Columns	Rating
Values	Count of Title

5. Add a border to the visualization.

Create a date oriented (hierarchical based) matrix

In this exercise you will create a date-based matrix for the Northwind company showing total sales per year, quarter, month and day per product category.



1. Open the **Northwind.pbix** report.

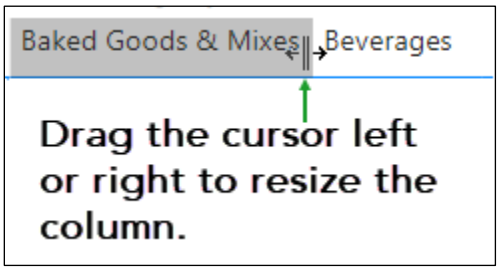
2. Create a new page.

3. Rename the page "Sales per Product Category"

4. Add a matrix visualization with the following properties.

Property	Setting
Rows	Orders, Order Date
Columns	Products, Category
Values	Extended Price
Values Rename for this visual	Total Sales
Border	On
Title	On
Title text	Sales per Product Category

5. As desired, in the column heading row, resize the column by positioning the cursor to the right of the column, and then dragging to the left or right.



6. Click the buttons to the left of the year, quarter, etc. to drill down to the next level in the date hierarchy.

Practice

1. In the Northwind report create a new page with a matrix that shows **total revenue** (Order Details, Extended price) per **Sales rep** (Employees, Last Name) per **product category** (Products, Category).

   Rename the page "**Revenue per Sales Rep per Product Category**".

   Save the report.

2. In the Movies report create a new page named "**Movies per year**".

   Create a matrix visualization that counts the number of **movies** per **year** per **category**.

   Create a second matrix visualization that counts the number of **movies** per **year** per **rating**.

   Save the report.

I. Gauge

Create a table and four gauges

In this exercise you will create a report with four gauges, for the following sales performance for the North, East, Source, and West territories of ABC Distributors.

Territory	Actual	Goal
North	$2,548	$3,500
East	$3,506	$3,500
South	$1,908	$3,500
West	$4,371	$3,500

Your report will include both a table and a gauge. The table will show the data for each territory, plus a calculation showing the percent of goal achieved by each territory.

1. Create a new Power BI report.

2. Load data from the two sheets in the **ABC Distributors.xlsx** file.

3. Rename the page "**Actual vs Goal per Territory**".

4. Add a table visualization that shows, for the Actual vs Goal table, the **Territory**, **Actual**, and **Goal** fields.

5. Create a new column with the following formula.

```
Pct of Goal = DIVIDE( [Actual] , [Goal], 0)
```

   This formula uses the Divide function which divides the first argument by the second argument. This function also prevents Divide by Zero errors that will occur if the second argument is ever zero. The third argument tells the computer what to show instead of the Divide by Zero error.

6. Select the **Pct of Goal** field, and then using the ribbon's Column tools tab, change the format to a Percentage with two decimal places.

7. Add the **Pct of Goal** field to the table.

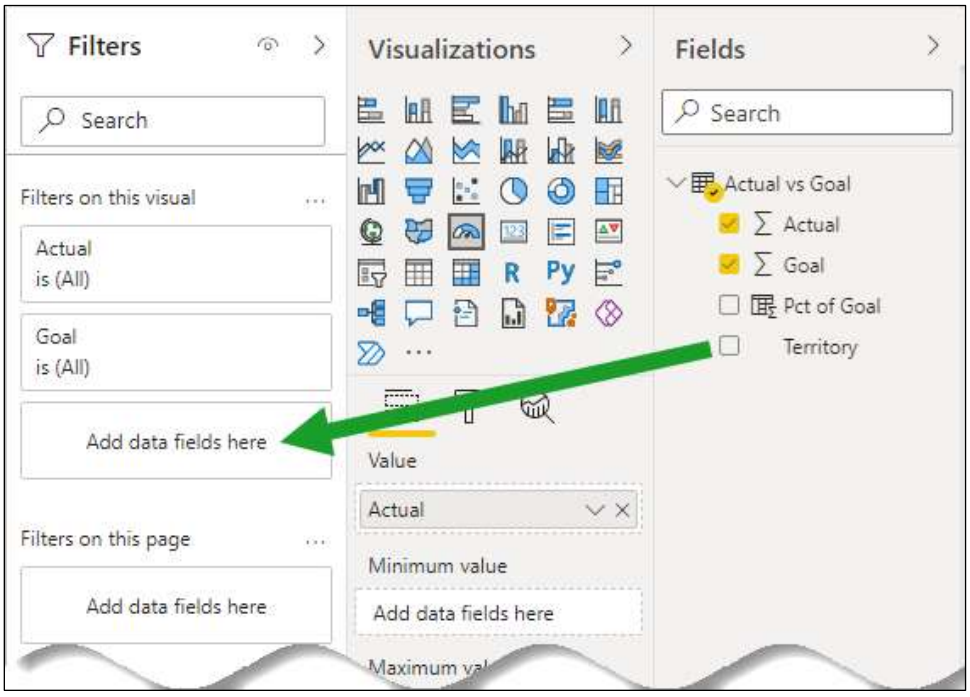
8. Add a Gauge visual with the following properties.

Property	Setting
Value	Actual
Target Value	Goal

Notice the Gauge is using the sum of all territories.

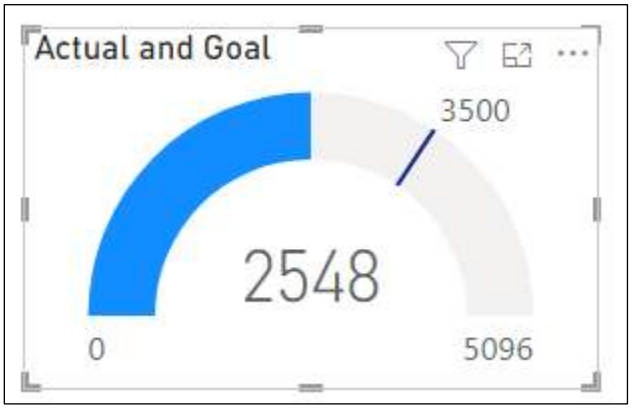
Notice that unfortunately the Gauge does not have a Small multiples option. Instead, we will have to create four separate gauges each with a different filter.

9. Drag the Territory field to the Add data fields here section of the Filters on this visual.



10. In the filter select **North**.

11. Notice the following:



- The gauge now shows the actual value, 2548, for the North territory in both the blue shaded curve and in the center of the visual.
- The gauge shows the goal with a line and its value of 3500.
- Because we did not include a Maximum value, the gauge sets the maximum as double the actual value.

12. Set the following properties for the gauge.

Property	Setting
Gauge axis, Max	5000
Border	On

13. Expand the Title section.

14. Click the conditional formatting button,  , next to Title text.

15. In the dialog box, click the drop down for the **Based on field** option, and choose **Territory**.

16. In the Summarization box choose **First**.

17. Click **OK**.

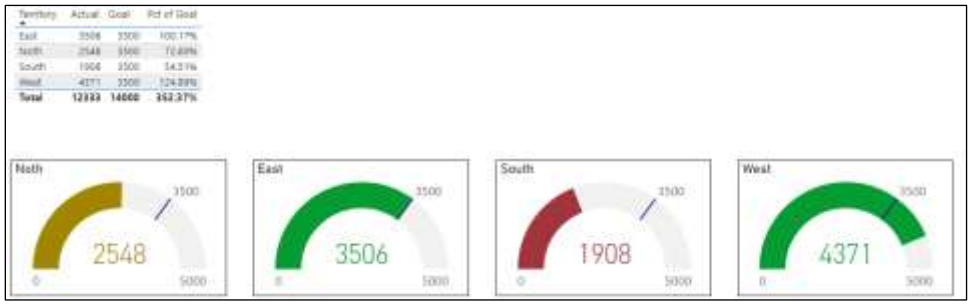
18. Duplicate the visual and place the duplicate on the right.

19. Then, in the fields pane, change the Territory filter to East.

20. Repeat the process for South and West.

21. Compare your screen with the image at the beginning of this exercise.

Changing gauge colors



You will now change the colors of the gauge based on the territory's percent of goal as shown below.

Percent of Goal	Color	Apply to
< 60%	Red	South
< 100%	Yellow	North
>100%	Green	East and West

1. Select the **North**'s visualization.

2. Expand the **data colors** section and set the **fill color** to a **dark yellow**.

3. Expand the **callout value** and set the **color** to the same **dark yellow**.

4. Consult the table above and repeat the process for the other three gauges.

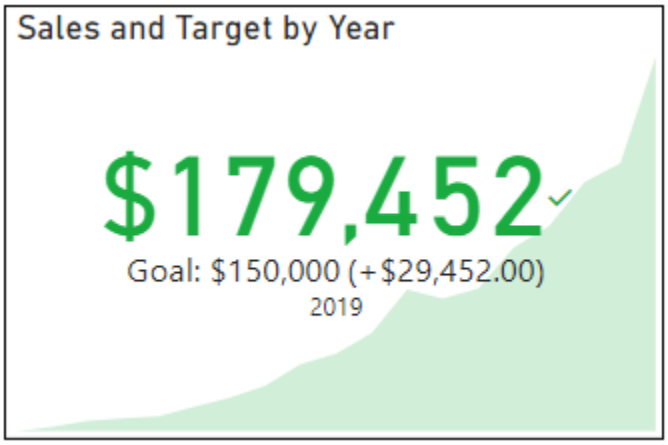
J. KPI

In this exercise you will create a KPI (key performance indicator) for the Annual Sales table of the ABC Distributors company. In a year when sales exceed the target the KPI will display green text, and red text when the sales fall short.

You will also add a table. The table will show the year, sales, target and the difference between the sales and target. You will make a negative difference appear in red by apply conditional formatting, and positive difference will appear in green.

Finally, you will add a slider so you can pick a range of years and have the KPI visual show an area chart for the selected years.

Create the KPI

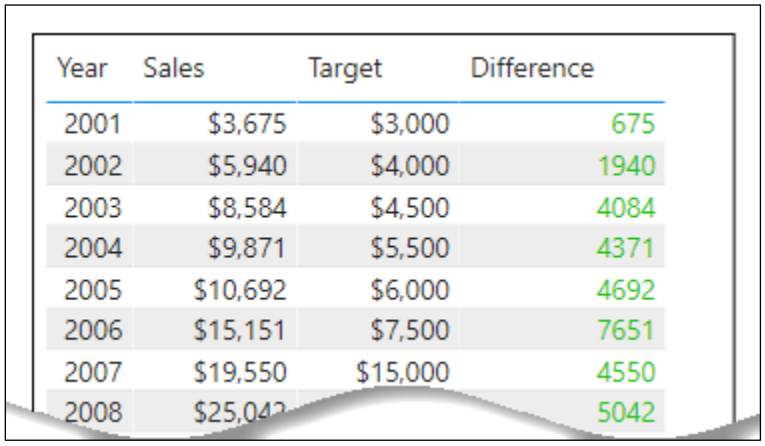


1. Add a new page to the ABC Distributors report.
2. Name the page "**Sales vs Target**".
3. Add a KPI to the report.
4. Expand the Annual Sales table
5. Set the following properties:

Property	Setting
Indicator	Sales
Trend axis	Year
Target goals	Target
Indicator, Display units	None
Trend axis	On

Property	Setting
Goal, Goal	On
Goal, Label	Value
Color coding, Direction	High is good
Date	On
Border	On

Create the table

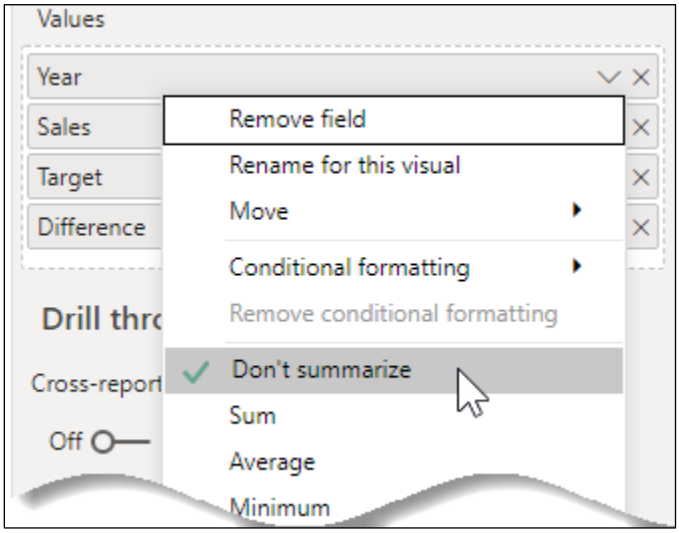


Year	Sales	Target	Difference
2001	$3,675	$3,000	675
2002	$5,940	$4,000	1940
2003	$8,584	$4,500	4084
2004	$9,871	$5,500	4371
2005	$10,692	$6,000	4692
2006	$15,151	$7,500	7651
2007	$19,550	$15,000	4550
2008	$25,042		5042

1. **Add a table visualization** to the page.

2. Add the **Year**, **Sales**, and **Target** fields to the table.

3. In the **Values** well, expand the **Year** field, and choose **Don't summarize**.



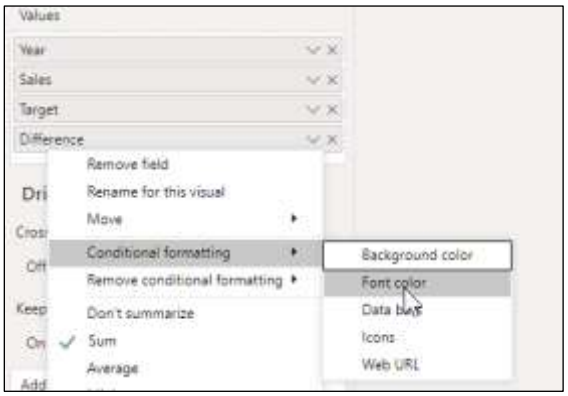
4. Add a new column to Annual Sales table with the following calculation.

```
Difference = [Sales] - [Target]
```

5. In the fields pane, select the **Difference** column, and then on the ribbon's **Column tools** tab set the format to **Currency** with **0 decimal** places.

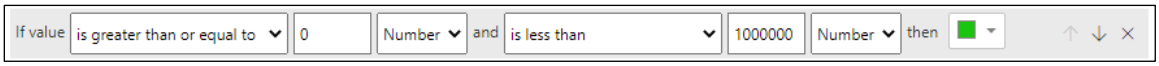
6. **Add** the **Difference** column to the table.

7. In the **Values** well expand the **Difference** value, choose **Conditional formatting**, and then choose **Font color**.



8. From the Format by drop down choose **Rules**.

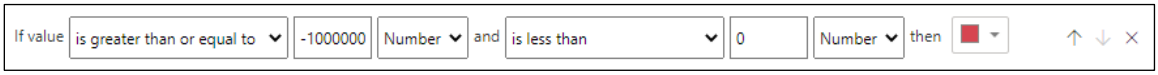
9. In the If value… line, set the following options:



Option	Setting
If value, Drop down	is greater than or equal to
If value, Value	0
If value, Percent or Number	Number
and, Drop down	is less than
and, Value	1000000 *(one million)*
and, Percent or Number	Number
then Color	Green

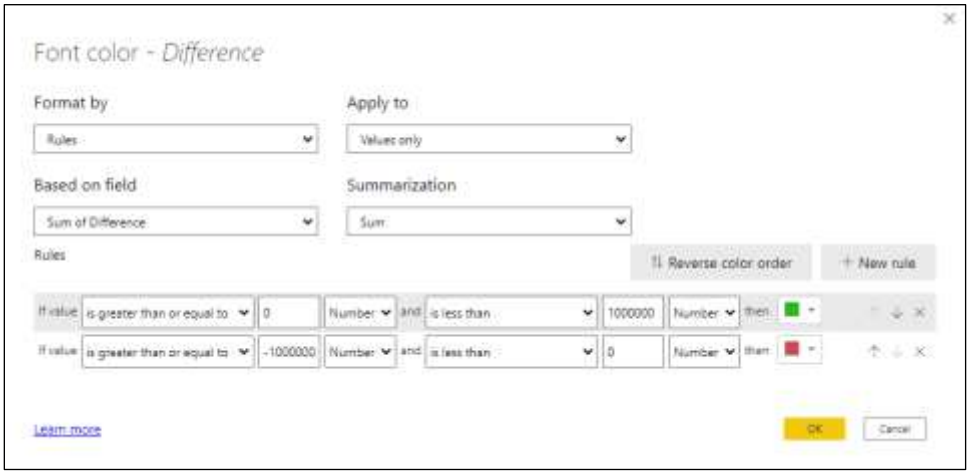
10. Click the **+ New rule** button.

11. Set the following options:



Option	Setting
If value, Drop down	is greater than or equal to
If value, Value	-1000000 *(negative one million)*
If value, Percent or Number	Number
and, Drop down	is less than
and, Value	0
and, Percent or Number	Number
then Color	Red

12. Verify the screen looks like the one below.



13. Click **OK**.

14. Compare your table with the screen shot below.

Year	Sales	Target	Difference
2001	$3,675	$3,000	$675
2002	$5,940	$4,000	$1,940
2003	$8,584	$4,500	$4,084
2004	$9,871	$5,500	$4,371
2005	$10,692	$6,000	$4,692
2006	$15,151	$7,500	$7,651
2007	$19,550	$15,000	$4,550
2008	$25,042	$20,000	$5,042
2009	$35,308	$35,000	$308
2010	$40,217	$45,000	($4,783)
2011	$49,725	$60,000	($10,275)
2012	$70,115	$75,000	($4,885)
2013	$65,862	$80,000	($14,138)
2014	$70,675	$85,000	($14,325)
2015	$89,834	$90,000	($166)
2016	$100,072	$100,000	$72
2017	$120,695	$110,000	$10,695
2018	$129,485	$120,000	$9,485
2019	$179,452	$150,000	$29,452
Total	**$1,049,945**	**$1,015,500**	**$34,445**

Add the year slicer

1. Add a slicer to the page.
2. Add the Year field to the slicer.
3. If necessary, change the slicer type to **between**

Test the report

1. Using the table below, set the slicer to the years shown, and notice the results.

Years	KPI shows
2001 to 2019	• The text displayed is for the last year of the slicer's date range. • The text is in green because 2019 had a positive difference. • An area chart representing the data from 2001 to 2019. • The chart is also green because 2019 exceeded the target.
2010 to 2015	• The text displayed is for the last year of the slicer's date range. • The text is in red because 2014 had a negative difference. • An area chart representing the data from 2010 to 2014. • The chart is also red because 2014 did not meet the target.

2. Reset the slicer to 2001 to 2019.
3. In the table click on the year 2011.

   Notice the KPI is in red because 2011 had a negative difference.

   Notice the KPI does not show a trend (area chart) because only one year was selected.

4. Control-click on 2016.

   Notice the KPI is in green because the last year 2016 had a positive difference.

   Notice the KPI show a trend (area chart) for the two selected years.

5. Experiment selecting data from the table and/or slicer and see the results in the KPI.

6. Save your report as **ABC Distributors**.

Power KPI and Power KPI Matrix

Power KPI and Power KPI Matrix are two additional visuals from Microsoft that you can add from AppSource.

4. Additional Visualization Objects

Complete this unit and you'll be able to:

A. Insert a text box

B. Insert a shape

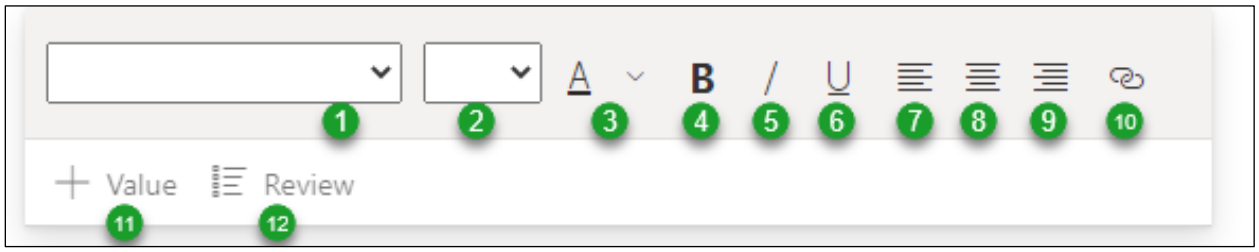
C. Insert an image

D. Create a bookmark

E. Insert a button

A. Inserting a text box

Like many other programs Microsoft Power BI allows you to insert text via a text box. That text can be formatted using the text box toolbar.

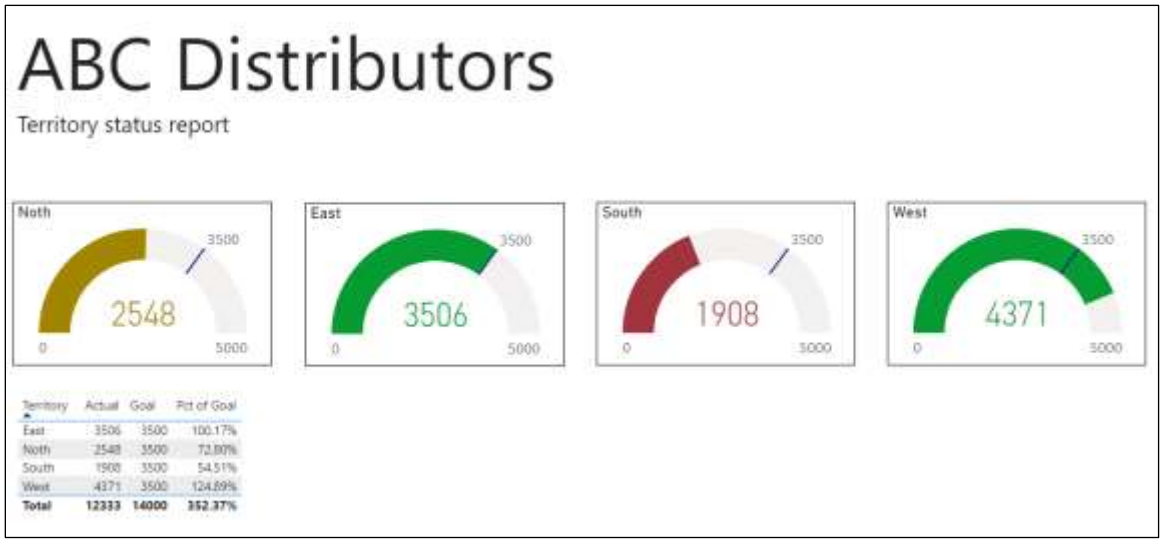


Using the toolbar, you can:

1. Change the **font**.
2. Change the **font size**.
3. Change the **font color**.
4. **Bold** the text.
5. **Italicize** the text.
6. **Underline** the text.
7. **Left justify** the text.
8. **Center justify** the text.
9. **Right justify** the text.
10. Turn the text into a **hyperlink**.
11. Make the text dynamic by inserting a variable value.
12. Review and edit previously created variable values.

Exercise: Create a text box

In this exercise you will add a simple text box to our report.



1. If necessary, open the **ABC Distributors** report you created earlier.

2. On the first page, Actual vs Goal per Territory, move the table below the first gauge.

3. From the ribbon select **Home**, **Insert**, **Text box**.

   Notice a new text box is placed on the page, and the text box tool bar is also displayed.

4. Resize the text box to fit the width of the page.

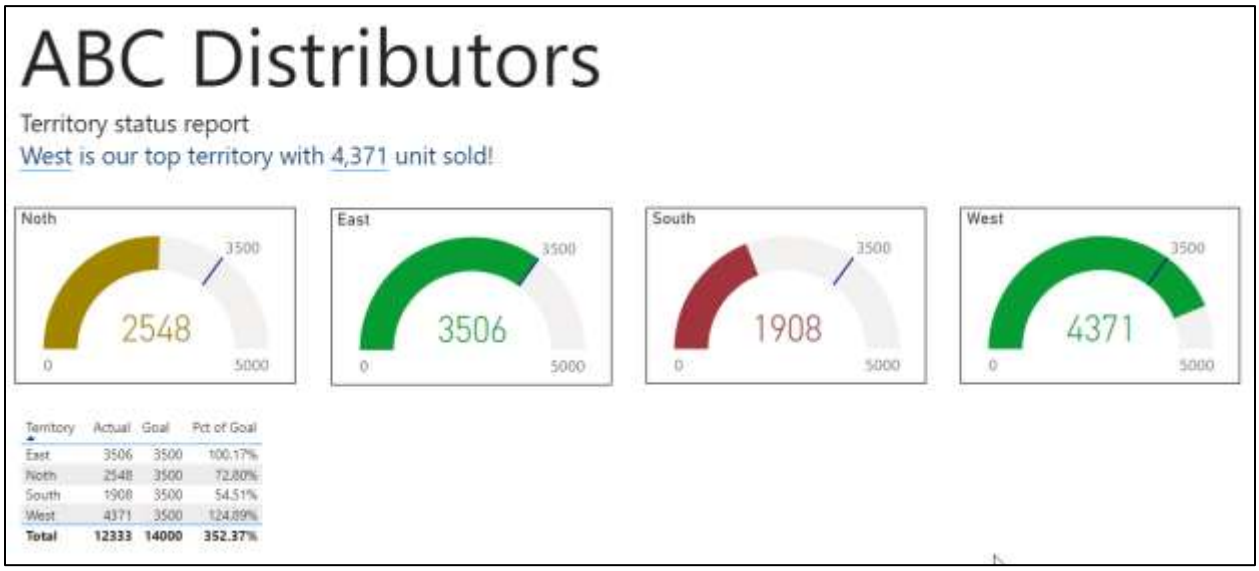
5. Change the **font size** to **60.**

6. In the text box type `ABC Distributors` then press enter.

7. Set the **font size** to **18** and type `Territory status report`.

Exercise: Add a dynamic text box

In this exercise you will add a line announcing the top territory. You will then create a calculation for the top actual amount, and then create another calculation for the name of the territory with the top actual amount. Finally, you will add these two calculations into the text box.



1. In the text box, add another return, and then type the following:

   ```
   X is our top territory with y units sold.
   ```

2. Create our first calculation by selecting from the ribbon **Home**, **Calculations**, **New Measure**.

   Remember a **measure** is as an aggregate calculation that is applied to a whole table, as opposed to a column which is a calculation that is applied to each row of a table.

3. Enter the following formula and press enter.

   ```
   Top Territory Actual = Max('Actual vs Goal'[Actual])
   ```

   This calculation uses the Max function to find the largest value for the selected column for the selected table.

4. Create our second calculation by again selecting from the ribbon **Home**, **Calculations**, **New Measure**.

5. Enter the following formula and press enter.

```
Top Territory Name = LOOKUPVALUE(
       'Actual vs Goal'[Territory],
       'Actual vs Goal'[Actual],
       [Top Territory Actual])
```

This calculation uses the LookupValue function to return the Territory Name that has an Actual value that matches the Top Territory Actual measure's value.

6. Click inside the text box, then highlight the x at the beginning of the third line.

7. Click the [**+ value**] button in the text box tool bar.

   Notice a blue box is inserted before the x.

8. Enter the following:

Property	Setting
How would you calculate this value	top territory name
Name your value	Top Territory Name

9. Click **Save**.

10. Delete the x.

11. Inside the text box, highlight the y in the third line.

12. Click the [**+ value**] button in the text box tool bar.

Notice a blue box is inserted before the y.

13. Enter the following:

Property	Setting
How would you calculate this value	top territory actual
Name your value	Top Territory Actual Value

14. Click **Save**.

15. Delete the y.

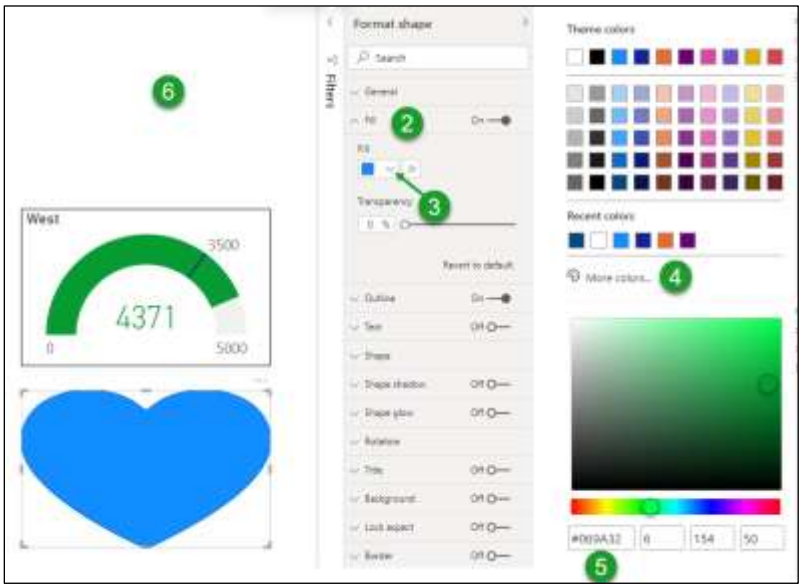
16. Save your report.

B. Inserting a shape

Like many other programs Microsoft Power BI allows you to insert shapes. In this exercise you will insert a green heart inside the west's gauge to show they are the top territory.

1. From the ribbon select **Insert**, **Elements**, **Shapes**, and then select the **heart**.

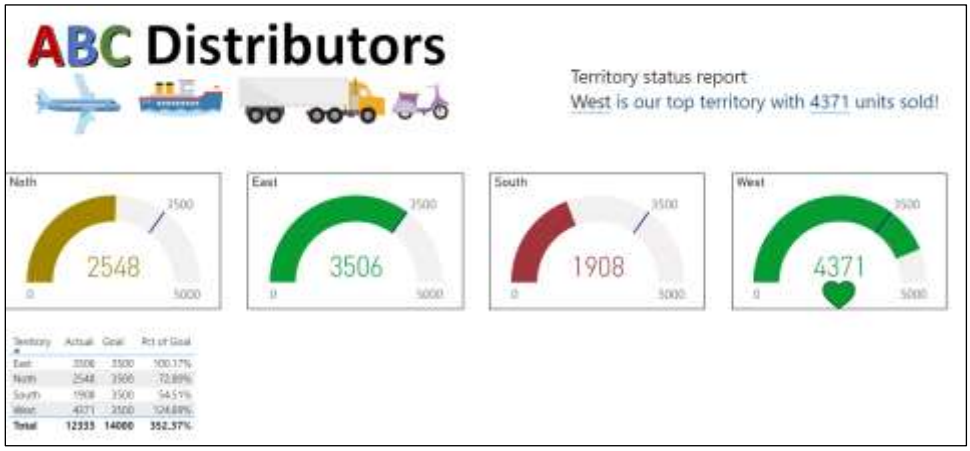
   A large blue heart is added to the page.



2. In the **Format** pane, expand the **Fill** section.
3. Click the **Fill** color drop down.
4. Click **More colors…** option.
5. Enter **#069A32** in the hexadecimal box.
6. Click back on the chart.
7. Using the shape's corner size handles, shrink the heart.
8. Position the heart inside the west's gauge.
9. In the **Format** pane, expand the **Outline** section.
10. Change the **Outline color** to **black**.

C. Inserting an image

Microsoft Power BI allows you to insert images. In this exercise you will insert a logo for ABC Distributors.



1. Be sure you have the ABC Distributors report open on the first page.
2. Edit the text box to remove the first line.
3. Move the text box to the right side of the page.
4. From the ribbon select **Insert**, **Elements**, **Image**.
5. Select the **ABC Distributors Logo.png** file from the exercise folder and click **Open**.
6. Drag and resize the image to position it in the top left corner of the page.

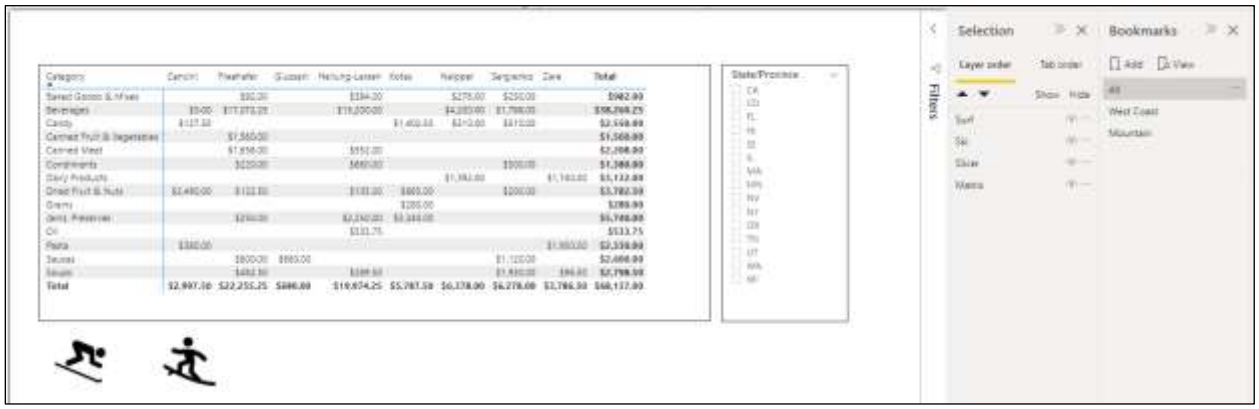
D. Adding bookmarks

In a physical book, a bookmark keeps track of the page number where you last read. In your web browser a bookmark keeps track of webpages you were viewing. In Power BI, bookmarks keep track of the current configurations of a report page, so you can easily return to this view later. The saved configurations include:

- The page you are viewing
- Filters
- Slicers
- Sort order
- Visibility of specific objects via the selection pane

In this exercise you will create some bookmarks for the third page of the Northwind report. To demonstrate the features of the bookmarks, you will add a slicer for the State/Providence field, and two images: a skier and a surfer. You will then setup three bookmarks:

- The first will be named All and will show data for all states. It will also show both images.
- The second will be named West Coast and will only show data for WA, OR, and CA. It will also show the surfer image but not the skier.
- The third will be named Mountain and will only show data for CO, ID, and UT. It will also show the skier image, but not the surfer.



Add the slicer

1. Open the **Northwind** report you created earlier, and go to the third page, "**Revenue per Sales Rep per Product Category**".
2. Add a **slicer** to the right side of the page for the **Customers**, **State/Province** field.

Add the images

1. From the ribbon select **View**, **Show panes**, **Selection**.
2. From the exercise folder, add the **Ski.png** image below the matrix.
3. In the selection pane, double click the image and rename it "Ski".
4. From the exercise folder, add the **Surf.png** image to the right side of the ski image.
5. In the selection pane, double click the image and rename it "**Surf**".

Add the bookmarks

1. From the ribbon select **View**, **Show panes**, **Bookmarks**.
2. In the bookmarks pane, click **Add**.

   Notice Bookmark 1 is added.
3. **Right click** on the bookmark, choose **Rename**, and enter the name "**All**".
4. In the slicer, select **CA**, **OR**, and **WA**.
5. In the selection pane, click the **eyeball** next to **Ski** to hide it.
6. Add a new bookmark and change its name to "**West Coast**".
7. Click on the **All** bookmark.

   Notice the page returns to showing all data, and both images.
8. Click on the **West Coast** bookmark.

   Notice the page again shows West Coast data, and the surf image is displayed.

9. Practice: Create a bookmark named "Mountain". It should have the slicer set to CO, ID, and UT and should display the Ski image.

10. Go to a different page in the report.

11. Click on any of the three bookmarks.

    The screen should automatically move to the third page, show the appropriate images, and set the slicer to the appropriate filter.

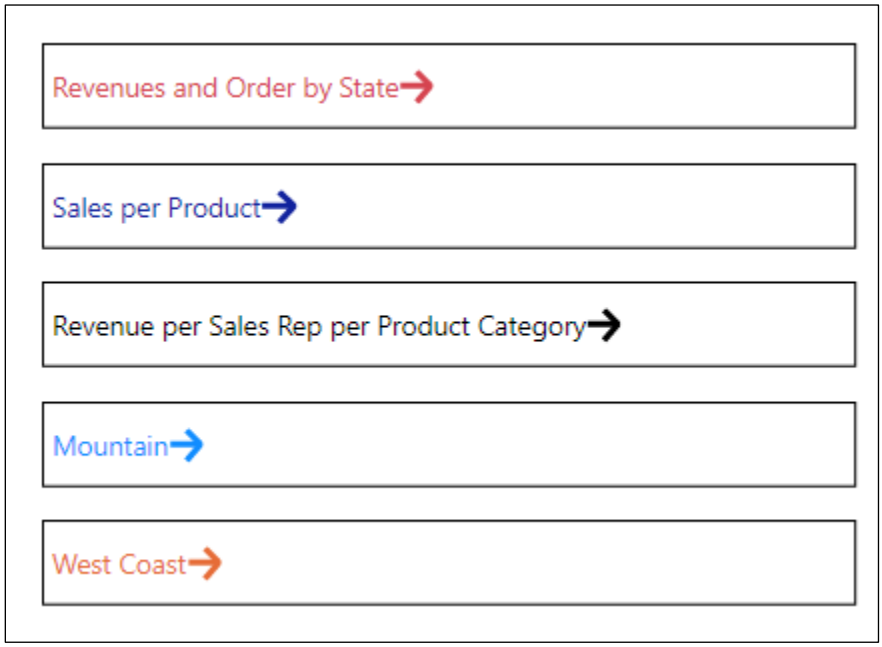
E. Inserting a button

Buttons are like shapes, image and text boxes but with one major advantage; buttons have actions that make them interactive. These actions can make your report behave like an app where users can easily interact with the content.

Below is a list of the actions you can assign to a button:

- Back – Like a back button in a browser, a button with the back action will return to the previous view.
- Bookmark – Performs the action associated with a designated bookmark.
- Drill through – Moves the user to a drill-through page with data that is filtered per a specification, all without the use of a bookmark.
- Page navigation – Moves the user to a specified page in the report.
- Q&A – Opens a window where users can type in questions that will cause Power BI's artificial intelligence to answer the question with a visualization.
- Web URL – Opens a web page in a new browser window.

Adding Buttons



In this exercise you will create a new "Home" page to your report, and add the following buttons to it:

Button Type	Button Text	Action Type	Results
Right Arrow	Revenues and Order by State	Page Navigation	Moves to the Revenue and Order by State page.
Right Arrow	Sales per Product	Page Navigation	Moves to the Sales per Product page.
Right Arrow	Revenue per Sales Rep per Product Category	Bookmark	Applies the All bookmark to let users view all the data on the Sales Rep per Product Category page.
Right Arrow	Mountain	Bookmark	Applies the Mountain bookmark.
Right Arrow	West Coast	Bookmark	Applies the West Coast bookmark.

1. Add a new page to the report.
2. Move the new page to page 1.
3. Rename the new page "Home".
4. From the ribbon select Insert, Elements, Buttons, and then choose the Right Arrow button.

5. Apply the following properties in the Format button area.

Property	Setting
Action	On
Action, Type	Page Navigation
Action, Destination	Revenues and Order by State
Text	On
Text, Text	Revenues and Order by State
Text, Color	Red
Text, Vertical Alignment	Middle
Text, Horizontal Alignment	Left
Icon	On
Icon, Icon placement	Right of text
Icon, Line color	Red
Border	On
General, Width	400

6. Duplicate the button (copy and paste) and place the new button below the first. Then change the properties as shown below.

Property	Setting
Action, Destination	Sales per Product
Text, Text	Sales per Product
Text, Color	Dark Blue
Icon, Line color	Dark Blue

7. Duplicate the button again and place the new button below the second. Then change the properties as shown below.

Property	Setting
Action, Type	Bookmark
Action, Bookmark	All
Text, Text	Revenue per Sales Rep per Product Category
Text, Color	Black
Icon, Line color	Black

8. Duplicate the Revenue per Sales Rep per Product Category button and place the new button below the third. Then change the properties as shown below.

Property	Setting
Action, Type	Bookmark
Action, Bookmark	Mountain
Text, Text	Mountain
Text, Color	Blue
Icon, Line color	Blue

9. Duplicate the Mountain button and place the new button below the third. Then change the properties as shown below.

Property	Setting
Action, Type	Bookmark
Action, Bookmark	West Coast
Text, Text	West Coast
Text, Color	Orange
Icon, Line color	Orange

10. Test each button by Ctrl Clicking on it.

11. As time permits play with other properties.

12. Save the report.

5. Interactive Tools

Complete this unit and you'll be able to:

A. Spotlight a visualization

B. Edit interactions

C. Determine which filters and slicers are affecting a visualization

D. Turn on Focus Mode

E. Use the Show as a Table feature

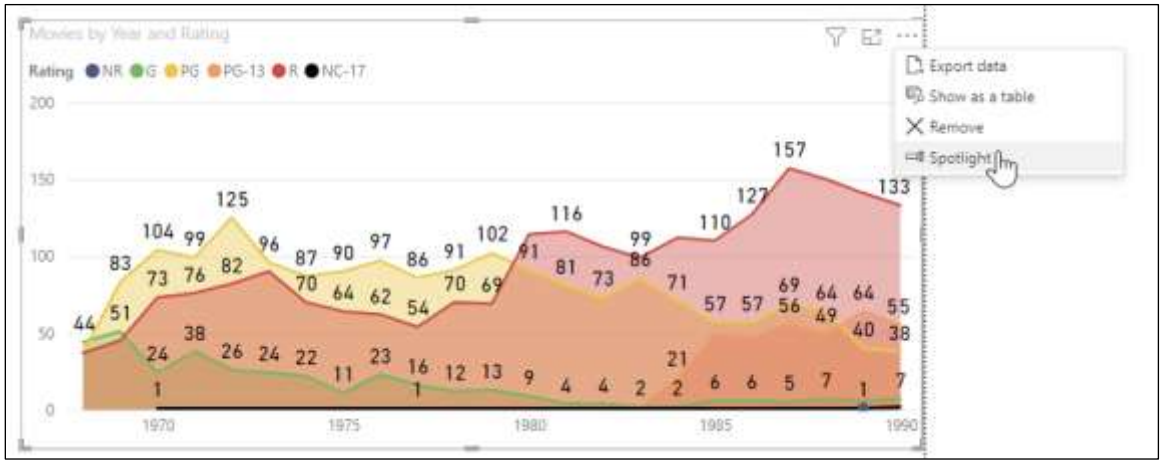
F. Export data from a visualization

G. Sort a visualization

H. Apply drill downs

A. Spotlights

In the theater when a director wants the audience to focus on the hero and heroine, , all the lights on the stage are dimmed and a spotlight is focused on the heroic couple. Spotlights in Power BI can also be used to draw viewers attention to a specific visual. In this exercise you will use the spotlight feature in the movies report.



1. **Open** the **Movies** report and select the first page, "Column Lines and Area Charts."

2. Click on the chart in the bottom right corner.

3. Click the three dots in the top right corner of the visualization.

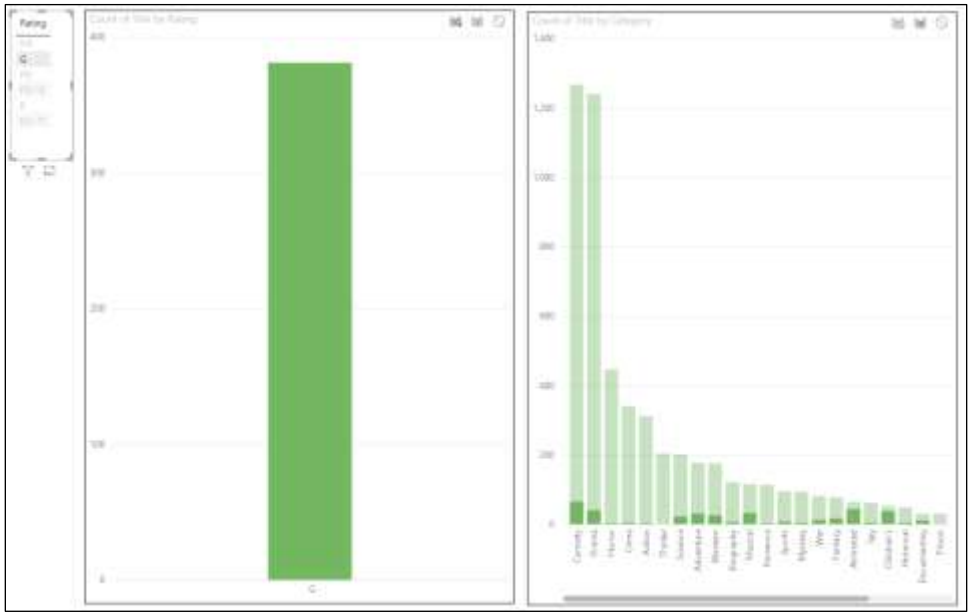
4. Choose Spotlight.

5. Notice the other charts are dimmed, thus attracting viewers to the one chart.

   Note: The spotlight is only temporary. Once you click somewhere else, the spotlight is removed.

B. Edit Interactions

As a designer of Power BI reports, you can control the interactions between various visualizations. Specifically, you can decide if a visualization should allow another to filter it, highlight it, or have no effect on it. This is done through the Edit Interactions feature on the ribbon's Format menu.



1. Add a new page to the Movies report. Name the page **2 Column Charts**.

2. Add a table for the list of ratings.

3. Create a column chart with the following properties:

Property	Setting
Axis	Rating
Values	Count of Title

4. Create a second column chart with the following properties:

Property	Setting
Axis	Category
Values	Count of Title

5. Select the table visualization.

6. From the ribbon select **Format**, **Interactions**, **Edit Interactions** to turn on the Edit interactions mode.

   Notice the column charts now display three icons that control the interactions between the table visualization and the other visualization.

Icon	Name	Meaning
	Filter	Data in this visualization can be filtered by the table visualization.
	Highlight	Data in this visualization can be highlighted by the table visualization.
	None	This visualization will not be affected by a selection in the table visualization.

   These three icons are like three radio buttons, so when you click one icon ON for a chart, you have simultaneously set the other two icons OFF.

7. For the Rating column chart click the filter icon  .

8. For the Category column chart click the filter icon  .

9. From the ribbon select **Format**, **Interactions**, **Edit Interactions** to turn off the Edit interactions mode.

10. In the Rating table click G.

    Notice that the rating chart only displays the G rated bar. Notice the Category chart highlights the portion of the bars that represent the G rated movies.

11. In the Rating table click G.

    Notice both charts return to normal.

12. Experiment with other selections in the rating table and see the results.

13. Save the report.

C. Filters and slicers affecting this visualization

Sometimes you need to know which slicers and other objects are affecting a visualization. The **Filters and slicers affecting this visualization** tool can provide this information. In this exercise you will add a slicer to the page and then use the **Filters and slicers affecting this visualization** tool.

1. Add a slicer based on the Category field.

2. In the slicer select Comedy.

3. In the Rating table click PG.

4. In the rating column chart hover the mouse over the funnel in the top right corner of the chart.

   Notice a tip window appears explaining that this chart is filtered by the Comedy Category and only includes the PG ratings.

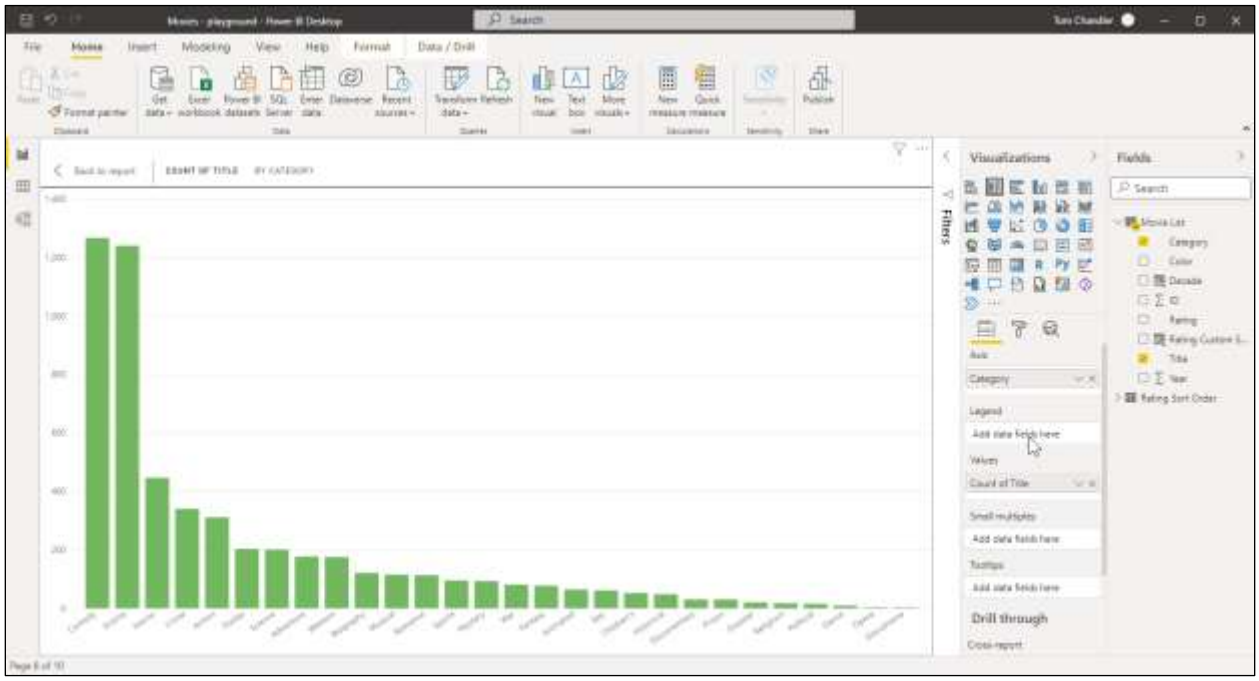
5. Clear the slicer.

6. Reselect PG in the rating table, thus enabling all the ratings.

7. Save your report.

D. Focus Mode

The Power BI focus mode allows you to zoom in and focus on a visualization.



1. Hover the mouse over the Category column chart.

2. Click the focus mode icon .

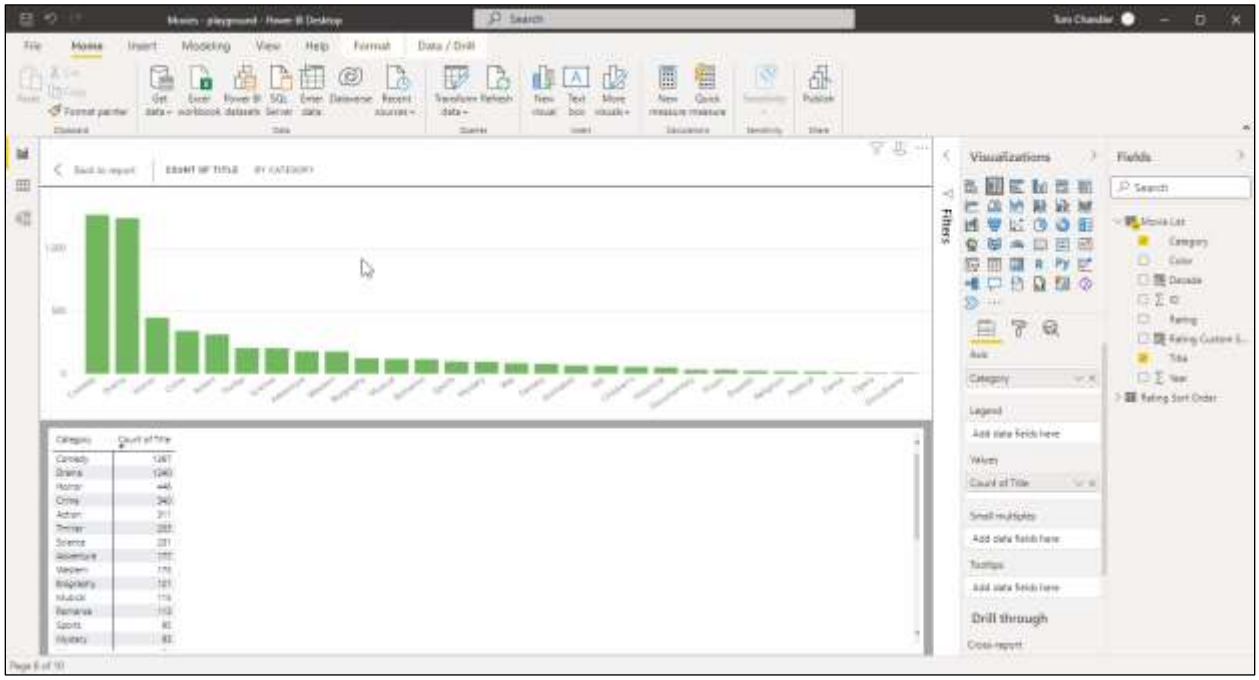
   Notice the screen now zooms in on, and only shows, the Category column chart; all other charts are temporarily hidden.

   Notice the Filters, Visualizations, and Fields panes are still visible. While in Focus mode you can still modify the visualization as needed.

3. In the top left corner click the **Back to report** button to exit focus mode.

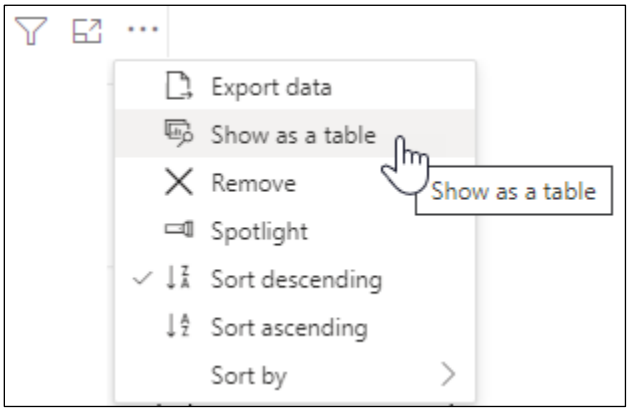
E. Show as a table

The Show as a table feature is similar to focus mode, but it also shows you the data in a table.



1. Again, hover the mouse over the Category column chart.

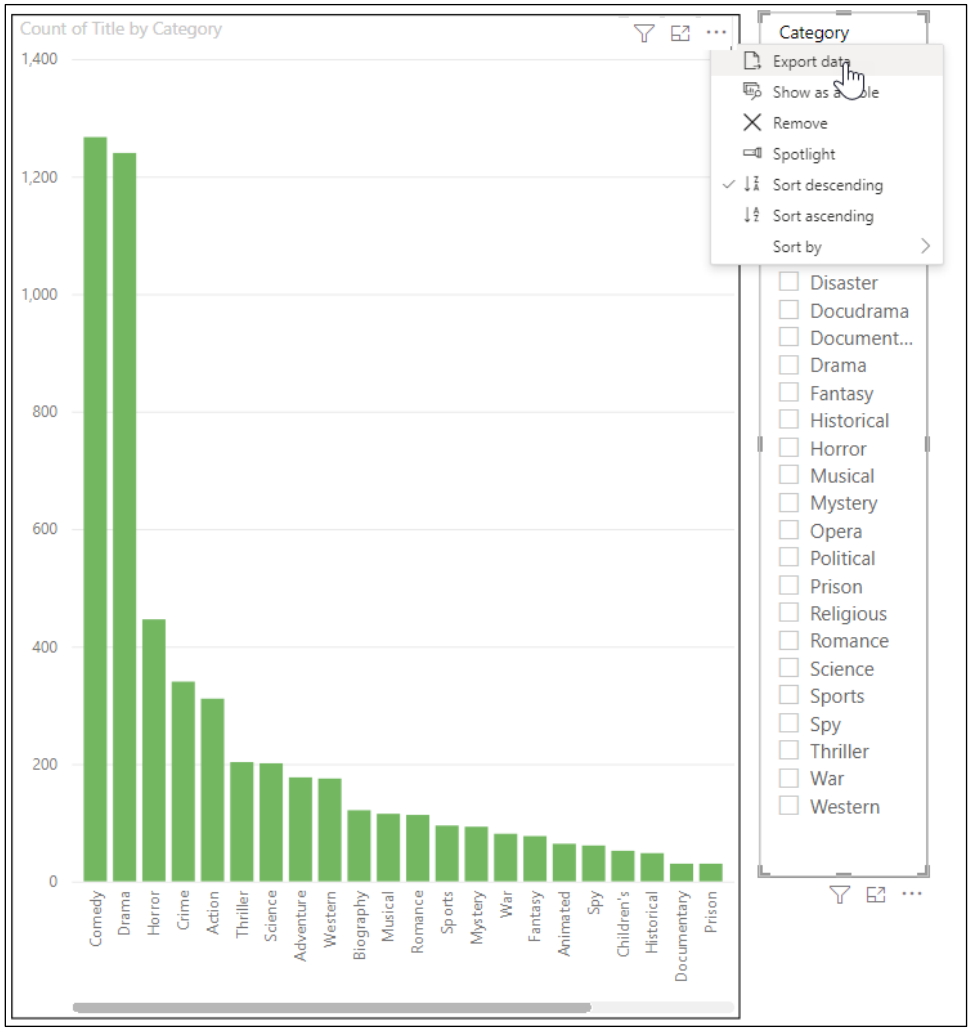
2. In the top right corner click the three dots, then click **Show as a table**.



3. Click the **Back to report** button to exit the Show as a table feature.

F. Export Data

Power BI gives you the ability to export the summary data from which a visualization is made. This is the same data that appears when you use the Show as a table feature. The data will be saved as a CSV file which can then be opened in Excel and many other programs.



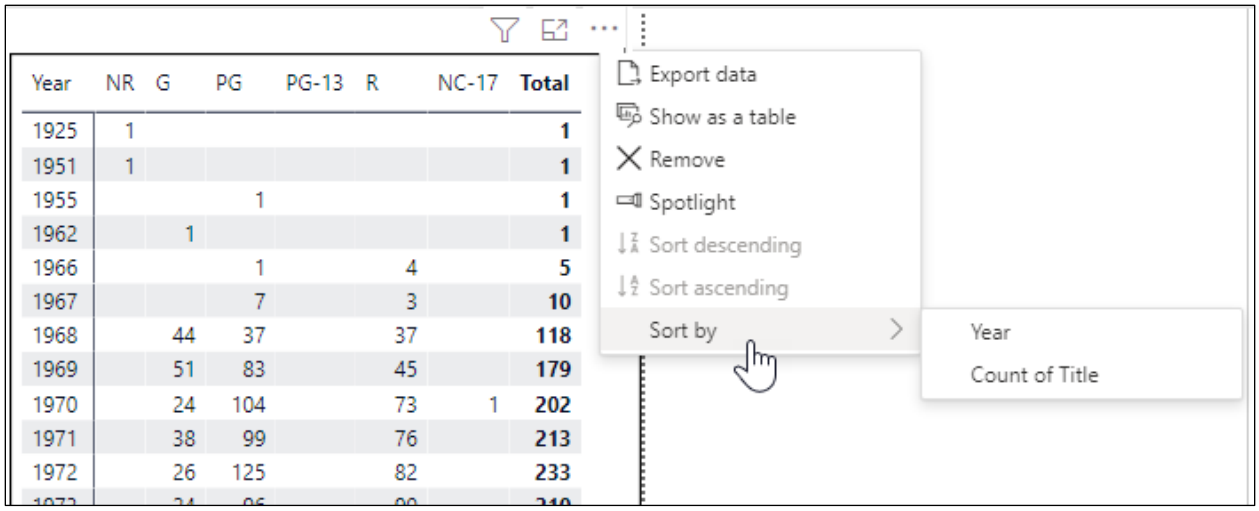
1. Again, hover the mouse over the Category column chart.

2. In the top right corner, click the three dots, then click **Export data**.

3. When prompted **choose a folder**, and **enter a filename**, then click **Save**.

G. Sorting

Depending on the visualization and its data, Power BI gives you options for how the data should be sorted. In this exercise you will experiment with sorting a visualization on the Movies per year page.



1. Move to the **Movies per year** page you created earlier.

2. Hove the mouse over the top right corner, click the **three dots**, and choose **Sort by**.

   Here you can see the fields from which this report can be sorted.

3. Choose the count of title field.

   Notice the report now sorts the results by the total column instead of the year column.

4. Again, hover the mouse over the top right corner, click the **three dots**, and choose **Sort Ascending**.

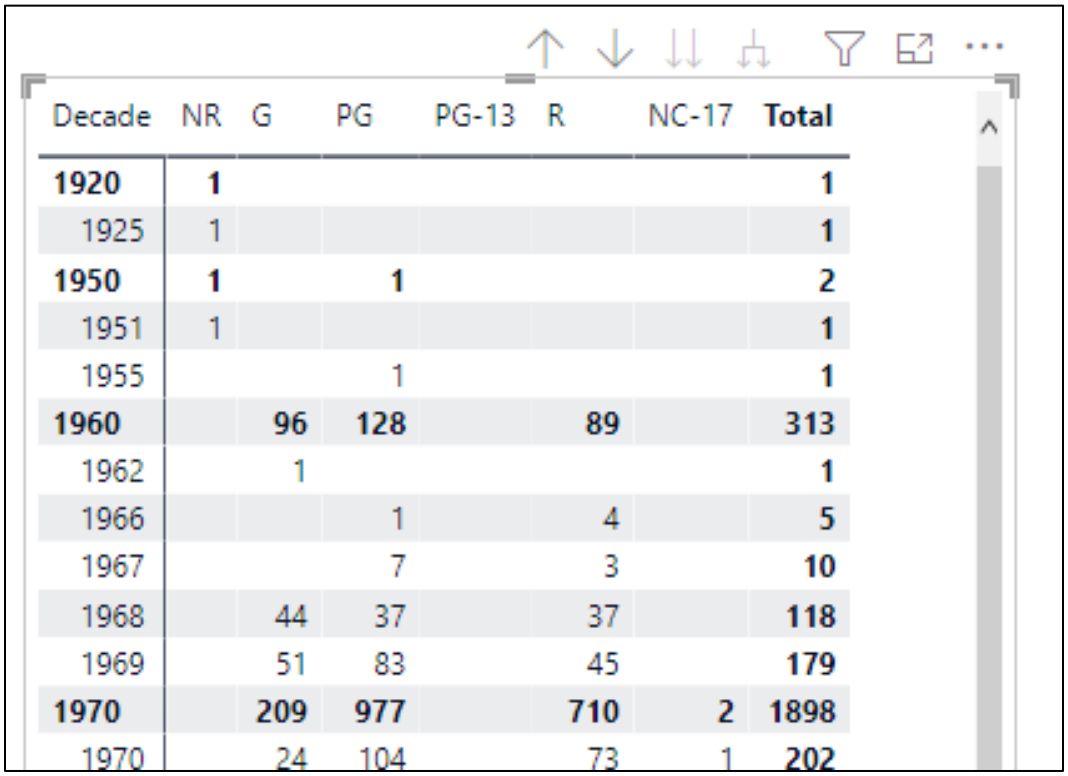
   The report now displays the records with the least number of total movies first.

5. Practice: Sort the visualization by year in descending order.

H. Drill down

Power BI gives you the ability to create a report with a high-level of detail, and then "drill down" to see a lower-level details.

In this exercise you will create a new column named decade. It will be based on the year the movie was made. You will then create a matrix report based on the decade and then see how to drill down to see the report per year.



Decade	NR	G	PG	PG-13	R	NC-17	Total
1920	**1**						**1**
1925	1						**1**
1950	**1**		**1**				**2**
1951	1						**1**
1955			1				**1**
1960		**96**	**128**		**89**		**313**
1962		1					**1**
1966			1		4		**5**
1967			7		3		**10**
1968		44	37		37		**118**
1969		51	83		45		**179**
1970		**209**	**977**		**710**	**2**	**1898**
1970		24	104		73	1	**202**

Create a calculated Decade field

1. From the ribbon select **Modeling**, **Calculations**, **New Column**.

2. Enter the following formula:

```
Decade = INT([Year]/10) * 10
```

Create a matrix visualization

1. Add a new page to the report named "Movies by decade".

2. Resize the matrix so it takes up most of the page.

3. Add a matrix visualization with the following properties.

Property	Setting
Rows	Decade
Columns	Rating
Values	Count of Title

4. Notice the bottom of the visualization includes these hierarchical icons.

Icon	Name
	Drill up
	Drill down
	Go to the next level in the hierarchy
	Expand all down one level in the hierarchy

5. Double click 1960.

   Notice nothing happens.

6. Click the drill down button  .

   This turns on the ability to drill down. Notice the icon has changed to  .

7. Double click 1960.

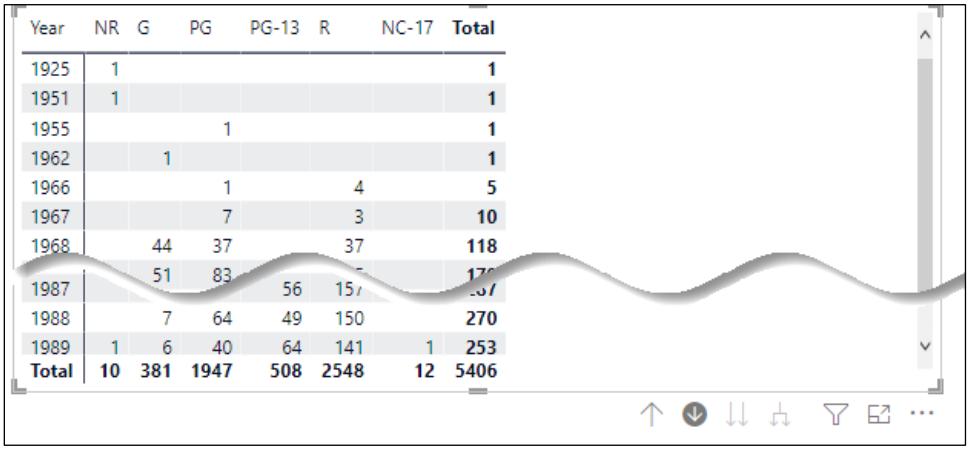
   Notice the screen now displays the 1960 decade with details for each year that has a data.

Decade	G	PG	R	Total
1960	**96**	**128**	**89**	**313**
1962	1			**1**
1966		1	4	**5**
1967		7	3	**10**
1968	44	37	37	**118**
1969	51	83	45	**179**
Total	**96**	**128**	**89**	**313**

8. Click the drill up icon to return to the decade only report.

9. Click the **Go to the next level in the hierarchy** button.

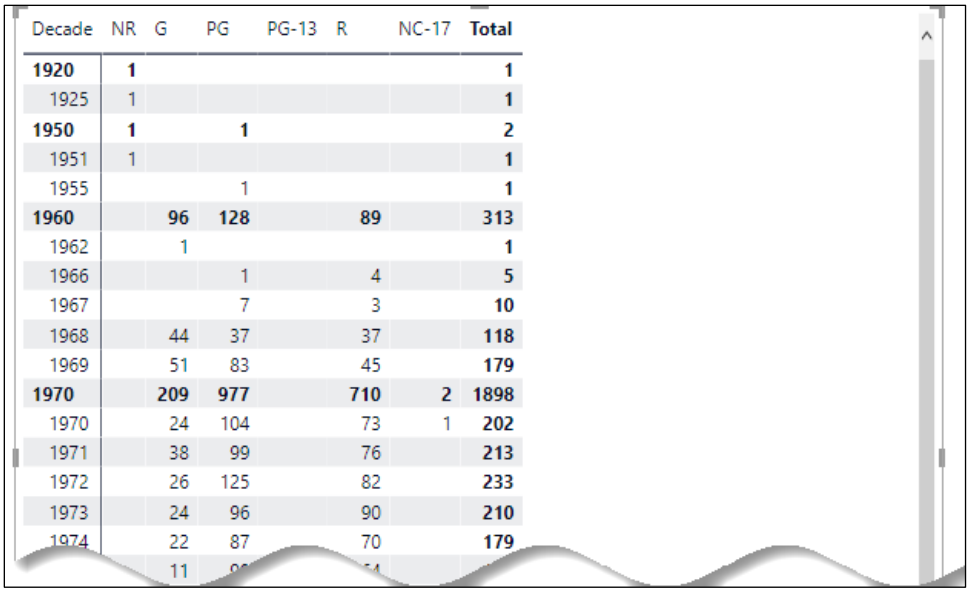
   Notice the screen now displays the report by year instead of decade.



10. Click the drill up icon to return to the decade report.

11. Click the **Expand all down one level in the hierarchy** button.

    Notice the report displays both decade and year data.



Decade	NR	G	PG	PG-13	R	NC-17	Total
1920	**1**						**1**
1925	1						**1**
1950	**1**		**1**				**2**
1951	1						**1**
1955			1				**1**
1960		**96**	**128**		**89**		**313**
1962		1					**1**
1966			1		4		**5**
1967			7		3		**10**
1968		44	37		37		**118**
1969		51	83		45		**179**
1970		**209**	**977**		**710**	**2**	**1898**
1970		24	104		73	1	**202**
1971		38	99		76		**213**
1972		26	125		82		**233**
1973		24	96		90		**210**
1974		22	87		70		**179**
		11					

12. Click the drill up icon to return to the decade only report.

13. Save your report.

6. Creation Tools

Complete this unit and you'll be able to:

A. Change the size of a page

B. Change the page view

C. Switch to a mobile layout

D. Apply themes

E. Easily align objects

F. Lock objects

G. Use the selection pane

A. Page Size

By default, Power BI assumes you are using a wide monitor with a 16:9 aspect ratio. If necessary you can change the page size as follows:

1. Open your report.
2. Click outside a visualization so your cursor is on the page.
3. From the visualizations pane, click the format properties button.
4. Expand the Page size section.
5. Click the type drop down and select the desired page size.

   Notice the drop down includes multiple options which are explained below:

Option	Width x Height in Pixels	Comments
4:3	920 x 720	This was the original standard size used by televisions in the 1940s.
16:9	1280 x 720	To compete with televisions, movie theaters used this wider screen size. Today's wide screen monitors are 1280 x 720 or 1920 x 1080, both of which use the 16:9 aspect ratio. Most computers today use this standard. It is therefore the default in Power BI.
Letter	816 x 1056	Sized for standard 8 ½" x 11" paper.
Tooltip	320 x 240	Size for a popup tooltip box.

B. Page View

Page size and page view are different. By analogy, page size and page view are somewhat like Microsoft Word's paper size and zoom. If you want to change your page view (zoom) do the following:

1. From the ribbon select **View**, **Scale to fit**, **Page View**, then select one of the following options:

Option	Result
Fit to page	Zoom is changed so the whole page (width and height) fits in the application window.
Fit to width	Zoom is changed so the width fits in the application window. You may have to scroll down to see the rest of the page.
Actual Size	Zoom is changed to 100% so, no matter the size of the application window, the image will be the same size. If you have a small application window then you will most likely need to scroll in order to see the rest of the image. In a large application window scrolling may not be needed.

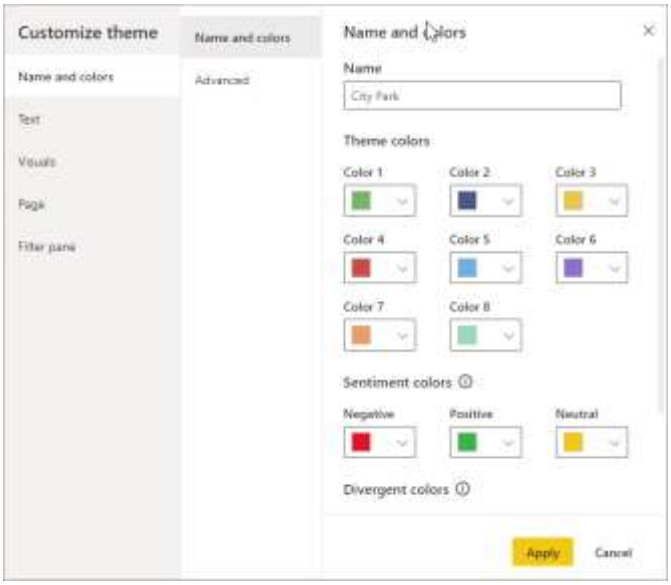
C. Themes

As we saw briefly in the exercise about ribbon charts, you can quickly change colors by applying a report theme. The ribbon's **View** menu lets you pick one of many prebuilt themes.

To select a theme, click the drop down and choose the desired theme.

By selecting **Customize current theme** you can do the following:

Name and colors	Set the name of the theme, and the default colors it uses.
Text	Change the default font, font size, and font color of various properties.
Visuals	Set the defaults settings for visuals including background color, border, header and tool tips
Page	Change the wallpaper and page background color.
Filter Pane	Set the colors and fonts of the filter pane, available filter cards, and the applied filter cards.



After you apply your changes, you can optionally save your theme to a JSON file using the **Save current theme** option.

The **Browse for theme** option allows you to open a JSON file that has the theme you or someone else created.

The **Theme gallery** option will open a webpage where Microsoft has multiple themes that you can preview and download.

If you would like to learn more about JSON files and creating a theme, click the **How to create a theme** option. It will take you to a detailed Microsoft web page about using and creating themes.

D. Aligning and grouping objects

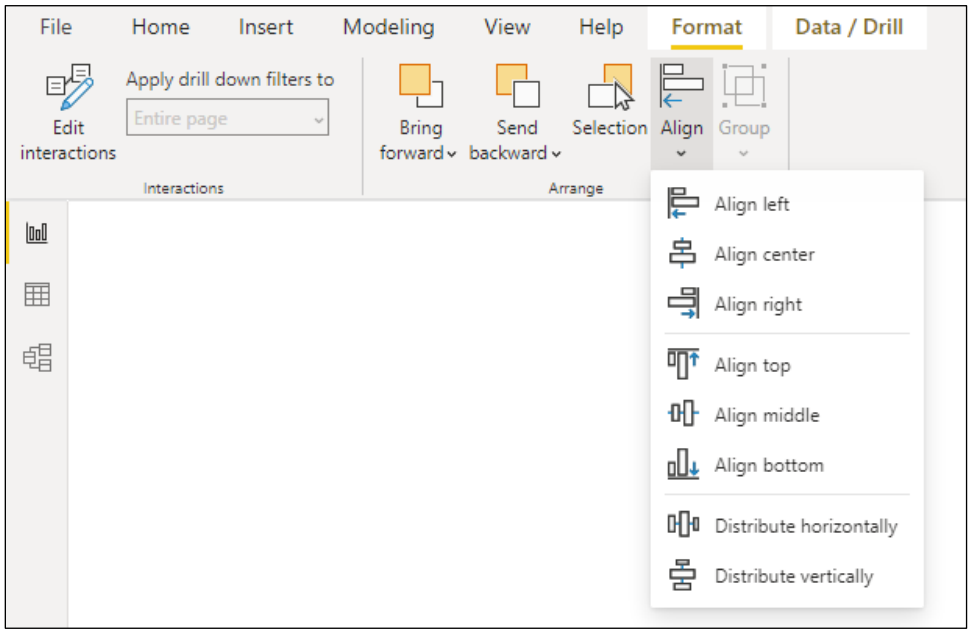
The ribbon's **View** menu and the ribbon's **Format** menu includes options that can help you align objects on your screen:

View menu

The **View, Gridlines** option will display gridlines that help you visually align your objects on the screen.

The **View, Snap to Grid** option can help you align objects on the page because, when checked, each object you move on the screen will have to line up with the dots of the grid. The snapping even take place if you choose to not show the gridlines.

Format Menu



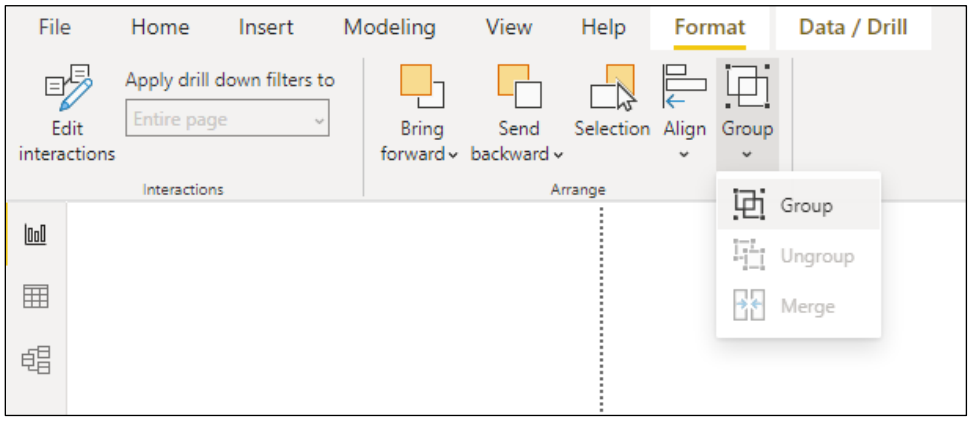
The format menu is displayed when one or more objects are selected.

When one object is selected, then the **Format, Align** button will let you align objects on the left, center, right, top, middle, or bottom or the page.

When multiple objects are selected, then the **Format, Align** button will let you align objects relative to each other.

When multiple objects are selected, the **Format, Align** button also lets you distribute the objects evenly on either the horizontal or vertical axis.

Grouping objects



You can use the **Format** menu's **Group** button to **Group** multiple objects together so they will act as one object. If you need to ungroup previously grouped objects, you can do so with **Format**, **Group**, **Ungroup**.

E. Lock Objects

Once you have placed an object right where you want it, you may want to lock it in place so it cannot be moved. This can be done with the ribbon's **View**, **Lock objects** check box. Once locked, the objects can be modified via the format pane, but they cannot be moved with the mouse.

F. Selection pane

The selection pane is a pane that displays the various objects (visuals, shapes, text boxes, etc.) that belong to a page. With the selection pane you can hide or show each object, change layer order (i.e. move one object in front of another), and change tab order.

View the selection pane

To view the selection pane, from the ribbon choose **View**, **Show pages**, **Selection**.

Hide an object

1. Select the object you wish to hide.
2. In the selection pane, click the eyeball next to the object.

Unhide an object

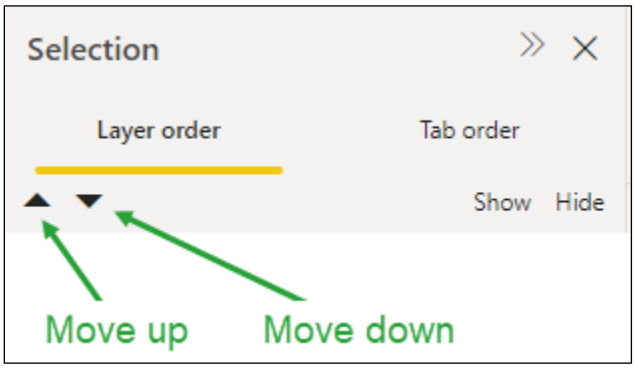
1. In the selection pane, find the object you wish to unhide.
2. In the selection pane, click the eyeball next to the object.

Change layer order

Power BI offers two methods for changing layer order.

Change layer order with the selection pane

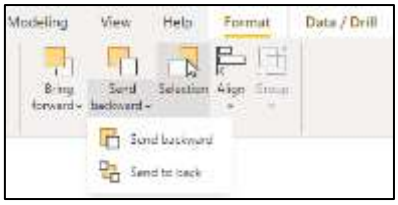
1. Select the desired object.
2. In the selection pane, click the black up arrow to move it toward the top of the list, or click the black down arrow to move it toward the bottom of the list.



Change layer with the format menu

1. Select the desired object.

2. Select one of the following the ribbon's Format menu:



Menu option	Menu path	Result
Bring forward	Format, Bring Forward, Bring Forward	Moves the object in front of another
Bring to front	Format, Bring Forward, Bring to front	Moves the object in front of all other objects
Send backward	Format, Send backward, Send backward	Moves the object behind another
Send to back	Format, Send backward, Send to back	Moves the object behind all other objects

Change tab order

When viewing a page in the Power BI service, many viewers prefer to use the keyboard's tab key to move from one visual to another. The default tab order is defined by the order in which you placed the objects on the page. Using the selection pane, you can change the tab order.

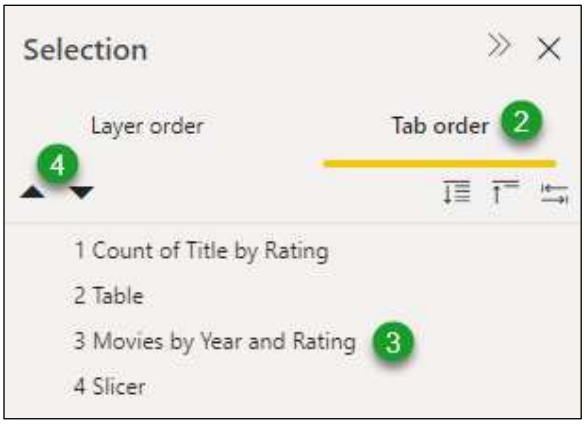
1. Display the selection pane (from the ribbon choose **View**, **Show pages**, **Selection**).

2. Click the Tab order link at the top of the selection pane.

   Notice each object is listed with a number to its left.

3. Select the desired object.

4. Click the black up or down arrow to promote or demote the object's tab order.



G. Mobile layout

You can quickly convert a desktop / browser-based Power BI report into a set of mobile-friendly visualizations using the Mobile layout view. This view allows you to drag and drop visualizations from your report page onto the image of a mobile phone. From there you can resize and reposition the visuals as desired.

1. Open your Power BI report.
2. From the ribbon select **View**, **Mobile**, **Mobile layout**.

   The screen will show you the image of a mobile phone, and a pane named Page visuals.
3. Select a visual from the pane and drag it onto the image of the mobile phone.
4. Click inside the added visual and move it to the desired location.
5. As needed, resize the visual using the size handles.
6. As needed, scroll up or down using the scroll bar on the right side of the mobile phone's image.
7. Repeat steps 3 through 6 for each visual.
8. From the ribbon select **View**, **Mobile**, **Mobile layout** to return to your regular report view.

7. DAX Calculations and Measures

Complete this unit and you'll be able to:

A. Define DAX

B. Create calculations

C. Create measures

A. What is DAX?

DAX is an acronym for Data Analysis Expressions.

DAX is a formula language

DAX is a formula language composed of a library of functions and operators that can be used to build formulas. DAX is used in multiple Microsoft Products including Power BI, Analysis Services and Power Pivot.[2]

DAX is not a programming language

DAX is not a programming language.[3] To understand the difference between a formula language and a programming language, consider the difference between Excel macros and Excel formulas and functions. Excel's VBA macro language is an object-oriented programming language that can sequentially execute commands that will manipulate your spreadsheet. In contrast, Excel's formulas and functions perform calculations that affect the cells or group of cells to which they are applied.

What can I do with DAX?

DAX allows you to add three types of calculations to your data model:

- Calculated tables
- Calculated columns
- Measures

DAX Calculations

Whether it is a calculated table, column, or measure the calculation has three parts: the name of the calculation, an equal sign, and the formula.

```
Calculation Name = Dax formula
```

The DAX formula can include the following:

- DAX functions
- Operators
- References to model objects
- Variables
- Constants
- White space

[2] See https://docs.microsoft.com/en-us/dax/
[3] See https://en.wikipedia.org/wiki/Data_Analysis_Expressions

Where can I learn more about DAX?

You can learn more about DAX from multiple resources including:

- Microsoft's Data Analysis Expressions (DAX) Reference found at https://docs.microsoft.com/en-us/dax/.
- Microsoft's DAX function Reference gives detail information about over 250 DAX functions. For each function it shares its syntax, parameters, return values, and examples. It can be viewed at https://docs.microsoft.com/en-us/dax/dax-function-reference.
- The DAX Guide is a project of SQLBI which includes a reference to DAX functions, operators, statements, data types. It is found at https://dax.guide/.
- Use DAX in Power BI Desktop is a 2 hour 27 minute, seven-module learning path provided by Microsoft. It can be found at https://docs.microsoft.com/en-us/learn/paths/dax-power-bi/.
- Introducing DAX Video Course is a free 150 minute "introductory video course about the DAX language" provided by SQLBI. You can learn more at https://www.sqlbi.com/p/introducing-dax-video-course/.

B. DAX Functions

Like functions in many other programs, each DAX function has a function name followed by parenthesis. Most functions have arguments, but some do not. Even when there are no arguments, the parentheses are required. Some arguments are required, and others are optional. Either way, the arguments are always listed inside the parenthesis, and are separated by commas.

```
FunctionName(Argument1, Argument2, …)
```

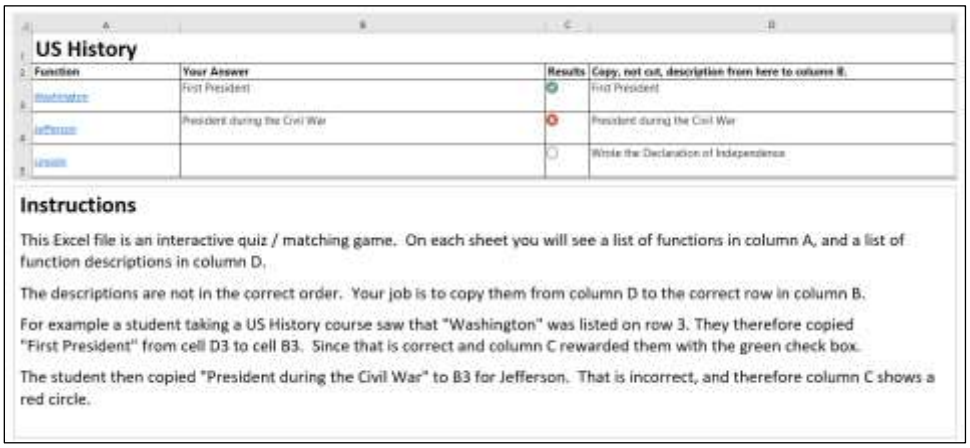
To help decrease the DAX learning curve, Microsoft intentionally used many of the same function names that are found in Excel.

Exercise – Match that Function

To help you learn and familiarize yourself with many of the DAX functions, you will perform the following Match that Function exercise.

1. From your exercise files folder, open Match that Function.xlsx.

2. Read the instructions on the Instructions sheet.



3. Select the Aggregate sheet, and for each function name copy the description from column D to the correct row in column B.

   Note: DO NOT CUT AND PASTE THE DESCRIPTION.
   Cutting (CTRL X) and pasting will ruin the conditional formatting in column C.

4. Repeat the process for the other sheets.

C. DAX Operators

As listed below, DAX uses many of the same mathematical, logical, and text operators as Excel.

Operator	Mathematical Operation
+	Add
-	Subtract
*	Multiply
/	Divide
^	Raise to the power of an exponent

Operator	Textual Operation
&	Concatenate

Operator	Comparison Operation
=	Equals
==	Exactly equals
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

Operator	Logical Operation
&&	AND
||	OR
IN	In a list
NOT	Reverses the logical operation

D. DAX Variables

When you create a DAX formula you can create variables that store the result of an expression. You can then use that variable name in the formula.

To declare a variable, use the keyword VAR followed by the variable name, an equal sign, and the variables expression (value or calculation).

```
VAR <name> = <expression>
```

When a variable is declared in a DAX calculation, the calculation must explicitly state the value that is to be returned.

For example, the following DAX calculation creates a measure named Three. It creates two variables named **One** and **Two**. It then uses the keyword **Return** to tell the computer that the results of the measure should be the value of the One variable plus the value of the Two variable.

```
1  Three =
2      VAR One = 1
3      VAR Two = 2
4      Return
5          One + Two
```

The variable name can use upper- or lower-case letters, and digits between 0 and 9. No spaces are allowed. The variable name must not start with a digit. You may also use a _ (single underscore) or __ (double underscore) as variable prefix.

You may not use reserved keywords (words like VAR and Return that show up in blue) for your variable name. Names of existing tables are also not allowed for variable names.

E. Measures

In this exercise you will learn about three different types of measures: Implicit Measures, Explicit Measures, and Quick Measures. You will see that an Implicit Measure is automatically calculated based on a field's default summary function, and that an Explicit Measure is based on a formula you create. You will also see that the Quick Measures feature is a way to get Power BI to help you create the formula for an Explicit Measure.

1. If necessary, open the AdventureWorks.pbix report.

2. Create a new page and name it "**Measures**".

Add an Implicit Measure

1. Add a **Matrix** visualization.

2. From the Sales table add the Extended Price field to the Values well.

3. In the values well, click the Extended Price drop down and notice Sum is selected.

   The sum of the Extended Price is an *implicit* measure because you did not explicitly request the sum function.

Change the default measure

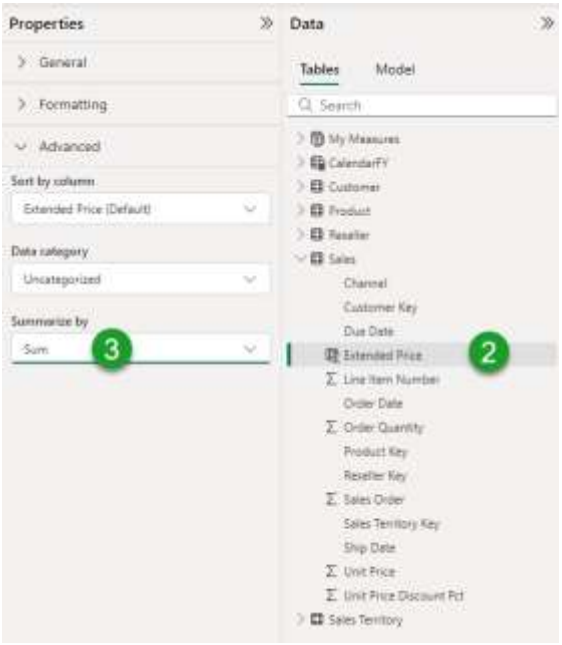
1. Switch to the model view.

2. In the fields pane, expand the **Sales** table and select the **Extended Price** field.

3. If necessary, expand the **Properties pane**, then **scroll down** and **expand** the **Advanced** section.

   Notice the **Summarize by** property is set to **Sum**.

   This setting determined our implicit calculation.



4. Click the Summarized by drop down and change the default to **Max**.

5. Return to the **Report** view.

   Notice the label now explicitly states it is the **Sum of Extended Price**.

6. In the Values well, click the x to **remove** the **Extended Price**.

7. **Add** the **Extended Price** again, and then click the drop down.

   Notice the field is now set to return the Maximum value.

8. Return to the **Model** view and change the default back to **Sum**.

Examine an Explicit Measure

The measure we did in earlier chapters are explicit measures.

1. Return to the **Report** view.

2. Click on the **My Measures**, **Totals Sales** measure.

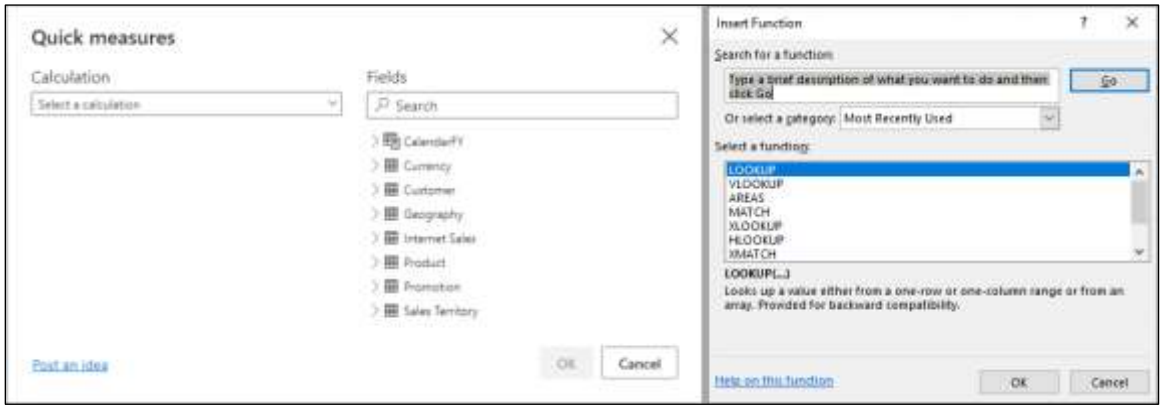
3. Notice the following formula explicitly tells Power BI that you want the sum of the Sales table's Extended Price column.

```
Total Sales = Sum(Sales[Extended Price])
```

4. Add the **Total Sales** measure to the **values well**.

Create a Quick Measure

Creating a quick measure in Power BI is somewhat like inserting a function into Excel with the Insert Function dialog box. As you will see the Calculation drop down gives you many measures that you can create with this wizard like dialog box.



1. Right click on the **My Measures** table and choose **New quick measure**.

2. Click the **Select a calculation** drop down arrow and scroll down.

   Notice that Power BI gives you the following categories.
   - Aggregates per category
   - Filters
   - Time Intelligence
   - Totals
   - Mathematical operations
   - Text

3. In the Time Intelligence category, choose the **Month to-date total**.

4. Drag the **Sales, Extended Price** field to the **Base value** well.

5. **Expand** the **CalendarFY** table and drag the **Date** field to the Date well.

6. Click **Add**.

7. Select the new **Extended Price MTD** field and on the Measure Tools ribbon change the format to **Currency** with **2 decimal places**.

8. Add the **Extended Price MTD** field to the values well of the matrix.

9. From the **CalendarFY** table, drag the **FiscalYear Hierarchy** to the **Rows well**.

The matrix now shows years.

10. Click the **Expand all down one level in the hierarchy** button .

The matrix now shows quarters.

11. Click the **Expand all down one level in the hierarchy** button .

The matrix now shows months.

12. Click the **Expand all down one level in the hierarchy** button .

The matrix now shows days.

Notice the Extended Price MTD total is increasing each day, and then starts again from zero the following month.

Fiscal Year	Max of Extended Price	Total Sales	Extended Price MTD
29	6,998.32	$116,319	$1,324,774.27
30	12,149.94	$71,222	$1,395,995.83
31	3,578.27	$27,361	$1,423,357.20
Sep	**21,532.39**	**$2,524,495**	**$2,524,494.67**
1	8,159.96	$38,547	$38,546.71
2	3,578.27	$20,717	$59,263.55
3	10,124.95	$107,653	$166,916.76
4	4,079.98	$16,649	$183,565.57