

Power BI Desktop

Power BI Desktop - Accessing Data

In this section, you will import VanArsdel and its competitors USA sales data. Then you import and merging sales data from other countries.

Power BI Desktop - Get Data

Let's start with looking at the data files. The dataset contains sales data of VanArsdel and other competitors. We have 7 years of transaction data by day, product and zip code for each manufacturer. We are going to analyze data from 7 countries.

USA sales data is in a csv file located in /Data/USSales folder.

Sales of all other countries is in /Data/InternationalSales folder. Each countries sales data is in a csv file in this folder.

Product, Geography and Manufacturer information in an excel file in /Data/USSales/bi_dimensions.xlsx.

1. Open [/Data/USSales/bi_dimensions.xlsx](#). Notice the first sheet has **Product** information. The sheet has a header and product data is in a named table. Also notice Category column has a bunch of empty cells.

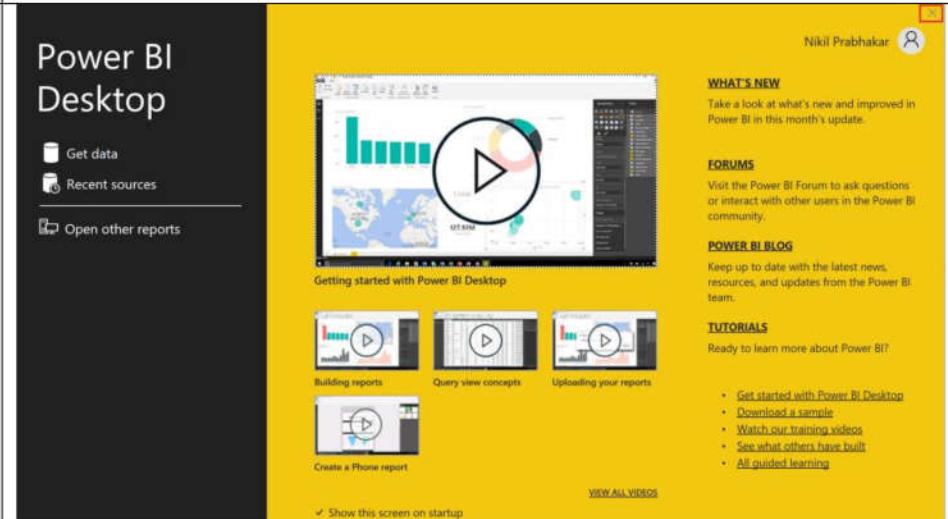
A	B	C	D	E	F
1	Source: Public Database				
2	Last Updated: Monday, February 1, 2016				
3					
4	Zip	City	State	Region	District
5	22654	Star Tannery, VA, USA	VA	East	District #07 USA
6	22655	Stephens City, VA, USA	VA	East	District #07 USA
7	22656	Stephenson, VA, USA	VA	East	District #07 USA
8	22657	Strasburg, VA, USA	VA	East	District #07 USA
9	22660	Toms Brook, VA, USA	VA	East	District #07 USA
10	22663	White Post, VA, USA	VA	East	District #07 USA
11	22664	Woodstock, VA, USA	VA	East	District #07 USA
12	22701	Culpeper, VA, USA	VA	East	District #07 USA
13	22709	Aroda, VA, USA	VA	East	District #07 USA
14	22711	Banco, VA, USA	VA	East	District #07 USA
15	22712	Bealeton, VA, USA	VA	East	District #07 USA

Manufacturer sheet has data laid out across the sheet and with no column headers and it has a couple of blank rows and a note in row 7.

Geo sheet has geography information. The first couple of rows has data details. Actual data starts from row 4.

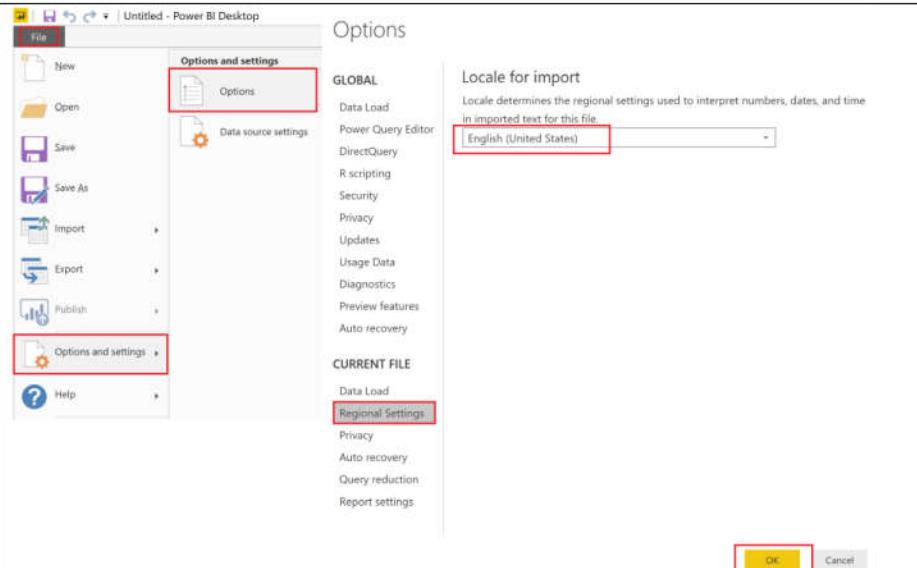
We will start by connecting to data from these different files and perform data cleaning and transformation operations.

2. If you don't have the **Power BI Desktop** open, launch it now.
3. Select **Already have a Power BI Account?** **Sign in** option.
4. **Sign in** using your Power BI credentials.
5. Startup screen opens. Click on **X** on the top right corner of the dialog to close it.



Let's set up the locale to US English, to make it convenient to go through the rest of this lab.

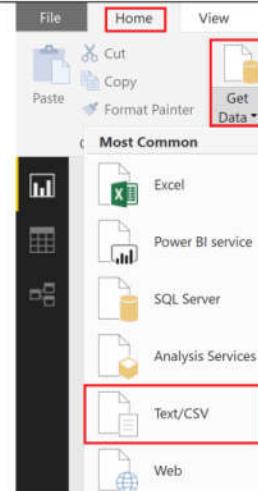
6. From the ribbon, select **File -> Options and settings -> Options**.
7. In the left panel of Options dialog, select **Regional Settings**.
8. From the **Locale** drop down select **English (United States)**.
9. Select **OK** to close the dialog.



First step is to [load data](#) to Power BI Desktop. We will load USA Sales data which is in comma separated value(CSV) files.

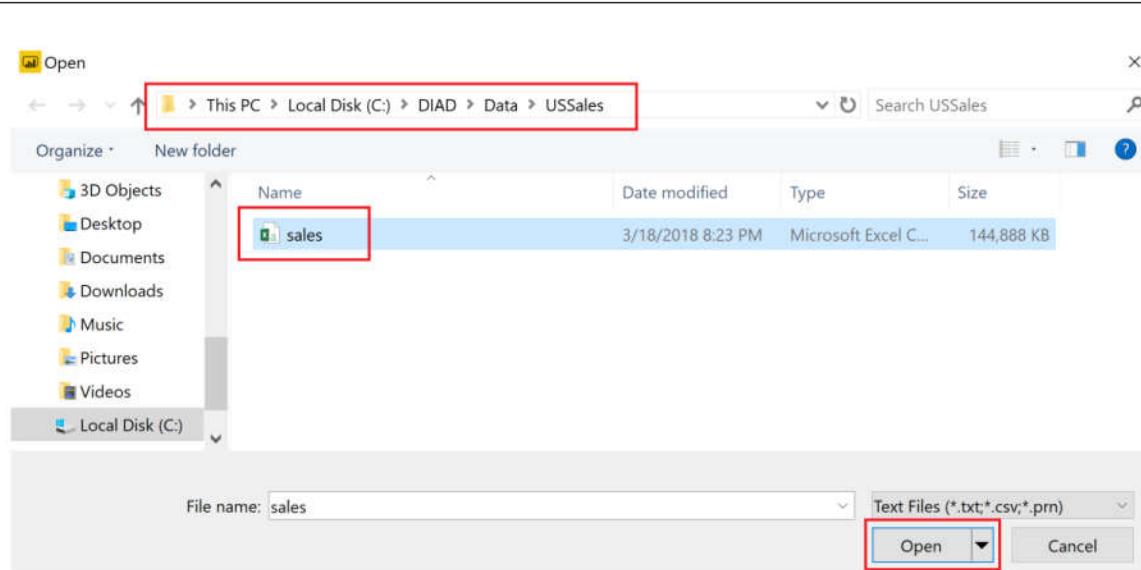
10. From the ribbon, select **Home -> Get Data**.
11. Select **Text/CSV**.

Note: Power BI Desktop has the capability to connect to 70+ data sources. We are using csv and excel data files in this lab for simplicity.



12. Browse to **DIAD\Data\USSales** folder and select **sales.csv**.

13. Click **Open**.



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

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

- **Load**, loads the data from the source into Power BI Desktop for you to start creating reports.
- **Edit** 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.

A screenshot of the Power BI 'Get Data' dialog box for 'sales.csv'. The dialog has sections for 'File Origin' (set to '1252: Western European (Windows)'), 'Delimiter' (set to 'Comma'), and 'Data Type Detection' (set to 'Based on first 200 rows'). Below these, a preview of the data is shown in a table with columns: ProductID, Date, Zip, Units, and Revenue. The preview data consists of 10 rows of sales information. At the bottom right of the dialog, there are three buttons: 'Load' (yellow), 'Edit' (red box), and 'Cancel'.

ProductID	Date	Zip	Units	Revenue
1076	1/20/2011	72638	1	254.5725
1076	1/21/2011	47577	1	254.5725
1076	1/28/2011	34653	1	254.5725
1076	1/31/2011	84014	1	254.5725
1076	2/1/2011	75070	1	254.5725
1076	2/1/2011	87031	1	254.5725
1076	2/3/2011	72019	1	254.5725
1076	2/3/2011	72086	1	254.5725
1076	2/3/2011	77089	2	509.145

- **Cancel** gets you back to the main canvas.

14. Click **Edit** as shown in the screenshot. A new window opens.

You should be in the Query Editor window as shown in the screenshot to the right. Query Editor is used to perform data shaping operations. Notice the sales file you connected to shows as a query in the left panel. You see a preview of the data in the center panel. Power BI predicts data type of each field (based on the first 200 rows) which is indicated next to the column header. In the right panel, steps that Query Editor performs are recorded.

Note: You will be bringing in sales data from other countries as well as performing certain data shaping operations.

15. Notice Power BI has set Zip field to data type Whole Number. To ensure that Zip codes which start with zero don't lose the leading zero, we will format them as text. Highlight the **Zip** column. From the ribbon, select **Home -> Data Type** and update it to **Text**.

16. **Change Column Type** dialog opens. Select **Replace Current** button which overwrites Power BI's predicted datatype.

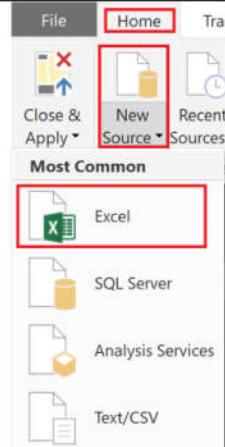
The image contains two screenshots of the Power Query Editor interface. Both screenshots show a table with columns: ProductID, Date, Zip, Units, and Revenue. The 'Zip' column is highlighted with a red border in both images.

Screenshot 1 (Top): The 'Data Type' dropdown for the Zip column is set to 'Whole Number'. The 'Properties' pane on the right shows the 'Name' as 'sales' and the 'Applied Steps' list contains a single step: 'Promoted Headers'. A callout bubble points to the Zip column with the text 'Zip missing leading zero'.

Screenshot 2 (Bottom): The 'Data Type' dropdown for the Zip column is now set to 'Text'. The 'Properties' pane on the right shows the 'Name' as 'sales' and the 'Applied Steps' list contains a step: 'Changed Type'. A callout bubble points to the Zip column with the text 'Leading zero added'.

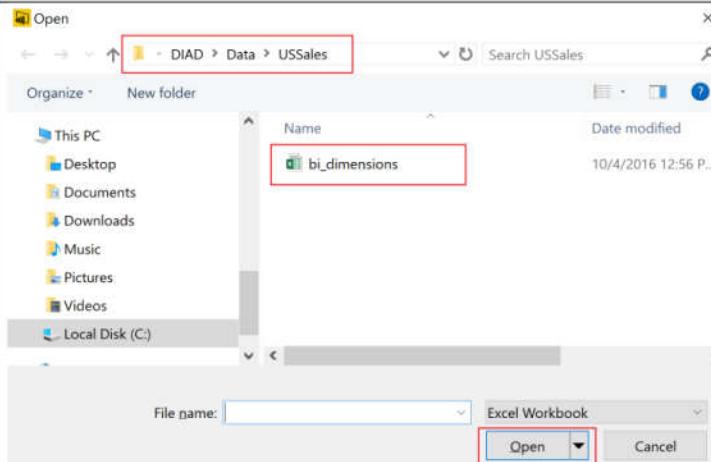
Now let's get the data that is in excel source file.

17. From the ribbon, select **Home** -> **New Source** -> **Excel**.



18. Browse to **DIAD\Data\USSales** folder and select **bi_dimensions.xlsx**.

Navigator dialog opens.



19. Navigator dialog lists 3 sheets that are in the excel workbook. It also lists the Product named table. **Select product** from the left panel and in preview panel notice the first row is the header. This is not part of the data.

20. **Unselect product** from the left panel. **Select Product_Table**. Notice this has only the contents of the named table. This is the data we need.

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

21. From the left panel, **select geo**. In the preview panel notice the first couple of rows are headers that are not part of the data. We will remove them shortly.

22. From the left panel, **select manufacturer**. In the preview panel notice the last couple of rows are footers that are not part of the data. We will remove them shortly.

23. Select **OK**. (Make sure Product_Table, geo and manufacturer are selected in the left panel)

Notice all 3 sheets are added as queries in the Query Editor.

Navigator

The Navigator dialog shows the 'product' query selected in the list of queries. The list includes 'Product Details', 'Product_Table', 'geo', 'manufacturer', and 'product'. The 'product' entry is highlighted with a red box.

product

Column1	Column2	Column3	Column4	Column5
ProductID	Product	Category	ManufacturerID	Price
1	Abbas MA-01 All Season	Mix		1 USD 412.13
2	Abbas MA-02 All Season			1 USD 329.78
3	Abbas MA-03 All Season			1 USD 963.38
4	Abbas MA-04 All Season			1 USD 828.98

Navigator

The Navigator dialog shows the 'Product_Table' query selected in the list of queries. The list includes 'Product Details', 'Product_Table', 'geo', 'manufacturer', and 'product'. The 'Product_Table' entry is highlighted with a red box.

Product_Table

ProductID	Product	Category	ManufacturerID	Price
1	Abbas MA-01 All Season	Mix		1 USD 412.13
2	Abbas MA-02 All Season			1 USD 329.78
3	Abbas MA-03 All Season			1 USD 963.38
4	Abbas MA-04 All Season			1 USD 828.98
5	Abbas MA-05 All Season			1 USD 745.5
7	Abbas MA-07 All Season			1 USD 451.45

Navigator

The Navigator dialog shows the 'geo' query selected in the list of queries. The list includes 'Product Details', 'Product_Table', 'geo', 'manufacturer', and 'product'. The 'geo' entry is highlighted with a red box.

geo

Column1	Column2	Column3	Column4	Column5	Column6
Source:	Public Database		null	null	nu
Last Updated:		2/1/2016	null	null	nu
Zip	City	State	Region	District	Country
22654	Star Tannery, VA, USA	VA	East	District #07	USA
22655	Stephens City, VA, USA	VA	East	District #07	USA

Navigator

The Navigator dialog shows the 'manufacturer' query selected in the list of queries. The list includes 'Product Details', 'Product_Table', 'geo', 'manufacturer', and 'product'. The 'manufacturer' entry is highlighted with a red box.

manufacturer

Column1	Column2	Column3
ManufacturerID		Co
Manufacturer	Abbas	1
Logo		All
		https://raw.githubusercontent.com/CharlesSterling/DiadManu/master/AI ht
List of Suppliers and Manufacturers		null
		null
		null

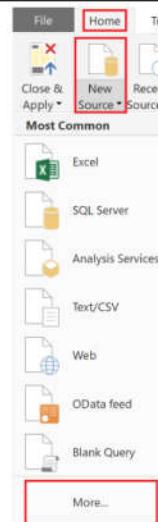
Power BI Desktop - Adding additional data

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 will each put their data.

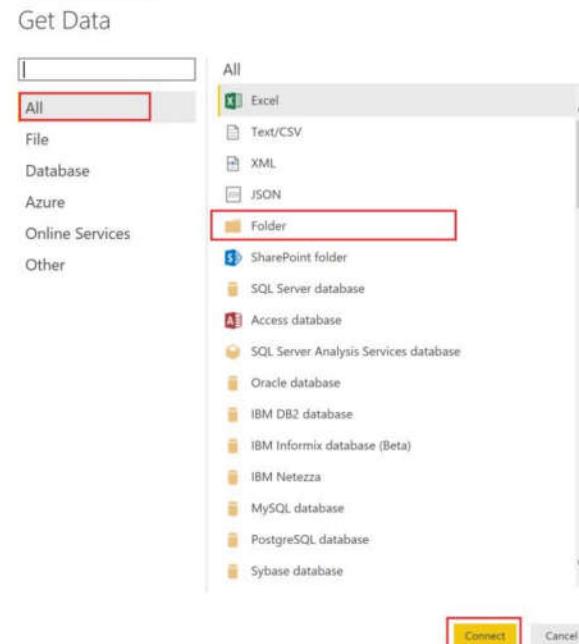
To analyze all the data together you will want to import the new data from each of the subsidiaries and combine it with the US Sales you loaded earlier.

24. Click on the **New Source** drop down in the Home menu tab of the Query Editor.
25. Select **More...** as shown in the figure.

Get Data dialog opens



26. In the Get Data dialog select **Folder** as shown in the diagram.
27. Click **Connect**.

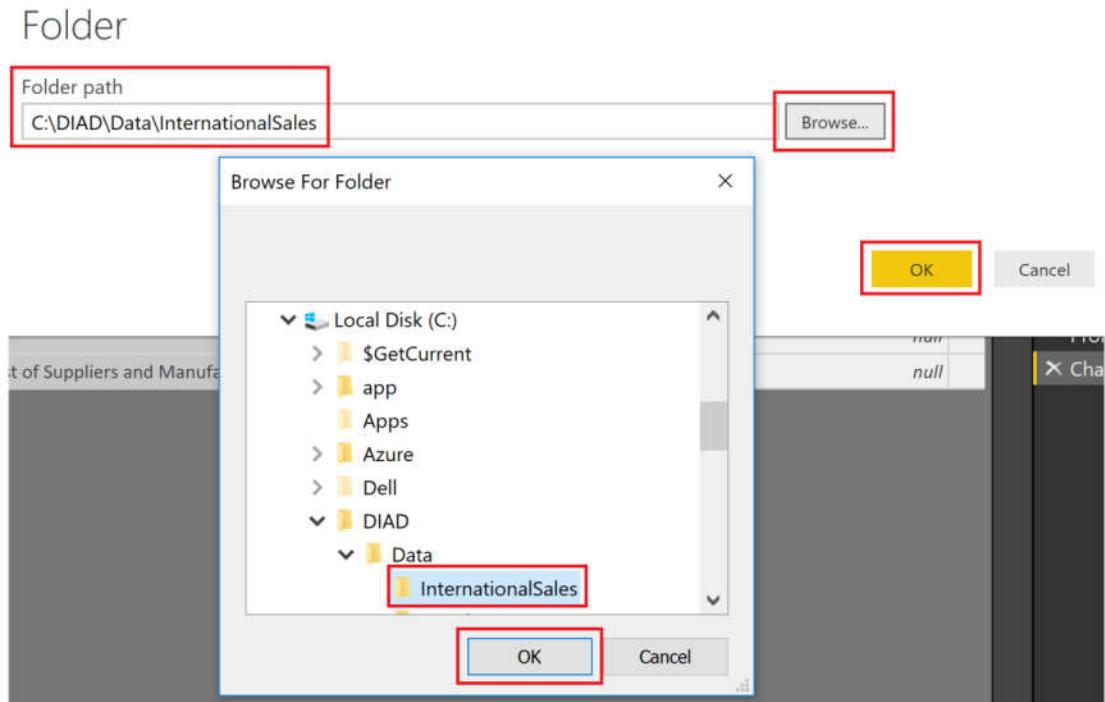


Folder dialog opens.

28. Click **Browse...** button.
29. In the **Browse for Folder** dialog navigate to the location where you unzipped the class files.
30. Open the **DIAD** folder.
31. Open the **Data** folder.
32. Select the **InternationalSales** folder.
33. Click **OK** (to close the **Browse for Folder** dialog box).
34. Click **OK** (to close the **Folder** dialog box).

Note: This approach, uses folders instead of individual files. This will load all files 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.



Dialog displays the list of files in the folder.

35. Since we want to combine data, click **Combine & Edit**.

Note: Date accessed, Date modified and Date created might be different compared to the dates displayed in the screenshot.

C:\DIAD\Data\InternationalSales								
Content	Name	Extension	Date accessed	Date modified	Date created	Attributes	Folder Path	
Binary	Australia.csv	.csv	3/28/2018 9:40:45 PM	3/14/2018 11:07:23 PM	3/28/2018 9:40:45 PM	Record	C:\DIAD\Data\InternationalSales\	
Binary	Canada.csv	.csv	3/28/2018 9:40:45 PM	2/13/2018 9:40:50 PM	3/28/2018 9:40:45 PM	Record	C:\DIAD\Data\InternationalSales\	
Binary	Germany.csv	.csv	3/28/2018 9:40:45 PM	2/13/2018 9:43:42 PM	3/28/2018 9:40:45 PM	Record	C:\DIAD\Data\InternationalSales\	
Binary	Japan.csv	.csv	3/28/2018 9:40:45 PM	2/13/2018 9:44:37 PM	3/28/2018 9:40:45 PM	Record	C:\DIAD\Data\InternationalSales\	
Binary	Mexico.csv	.csv	3/28/2018 9:40:45 PM	2/13/2018 9:45:22 PM	3/28/2018 9:40:45 PM	Record	C:\DIAD\Data\InternationalSales\	
Binary	Nigeria.csv	.csv	3/28/2018 9:40:46 PM	2/13/2018 9:46:39 PM	3/28/2018 9:40:46 PM	Record	C:\DIAD\Data\InternationalSales\	

Combine & Edit **Edit** **Cancel**

Combine Files dialog opens. By default, Power BI again detects 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 Delimiter option as Comma.

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

36. Select OK.

You will be in the **Query Editor** window with a new query called **InternationalSales**.

37. If you do not see the **Queries** pane on left, click on the > icon to expand.

38. If you do not see the Query Settings pane on the right as shown in the figure, click on **View** in the ribbon and click **Query Settings** to see the pane.

39. Click on the Query **InternationalSales**.

Combine Files

Specify the settings for each file. [Learn more](#)

Example File:
First file

File Origin

Delimiter

Data Type Detection

1252: Western European (Windows)

Comma

Based on first 200 rows

ProductID	Date	Zip	Units	Revenue	Country
1070	2017-01-18	2128	1	157.447500	Australia
1070	2017-04-02	2565	1	157.447500	Australia
1070	2017-04-25	4581	4	629.790000	Australia
1070	2017-04-26	1189	2	314.895000	Australia
1070	2017-04-26	3981	1	157.447500	Australia

Skip files with errors

OK

Cancel

The screenshot shows the Power BI Query Editor interface. The ribbon at the top has 'View' highlighted. The 'View' tab contains options like 'Formula Bar', 'Monospaced', 'Show whitespace', and 'Always allow'. Below the ribbon is the 'Layout' tab, which includes 'Data Preview', 'Columns', 'Parameters', 'Advanced', and 'Dependencies'. On the left, the 'Queries [9]' pane is expanded, showing 'Transform File from Internation...', 'Sample Query [2]', 'Transform Sample File from...', 'Other Queries [5]', and 'InternationalSales' (which is selected and highlighted in red). In the center, a data preview grid shows columns: Source.Name, ProductID, Date, Zip, Units, Revenue, and Country. The data consists of 13 rows of sales data for Australia. On the right, the 'QUERY SETTINGS' pane is open, showing 'PROPERTIES' (Name: InternationalSales, All properties) and 'APPLIED STEPS' (Source, Invoke Custom Function1, Renamed Columns1, Removed Other Columns1, Expanded Table Column1, Changed Type).

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

40. Highlight the **Zip** column and change the **Data Type** to **Text**.
41. **Change Column Type** dialog opens. Select **Replace Current** button.

IMPORTANT!
Changing the data type is a big deal to use later

A ^b c. Source.Name	ProductID	Date	Zip	Units	Revenue	Country
Australia.csv	1070	1/18/2017	2128	1	157.4475	Australia
Australia.csv	1070	4/2/2017	2565	1	157.4475	Australia
Australia.csv	1070	4/25/2017	4581	4	629.79	Australia
Australia.csv	1070	4/26/2017	1189	2	314.895	Australia
Australia.csv			3981	1	157.4475	Australia
Australia.csv			1189	1	157.4475	Australia
Australia.csv			5010	1	157.4475	Australia
Australia.csv			6646	1	157.4475	Australia
Australia.csv			7212	1	157.4475	Australia
Australia.csv			4423	1	157.4475	Australia
Australia.csv			2155	1	157.4475	Australia
Australia.csv			1128	1	157.4475	Australia
Australia.csv			2455	1	157.4475	Australia
Australia.csv			3194	1	157.4475	Australia

In Queries panel, notice Transform File from InternationalSales folder is created. This contains the function used to load each of the files in the folder.

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

A ^b c. Source.Name	ProductID	Date	Zip	Units	Revenue	Country
Australia.csv	1070	1/18/2017	2128	1	157.4475	Australia
Australia.csv	1070	4/2/2017	2565	1	157.4475	Australia
Australia.csv	1070	4/25/2017	4581	4	629.79	Australia
Australia.csv	1070	4/26/2017	1189	2	314.895	Australia
Australia.csv	1070	4/26/2017	3981	1	157.4475	Australia
Australia.csv	1070	4/30/2017	1189	1	157.4475	Australia
Australia.csv	1070	5/14/2017	5010	1	157.4475	Australia
Australia.csv	1070	5/27/2017	6646	1	157.4475	Australia
Australia.csv	1070	5/30/2017	7212	1	157.4475	Australia
Australia.csv	1070	6/7/2017	4423	1	157.4475	Australia
Australia.csv	1070	6/20/2017	2155	1	157.4475	Australia
Australia.csv	1070	6/27/2017	1128	1	157.4475	Australia
Australia.csv	1070	6/27/2017	2455	1	157.4475	Australia

42. We do not need Source.Name column.
 Select **Source.Name** column. From the ribbon, select **Home** -> **Remove Columns** -> **Remove Columns**.

Source.Name	ProductID	Date	Zip
Australia.csv	1070	1/18/2017	2128
Australia.csv	1070	4/2/2017	2565
Australia.csv	1070	4/25/2017	4581
Australia.csv	1070	4/26/2017	1189
Australia.csv	1070	4/26/2017	3981
Australia.csv	1070	4/30/2017	1189
Australia.csv	1070	5/14/2017	5010
Australia.csv	1070	5/27/2017	6646
Australia.csv	1070	5/30/2017	7212
Australia.csv	1070	6/7/2017	4423
Australia.csv	1070	6/20/2017	2155
Australia.csv	1070	6/27/2017	1128
Australia.csv	1070	6/27/2017	2455

43. Click on the drop down next to **Country** column to see the unique values.
 44. You will only see Australia as shown in the figure. Click on **Load more** to validate you have data from various countries included.

Country

- (Select All)
- Australia

Load more

You will see the countries, Australia, Canada, Germany, Japan, Mexico and Nigeria.

45. Click OK.

Note: You can perform various types of filters, sorting operations using the drop down to verify the imported data.

The screenshot shows the Power BI Desktop interface. On the left, the 'Queries [9]' pane is open, displaying a list of queries including 'Transform File from Internati...', 'Sample Query [2]', 'Other Queries [5]', and 'InternationalSales'. The 'InternationalSales' query is selected and highlighted with a red box. In the center, a table view displays data with columns: ProductID, Date, Zip, Units, Revenue, and Country. To the right of the table, a filter dialog is open for the 'Country' column. The dialog shows a list of countries: '(Select All)', 'Australia', 'Canada', 'Germany', 'Japan', 'Mexico', and 'Nigeria', all of which are checked. The 'OK' button at the bottom of the dialog is also highlighted with a red box.

Power BI Desktop – Data Preparation

In this section, we will explore methods to [transform data in the data model](#). 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 are combined. In other instances, groups of data are renamed so that they are more recognizable by end users and simplifies report writing.

Power BI Desktop - Renaming tables

The Query Editor window should appear as shown in the diagram.

- 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 on the ribbons.
- Select the options available on the ribbon – **Home, Transform, Add Column and View** to notice the various features available.

1. Under **Queries** panel, **minimize Transform** Files from InternationalSales folder.
2. Select each query name in the **Other Queries** section.
3. **Rename** them in the **Query Settings** -> **Properties** section as shown below:

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

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

The screenshot shows the Power BI Desktop interface with the Query Editor open. The ribbon at the top has 'View' selected. The 'Queries' pane on the left lists 'Transform File from InternationalSales [3]' and 'Other Queries [5]', with 'Sales' selected. The main area displays a table with columns: ProductID, Date, A#, Zip, Units, 1.2, Revenue. The 'QUERY SETTINGS' pane on the right shows the 'Properties' section with 'Name' set to 'Sales'. The 'Applied Steps' pane shows the transformation steps: Source (InternationalSales), Promoted Headers, and Changed Type.

Power BI Desktop – Using Fill feature

Some of the data provided is not in the right format. Power BI provides extensive transformation capabilities to clean and prepare the data to meet our needs. Let's start with Product query. Notice that Category column has a lot of null values. Looks like there are values in Category column only when the value changes. We need to fill it down to have values in each row.

4. From the left panel, select **Product** Query.
5. Select **Category** column.
6. From the ribbon select **Transform -> Fill -> Down**.

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

The screenshot shows the Power BI Desktop interface with two main sections: the left pane and the main workspace.

Left Pane: Shows the 'Queries [9]' list. The 'Product' query is selected and highlighted with a red box.

Main Workspace: Displays the 'Product' query table with columns: ProductID, Product, Category, ManufacturerID, and Price.

Data Preview: The 'Category' column contains several null values, which are highlighted with a red box. The first row has 'Mix' in the Category column.

Ribbon: The 'Transform' tab is selected, indicated by a red box. In the 'Fill' dropdown menu, the 'Down' option is selected, also indicated by a red box.

Table View: The table shows the result of the 'Fill Down' operation. All null values in the 'Category' column have been replaced by 'Mix', as shown in the preview.

Power BI Desktop – Using Split feature

In Product query, notice that the Product column. Looks like two fields 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.

7. From the left panel, select **Product** Query.
 8. Select **Product** column.
 9. From the ribbon select **Home -> Split Column -> By Delimiter**. Split Column by Delimiter dialog opens.
 10. In the dialog, make sure **Custom** is selected in the **Select or enter delimiter** dropdown.
- Note:** Select or enter delimiter dropdown has some of the standard delimiters like comma, colon, etc.
11. Notice in the text area, there is a hyphen (-). Power BI assumes we want to split by hyphen. **Remove hyphen symbol and enter pipe symbol (|)** as shown in the screenshot.
 12. Select **OK**.

Note: If the delimiter occurs multiple times, **Split at** section provides option to split only once (either left most or right most) or the column can be split on each occurrence of the delimiter.

In this scenario delimiter occurs only once, hence Product column is split into 2 columns.

The screenshot shows the Microsoft Power BI desktop application. On the left, the 'Queries [9]' pane is open, showing a hierarchy with 'Sales' selected. In the main workspace, a table named 'Sales' is displayed with columns: ProductID, Product, Category, ManufacturerID, and Price. The 'Product' column contains two rows: '1 Abbas MA-01|All Season' and '2 Abbas MA-02|All Season'. The 'Category' column shows 'Mix' for both rows. The 'ManufacturerID' and 'Price' columns show '1 USD 412.13' and '1 USD 329.78' respectively. A 'Split Column' icon is highlighted with a red box in the ribbon bar. A 'Split Column by Delimiter' dialog box is open in the foreground. It contains a dropdown 'Select or enter delimiter' set to 'Custom-' with a pipe symbol '|'. Below it, 'Split at' options are set to 'Each occurrence of the delimiter'. The 'OK' button is highlighted with a red box.

Power BI Desktop – Using Rename Column feature

Let's rename the columns.

13. Select **Product.1** column. Right click next to the column name.
14. Select **Rename** from the selection dialog.
15. **Rename** the field to **Product**.
16. Similarly rename **Product.2** to **Segment**.

The screenshot shows the Power BI Desktop interface with the 'Queries [9]' pane on the left. A query named 'Product' is selected. In the main area, a table is displayed with columns: ProductID, Product, Segment, Category, and Manufacturer. The 'Product' column is highlighted with a red box. A context menu is open over this column, also with a red box around it. The menu options include: Copy, Remove, Remove Other Columns, Duplicate Column, Add Column From Examples..., Remove Duplicates, Remove Errors, Change Type, Transform, Replace Values..., Replace Errors..., Split Column, Group By..., Fill, Unpivot Columns, Unpivot Other Columns, Unpivot Only Selected Columns, Rename... (which is highlighted with a red box), and Move.

ProductID	Product	Segment	Category	Manufacturer
1	Abbas MA-01	All Season		
2	Abbas MA-02	All Season		
3	Abbas MA-03	All Season		
4	Abbas MA-04	All Season		
5	Abbas MA-05	All Season		
6	Abbas MA-07	All Season		
7	Abbas MA-06	All Season		
8	Abbas MA-08	All Season		
9	Abbas MA-09	All Season		
10	Abbas MA-10	All Season		
11	Abbas MA-11	All Season		
12	Abbas MA-12	All Season		
13	Abbas MA-13	All Season		
14	Abbas MA-14	All Season		
15	Abbas MA-15	All Season		
16	Abbas MA-16	All Season		
17	Abbas MA-17	All Season		

Power BI Desktop – Using Column From Examples feature

In Product query, notice that the Price column. You will see price and currency concatenated into one field. To do any calculations we just need the numeric value. It will be good 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 a delimiter.

17. From the left panel, select **Product Query**.
18. From the ribbon, select **Add Column -> Column From Examples**.
19. In the **first row of Column1** enter the first Price value which is **412.13** and click enter
Notice as you enter, Power BI knows that you want to split Price column. The formula it uses is displayed as well.
20. **Double click** column header **Text After Delimiter** to rename it.
21. **Rename** the column to **MSRP**.
22. Click **OK** to apply the changes.

Queries [9]

Add Column From Examples

Enter sample values to create a new column (Ctrl+Enter to apply).
Transform: Text.AfterDelimiter([Price], ",")

	ProductID	Product	Product.2	Category	ManufacturerID	Price	MSRP
1	1 Abbas MA-01	All Season	Mix	1 USD 412.13	412.13		
2	2 Abbas MA-02	All Season	Mix	1 USD 329.78	329.78		
3	3 Abbas MA-03	All Season	Mix	1 USD 963.38	963.38		
4	4 Abbas MA-04	All Season	Mix	1 USD 828.98	828.98		
5	5 Abbas MA-05	All Season	Mix	1 USD 745.5	745.5		

OK Cancel

- Notice MSRP field is of data type text. It must be a decimal. Let's change it.
23. Select **ABC** in **MSRP** column.
 24. From the selection dialog, select **Decimal Number**.

Notice all the steps we performed on the Product query are being recorded under **APPLIED STEPS** in the right panel.

Properties

Name: Product

APPLIED STEPS

- Source
- Navigation
- Changed Type
- Filled Down
- Split Column by Delimiter
- Changed Type1
- Renamed Columns
- Inserted Text After Delimiter

Similarly, let's create a currency column.

25. From the left panel, select **Product** Query.

26. From the ribbon, select **Add Column -> Column From Examples**.

27. In the **first row of Column1** enter the first Currency value as **USD** and click enter

Notice as you enter, Power BI knows that you want to split Price column. The formula it uses is displayed as well.

28. Double click column header **Text After Delimiter** to rename it.

29. Rename the column to **Currency**.

30. Click **OK** to apply the changes.

The screenshot shows the 'Add Column From Examples' dialog in Power BI. The 'Transform' tab is selected. In the ribbon, the 'Column From Examples' button is highlighted with a red box. Below it, the 'Text' dropdown in the 'From Text' section is also highlighted with a red box. The main area shows a table with columns: ProductID, Product, Segment, Category, ManufacturerID, and Currency. The first row has 'USD' entered in the Currency column. The 'OK' button at the bottom right is highlighted with a red box.

Now that we have split Price into MSRP and Currency columns, we don't need Price column. Let's remove it.

31. From the left panel, select **Product** Query.

32. Right click next to **Price** column.

33. Select **Remove**.

The screenshot shows the 'Product' query table in Power BI. The 'Price' column is selected, and a context menu is open. The 'Remove' option is highlighted with a red box. Other options in the menu include 'Remove Other Columns', 'Duplicate Column', 'Add Column From Examples...', 'Remove Duplicates', 'Remove Errors', and 'Change Type'. The 'OK' button at the bottom right of the dialog is also highlighted with a red box.

Power BI Desktop – Using Add/Remove Rows feature

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

34. In the left panel, select **Geography** query.

35. From the ribbon, select **Home** > **Remove Rows** > **Remove Top Rows**.

36. Remove Top Rows dialog opens. Enter **3** in the text box, since we want to remove the top 2 informational data rows and the blank 3rd row.

37. Select **OK**.

Queries [9]

File Home Transform Add Column View Help

Close & Apply New Source Recent Data Data source settings Manage Parameters Refresh Preview Advanced Editor Properties Manage Query

Choose Columns Remove Columns Keep Rows Remove Rows

Remove Top Rows

Remove Bottom Rows Remove Alternate Rows Remove Duplicates Remove Blank Rows Remove Errors

Column1 ABC Column2 A^bc

1 Source: Public Database
2 Last Updated: 2/1/2016
3 null null null null null
4 Zip City State Region District Country
5 22654 Star Tannery, VA, USA VA East District #07 USA

Remove Top Rows

Specify how many rows to remove from the top.

Number of rows

3

OK Cancel

Notice the first row in **Geography** query now is the column header. So let's make it a header

38. With **Geography** query selected in the left panel, from the ribbon select **Home** > **Use First Row as Headers**

First Row as Headers

Notice column Zip is of data type number. Let's change it to text as we did earlier. If we don't we will see errors when we load the data.

39. Select **123** next to Zip Column. From the dialog, select **Text**.

40. Select **Replace Current** in the **Change Column Type** dialog.

File Home Transform Add Column View Help

Close & Apply New Source Recent Data Data source settings Manage Parameters Refresh Preview Advanced Editor Properties Manage Query

Choose Columns Remove Columns Keep Rows Remove Rows Reduce Rows Sort Transform

Use First Row as Headers

Queries [9]

123 Zip A^bc City A^bc State A^bc Region A^bc District A^bc Country

1 12 Decimal Number A, USA VA East District #07 USA
2 \$ Fixed decimal number /A, USA VA East District #07 USA
3 123 Whole Number , USA VA East District #07 USA
4 % Percentage JSA VA East District #07 USA
5 Date/Time , USA VA East District #07 USA
6 Date USA VA East District #07 USA
7 Time USA VA East District #07 USA
8 Date/Time/Timezone , USA VA East District #07 USA
9 Duration SA VA East District #07 USA
10 Text , VA East District #07 USA
True/False , VA East District #07 USA

41. From the left panel, select **Manufacturer** query. Notice the bottom 3 rows are not part of the data. Let's remove it

42. From the ribbon, select **Home** -> **Remove Rows** -> **Remove Bottom Rows**

43. Remove Bottom Rows dialog opens. Enter **3** in **Number of rows** text box.

44. Select **OK**.

The screenshot shows the Power BI Desktop interface. The ribbon is set to 'Home'. In the 'Queries' pane, the 'Manufacturer' query is selected. The main area shows a table with two columns: 'Column1' and 'Column2'. The data includes rows for ManufacturerID, Manufacturer, Logo, and three blank rows (4, 5, 6). A 'Remove Bottom Rows' dialog is open, prompting to specify how many rows to remove from the bottom. The 'Number of rows' input field contains '3'. The 'OK' button is highlighted with a red box.

Power BI Desktop – Using Transpose feature

45. From the left panel, select **Manufacturer** Query. Notice ManufacturerID, Manufacturer and Logo data is laid across in rows. And the header is not useful. We need to transpose the table to meet our needs.

46. From the ribbon select **Transform** -> **Transpose**.

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

The screenshot shows the Power BI Desktop interface with the ribbon set to 'Transform'. The 'Transpose' button in the ribbon is highlighted with a red box. The 'Queries' pane shows the 'Manufacturer' query selected. The main area displays a table with two columns, 'Column1' and 'Column2'. The data rows (1, 2, 3) now contain the values from the previous table's rows, effectively transposing the data. The 'OK' button is highlighted with a red box.

47. From the ribbon select Home -> Use First Row As Headers.

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

Notice on the right panel 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 clicking on the step. Steps can also be deleted by clicking on the X that appears to the left of the step.

The properties of each step can be reviewed by clicking on the gear to the right of the step.

The screenshot shows the Power BI Desktop interface. In the center, the 'Queries [9]' pane displays the 'Manufacturer' query. The first row of the table is highlighted with a yellow background, indicating it is the header. The ribbon at the top has 'Home' selected. On the far right, the 'APPLIED STEPS' pane lists the transformation 'Promoted Headers'.

Power BI Desktop – Using Append and Conditional Column feature

To analyze the Sales of all countries, it is convenient to have a single Sales table. Hence you want to append all the rows from **International Sales** to **Sales**.

48. Select Sales in the Queries window in the left panel as shown in the figure.

49. From the ribbon select Home -> Append Queries.

Append dialog 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.

50. Select International Sales from the drop down and click **OK**.

The screenshot shows the Power BI Desktop interface. In the center, the 'Queries [9]' pane displays the 'Sales' query. The 'Append' dialog box is open over the main area. It shows 'Two tables' selected and 'International Sales' chosen in the 'Table to append' dropdown. The 'OK' button is highlighted with a yellow box.

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

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

51. From the ribbon select **Add Column -> Conditional Column**.

52. In the **Add Conditional Column** dialog, enter name of the column as “**CountryName**”.

53. Select **Country** from the **Column Name** dropdown.

54. Select **equals** from the **Operator** dropdown.

55. Enter **null** in the **Values** text.

56. Enter **USA** in the **Output** text.

57. Select the dropdown under **Otherwise** and pick **Select a column** option.

58. Select **Country** from the column dropdown.

59. Click **OK**.

This reads, if Country equals null then the value is USA else value is that of Country.

ProductID	Date	Zip	Units	Revenue	Country
1076	1/20/2011	72638	1	254.5725	null
1076	1/21/2011	47577	1	254.5725	null
1076	1/28/2011	34653	1	254.5725	null
1076	1/31/2011	84014	1	254.5725	null
1076	2/1/2011	75070	1	254.5725	null
1076	2/1/2011	87031	1	254.5725	null
1076	2/3/2011	72019	1	254.5725	null
1076	2/3/2011	72086	1	254.5725	null

Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name
CountryName

Column Name	Operator	Value	Output
If	Country	equals	null
Then			USA

Otherwise

Country

Select a column

Parameter

OK Cancel

60. You will see the **CountryName** column in the Query editor window.

The screenshot shows the Power BI Query Editor interface. On the left, the 'Queries [9]' pane lists 'Transform File from Internati...' and 'Other Queries [5]'. Under 'Other Queries', 'Sales' is selected and highlighted with a red box. The main area displays a table with columns: ProductID, Date, Zip, Units, Revenue, Country, and CountryName. Rows 1-10 show 'Country' as null and 'CountryName' as USA. Row 11 shows 'Country' as null and 'CountryName' as USA. The 'APPLIED STEPS' pane on the right shows the steps: Source, Promoted Headers, Changed Type, Appended Query, and Added Conditional Column, with the last step highlighted by a red box.

The original **Country** column is only required as a temporary column. It is not required in the final table for analysis and can be removed.

61. Right click on the **Country** column and select **Remove** as shown in the figure.

We can now rename **CountryName** column to **Country**.

62. Right click on the **CountryName** column and rename to **Country**.

63. Using **Home -> Data Type**, change the **data type** of the **Country** column to type **Text**.

When the data is refreshed, it will process through all the "Applied Steps" that you have created.

The screenshot shows the Power BI Query Editor interface. The 'Sales' query is selected in the 'Queries [9]' pane. The main area shows the same table as before, but the 'Country' column is now highlighted with a red box. A context menu is open over the 'Country' column, with the 'Remove' option highlighted by a red box. Other options in the menu include Copy, Remove Other Columns, Duplicate Column, Add Column From Examples..., Remove Duplicates, Remove Errors, Change Type, and Transform.

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

64. At first, you will only see USA data. Click on **Load more** to validate you have data from all 7 countries.

65. Click **OK** to close this filter.

Typically, when exploring data, we load a subset of data. There are multiple ways to do this. From the ribbon, select **Home -> Keep Rows -> Keep Top Rows OR Home -> Keep Rows -> Keep Bottom Rows OR Home -> Keep Rows -> Keep Range of Rows**. You can use any of these options to filter down to a subset of data.

Our dataset has data from 2011 to 2017. For our analysis we want to start with the last 3 years of data (2015-2017). We don't know how many rows. We can filter by year to get the subset.

66. Select the **arrow** next to **Date** in **Sales** Query.

67. Select **Date Filters -> In the Previous...**

68. Filter Rows dialog opens. Enter **3** in the text box next to **is in the previous**.

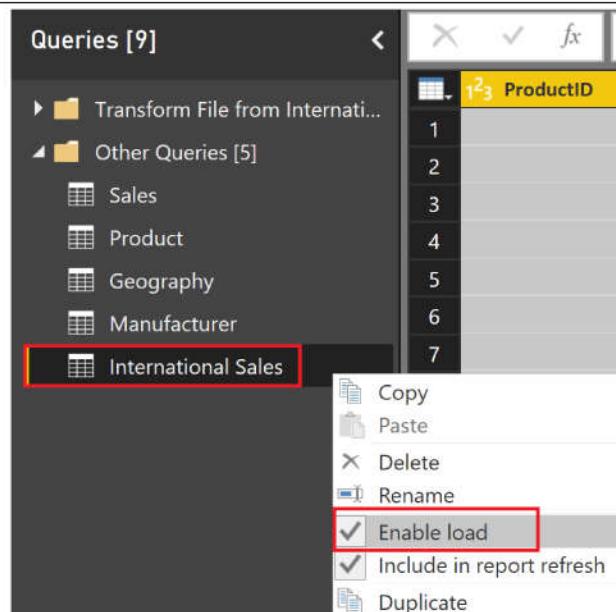
69. Select **years** from the dropdown.

70. Select **OK**.

Now that International Sales data is appended to Sales, we don't need the International Sales table to load to the data model. Let's prevent International Sales table from loading to the data model.

71. From the Queries panel on the left, select **International Sales** query.
72. Right click and select **Enable Load**. This will disable loading International Sales.

Note: The appropriate data from the International Sales table will load into 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.

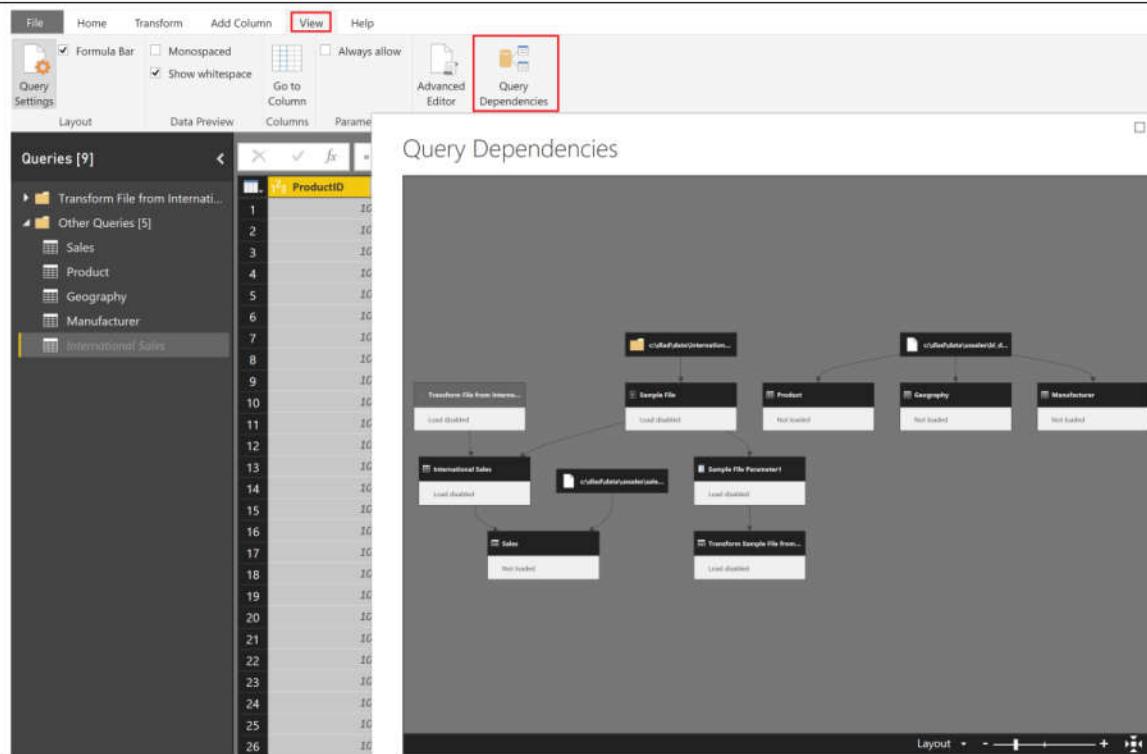


73. From the ribbon select View -> Query Dependencies.

This opens Query Dependencies dialog. The dialog shows the source of each of the query and dependencies. E.g. We see that Sales query has a csv file source and it has a dependency on International Sales query. This is a useful self-document that can be used to share knowledge with your team members.

74. Select Close in the dialog.

Query Dependencies view can be zoomed in and out as needed.



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

75. Click on File -> Close & Apply.

