Microsoft Power Platform

Power BI Dashboard in a Day Lab 1

Maintained by: Microsoft Corporation

Contents

Introduction

Today you will learn about various key features of the Power BI service. This is an introductory course intended to teach you how the author builds reports using the Power BI Desktop, create operational dashboards, and share content via the Power BI Service.

By the end of this lab, you will have learned:

- How to load data from Microsoft Excel and Comma-Separated Values (CSV) sources
- How to manipulate the data to prepare it for reporting
- How to prepare the tables in Power Query and load them into the model

Learning these steps will prepare you for the modeling exercises in Lab 2. Additionally, the results of this lab will be the starting point for Lab 2.

Power BI Desktop

Power BI Desktop – Accessing Data

In this section, you will import VanArsdel's and its competitors' USA sales data. You will then import and merge sales data from other countries.

Power BI Desktop – Get Data

Let's start by looking at the data files. The dataset contains sales data of VanArsdel and other competitors. We have seven years of transaction data by day, product, and zip code for each manufacturer. We are going to analyze data from seven countries.

USA sales data is in a CSV file located in the USSales subfolder within the Data folder (/Data/USSales).

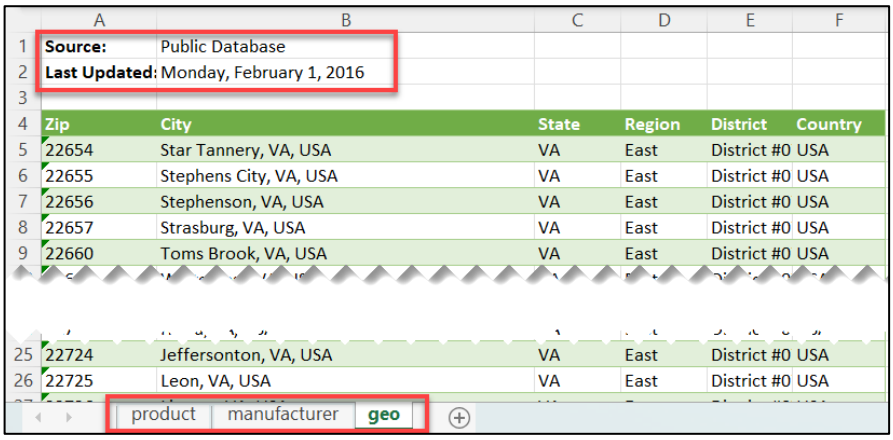
Sales of all other countries is in the InternationalSales subfolder within the Data folder (/Data/InternationalSales). Each country's sales data is in a CSV file in this folder.

Product, Geography, and Manufacturer information is in a Microsoft Excel file called bi_dimensions.xlsx in the USSales subfolder within the Data folder (/Data/USSales/).

1. Open the **bi_dimensions.xlsx** file. Notice that the first sheet has **Product** information. Within this sheet there is a header, and product data is in a named table. Also notice that the **Category** column has numerous empty cells.

The **Manufacturer sheet** has data laid out across the sheet, no column headers, several blank rows, and a note in row seven.

The **Geo sheet** has the geography information. The first few rows have data details. The actual data starts within row four.

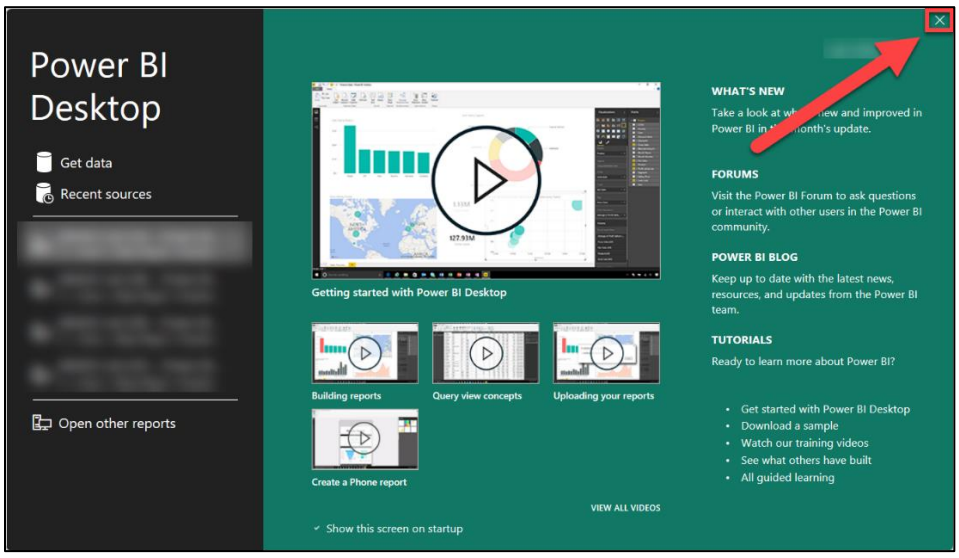


	A	B	C	D	E	F
1	**Source:**	Public Database				
2	**Last Updated:**	Monday, February 1, 2016				
3						
4	**Zip**	**City**	**State**	**Region**	**District**	**Country**
5	22654	Star Tannery, VA, USA	VA	East	District #0	USA
6	22655	Stephens City, VA, USA	VA	East	District #0	USA
7	22656	Stephenson, VA, USA	VA	East	District #0	USA
8	22657	Strasburg, VA, USA	VA	East	District #0	USA
9	22660	Toms Brook, VA, USA	VA	East	District #0	USA
25	22724	Jeffersonton, VA, USA	VA	East	District #0	USA
26	22725	Leon, VA, USA	VA	East	District #0	USA

product | manufacturer | geo

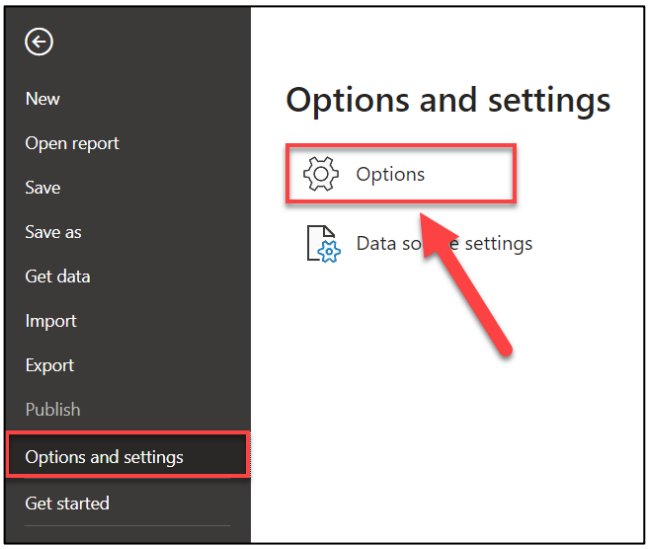
After reviewing the **bi_dimensions.xlsx** file, close **Microsoft Excel**. We will start by connecting data from these different sheets, and then perform data cleaning and transformation operations.

2. If you don't already have the **Power BI Desktop** open, launch it now.
3. If you have not signed into the **Power BI Desktop**, select the **Get started** option.
4. **Sign in** using your Power BI credentials.
5. You will see the startup screen open. Select the **X** in the top right corner of the dialog box to close it.

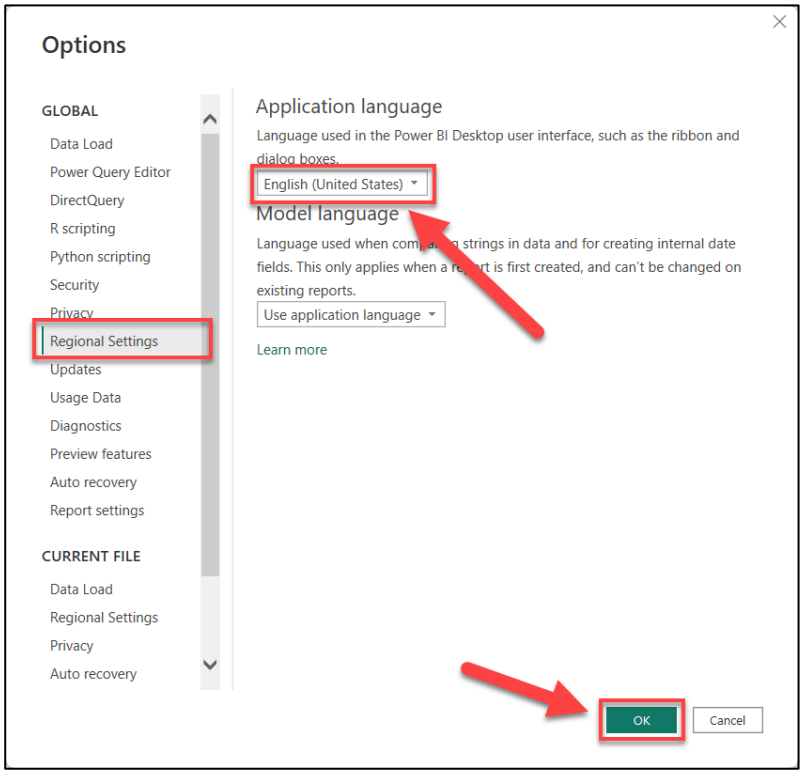


Next, let's set the **Locale** to US English to make it convenient in the rest of this lab.

6. From the ribbon, select **File**, then choose **Options and settings**. Then, select **Options**.

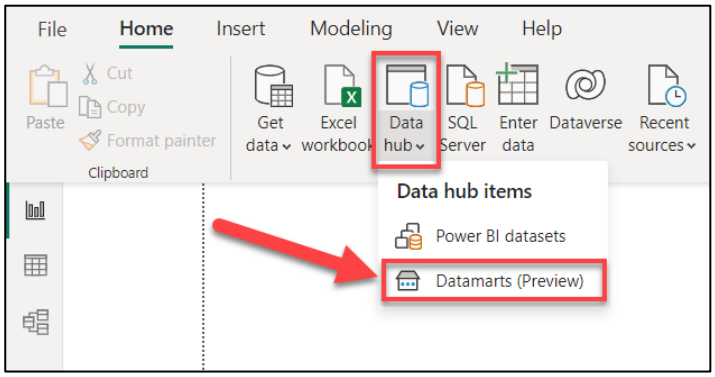


7. Within the pane to the left of the **Options** dialog box, select **Regional Settings** under **Current File**.
8. From the **Locale** drop-down, select **English (United States).**
9. Then, select **OK** to close the dialog box.



The next step is to load data into the **Power BI Desktop**.

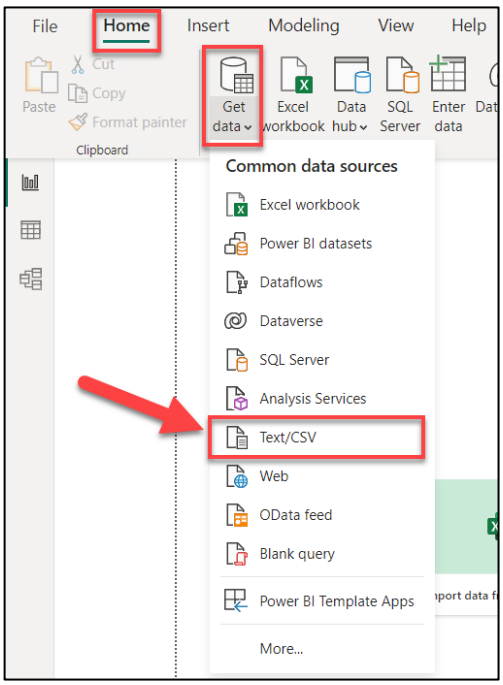
Note: Power BI Desktop has the capability to connect to 300+ data sources. The newest source to be added is Datamarts. You will not be using Datamarts in this class.



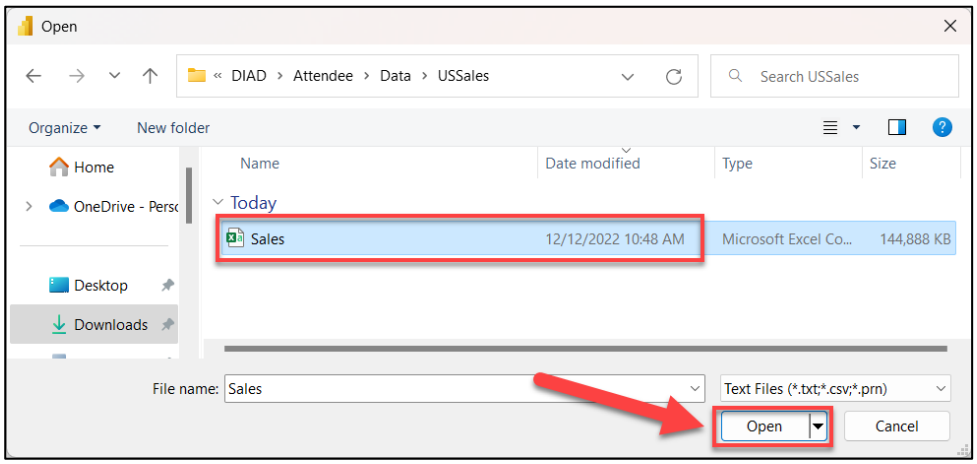
We are using CSV and Excel data files in this lab for simplicity. If you would like a full list of data sources, please visit this link: https://docs.microsoft.com/en-us/power-bi/connect-data/desktop-data-sources

Start by loading **USA Sales data** which is in a CSV file.

10. From the ribbon at the top of the screen, select the **Home** tab. Then, choose the **Get Data** drop-down.
11. Select **Text/CSV** from the options list.



12. Browse to the **DIAD** folder (this folder may be called **Attendee** if you did not rename it in Lab 0), double-click **Data**, double-click the **USSales** folder, and then select the **Sales.csv** file.
13. Then, select the **Open** button.



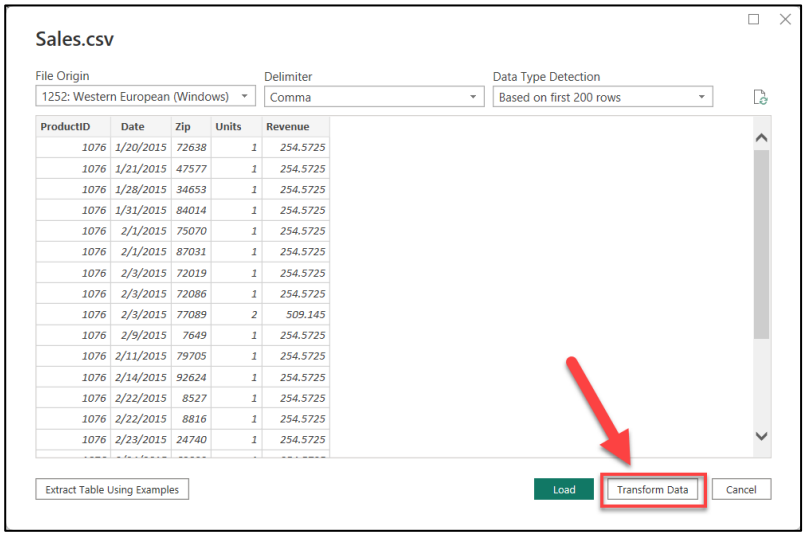
> **Note**: If your folder appears empty then this likely means you forgot to unzip your class files. Navigate to your location where you saved the class files and unzip the files by right-clicking on the .zip file, then select **Extract All**.

Power BI detects the data type within each column. There are options to detect the data type based on the first 200 rows, based on the entire dataset or to not detect the data. Since our dataset is large and it will take time and resources to scan the complete dataset, we will leave the default option of selecting the dataset based on the first 200 rows.

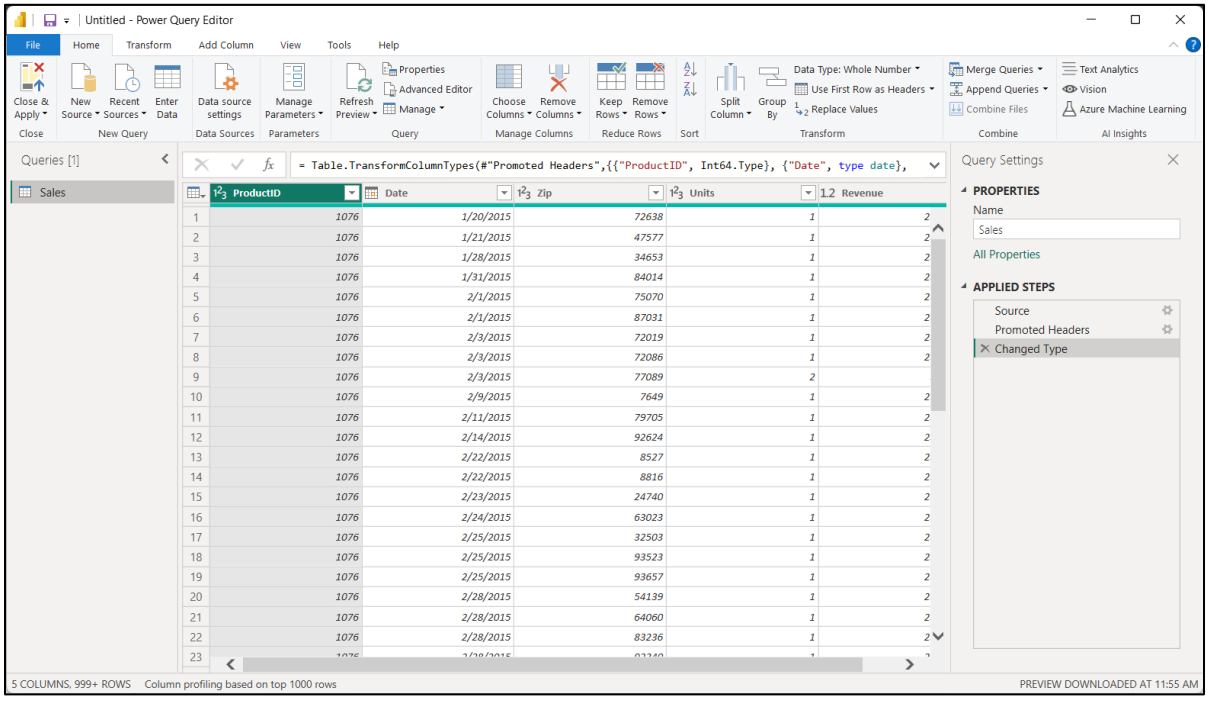
After completing your selection, you have three options – Load, Transform Data or Cancel.

- **Load** adds the data from the source into Power BI Desktop for you to start creating reports.
- **Transform Data** allows you to perform data shaping operations such as merging columns, adding additional columns, changing data types of columns as well as bringing in additional data.
- **Cancel** returns you back to the main canvas.

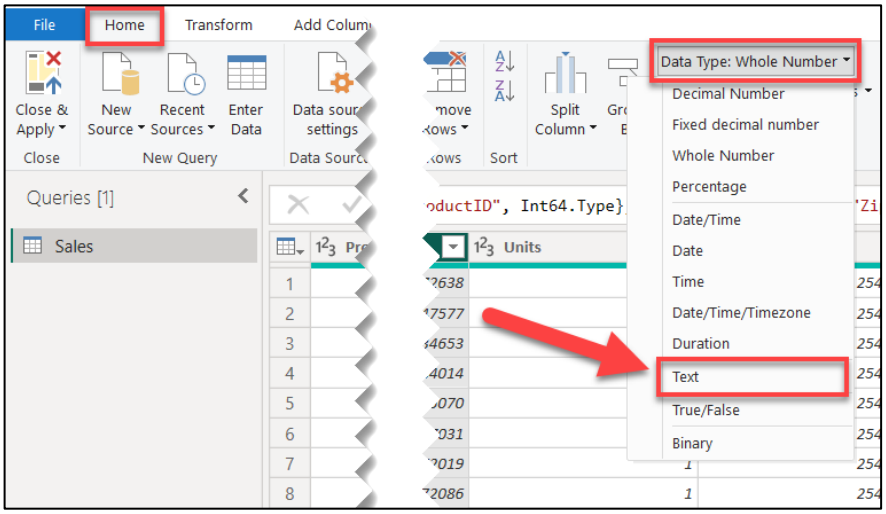
14. Within the **Sales.csv** dialog window, select the **Transform Data** button.



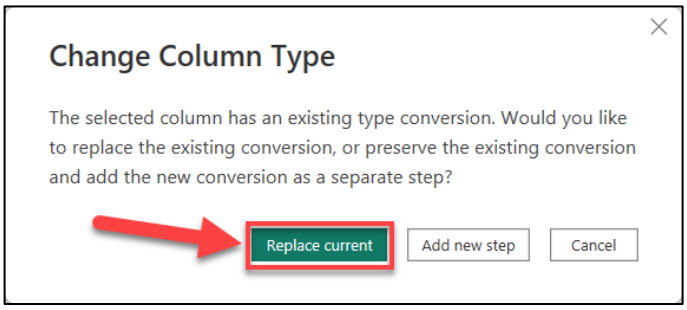
You should be in the Query Editor window as shown in the screenshot below. The Query Editor is used to perform data shaping operations. Notice that the sales file you connected shows as a query in the pane to the left of the screen. You can see a preview of the data in the center pane. Power BI predicts the data type of each field (based on the first 200 rows) as indicated by the icons to the right of each column header. Within the pane to the right of the screen, steps that the Query Editor performs are recorded within the APPLIED STEPS section.



15. Notice that Power BI has set the **Zip** field to the data type **Whole Number**. To ensure that the leading zero is not dropped from Zip codes that start with zero, we will format them as **Text**. To do this, select the **Zip** column. Then, from the ribbon, select the **Home** tab. From the menu at the top of the screen, within the **Transform** group, select the **Data Type** drop-down and choose the **Text** option.



16. The **Change Column Type** notification box opens. Select the **Replace current** button which overwrites Power BI's predicted data type.

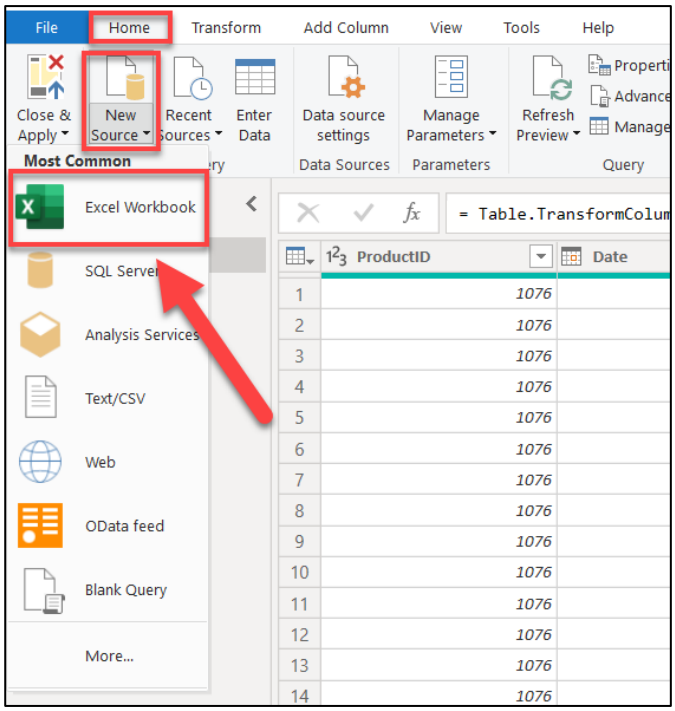


IMPORTANT!

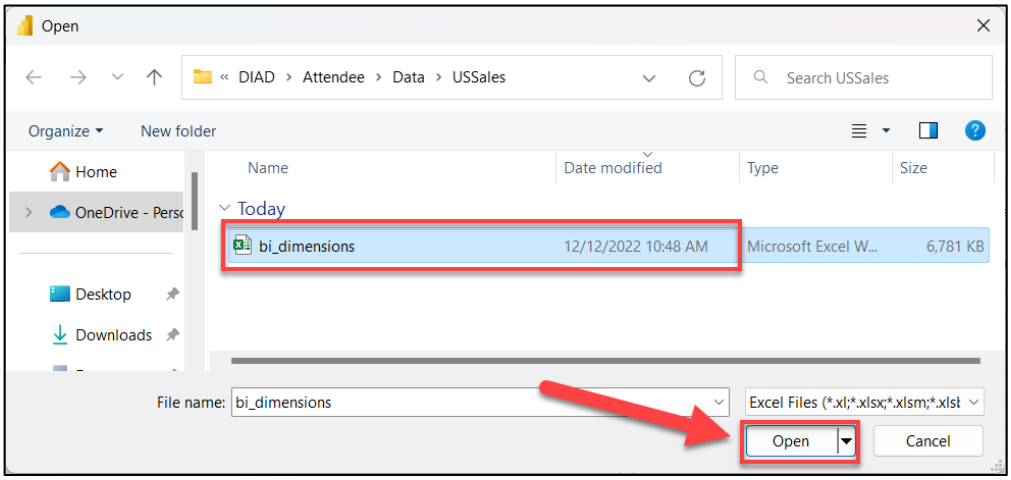
Missing these last two steps can impact your experience later.

Now, let's get the data that is in the Excel source file called **bi_dimensions.xlsx**.

17. From the ribbon at the top of the screen, select the **Home** tab, choose the **New Source** drop-down, and then select **Excel Workbook.**

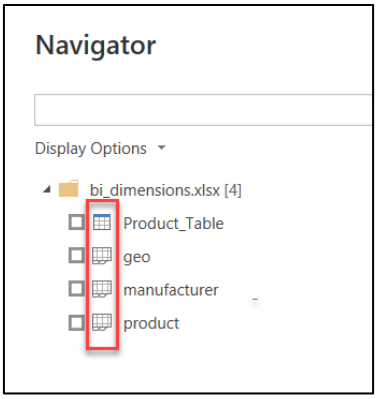


18. Browse to the **DIAD** folder, double-click **Data**, double-click the **USSales** folder, and then select **bi_dimensions.xlsx**.
19. Then, select the **Open** button. The **Navigator** dialog box will appear.

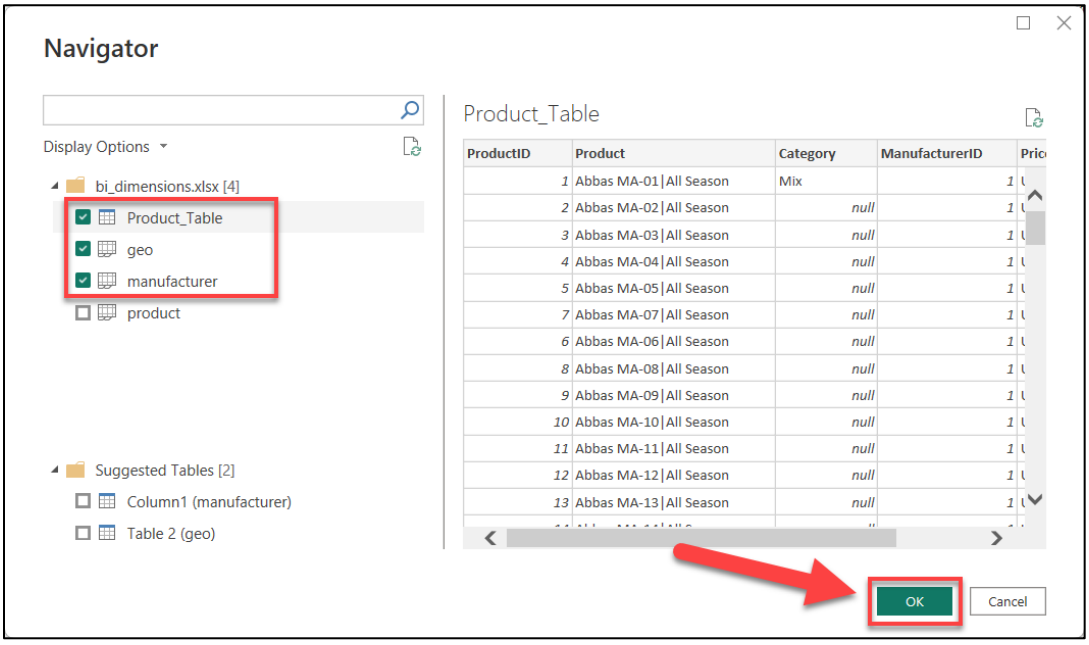


20. The **Navigator** dialog box lists three sheets that are within the Excel workbook. It also lists the **Product_Table** which is a pre-defined Excel table.

Note: Excel Table names are differentiated from Worksheet names by using different icons.



21. From the pane to the left, select the box to the left of **geo**. In the preview pane, notice that the first few rows are headers and are not part of the data. We will remove them shortly.
22. From the left pane, select the box to the left of **manufacturer**. In the preview pane, notice that the last couple of rows are footers and are not part of the data. We will remove them shortly.
23. From the left pane, select the box to the left of **Product_Table**. Notice the different icon indicates that this data is stored in an Excel table.
24. Make sure that **Product_Table**, **geo** and **manufacturer** are selected within the pane to the left, and then select **OK**. Notice that three sheets are added as queries in the Query Editor



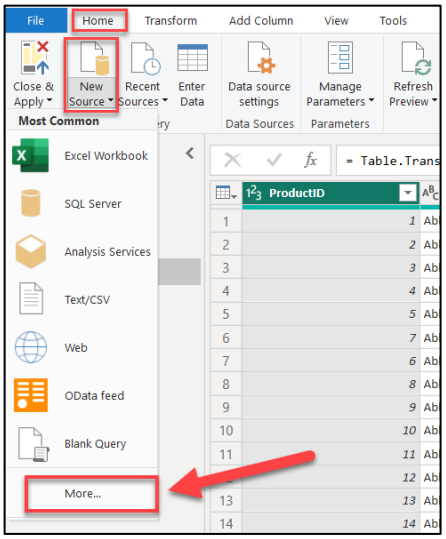
Power BI Desktop – Adding additional data

In this scenario, the international subsidiaries have agreed to provide their sales data so that the company's sales can be analyzed together. You've created a folder where they each put their data.

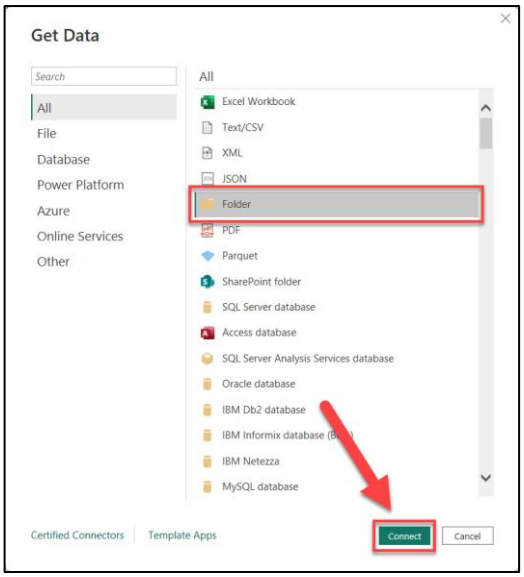
To analyze all the data together, you need to import the new data from each of the subsidiaries and combine it with the US Sales you loaded earlier.

You can load the files one at a time, like you loaded the US Sales data, but Power BI provides an easier way to load all the files in a folder together at once.

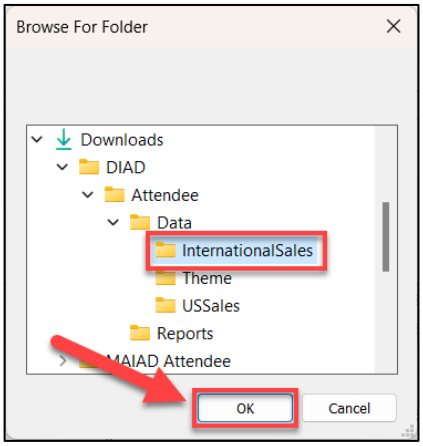
25. From the **Home** tab of the Query Editor, select the **New Source** drop-down menu.
26. Select **More…** from the options list. The **Get Data** dialog box will appear.



27. Within the **Get Data** dialog box, select **Folder** from the **All** list.
28. Then, select the **Connect** button and the **Folder** dialog box will open.



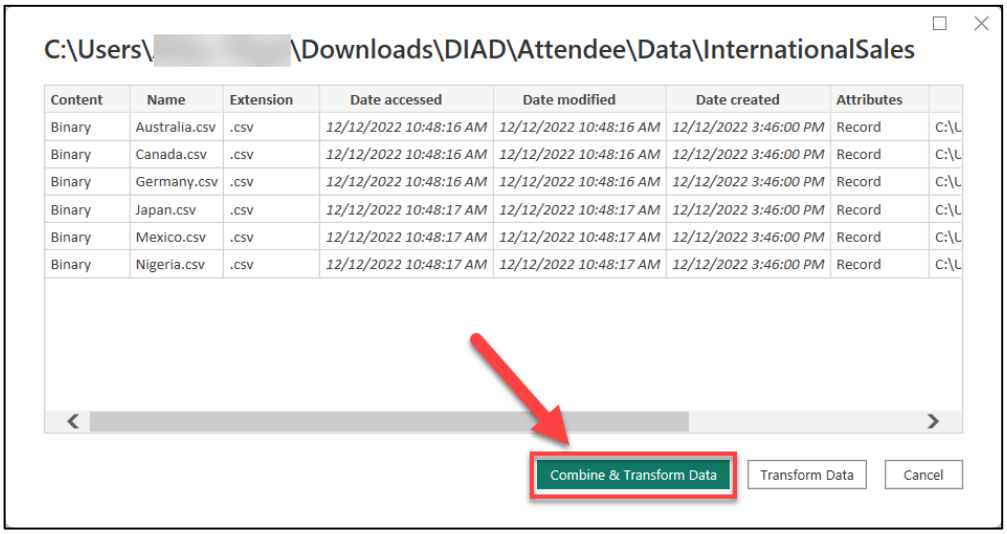
29. Within the Folder dialog box, select the **Browse…** button.
30. In the **Browse For Folder** dialog box, navigate to the location where you **unzipped** the class files.
31. Open the **DIAD** folder.
32. Open the **Data** folder.
33. Select the **InternationalSales** folder.
34. Select **OK** to close the **Browse for Folder** dialog box.
35. Select **OK** to close the **Folder** dialog box.



Note: This approach will load all the files located in the folder. This is useful when you have a group that puts files on an FTP site each month and you are not always sure of the names of the files or the number of files. All the files must be of the same file type with columns in the same order.

The selected folder dialog box will display the list of files within the folder.

36. Select the **Combine & Transform Data** button at the bottom of the dialog box.

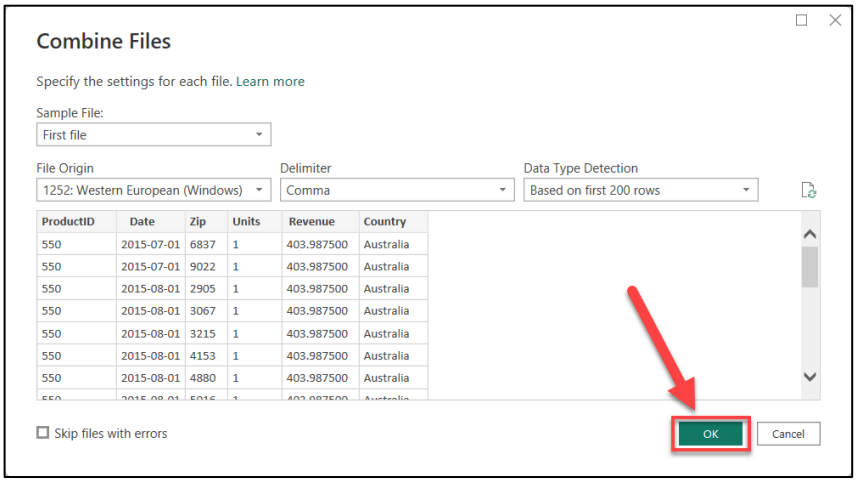


Note: The data in your file for **Date accessed**, **Date modified**, and **Date created** might be different than the dates displayed in the screenshot above.

The **Combine Files** dialog box will open. By default, Power BI will again detect the data type based on the first 200 rows. Notice there is an option to select various file Delimiters. The file we are working with is Comma delimited, so let's leave the default **Delimiter** option as **Comma**.

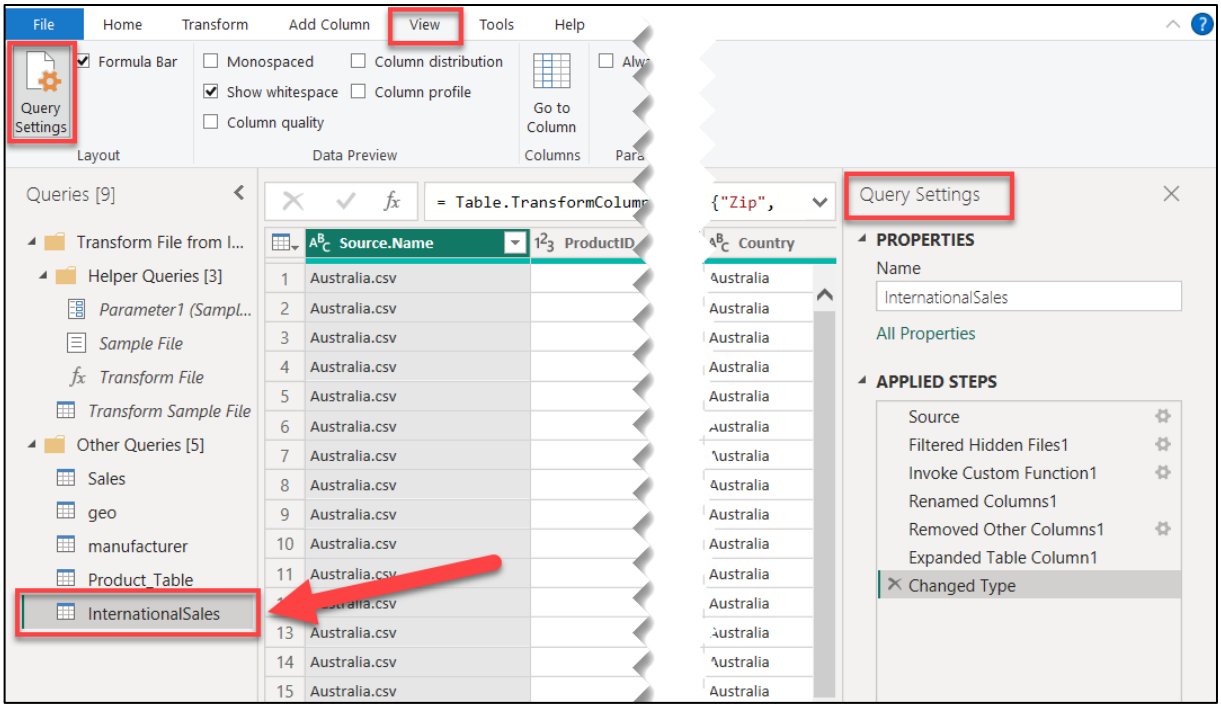
There is also an option to select each individual file in the folder (using **Example File** drop-down) to validate the format of the files.

37. Select the **OK** button located at the bottom of the dialog window.



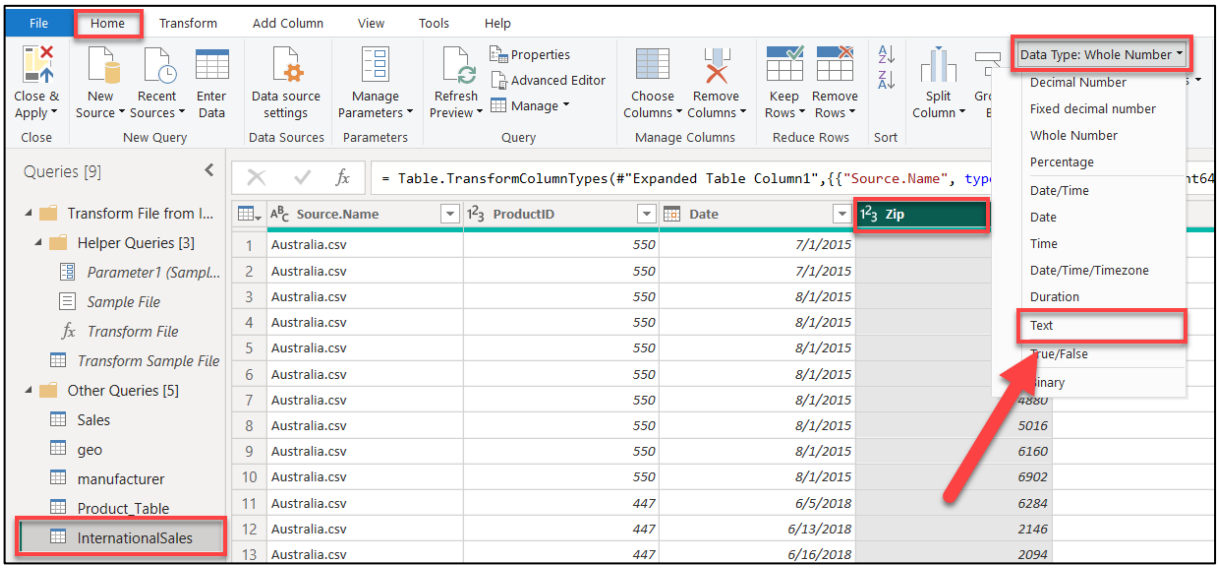
You will now be in the **Power Query Editor** window with a new query named **InternationalSales**.

38. If you do not see the **Queries** pane to the left of the screen, select the **>** (greater than) icon to expand and now view the Queries pane.
39. If you do not see the **Query Settings** pane on the right of the screen, select the **View** tab in the ribbon and choose **Query Settings** to view the pane.
40. Select the **InternationalSales** from the query pane on the left.



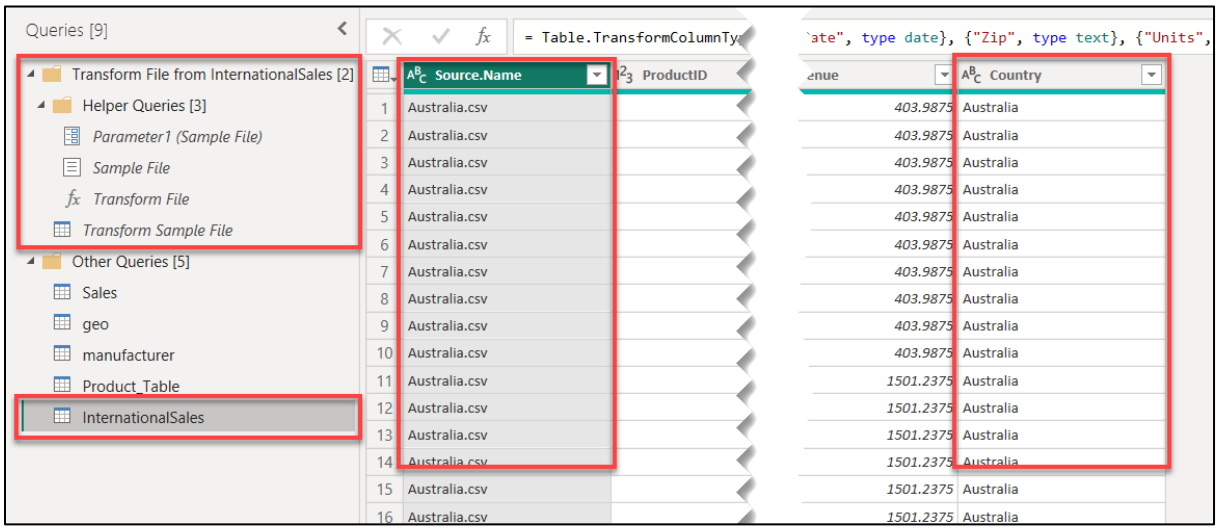
Notice that the **Zip** column is of the **Whole Number** type. Based on the first 200 rows, Power BI thinks the Zip column consists of whole numbers. But zip code could be alpha numeric in some countries or regions or contain leading zeros. If we do not change the data type, we will receive an error when we load the data shortly. So, let's change the Zip column to data type **Text**.

41. Select the **Zip** column within the **InternationalSales** query, and then change the **Data Type** to **Text** using the drop-down under the **Home** tab.
42. The **Change Column Type** dialog box will open. Select the **Replace Current** button.

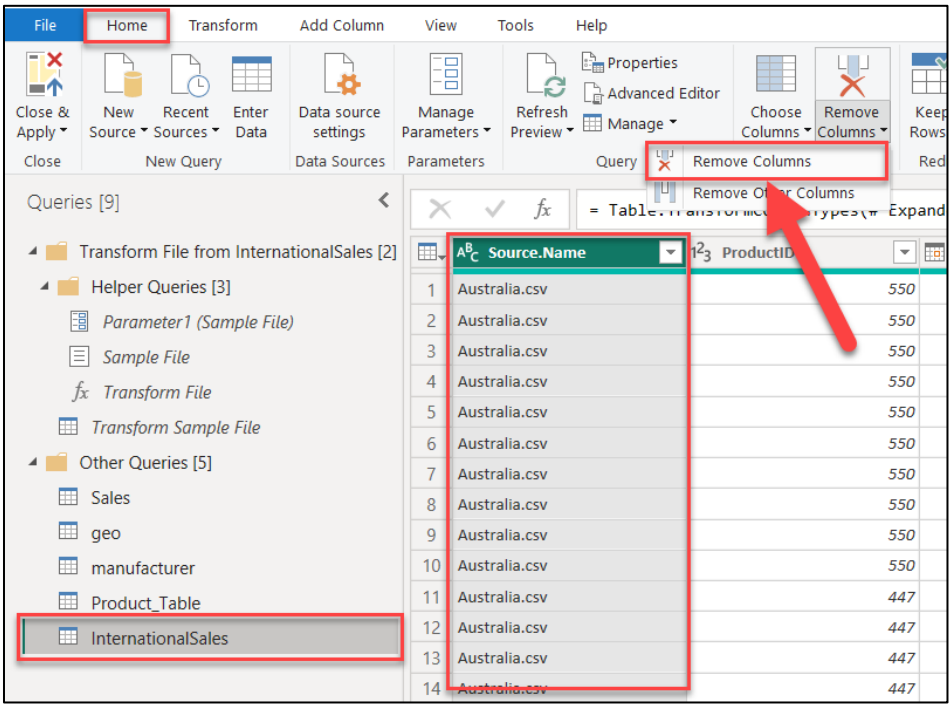


Within the **Queries** pane, notice that a **Transform File from the InternationalSales** folder is created. This contains the function used to load each of the files from the folder.

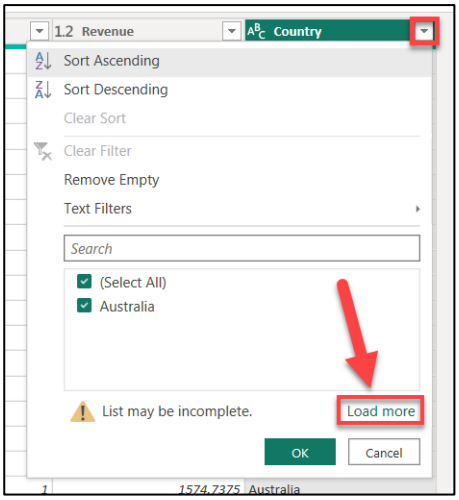
If you compare the **InternationalSales** and the **Sales** table, you will see the **InternationalSales** table contains two new columns: **Source.Name** and **Country**.



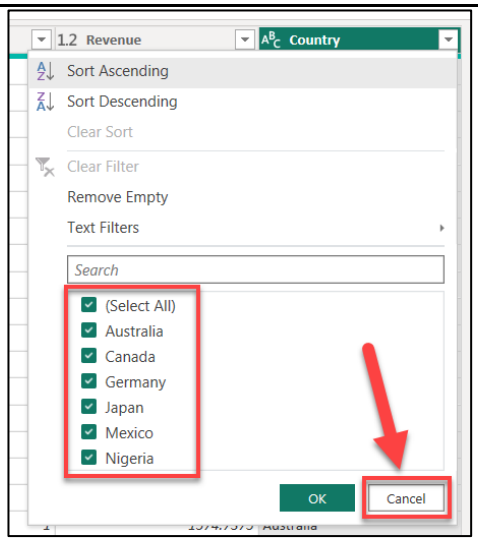
43. We do not need the **Source.Name** column within the **InternationalSales** query. Select the **Source.Name** column and from the ribbon, select the **Home** tab. Choose the **Remove Columns** drop-down, and then select **Remove Columns** again.



Note: You may find that Australia is the only country displayed. This due to the **Power Query Editor** displaying only the first 1000 rows of any data source. To validate you have the data from all country files you can *optionally* select the drop-down menu next to the **Country** column then select **Load more**.



You will now see that **Australia**, **Canada**, **Germany**, **Japan**, **Mexico**, and **Nigeria** are all selected.



If you did this optional step, select **Cancel**.

Power BI Desktop – Data Preparation

In this section, we will explore methods to transform data. Transforming the data by renaming tables, updating data types, and appending tables together ensures that the data is ready to be used for reporting. In some instances, this means cleaning the data up so that similar sets of data can be combined. In other instances, groups of data are renamed so that they are more easily recognized by end users and report writing is simplified.

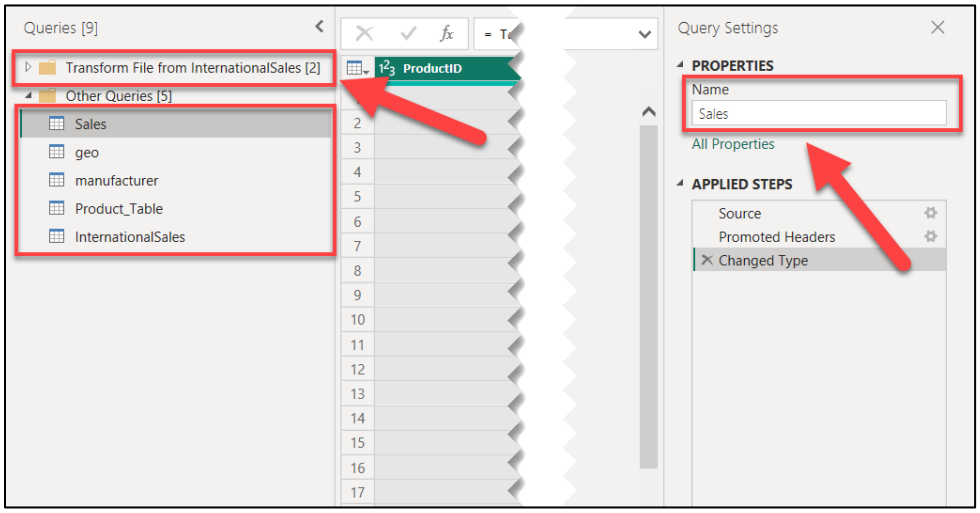
Power BI Desktop – Renaming tables

The Query Editor window should appear as shown below.

- If **formula bar** is disabled, you can turn on the formula bar from the **View** ribbon. This enables you to see the "M" code generated by each click within the ribbons.
- Select the options available within the ribbon - **Home**, **Transform**, **Add Column**, and **View** - to review the various features available.

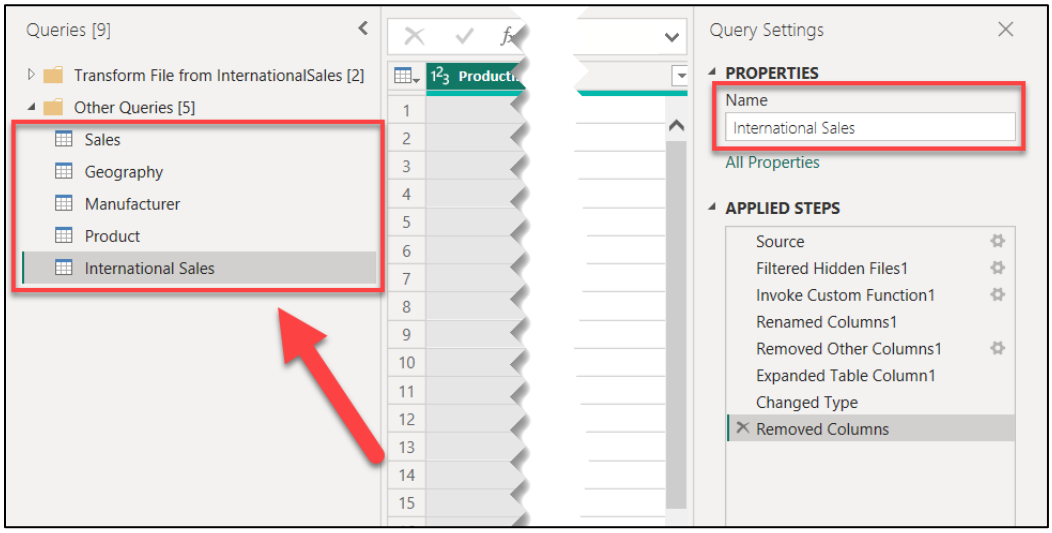
44. Within the **Queries** pane, minimize the **Transform Files from InternationalSales** folder.

45. Select each query name in the **Other Queries** section as you **rename** them in the **next step**.



46. Navigate to **Query Settings** pane to the right of the screen, and then the **Properties** section to rename the queries. **Rename** each query listed within the **Queries** pane to the left of the screen using the new names listed below. You will type the new name within the **Name** property of the **Query Settings** pane and then hit **Enter** on your keyboard. Notice that once the query has been named, it will also change within the Queries pane to the left of the screen.

Initial Name	Final Name
Sales	Sales
geo	Geography
manufacturer	Manufacturer
Product_Table	Product
InternationalSales	International Sales

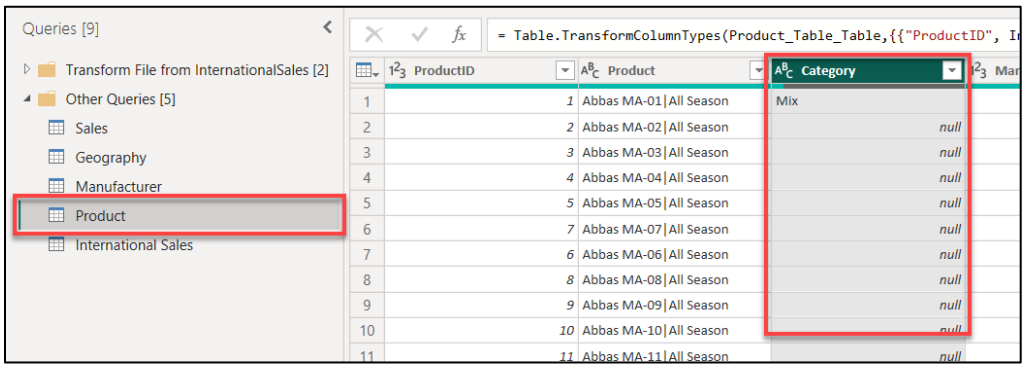


Note: It is a best practice to provide descriptive query and column names. These names are used in visuals and in the Q&A section of Power BI, which is covered in a later lab.

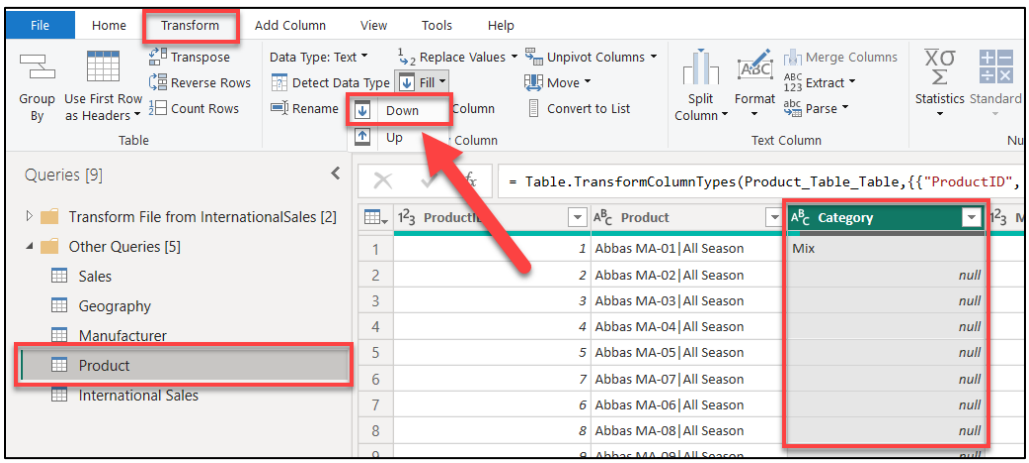
Power BI Desktop – Filling empty values

In our scenario, some of the data is not in the right format. Power BI provides extensive transformation capabilities to clean and prepare data to meet your needs. Let's start by selecting the **Product** query from the **Queries** pane.

Notice that the **Category** column has a lot of **null** values. Hover over the green/gray bar (known as the quality bar) below the column header. This allows you to easily identify errors and empty values in your data previews. It looks like there are values in the Category column only when the value changes. We need to provide data in this column so there are values in each row.



47. With the **Product** query selected from the **Queries** pane, select the **Category** column.
48. From the ribbon, select the **Transform** tab, choose the **Fill** drop-down, and then select the **Down** option.



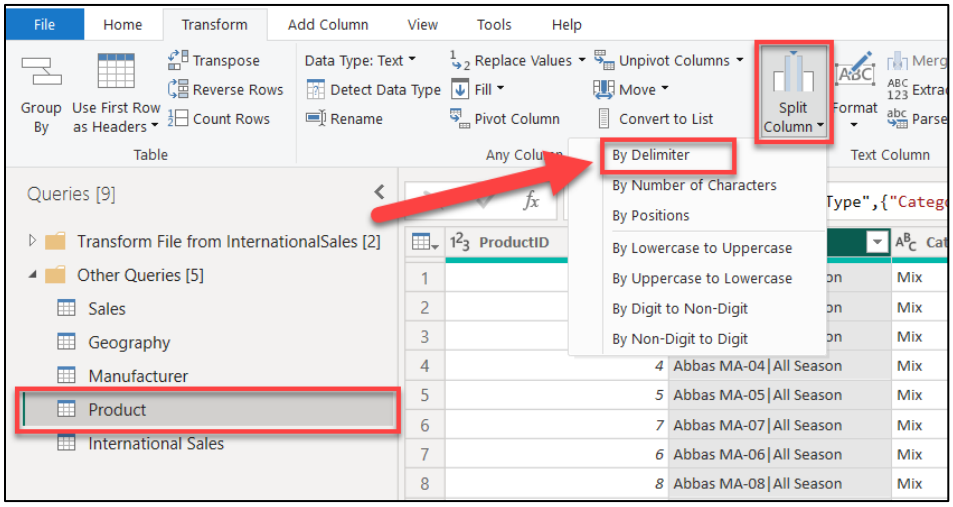
Notice how all the null values are filled with the appropriate Category values.

Note: The fill down operation takes a column and traverses through the values in it to fill any null values in the next rows until it finds a new value. This process continues on a row-by-row basis until there are no more values in that column.

Power BI Desktop – Splitting columns

In the **Product** query, notice the **Product** column. It looks like the product name and product segment are concatenated into one field with a pipe (|) separator. Let's **split** them into **two** columns. This will be useful when we build visuals so we can analyze based on both fields.

49. From the Queries pane to the left, ensure that the **Product** query is selected.
50. Select the **Product** column.
51. From the ribbon, select the **Transform** tab, choose **Split Column**, and then select **By Delimiter**. The **Split Column by Delimiter** dialog box opens.



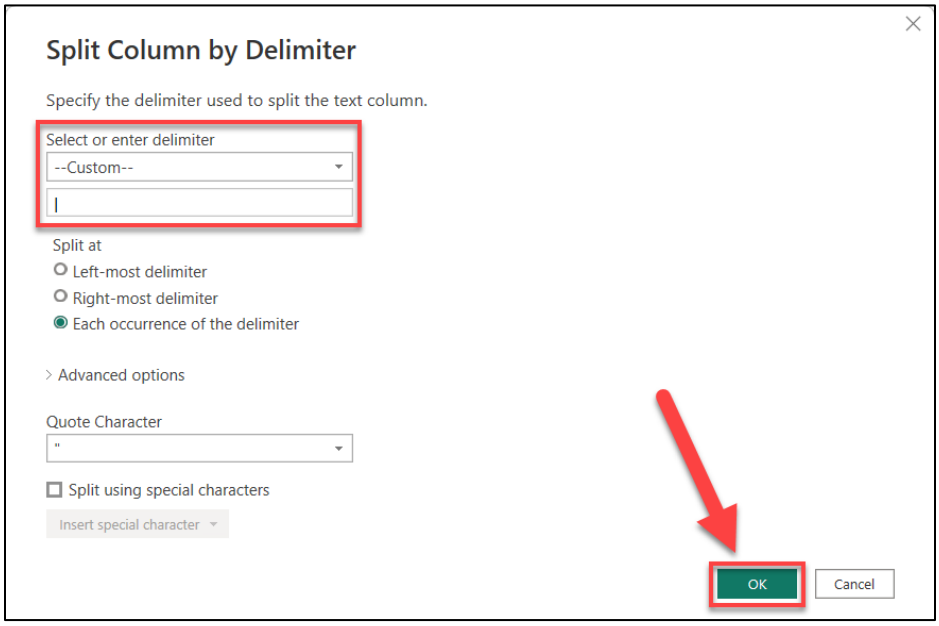
52. Within the dialog box, ensure that **Custom** is selected in the **Select or enter delimiter** drop-down menu.

> **Note**: The **Select or enter delimiter** drop-down menu has some of the standard delimiters like comma, colon, and so on.

53. Notice that within the text box, there is a **hyphen** (-). Power BI assumes we want to split by hyphen. **Remove** the hyphen symbol and enter the **pipe symbol** (**|**).

> **Note**: The pipe symbol is located in the upper right-hand corner of the keyboard below the Backspace button.

54. Then, select **OK**.



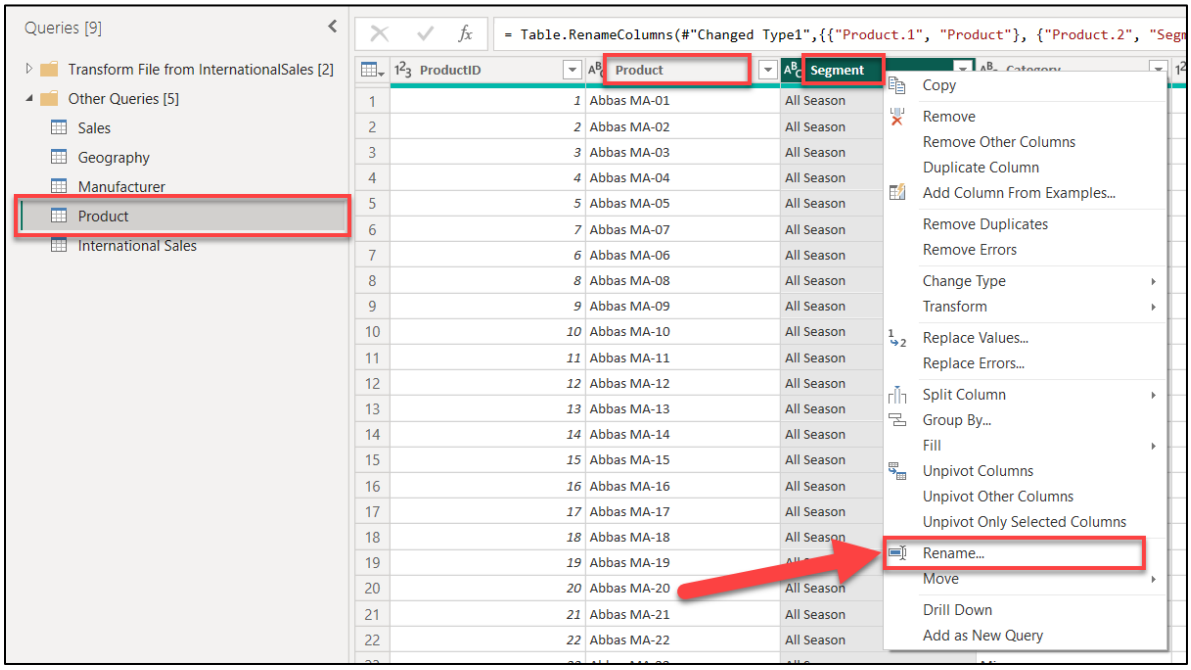
> **Note**: If the delimiter occurs multiple times, the **Split at** section provides the option to split only once (either left most or right most) or the option to split the column on each occurrence of the delimiter.
>
> In this scenario, the delimiter occurs only once, therefore the Product column is split into two columns.

Power BI Desktop – Renaming columns

Let's rename the columns.

55. Select the **Product.1** column, and then **right-click** next to the column name.
56. Choose **Rename…** from the options menu.
57. **Rename** the field to **Product**.

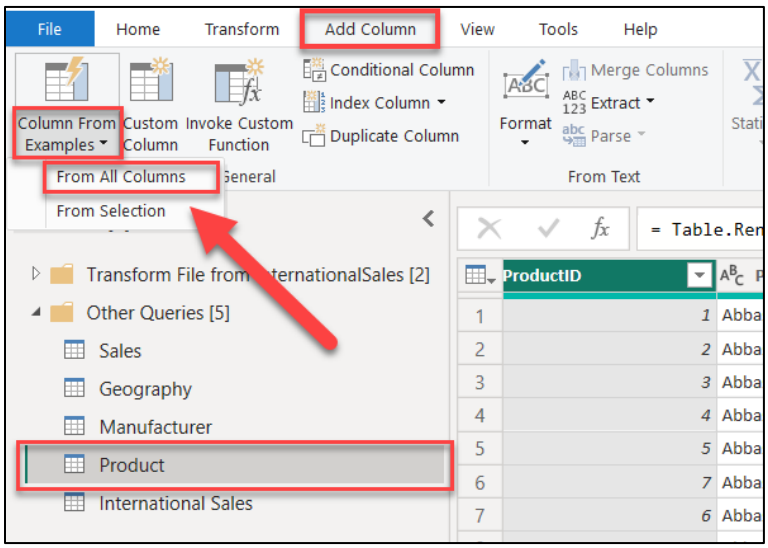
58. Using the same process as in steps 55-56, also rename **Product.2** to **Segment**.



Power BI Desktop – Using Column From Examples to split columns

In the **Product** query, notice that the **Price** column has price and currency concatenated into one field. To do any calculations we only need the numeric value. Therefore, we need to split this field into two columns. We can use the split feature like earlier or we can use **Column From Examples**. **Column From Examples** is handy in scenarios where the pattern is more complex than simply a delimiter.

59. From the **Queries** pane to the left of the screen, ensure that the **Product** query is selected.
60. From the ribbon at the top of the screen, select the **Add Column** tab, choose the **Column From Examples** drop-down, and then select **From All Columns**.

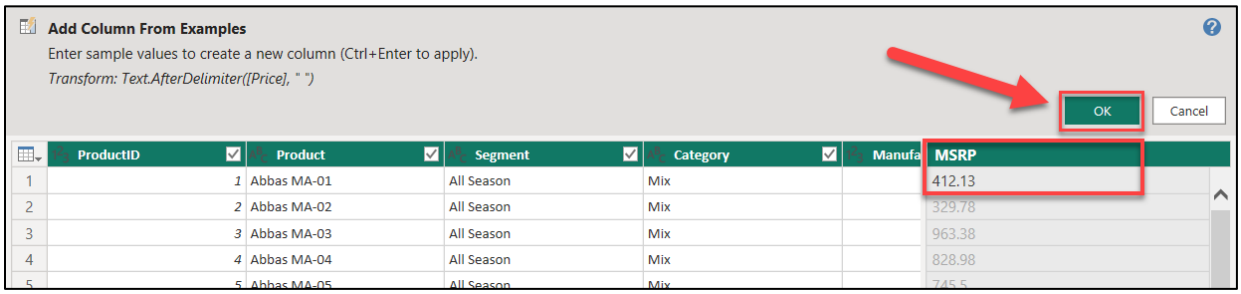


61. Within the **first row** of the newly added **Column1,** enter the first **Price** value, **412.13**, and hit **Enter** on your keyboard.

Notice after you hit Enter, Power BI knows that you want to split the **Price** column. The formula Power BI uses is displayed as well.

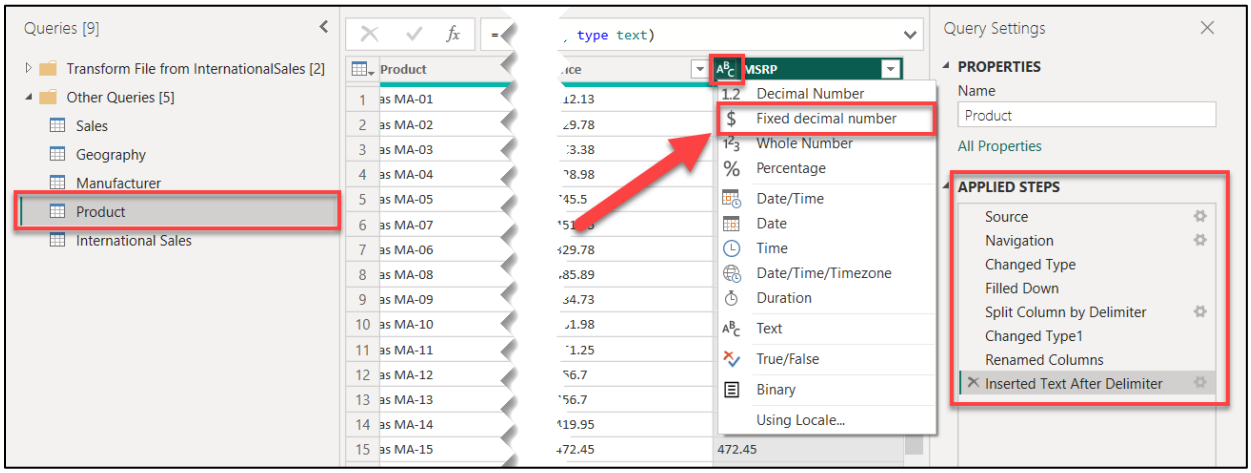
> **Note**: A common mistake that can occur here is the **Column From Example** feature may attempt to auto-type **USD 412.13** with the Intellisense feature**.** DO NOT accept this auto-typed value.

62. **Double-click** the column header **Text After Delimiter** to rename it.
63. **Rename** the column to **MSRP** and select **OK** to apply the changes.



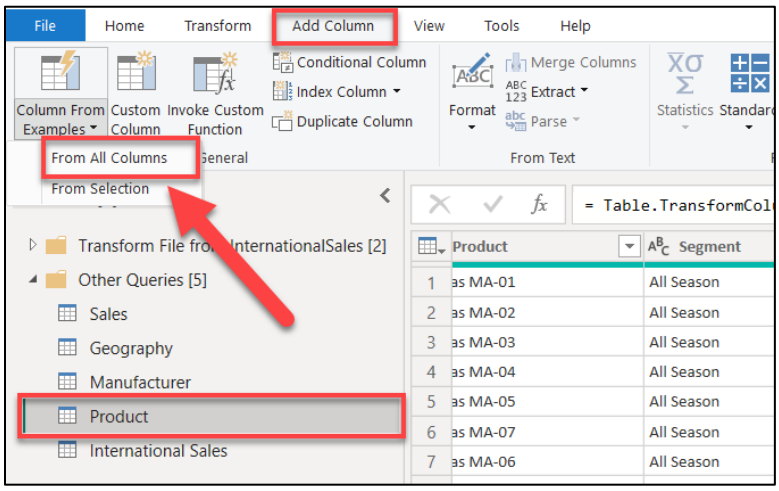
Notice that the **MSRP** field has a Data Type of **Text**. The Data Type that it needs to be is **Decimal**. Let's change it.

64. Select the **ABC** icon to the left of the **MSRP** column header.
65. From the menu, select **Fixed Decimal Number**. Notice that all the steps we performed on the Product query are being recorded under **APPLIED STEPS** in the right panel.

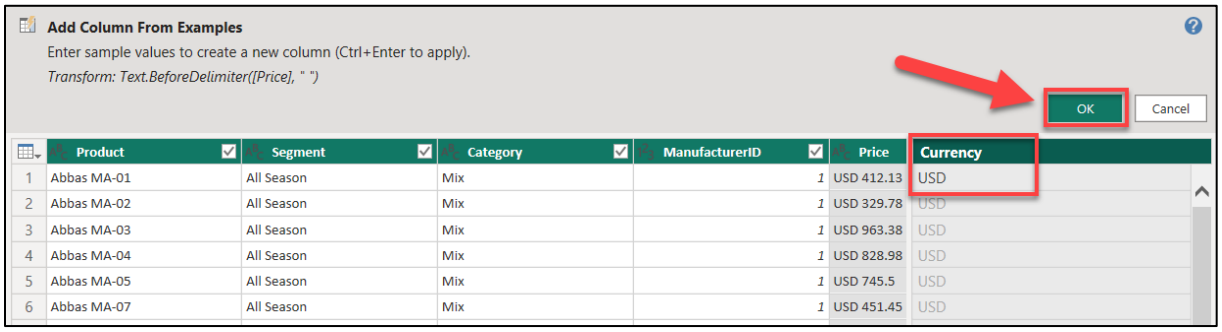


Now let's create a **Currency** column in the same way.

66. With the **Product** query selected, from the ribbon, select the **Add Column** tab, choose the **Column From Examples** drop-down, and then select **From All Columns**.



67. Within the **first row** of the newly added **Column1** enter the first **Currency** value as **USD** and then hit **Enter** on your keyboard.
68. **Rename** the column header **Text Before Delimiter** to now be named **Currency**.
69. Select **Ok** to apply the changes.

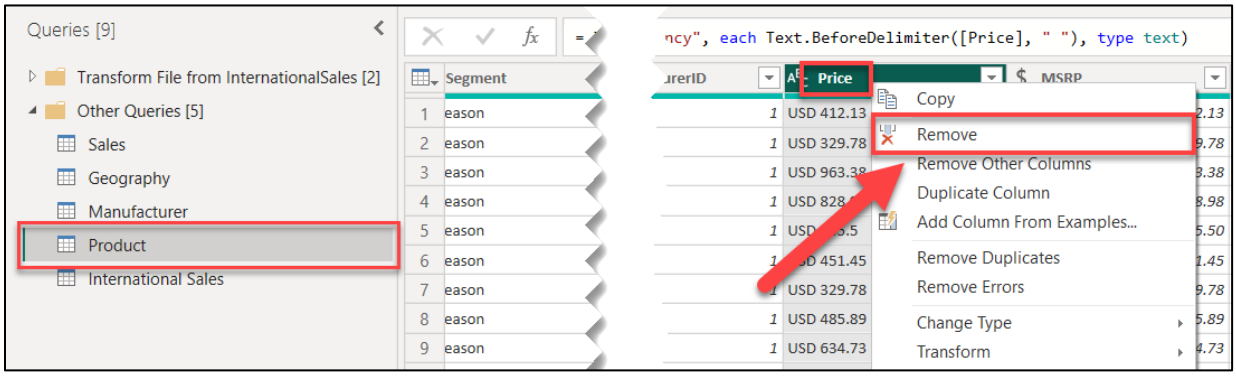


Notice that after you hit **Enter**, Power BI knows you want to split the **Price** column. The formula it uses is displayed above as well.

Now that we have split the **Price** column into the **MSRP** and **Currency** columns, we no longer need the original **Price** column. Let's remove it.

70. From the Queries pane to the left of the screen, select the **Product** Query.
71. **Right-click** on the **Price** column.

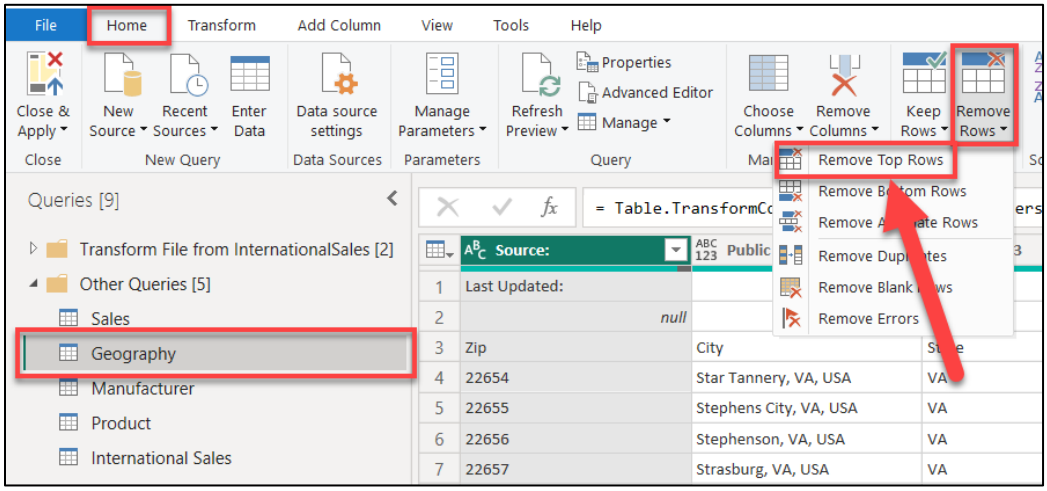
72. Select **Remove** from the options menu.



Power BI Desktop – Removing unwanted rows

In the **Geography** query, notice that the first two rows are informational. They are not part of the data. Similarly, in the Manufacturer query, the last couple of rows are not part of the data. Let's remove them so we have a clean dataset to work with.

73. Within the Queries pane to the left of the screen, select the **Geography** query.
74. From the ribbon, select the **Home** tab, choose the **Remove Rows** drop-down, and then select **Remove Top Rows**.

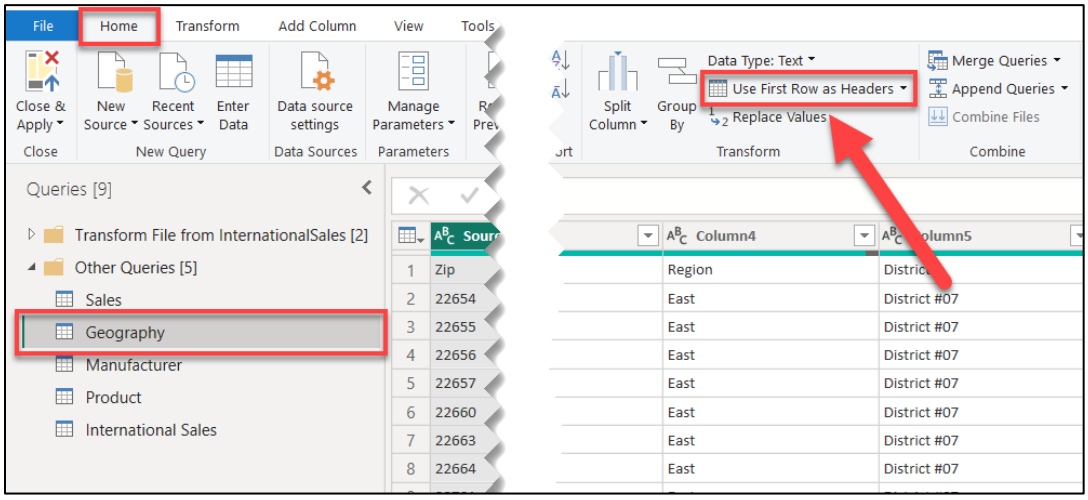


75. The **Remove Top Rows** dialog box opens. Enter **2** in the text box since we want to remove the top informational data row and the blank second row.

76. Then, select **OK**.

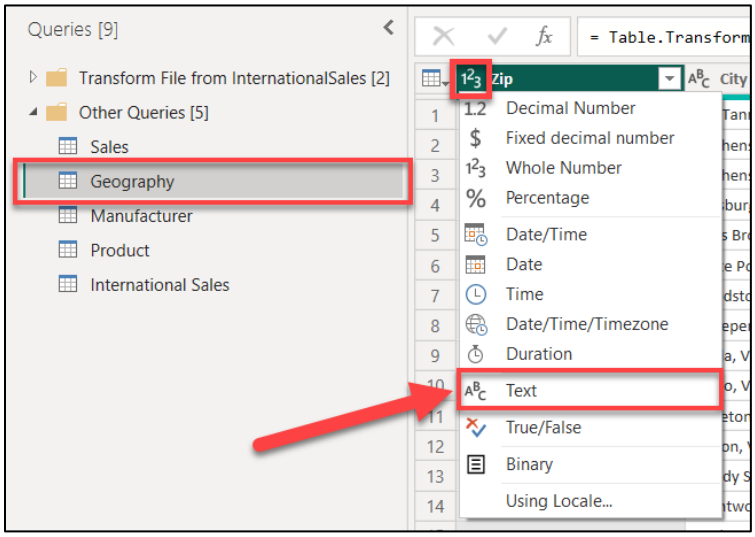
Notice the first row in the Geography query contains the column headers. Let's move them into the column header position.

77. With the **Geography** query selected in the **Queries** pane, from the ribbon at the top of the screen, select the **Home** tab, and then choose **Use First Row as Headers.**

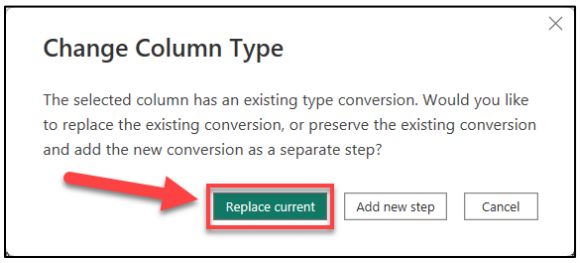


With that step, Power BI will predict the data type of each field again. Notice that the column **Zip** was changed to the **Number** Data Type. Let's change it to **Text** again as we did earlier. If we don't, we will see errors when we load the data.

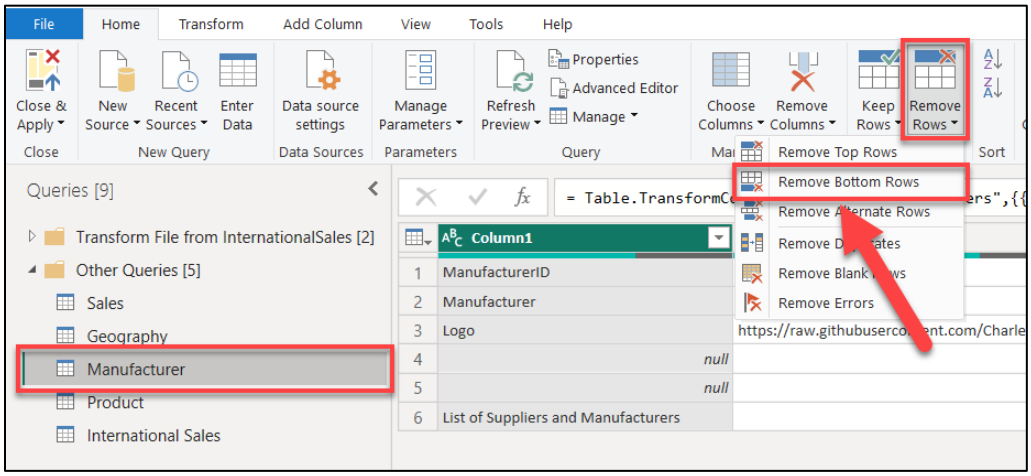
78. Select the **123** icon to the left of the **Zip** column header. From the options menu, select **Text**.



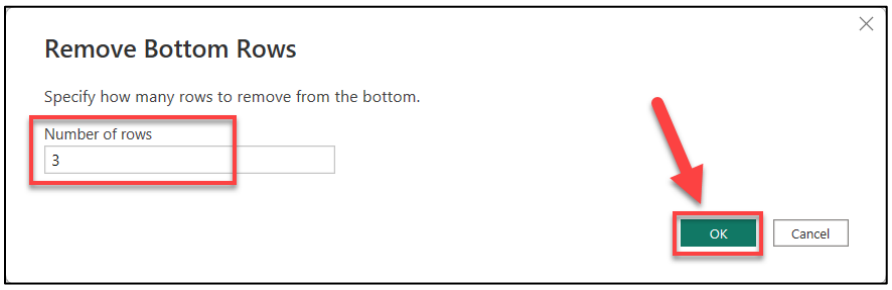
79. Select **Replace Current** in the **Change Column Type** dialog box.



80. From the **Queries** pane, select the **Manufacturer** query. Notice the bottom three rows are not part of the data. Let's remove them.
81. From the ribbon, select the **Home** tab, choose the **Remove Rows** drop-down, and then select **Remove Bottom Rows.**

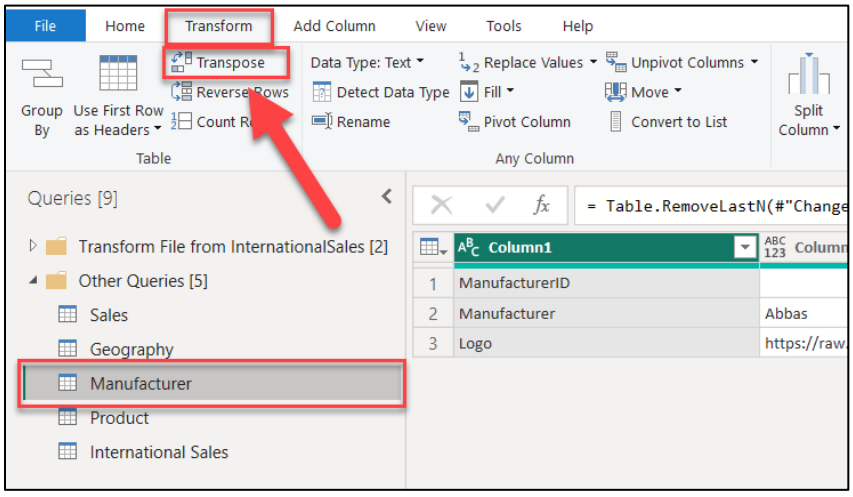


82. The **Remove Bottom Rows** dialog box opens. Enter **3** in the **Number of rows** text box.
83. Then, select **OK**.



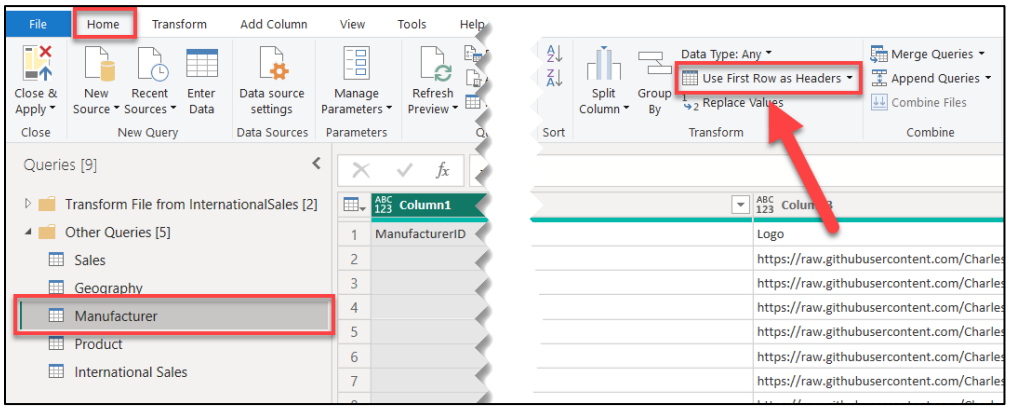
Power BI Desktop – Transposing data

84. From the Queries pane to the left of the screen, select the **Manufacturer** Query. Notice that the **ManufacturerID**, **Manufacturer**, and **Logo** data are laid across in rows. Also notice that the header is not useful. We need to transpose the table to meet our needs.
85. From the ribbon at the top of the screen, select the **Transform** tab, then choose **Transpose**.

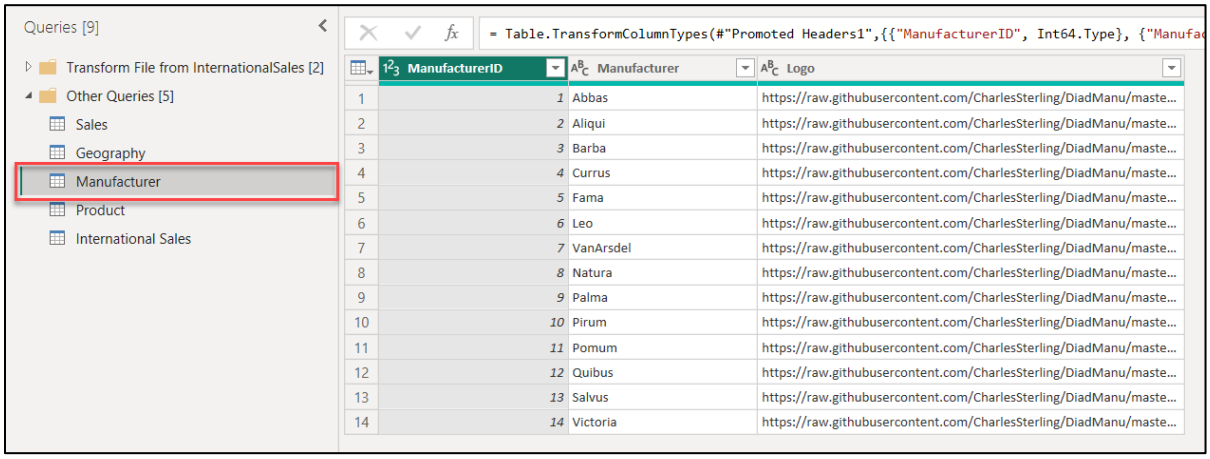


Notice that this transposes the data into columns. Now we need the first row to be the header.

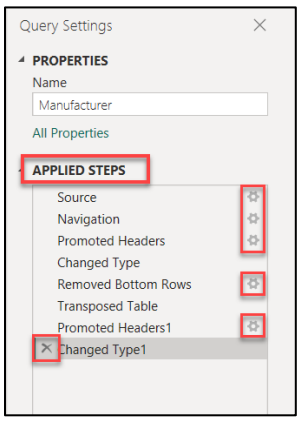
86. From the ribbon at the top of the screen, select the **Home** tab, and then choose the **Use First Row as Headers** button.



Notice that now the **Manufacturer** table is laid out the way we need it with a header and values along columns.



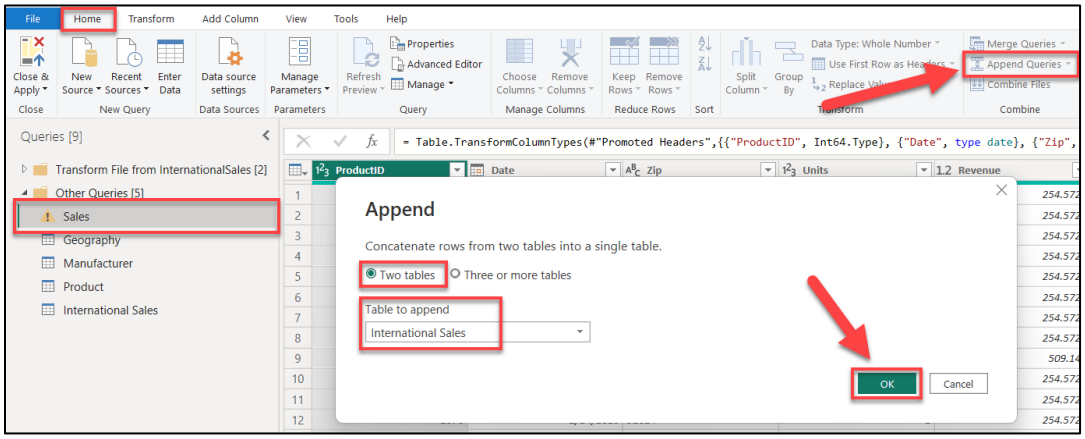
Also, notice that with the **Query Settings** pane, under **APPLIED STEPS**, you will see the list of transformations and steps that have been applied. You can navigate through each change made to the data by selecting the step. Steps can also be deleted by choosing the **X** that appears to the left of the step. The properties of each step can be reviewed by selecting the **gear** to the right of the step.



Power BI Desktop – Appending queries

To analyze the sales of all countries, it is convenient to have a single **Sales** table. To do this, you need to append all the rows from the **International Sales** query to the **Sales** query.

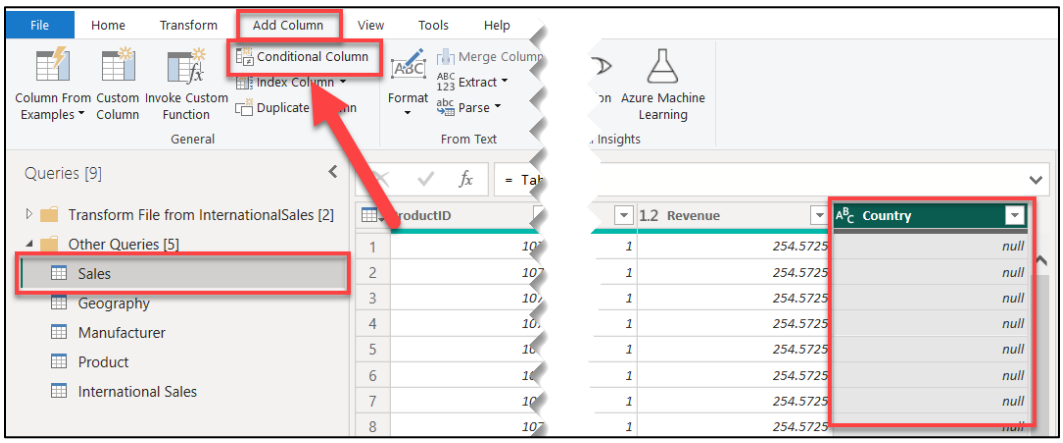
87. Within the **Queries** pane to the left of the screen, select the **Sales** query.
88. From the ribbon at the top of the screen, select the **Home** tab, and then choose **Append Queries**.
89. The **Append** dialog box opens. There is an option to append **Two tables** or **Three or more tables**. Leave **Two tables** selected since we are appending just two tables.
90. From the **Table to append** drop-down, select **International Sales**. Then, select **OK**.



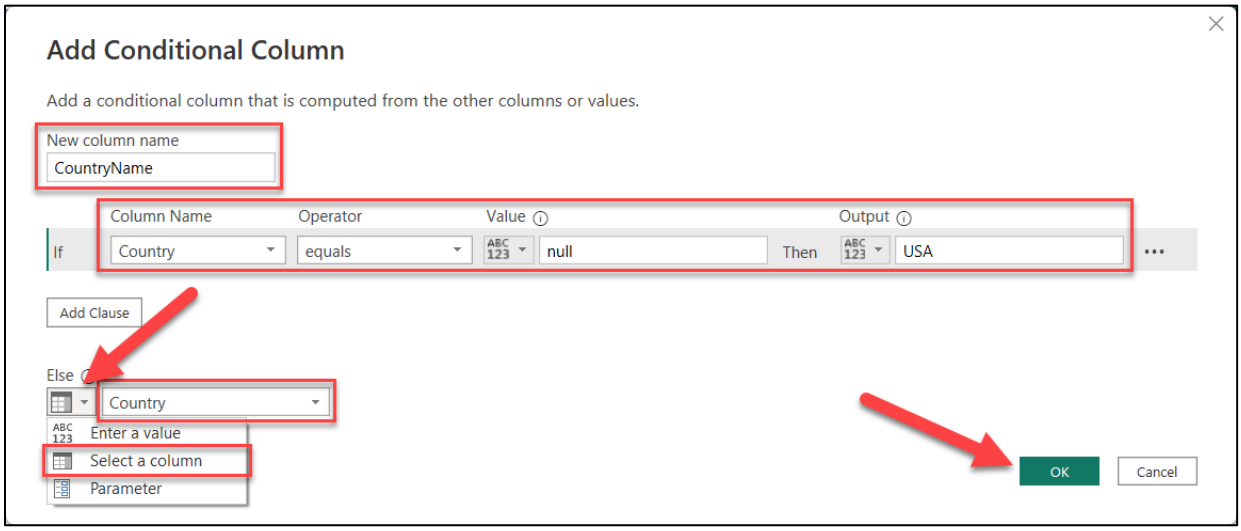
You will now see a new column in the **Sales** table called **Country**. Since the **International Sales** query had the additional column for **Country**, the Power Query Editor added the **Country** column to the newly updated **Sales** table when it loaded the values from the **International Sales** query.

You will see **null** values within the **Country** column by default for the **Sales** table rows because that column did not exist for the table with USA data. We will now add the value **USA** as a data shaping operation.

91. From the ribbon at the top of the screen, select the **Add Column** tab, and then choose **Conditional Column**.

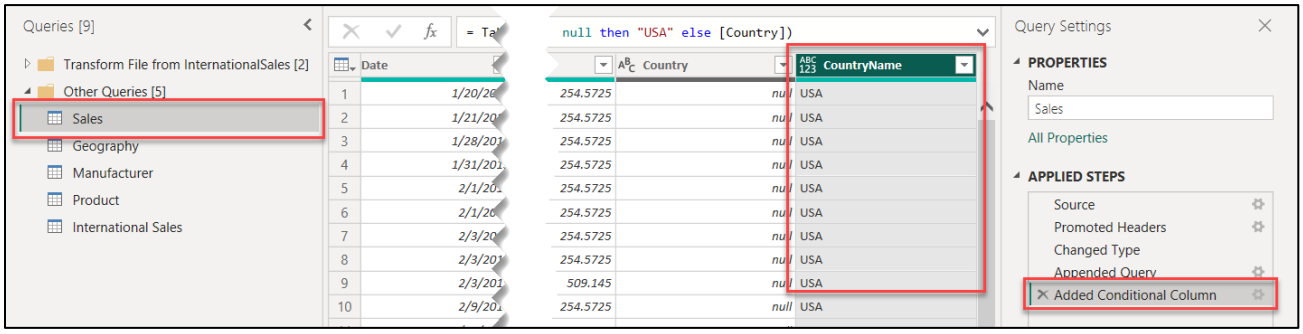


92. In the **Add Conditional Column** dialog box, enter the name of the column as **CountryName**.
93. Select **Country** from the **Column Name** drop-down menu.
94. Choose **equals** from the **Operator** drop-down menu.
95. Enter **null** in the **Value** text box.
96. Enter **USA** in the **Output** text box.
97. Select the ABC 123 drop-down menu under **Else** and then choose the **Select a column** option.
98. Choose **Country** from the column drop-down menu.
99. Then select **OK**.



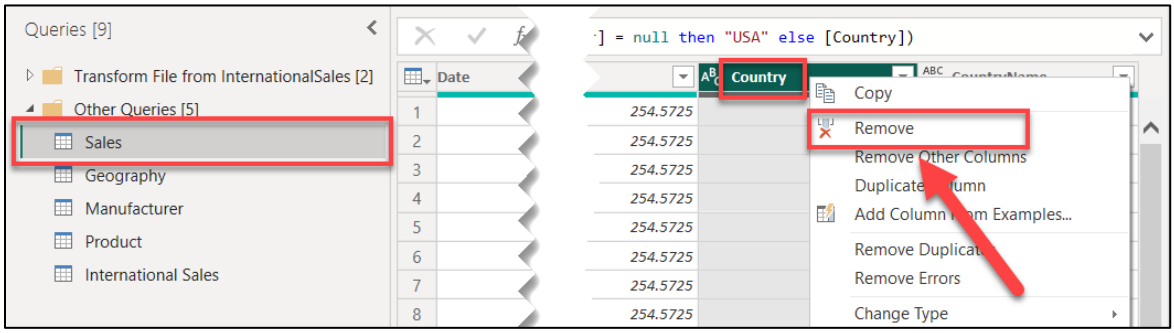
This reads: If the current Country value is equal to null, then the value should return USA; otherwise, if the value is not null, then use the current Country value.

100. You will see the **CountryName** column in the Query editor window. Notice that within the **APPLIED STEPS** list, it has added to the list the action you just completed.



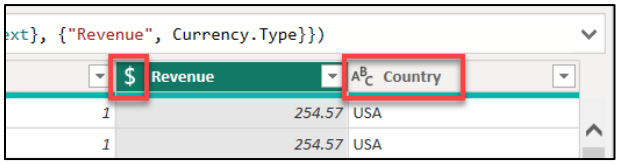
The original **Country** column containing the null values is no longer needed and can be removed from the final table for analysis.

101. **Right-click** on the **Country** column and select **Remove** from the options menu.



With this column now removed, we can now **rename** the **CountryName** column to **Country**.

102. **Right-click** on the **CountryName** column and **rename** it to **Country**.
103. Select the **Data Type icon** to the left of the **Country** column header and change the **Data Type** to **Text**.
104. Next, select the **Data Type icon** to the left of the **Revenue** column header and change the **Data Type** to **Fixed decimal number** because it is a currency field.



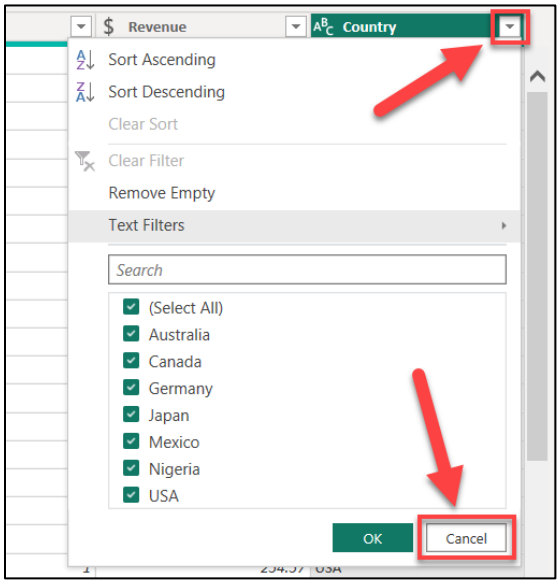
Note: The difference between a Fixed decimal number and a Decimal number is related to the length and precision of the decimal places. https://learn.microsoft.com/en-us/power-bi/connect-data/desktop-data-types#number-types

When the data is refreshed, it will process through all the **APPLIED STEPS** that you have created.

The newly named **Country** column will have names for **all countries**, including the USA. You can validate this by selecting the drop-down menu next to the **Country** column to see the unique values.

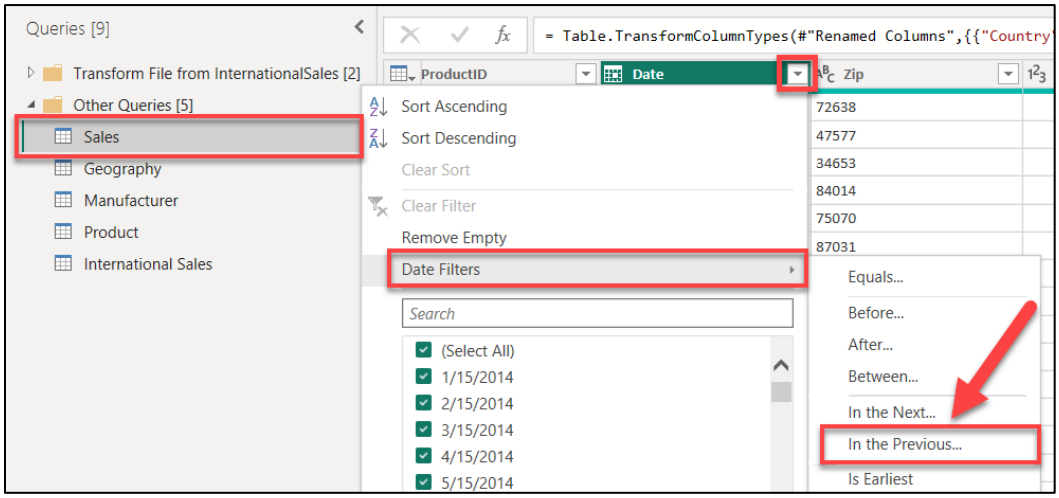
105. At first, you will only see USA data. Select the **drop-down arrow** to the right of the **Country** column header. Select **Load more** to validate your data from all seven countries.

106. Select **Cancel** to close this filter. You **do not** need to apply this filter to the data.



Often, when exploring data, we load a subset of data to test the results. Our dataset has data from 2014 to 2021. For our analysis we want to start with the last three years of data (2019-2021). We don't yet know how many rows will result. We can filter by year to get the subset.

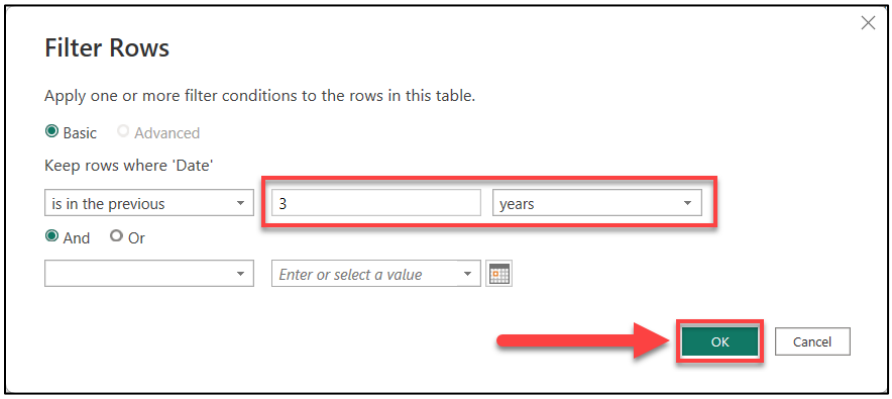
107. Ensure that the **Sales** query is selected within the **Queries** pane to the left of the screen. Select the **drop-down** to the right of the **Date** column header.
108. Select **Date Filters** and then choose **In the Previous…**



109. The **Filter Rows** dialog box opens. Enter **3** in the first text box to the right of **is in the previous**.
110. Select **years** from the drop-down menu displaying the intervals.

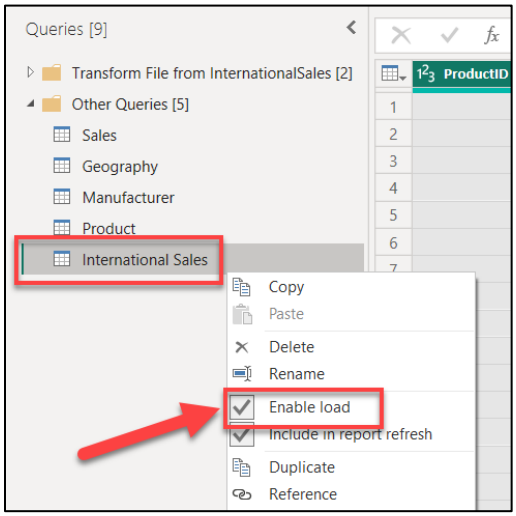
Note: This is saying that we want the three years prior to the current year.

111. Then, select **OK**.



Now that the **International Sales** data is appended to the **Sales** query, in order to avoid duplicating data we should suppress the **International Sales** table from loading into the data model.

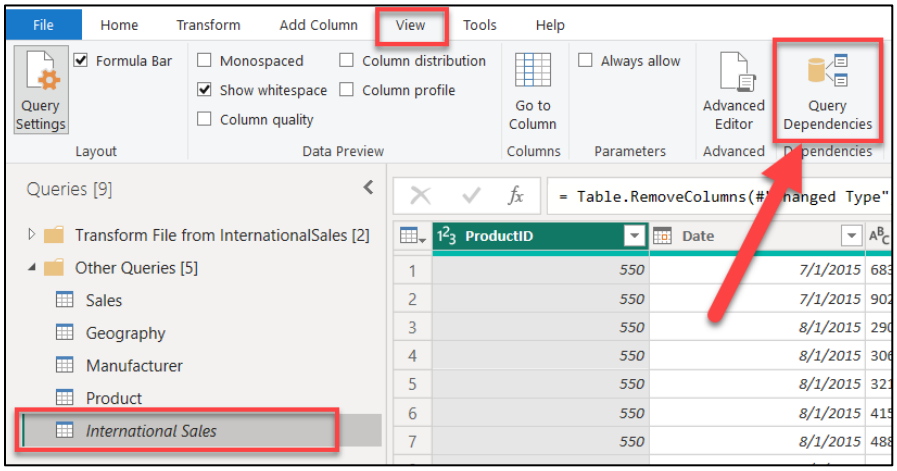
112. From the **Queries** pane to the left of the screen, select the **International Sales** query.
113. **Right-click** on the **International Sales** query within the Queries pane, and then choose **Enable Load** to **deselect** this setting. This will disable loading of the International Sales query into the data model. (You will see the name of this query become italicized in the Queries pane)



> **Note:** The appropriate data from the International Sales table will load onto the Sales table each time the model is refreshed. By removing the International Sales table, we are preventing duplicate data from loading into the model and increasing its file size. In some instances, storing very large amounts of data affects the data model performance.

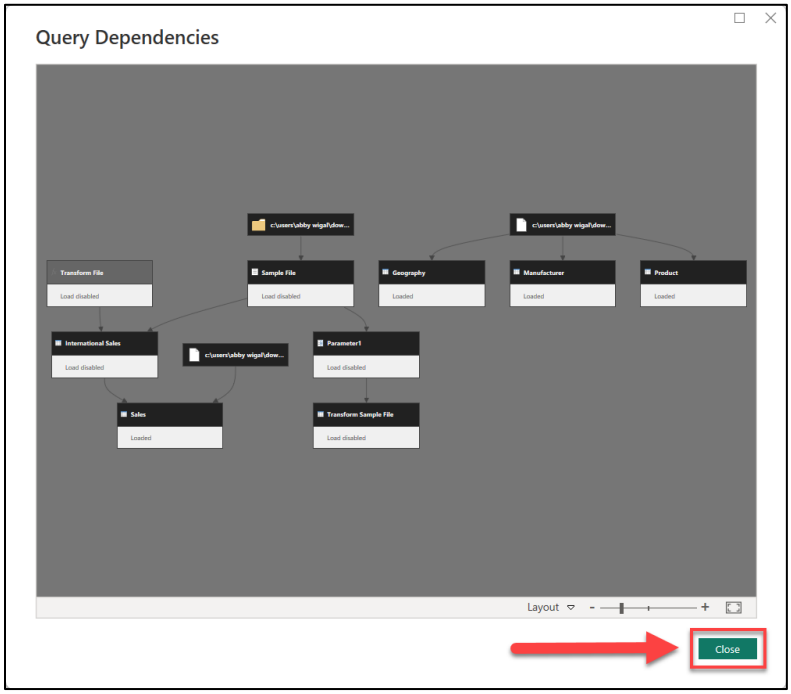
114. You *may* receive a message about **Possible Data Loss Warning**. If so, select **Continue** when this warning appears.

115. Next from the ribbon, select the **View** tab and then choose **Query Dependencies**.



This opens the **Query Dependencies** dialog box. The dialog box shows the source of each query and its dependencies. For example, we see that the **Sales** query has a **CSV file source** and a dependency on the **International Sales** query. This is a useful information to share knowledge with your team members.

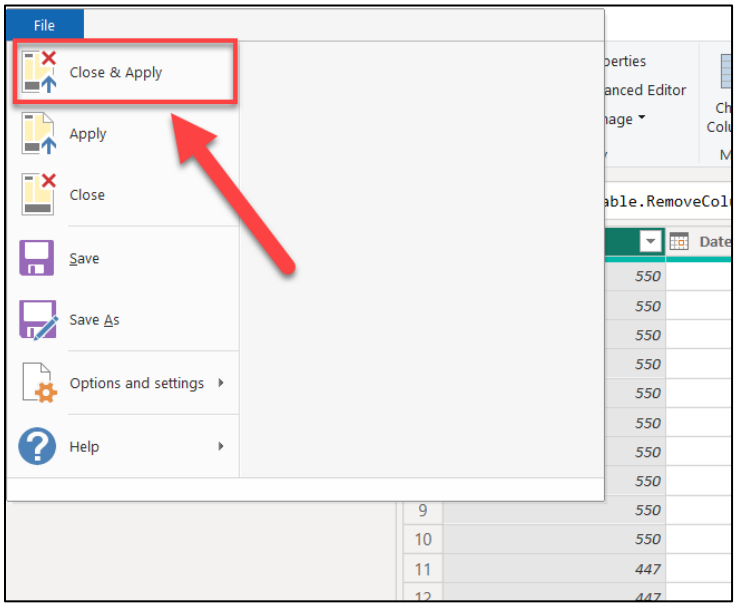
116. Select **Close** at the bottom of the dialog box.



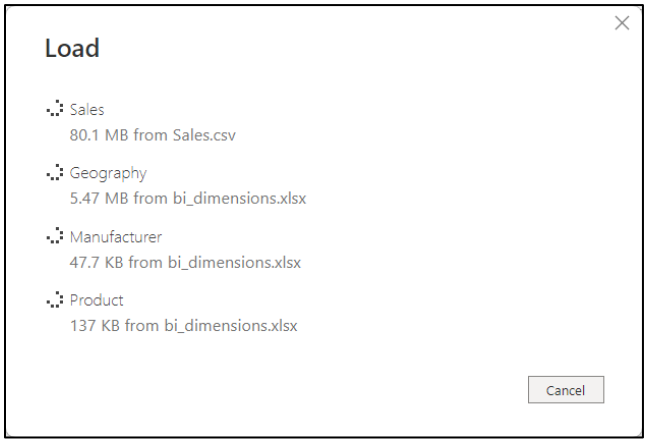
Note: That you can zoom in and out of the **Query Dependencies** view as needed.

You have now successfully completed import and data shaping operations and are ready to load the data into the Power BI Desktop data model to visualize the data.

117. From the ribbon at the top of the screen, select the **File** tab, then choose **Close & Apply**. This will close out the power query window and apply all changes

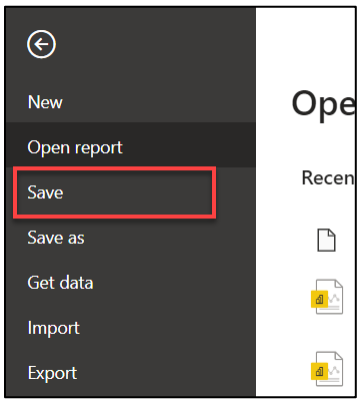


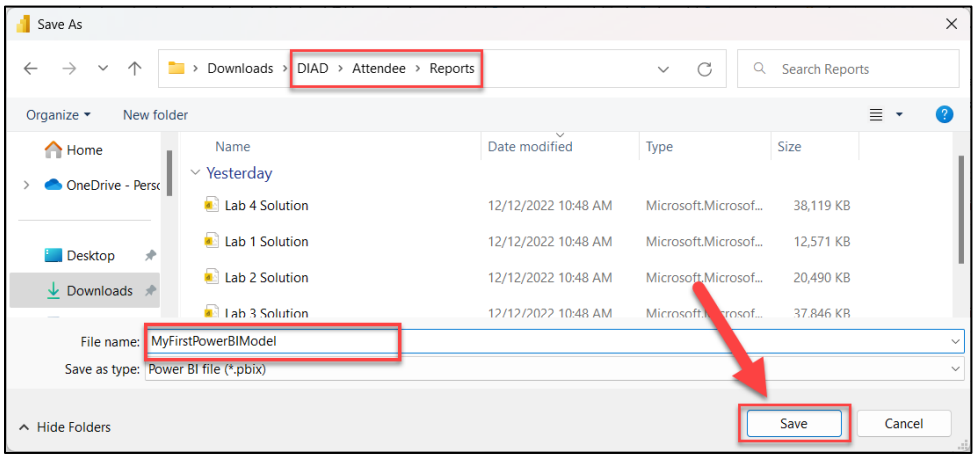
All the data will be loaded in memory in the Power BI Desktop. You will see the progress dialog box with the number of rows being loaded in each table as shown in the Figure. Once the load completes the results of this Power BI Desktop file will be used in Lab 02.



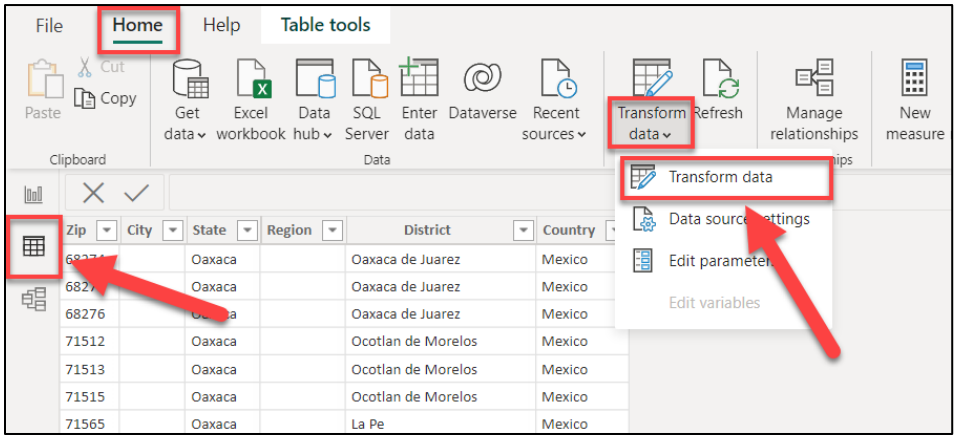
Note: It may take several minutes to load all the tables.

118. Once the data has finished loading, select the **File** tab from the ribbon at the top of the screen.
119. Then, from the options menu to the left, select **Save** to save the file. Name the file as **MyFirstPowerBIModel**. Save the file within the **DIAD Reports** (**\DIAD\Reports**) folder.



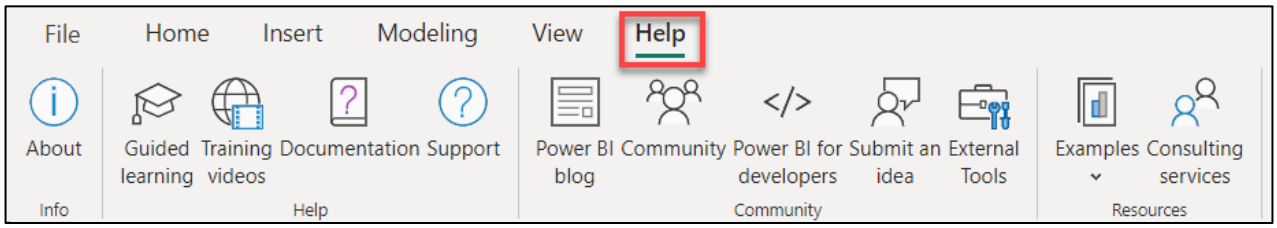


120. Within the **navigation pane** to the left of the screen, select the **Data** icon to view the data that was loaded. If you need to open **Power Query editor** again, navigate to **Home** → **Transform Data** → **Transform data**.



References

Dashboard in a Day introduces you to some of the key functions available in Power BI. In the ribbon of the Power BI Desktop, the Help section has links to some great resources.



Here are a few more resources that will help you with your next steps with Power BI.

- Getting started: http://powerbi.com
- Power BI Desktop: https://powerbi.microsoft.com/desktop
- Power BI Mobile: https://powerbi.microsoft.com/mobile
- Community site https://community.powerbi.com/
- Power BI Getting started support page: https://support.powerbi.com/knowledgebase/articles/430814-get-started-with-power-bi
- Support site https://support.powerbi.com/
- Feature requests https://ideas.powerbi.com/forums/265200-power-bi-ideas
- New ideas for using Power BI https://aka.ms/PBI_Comm_Ideas
- Power BI Courses http://aka.ms/pbi-create-reports
- Power Platform https://powerplatform.microsoft.com/en-us/instructor-led-training/
- Power Apps Business Apps | Microsoft Power Apps
- Power Automate Power Automate | Microsoft Power Platform
- Dataverse What is Microsoft Dataverse? - Power Apps | Microsoft Docs

COPYING OR REPRODUCTION OF THE DEMO/LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS DEMO/LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS DEMO/LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCITONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK. If you give feedback about the technology features, functionality and/or concepts described in this demo/lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE DEMO/LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF DEMO/ LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE DEMO/LAB FOR ANY PURPOSE.

DISCLAIMER

This demo/lab contains only a portion of new features and enhancements in Microsoft Power BI. Some of the features might change in future releases of the product. In this demo/lab, you will learn about some, but not all, new features.