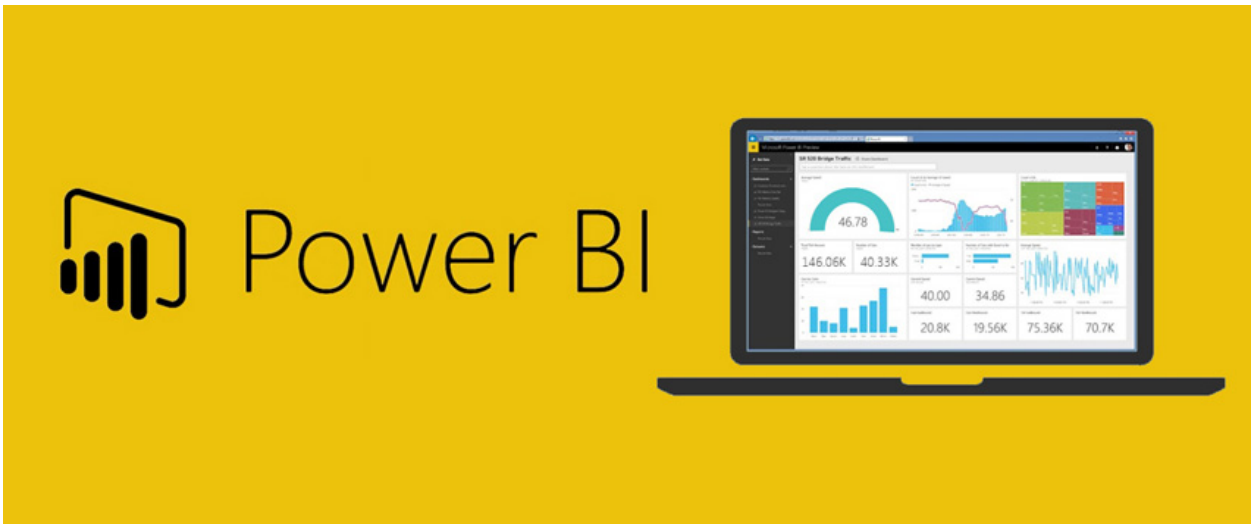
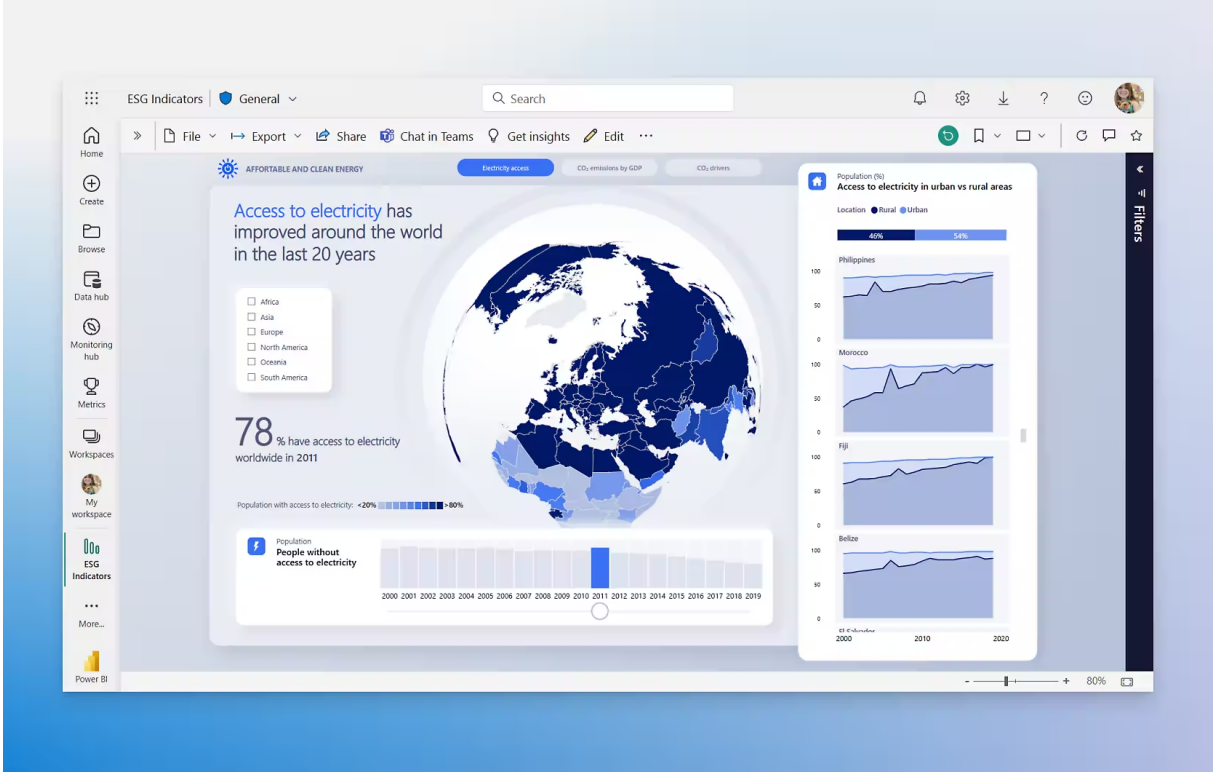


Empower Your Data: Power BI Code Camp for Beginners

By Wayne Dayata and Sabrinah Yap | May 25, 2024

Introduction

Power BI is a powerful tool used for generating business insights and aiding decision-making through interactive visualizations and business intelligence capabilities. It is instrumental in transforming raw data into meaningful information, enabling businesses to make strategic decisions based on data-driven insights. Whether you're a business owner, a data analyst, or someone interested in data visualization, this workshop will provide you with the fundamental skills needed to navigate and utilize Power BI effectively.

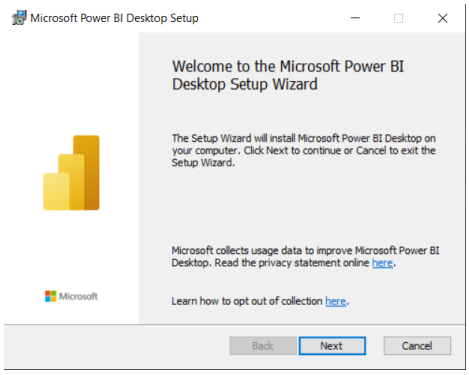


Key features of Power BI:

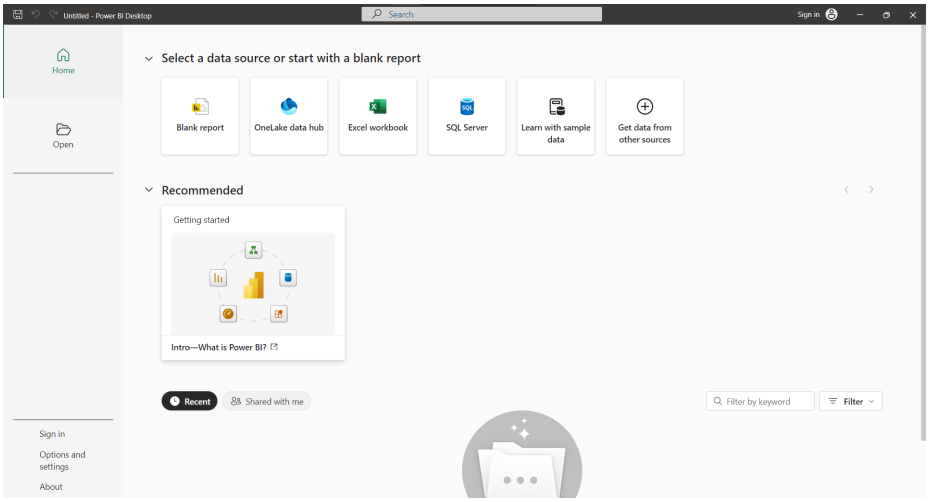
- Quick set up compared to traditional BI or other tools
- Interactive visualizations
- Supports different data sources
- Ability to push to web (app.powerbi.com)
- Scalable and flexible
- Accessible in any platform and device

Setup and Installation:

1. Download the resource files for the workshop here: https://bit.ly/PowerBIWorkshopDatasets
2. Download the Power BI Desktop application. You may use the web version but that will require you to set up an account and start a free trial for MS Fabric.



2. Once installation is done, launch the application. The initial display is as follows:



Great, we can now proceed and perform some magic on our data!

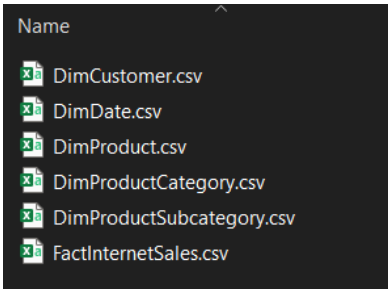
▼ Part 1: Basics

Objectives:

1. Get the data to a data source
2. Model the data through relations
3. Report on the data through visualizations

▼ Step 0: Prepare the data

Open the Part 1 folder, which should contain six (6) CSV files inside it.

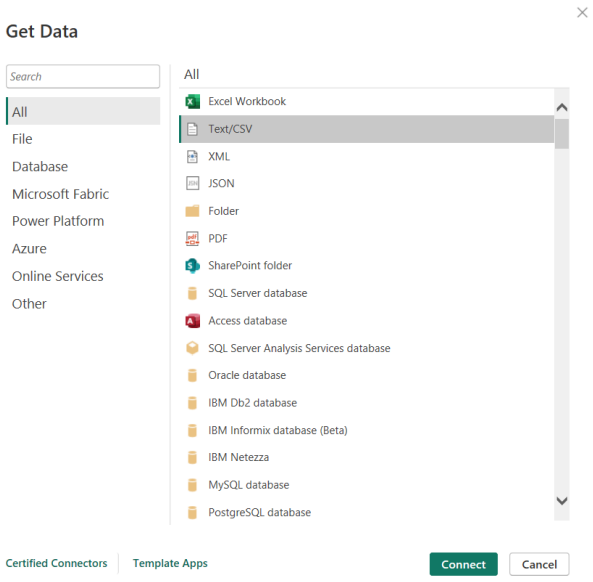


💡 **Dimension tables** contain descriptive attributes that are typically textual fields relevant to the business.

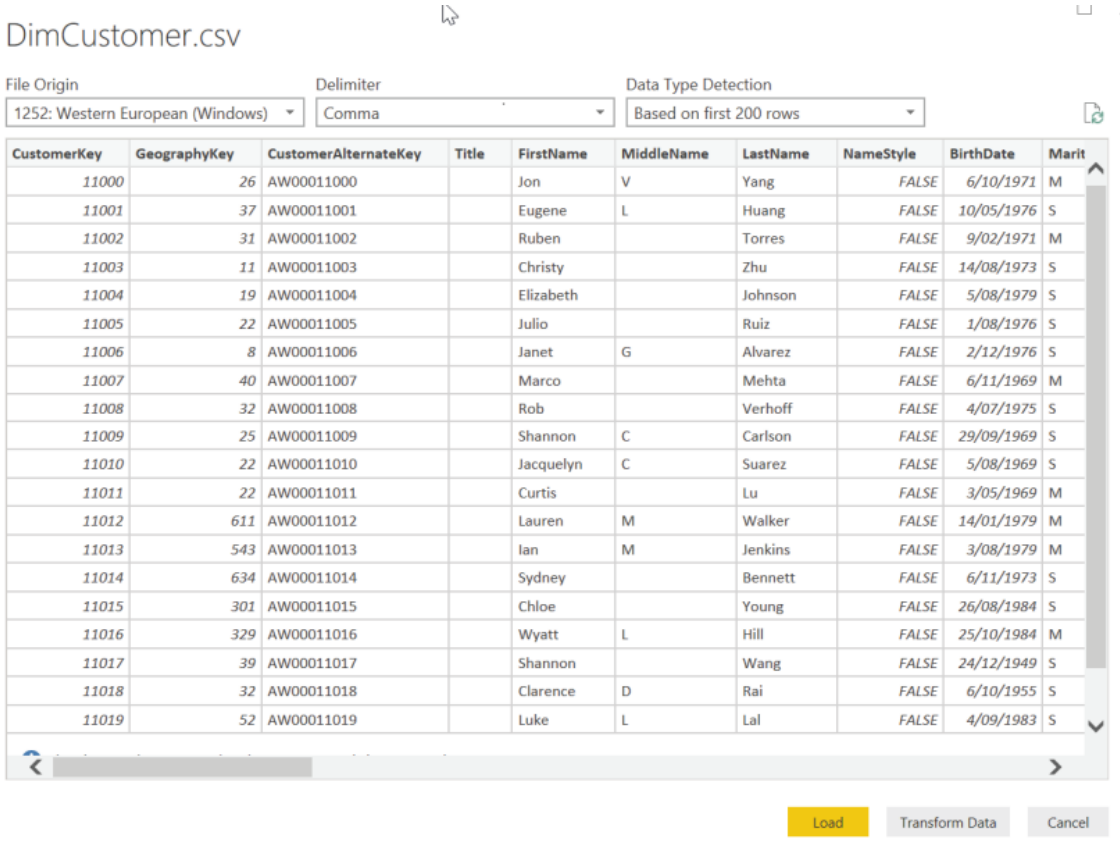
Fact tables, on the other hand, contain measurable, quantitative data that can be analyzed.

▼ Step 1: Load the data

1. In the initial screen, select **Get data from other sources**.
2. Select **Text/CSV**, and then click **Connect.**



3. Browse to the unzipped data files you save earlier, select the first file **DimCustomer.csv**, and click **Open**. the window below then appears. you may browse through and study the columns.



CustomerKey	GeographyKey	CustomerAlternateKey	Title	FirstName	MiddleName	LastName	NameStyle	BirthDate	Marit
11000	26	AW00011000		Jon	V	Yang	FALSE	6/10/1971	M
11001	37	AW00011001		Eugene	L	Huang	FALSE	10/05/1976	S
11002	31	AW00011002		Ruben		Torres	FALSE	9/02/1971	M
11003	11	AW00011003		Christy		Zhu	FALSE	14/08/1973	S
11004	19	AW00011004		Elizabeth		Johnson	FALSE	5/08/1979	S
11005	22	AW00011005		Julio		Ruiz	FALSE	1/08/1976	S
11006	8	AW00011006		Janet	G	Alvarez	FALSE	2/12/1976	S
11007	40	AW00011007		Marco		Mehta	FALSE	6/11/1969	M
11008	32	AW00011008		Rob		Verhoff	FALSE	4/07/1975	S
11009	25	AW00011009		Shannon	C	Carlson	FALSE	29/09/1969	S
11010	22	AW00011010		Jacquelyn	C	Suarez	FALSE	5/08/1969	S
11011	22	AW00011011		Curtis		Lu	FALSE	3/05/1969	M
11012	611	AW00011012		Lauren	M	Walker	FALSE	14/01/1979	M
11013	543	AW00011013		Ian	M	Jenkins	FALSE	3/08/1979	M
11014	634	AW00011014		Sydney		Bennett	FALSE	6/11/1973	S
11015	301	AW00011015		Chloe		Young	FALSE	26/08/1984	S
11016	329	AW00011016		Wyatt	L	Hill	FALSE	25/10/1984	M
11017	39	AW00011017		Shannon		Wang	FALSE	24/12/1949	S
11018	32	AW00011018		Clarence	D	Rai	FALSE	6/10/1955	S
11019	52	AW00011019		Luke	L	Lal	FALSE	4/09/1983	S

A **table** is a structured set of data made up of rows (**records**) and columns (**fields**).

4. In the figure above, you can see the options Load and Transform Data. For now, click **Load** to import the data directly into Power BI. This **imports the data** as it is, and loads it into Power BI Desktop.

Clicking Transform Data will take you to **Power Query Editor**. This allows you to perform any required changes on the file before importing into Power BI, such as changing table name, column name, data type of a column, removing data or deleting columns, and so on.

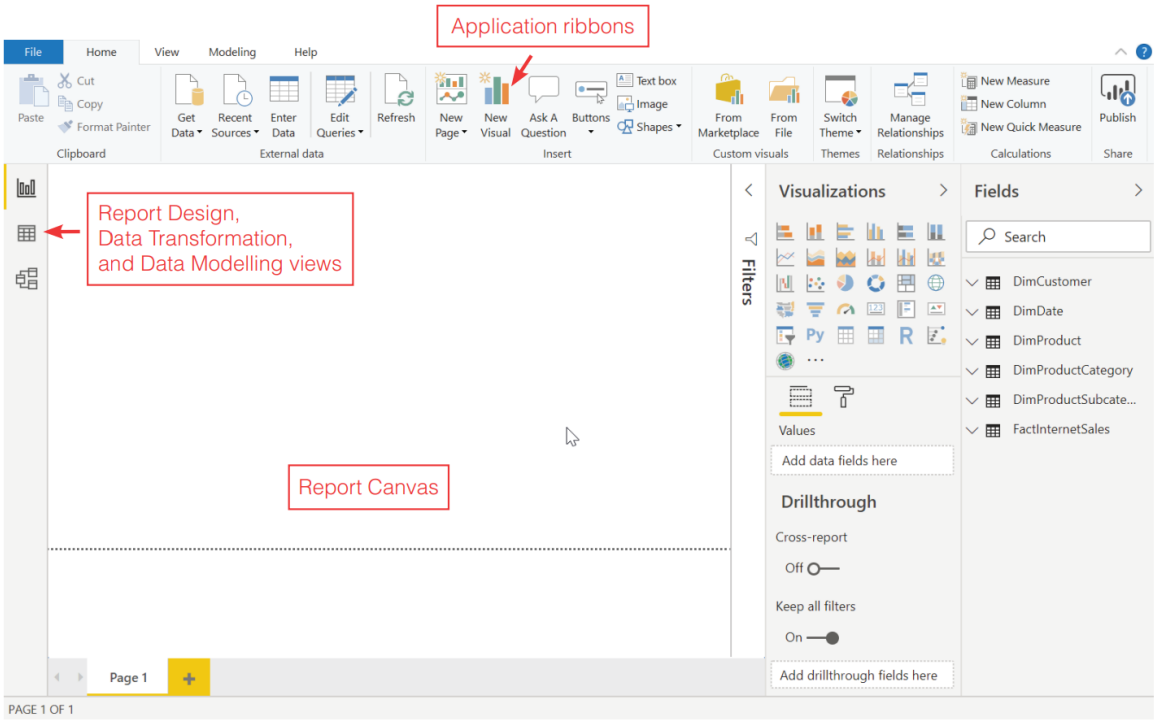
5. Repeat the same process and load all other sample files (**DimDate, FactInternetSales, DimProduct, DimProductCategory, and DimProductSubCategory**). Note that you will have to load these one at a time.
6. **Save** the Power BI Desktop model.

> 💡 Saving the model only saves the data ingestion and transformation process and not including the data sources. This makes any changes to the original data source affect the model and the data displayed.

Hooray, you have now loaded your data into the Power BI model and you have a blank canvas to start working with.

▼ Step 2: Explore the Power BI Desktop Application Interface

Let's take a moment to explore this report canvas, the Power BI Desktop application:

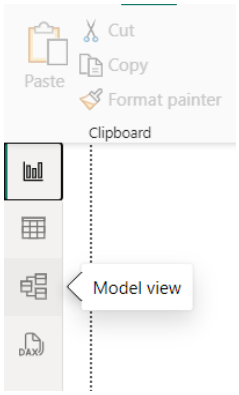


1. The left menu is used to switch between **Report Design**, **Data Transformations**, **Data Modelling** (creating relationships within your data), and **DAX Querying** (returning specific data parts or performing data operations).
2. The **Report Canvas** is for Visual Design and Layout.
3. The **Application ribbons** contain all options and settings, visual or page level properties, and other settings configuration.
4. The report building panes at the right contain all the components that may be added to a report. You can:
   a. Select fields and data from imported tables on the **Fields** pane.
   b. Select different ways to display this data from the

Visualizations pane.
c. Apply filtering to the data in the **Filters** pane.

▼ Step 3: Modelling data

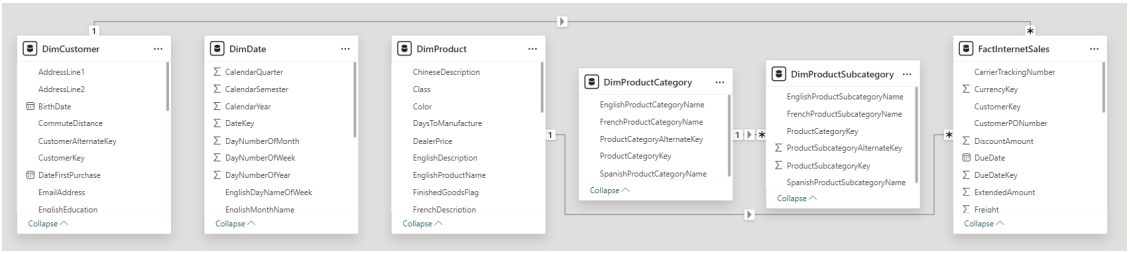
1. Select the **Data Modelling** view.



Once the model view is loaded, you will notice the following:

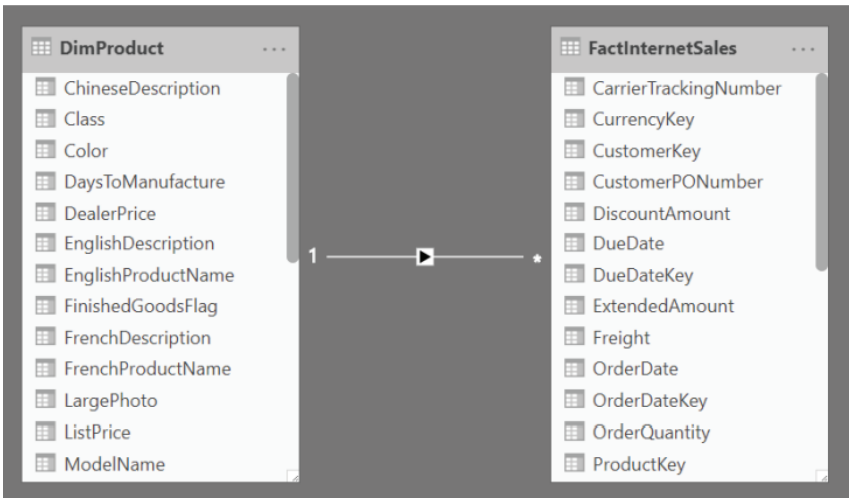
- **Tables** and their column names
- **Relationships** between tables

In addition, there are some relationships that are automatically created.



▼ Exploring about relationships:

In real-life scenarios, we will be working with databases that will contain **multiple tables**. These tables are connected, forming **relationships** like the one shown below.



Here are the things that should be noted:

1. **Line**: Represents the relationship between the tables.
2. **Arrow**: Indicates which direction filtering will occur.

   Here, If **DimProduct** is filtered on a particular value, **FactInternetSales** will also be filtered to only show records related to the selected value.
3. **"One" (1) side**: The relationship works off a single **unique** record on the **DimProduct** table.
4. **"Many" (*) side**: The relationship links to many records on the **FactInternetSales** table.

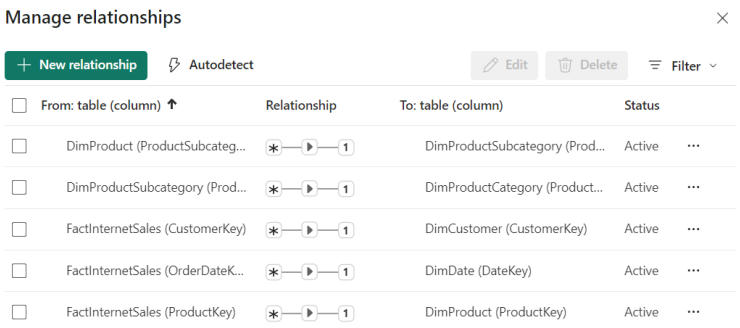
> 💡 **Well-defined relationships** between tables ensure the consistency and accuracy of the data → more efficient and accurate data extraction and analysis by establishing clear connections between related data points.

2. From the top menu, select **Manage Relationships**.
3. Click **+ New relationship**.
4. Input the following details in the new relationship dialog:
   a. From table: **DimProduct**
   b. In the first table, click to highlight the column **ProductSubcategoryKey.**
   c. To table: **DimProductSubcategory**
   d. In the second table, click to highlight the column **ProductSubcategoryKey.**
   e. Cardinality: **Many to one (* : 1)**
   f. Cross-filter direction: **Single**

g. Check the **"Make this relationship active"** checkbox.

5. Repeat steps 3 and 4 for another many to one relationship between **FactInternetSales (OrderDateKey)** **and** **DimDate (DateKey)**.

   Your relationships should look like this:



6. Click **Close** to save the relationships, then **save** the model.

At this point, you can begin creating visuals on the reporting canvas.

▼ Step 4: Creating the report

Now that the data has all been uploaded and modeled, we can begin reporting on that data.

▼ Building basic visualizations

Let's start with a graph to support a simple business case: *Our Sales Manager would like to track the value of our website bike sales over the last few years*.

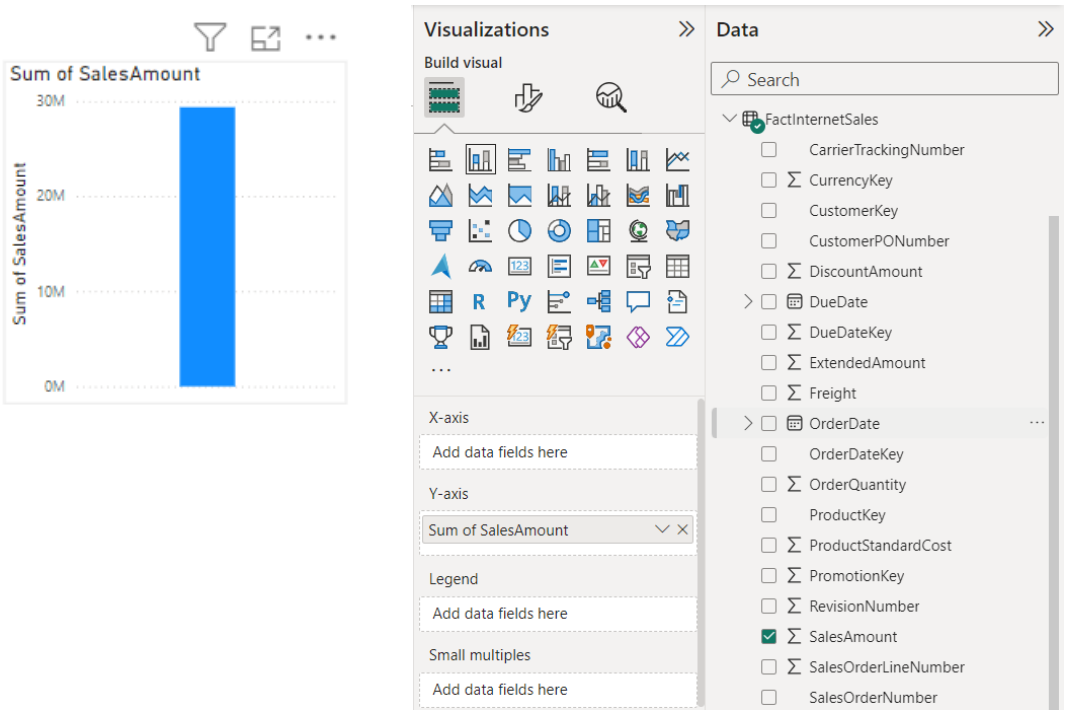
1. Switch to the **Report** view.

2. In the Visualization Pane, select the **Stacked Column Chart.** This creates a placeholder graphic to the Report Canvas.

3. Populate the column chart. Using the Data Pane on the right of the screen, complete the following steps:

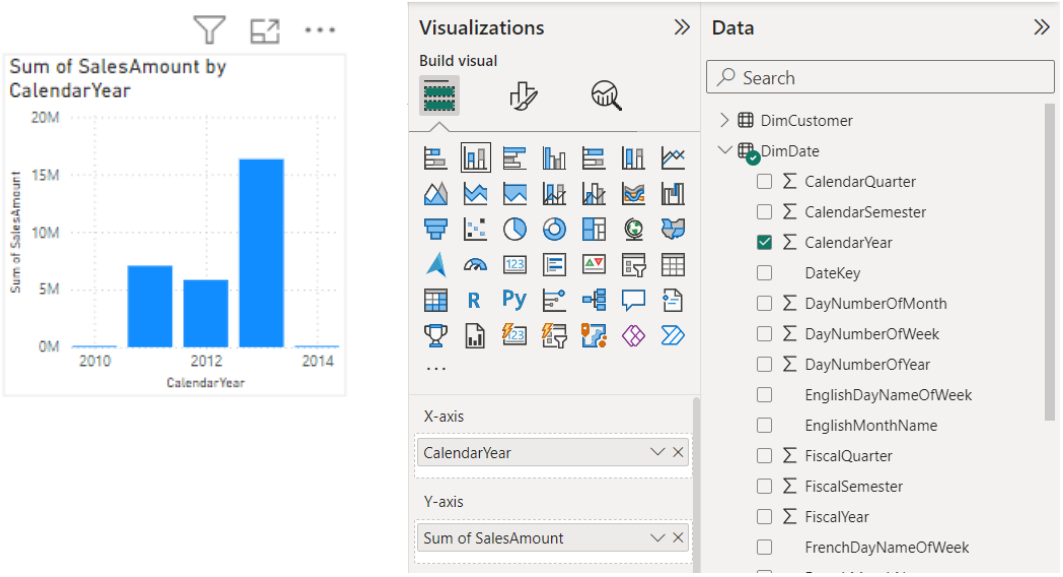
   a. Go to **FactInternetSales** and click the arrow to expand the list.

   b. Click and drag the **SalesAmount** to the space below "Y-axis".



c. Go to **DimDate** and click the arrow to expand the list.

d. Click and drag the **CalendarYear** to the space below "X-axis".

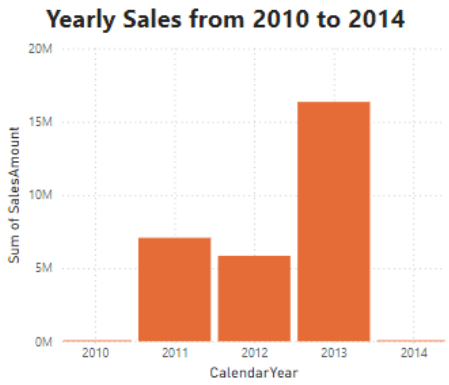


4. Style your visualization by navigating to the **Format visual** menu of the Visualization pane.

   a. Change the color of the columns: Expand the **Columns** menu, then expand the **Color** menu and pick another color.

b. Adjust column spacing: Still in the **Columns** menu, expand the layout menu and adjust the spacing between columns.

c. Change the visualization title: Switch to the **General** tab, expand the **Title** menu, then change the title and its font styles.



You may hover over the chart's columns to display the specific values.

Congratulations, you just created your first chart!

Next, our Marketing Manager would like to know the *occupation of the customers who order the most stock*, to target with a new campaign.

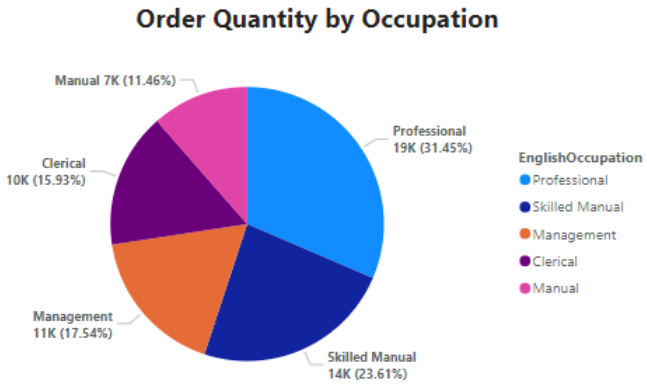
1. Switch to the **Report** view. Make sure none of the charts in the canvas is selected.

2. In the Visualization Pane, select the **Pie Chart.** This creates a placeholder graphic to the Report Canvas.

3. Populate the pie chart. Using the Data Pane on the right of the screen, complete the following steps:
   a. Go to **FactInternetSales** and click the arrow to expand the list.
   b. Click and drag the **OrderQuantity** to the space below "Values".
   c. Go to **DimCustomer** and click the arrow to expand the list.
   d. Click and drag the **EnglishOccupation** to the space below "Legend".

4. Style your visualization by navigating to the **Format visual** menu of the Visualization pane.
    - Some areas you may format include the title, chart padding, text styles, legend positioning, and label contents.

Your visualization may look closer to this output:



> 💡 Try to click over a given slice and observe its effects in the other visuals. What do you notice?

Save your output.

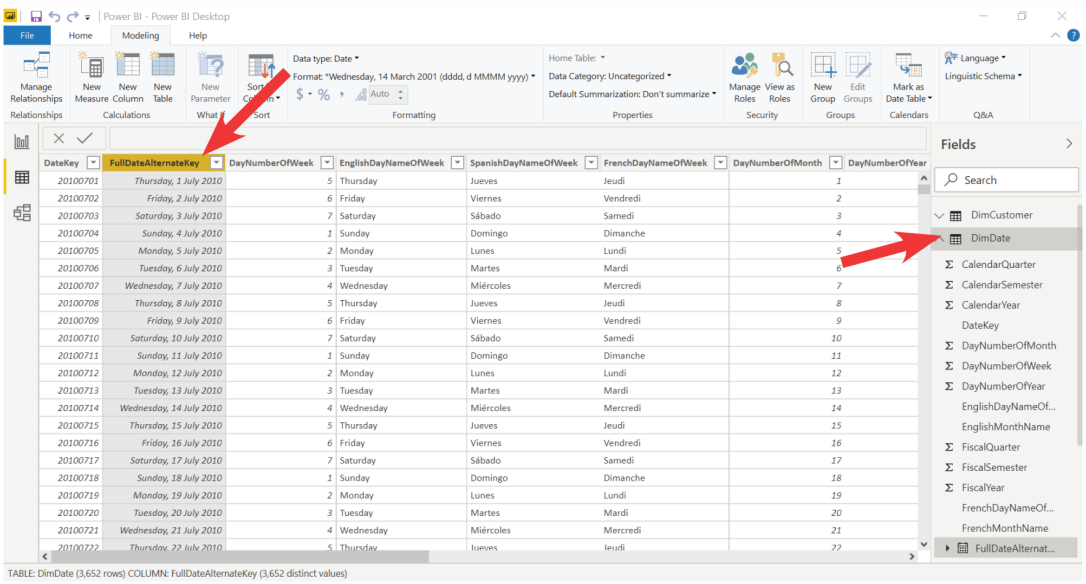
▼ Adding insights with new measures

We'll now create a table that displays sales amount and sales amount YTD.

Time intelligence provides richer analytics by providing measures that simplify timebased reporting, for example, time-based calculations for **Year-to-Date (YTD), Month-to-Date (MTD)**, and **Previous Period** comparisons.

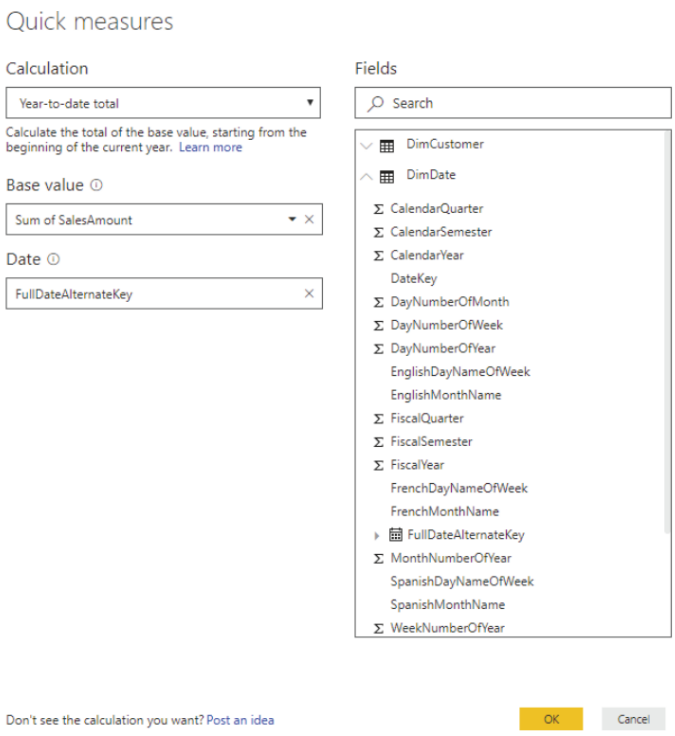
In this example, we will be using the DimDate dimension for the time intelligence reporting. Follow the steps below:

1. Switch to the **Data Transformation** view.
2. Select the **DimDate** table from the Fields pane on the right.
3. Check that the **FullDateAlternateKey** column has its Data Type set to **Date**.



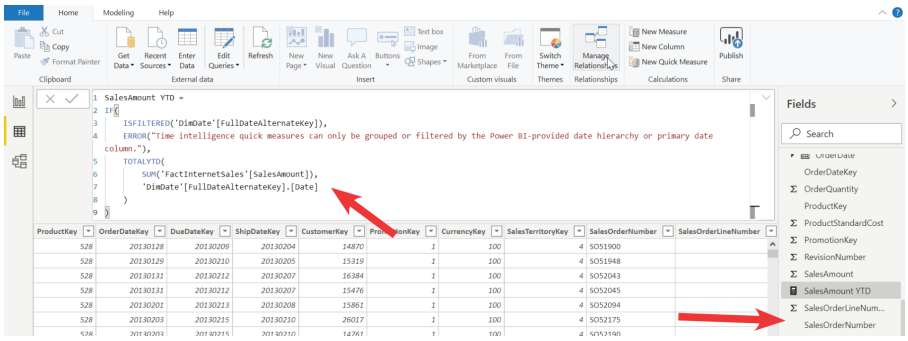
It is always important to **verify the data types** before modeling or performing any time intelligence computations.

4. Select the **FactInternetSales** table in the Fields pane.
5. On the Home ribbon, click Quick Measure.
6. In the Quick Measures dialog, do the following:
    a. For **Calculation:** select the Year-to-date total under the **Time Intelligence** Category.
    b. From the fields list on the right, go to **FactInternetSales** table and then drag **SalesAmount** into the **Base value** box.
    c. From the fields list on the right, go to **DimDate** table and then drag **FullDateAlternativeKey** into the **Base value** box.



d. Click **OK** once the details are correct.

Great, you made Power BI generate the new measure which can now be seen in the Fields pane, together with the generated DAX code!



Let's now create a visualization using the new measure we just made:

1. Switch to the **Report** view. Make sure none of the charts in the canvas is selected.

2. In the Visualization Pane, select **Table.**

3. Populate the table. Using the Data Pane on the right of the screen, complete the following steps:
    a. From **DimDate** expand **FullDateAlternateKey** and tick **Date Hierarchy**, **Year**, and **Quarter**.
    b. From **FactInternetSales**, tick **SalesAmount** **and** **SalesAmountYTD**.
4. Style and resize your table by navigating to the **Format visual** menu of the Visualization pane.

The Report Canvas now shows a table of Sales values, by Quarter, with the accumulated Year to Date values alongside.

Year-To-Date Sales Amount

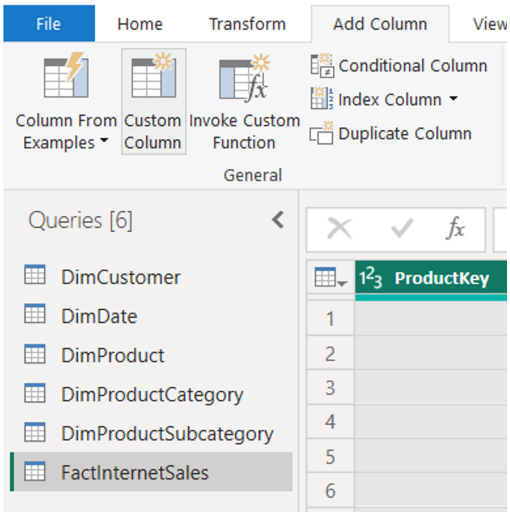
Year	Quarter	Sum of SalesAmount	SalesAmount YTD
2013	Qtr 4	5,328,573.76	16,351,550.34
2013	Qtr 3	4,370,237.06	11,022,976.58
2013	Qtr 2	3,973,793.48	6,652,739.52
2013	Qtr 1	2,678,946.04	2,678,946.04
2011	Qtr 4	2,038,185.32	7,075,525.93
2011	Qtr 3	1,814,387.99	5,037,340.61
2011	Qtr 2	1,801,595.14	3,222,952.62
2012	Qtr 4	1,697,617.17	5,842,485.20
2012	Qtr 3	1,454,653.06	4,144,868.03
2011	Qtr 1	1,421,357.48	1,421,357.48
2012	Qtr 1	1,375,841.32	1,375,841.32
2012	Qtr 2	1,314,373.65	2,690,214.97
2014	Qtr 1	45,694.72	45,694.72
2010	Qtr 4	43,421.04	43,421.04
2014	Qtr 2		45,694.72
2014	Qtr 3		45,694.72
2014	Qtr 4		45,694.72
Total		**29,358,677.22**	**45,694.72**

Save your output.

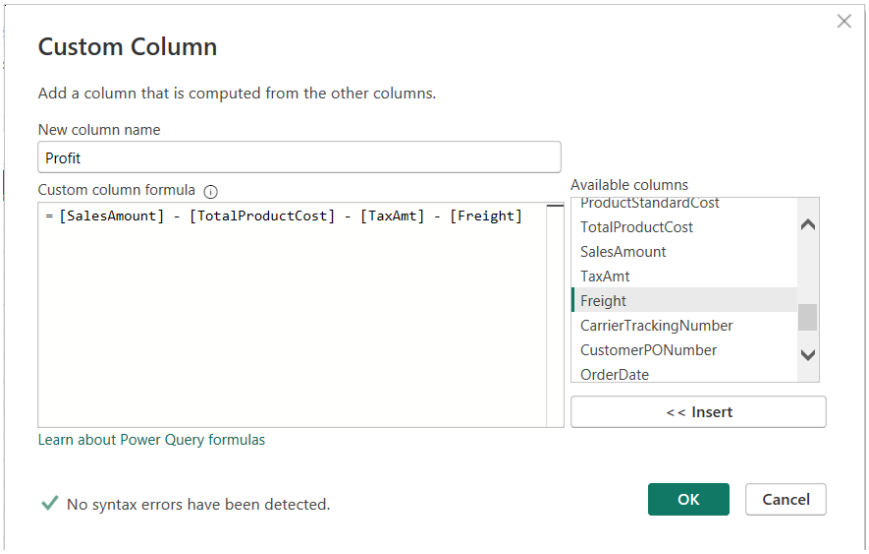
▼ Adding a new column to sales data

Now your manager wants to visualize the profit of your business across the different time spans. With this, you need to determine the profit, which is not readily available in your data model. However, we can perform calculations in the data transformation view.

1. In the reports view, select **Transform Data**. This load the **Power Query Editor**.
2. At the left pane, select the **FactInternetSales** table.
3. On the navigation pane, click the **Add Column** tab, then click **Custom Column**.



4. Name the column **Profit**, and create the formula [SalesAmount] - [TotalProductCost] - [TaxAmt] - [Freight]. Click **OK** once done.



5. Observe the new column that is created at the end of the **FactInternetSales** table. Pay attention to the icon besides the **Profit** column.

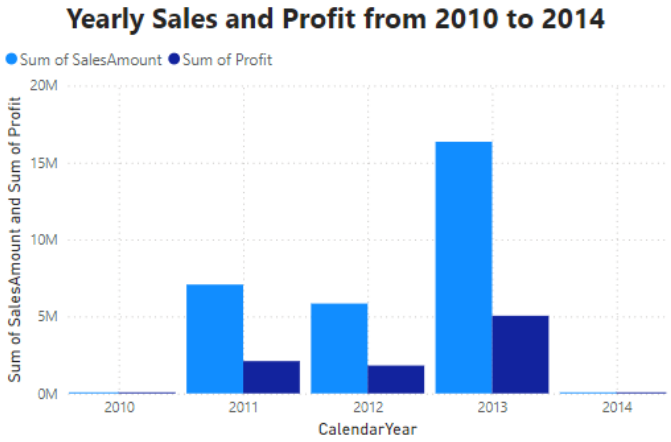
This shows a mixed data type. Click the icon and change to decimal by selecting **Decimal Number** in the dropdown.

6. Save your changes.

You now have a working profit column. Let's now use it in our report by updating the chart and table.

1. Switch to the **Report** view.
2. Select the column chart by clicking it once.
3. Go to the Visualizations pane and select the **clustered column chart**. This updates the selected chart to the selected type.

4. From the Data pane, click and drag the newly created **Profit** column under **FactInternetSales** table to right below the "Sum of SalesAmount" entry under "Y-axis".
5. Resize the visualization to enlarge the chart and adjust the title.



6. Add a new quick measure for Profit YTD.
    a. Click **Quick Measure**.
    b. In the "Select a calculation" dropdown, select **Year-to-date** total.
    c. Drag **Profit** to the base value entry.
    d. Drag **FullDateAlternativeKey** to the Date entry.
    e. Click **Add** to add the new measure.
7. Go back to the report view and select the table by clicking it once.
8. Add the measures **Profit** and **Profit YTD** to the columns of the table under the Visualizations pane.

9. Resize the visualization to enlarge the table and adjust the title.

Year-To-Date Sales Amount and Profit

Year	Quarter	Sum of SalesAmount	SalesAmount YTD	Sum of Profit	Profit YTD
2010	Qtr 4	43,421.04	43,421.04	13,289.76	13,289.76
2011	Qtr 1	1,421,357.48	1,421,357.48	420,825.77	420,825.77
2011	Qtr 2	1,801,595.14	3,222,952.62	534,521.05	955,346.82
2011	Qtr 3	1,814,387.99	5,037,340.61	537,782.75	1,493,129.57
2011	Qtr 4	2,038,185.32	7,075,525.93	608,003.82	2,101,133.39
2012	Qtr 1	1,375,841.32	1,375,841.32	413,372.50	413,372.50
2012	Qtr 2	1,314,373.65	2,690,214.97	405,680.21	819,052.71
2012	Qtr 3	1,454,653.06	4,144,868.03	457,548.87	1,276,601.57
2012	Qtr 4	1,697,617.17	5,842,485.20	537,944.36	1,814,545.93
2013	Qtr 1	2,678,946.04	2,678,946.04	828,757.12	828,757.12
2013	Qtr 2	3,973,793.48	6,652,739.52	1,224,627.58	2,053,384.70
2013	Qtr 3	4,370,237.06	11,022,976.58	1,348,696.63	3,402,081.33
2013	Qtr 4	5,328,573.76	16,351,550.34	1,646,414.48	5,048,495.81
2014	Qtr 1	45,694.72	45,694.72	20,754.91	20,754.91
2014	Qtr 2		45,694.72		20,754.91
2014	Qtr 3		45,694.72		20,754.91
2014	Qtr 4		45,694.72		20,754.91
Total		**29,358,677.22**	**45,694.72**	**8,998,219.81**	**20,754.91**

Save your output.

▼ Adding additional elements

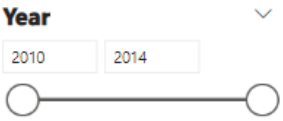
Now you might want to filter all the visuals in the page to display for a specific year (and make the others disappear). Fortunately, Power BI also has a feature for you to achieve this seamlessly, by using **Slicers**.

1. Switch to the **Report** view. Make sure none of the charts in the canvas is selected.

2. In the Visualization Pane, select **Slicer.**

3. Populate the filter. From **DimDate** expand **FullDateAlternateKey**, then drag **Year** to the fields entry.
4. Expand the filters pane and then expand the **FullDateAlternateKey - Year** field.
5. Switch the filtering type to **Basic filtering** then check years **2010, 2011, 2012, 2013, and 2014**.

Your output will look similar to this. Notice that the ranges are now between 2010 and 2014. Toggle the slicer and observe the changes in the report.



📝 **EXERCISE**: Create another slicer that filters the report by product subcategory.
- Use the **EnglishProductCategoryName** field.
- Exclude the **(Blank)** entry in the filter pane for the slicer.

Sometimes, we just want to display a certain value outright especially if it is of great impact in the report, such as a grand total or general average value. We can do this using Power BI's **Card** visual.

1. .Switch to the **Report** view. Make sure none of the charts in the canvas is selected.

2. In the Visualization Pane, select **Card.**

3. Populate the card. Drag **Profit** from **FactInternetSales** to the field entry.
4. Click the arrow besides the **Profit** and select "Rename for this visual". Rename the field as **Total Profit.**
5. Format the card to display the full amount.
    a. Go to Format visual, then expand the **callout value** menu.
    b. Set the display units to **None**. This displays the exact amount.
    c. Set the number of decimal places to **2**.
    d. Adjust the font style and size to fit the display.

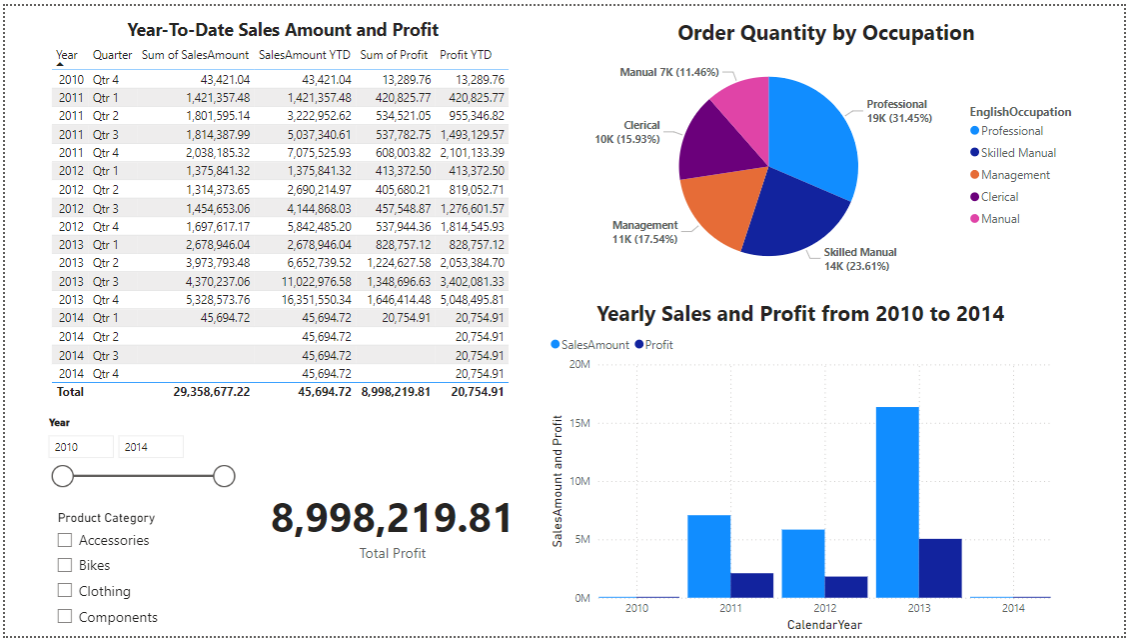
Your output will look similar to this.

> 💡 Try to toggle the slicers you created. Notice that the value in the card changes as well. How flexible your report is indeed!

Congratulations, you have now completed your first Power BI Report!

Save your output.

▼ Reference dashboard



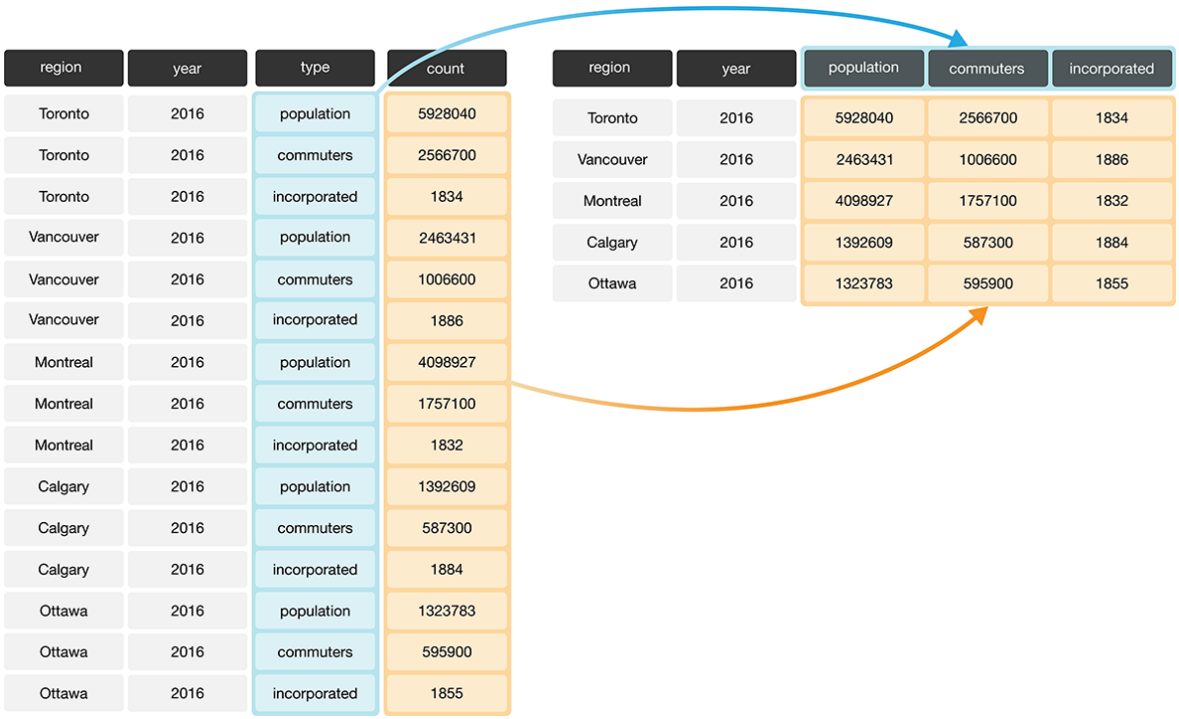
▼ Part 2: Data Transformation Deep Dive

In this part, we will be focusing on an Excel file with a data formatting that cannot be directly be used in visualization tools like Power BI. Here, we learn how to **transform the data** entirely using Power BI to have it ready for modeling.

▼ Step 0: Explore the data

1. Open the Excel workbook inside the Part 2 folder and take a look at the data.
2. Notice that there are currency values scattered across multiple columns, with each column storing values for a given year.

> 💡 What you are seeing is an example of **wide data,** a format often used for tabular presentation. However, if we need to perform a generalized analysis, we need to strictly store each measure in a single column. That means all budget values should reside in one column, making the table a **long data** or **"tidy" data** format.



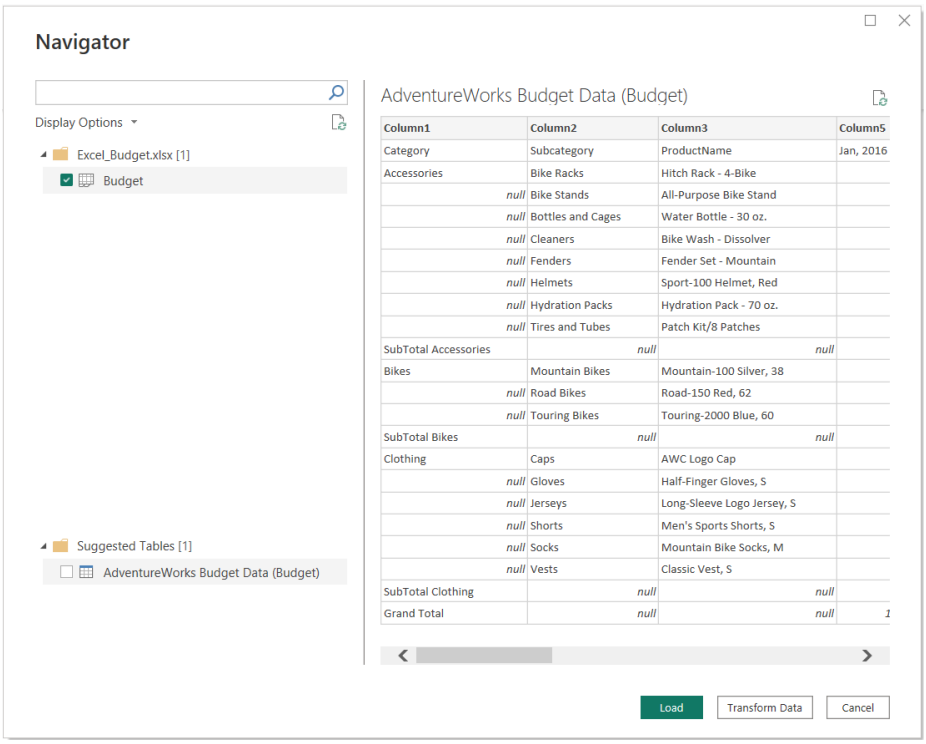
region	year	type	count
Toronto	2016	population	5928040
Toronto	2016	commuters	2566700
Toronto	2016	incorporated	1834
Vancouver	2016	population	2463431
Vancouver	2016	commuters	1006600
Vancouver	2016	incorporated	1886
Montreal	2016	population	4098927
Montreal	2016	commuters	1757100
Montreal	2016	incorporated	1832
Calgary	2016	population	1392609
Calgary	2016	commuters	587300
Calgary	2016	incorporated	1884
Ottawa	2016	population	1323783
Ottawa	2016	commuters	595900
Ottawa	2016	incorporated	1855

region	year	population	commuters	incorporated
Toronto	2016	5928040	2566700	1834
Vancouver	2016	2463431	1006600	1886
Montreal	2016	4098927	1757100	1832
Calgary	2016	1392609	587300	1884
Ottawa	2016	1323783	595900	1855

We will be performing the transformation here in Power BI. Relax and see the magic!

▼ Step 1: Load the data

1. Open a new Power BI notebook.
2. Import the given Excel workbook.
3. Select the Budget sheet and verify the contents previewed.
4. Click **“Transform data”**.



▼ Step 2: Transform the data

Make sure you are in the **Power Query Editor** before moving into the steps below.

1. **Remove** the first rows before the actual header row.
2. **Promote** the current first row as header.

	Category	Subcategory	ProductName	ProductKey	Jan, 2016	Feb, 2016
1	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	1131	2635

3. **Remove** the rows with **subtotal/grand total** values.
4. **Remove** the **Grand total** column.
5. **Fill** the first column **(Category)** downwards.
6. **Unpivot** the **Amount** columns.
7. Adjust and verify the **data types** of the **date** and **amount** columns.
8. Create new columns: **Year, Year-Month, Year-Month code**.
9. **Create a new unique key** for each row combining the **Year-Month code** and **Product code**.
10. Verify the result and save your work.

> 💡 **TIP**: You may view the **APPLIED STEPS** under the Query settings pane at the right to view your edit history in the table you are transforming.

Your final output should look close to this for your reference:

▼ **See sample result**

	Category	Subcategory	ProductName	ProductKey	Year Month	Amount
1	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/01/2016	1131
2	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/02/2016	2635
3	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/03/2016	4134
4	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/04/2016	2179
5	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/05/2016	2637
6	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/06/2016	3279
7	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/07/2016	2218
8	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/08/2016	3287
9	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/09/2016	3885
10	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/10/2016	2484
11	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/11/2016	5441
12	Accessories	Bike Racks	Hitch Rack - 4-Bike	483	01/12/2016	3551
13	Accessories	Bike Stands	All-Purpose Bike Stand	486	01/01/2016	666
14	Accessories	Bike Stands	All-Purpose Bike Stand	486	01/02/2016	3695

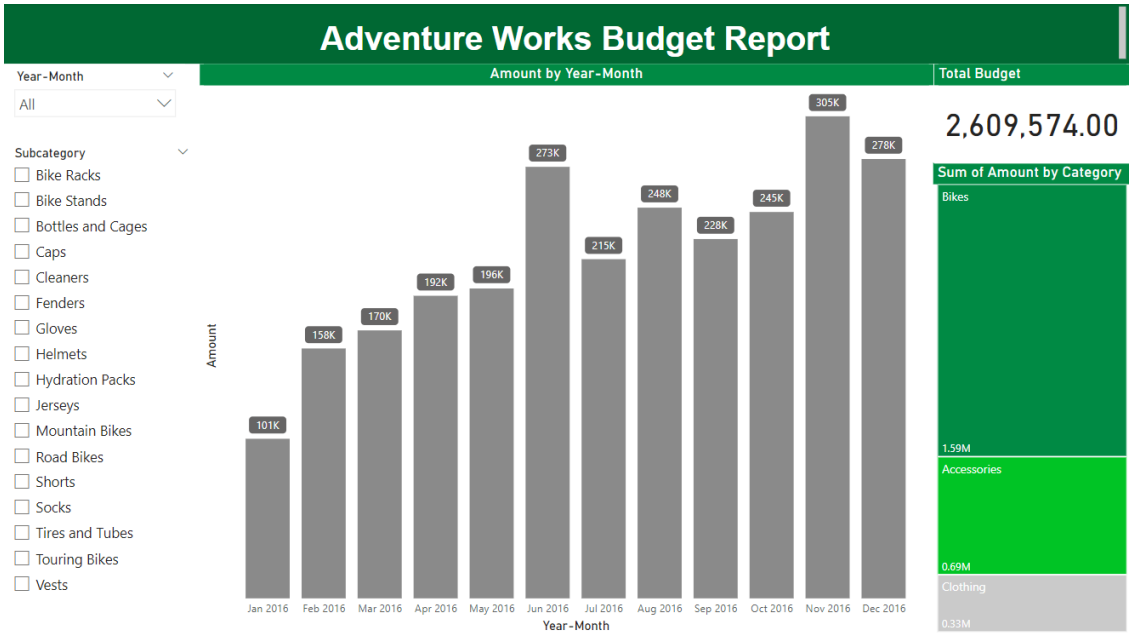
Amount	Source	Year	Year-Month	Year-Month Code	RecordKey
1131	Budget	2016	Jan 2016	201601	201601483
2635	Budget	2016	Feb 2016	201602	201602483
4134	Budget	2016	Mar 2016	201603	201603483
2179	Budget	2016	Apr 2016	201604	201604483
2637	Budget	2016	May 2016	201605	201605483
3279	Budget	2016	Jun 2016	201606	201606483
2218	Budget	2016	Jul 2016	201607	201607483
3287	Budget	2016	Aug 2016	201608	201608483
3885	Budget	2016	Sep 2016	201609	201609483
2484	Budget	2016	Oct 2016	201610	201610483
5441	Budget	2016	Nov 2016	201611	201611483
3551	Budget	2016	Dec 2016	201612	201612483
666	Budget	2016	Jan 2016	201601	201601486

▼ Step 3: Visualize the data

1. Switch to the report view.
2. Insert the following elements into your report:
    a. A **title** for your report.
    b. A **column** chart reflecting the monthly budget by Year-Month.
    c. A **treemap** reflecting the monthly budget by category.
    d. A **card** reflecting the total budget.
    e. A **slicer** with a dropdown for the Year - Month selection
    f. A **slicer** with checkboxes for the Subcategory selection.
3. Style and format your visuals.
4. Examine your report and save your work.

Your final output should look close to this for your reference:

▼ **See sample dashboard**



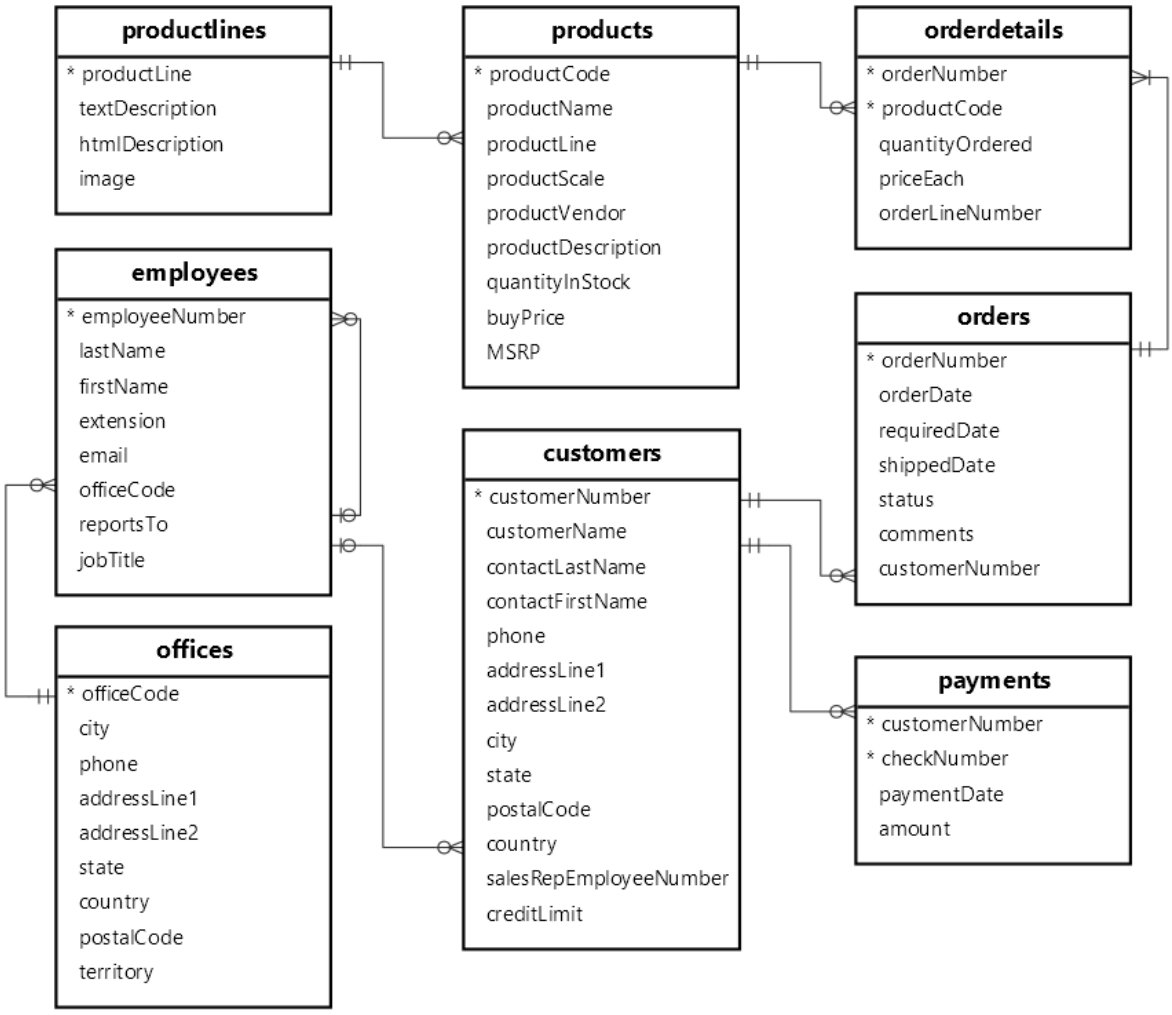
▼ Practice: Classic Models

The classic models dataset, originally the sample MySQL database, is one that is commonly used to train students on database querying. This can be also used for analytics modeling using Power BI to derive as much visuals as you can and infer unique insights out of the data.

Schema:

- **customers**: stores customer's data.
- **products**: stores a list of scale model cars.
- **productlines**: stores a list of product lines.
- **orders**: stores sales orders placed by customers.
- **orderdetails**: stores sales order line items for every sales order.
- **payments**: stores payments made by customers based on their accounts.
- **employees**: stores employee information and the organization structure such as who reports to whom.
- **offices**: stores sales office data.

Entity Relationship Diagram:



Instructions:

1. Open the SQL file in the Practice folder. (Source: **Dataset link**).
2. Open a new Power BI project and import the SQL file. Verify the column data types and do transformations if necessary.
3. Establish and verify the relationships as shown in the diagram.
4. Create a meaningful dashboard out of the data model. You may use **this link** for some guide questions.
5. Enjoy!

Want more practice? Check out this website for more dataset sources!