



POWER BI



Contents

What is Power BI?	2
When did it start?	2
Key components of Power BI	2
How is Power BI charged?.....	2
Connecting to data.....	2
Exploring Power BI Dashboard.....	3
Installing and starting Power BI Desktop	3
Learn by Example	6
Step by step.....	6
Exploring Power BI	13
Connect to data sources	14
Connect to databases, files, folders.....	14
Connect to Oracle Database	14
Connect to SQL Server	14
Import from Excel	15
Connect to SQL Azure, Big Data, SQL Server Analysis Services (SSAS).....	15
Perform transformations	16
Basic transformations	16
Transform Example #1 – Use First Row as Headers.....	17
Transform Example #2 – Remove Columns	18
Transform Example #3 – Change type	18
Transform Example #4 – Add Column from Examples	18
Advanced data transformation options.....	19
Transform Example #4 – Conditional Columns.....	19
Transform Example #4 – Fill Down	20
Transform Example #5 – Unpivot	20
Merging Queries	22
Appending Queries	24
Cleanse data.....	25
Manage incomplete data	25
Meet data Quality requirements	25
Handling error values.....	25

What is Power BI?

Power BI is a suite of tools from Microsoft that allows you to extract data from variety of sources, transform it and then load into a data model for analysis and visualization.

When did it start?

Power BI was officially released as a product by itself in July 2015. The tools that are offered by Power BI appeared in various versions of Excel since 2010 as Power Query, Power Pivot and Power View

Key components of Power BI

Power BI Desktop - The Windows-desktop-based application for PCs and desktops.

Power BI (Service) - The SaaS (Software as a Service) based online service (formerly known as Power BI for Office 365, now referred to as PowerBI.com or simply Power BI).

Power BI Mobile Apps - The Power BI Mobile apps for Android and iOS devices, as well as for Windows phones and tablets.

Power BI Gateways - Gateways used to sync external data in and out of Power BI.

Power BI Embedded - Power BI REST API can be used to build dashboards and reports into the custom applications that serves Power BI users, as well as non-Power BI users.

Power BI Report Server - Provides a localized way for storing and managing Power BI reports.

Power BI Visuals Marketplace - A marketplace of custom visuals and R-powered visuals.

How is Power BI charged?

Author	Share and collaborate	Scale large deployments
Power BI Desktop Free	Power BI Pro \$9.99 per user per month	Power BI Premium Capacity pricing per node per month
Connect to hundreds of data sources Clean and prepare data using visual tools Analyze and build stunning reports with custom visualizations Publish to the Power BI service Embed in public websites	Build dashboards that deliver a 360-degree, real-time view of the business Keep data up-to-date automatically, including on-premises sources Collaborate on shared data Audit and govern how data is accessed and used Package content and distribute to users with apps	Gain dedicated capacity you allocate, scale, and control Distribute and embed content without purchasing per-user licenses Publish reports on-premises with Power BI Report Server Unlock more capacity and higher limits for your Pro users <small>Licensing information for Power BI Report Server and for embedding analytics with Power BI Premium.</small>

Connecting to data

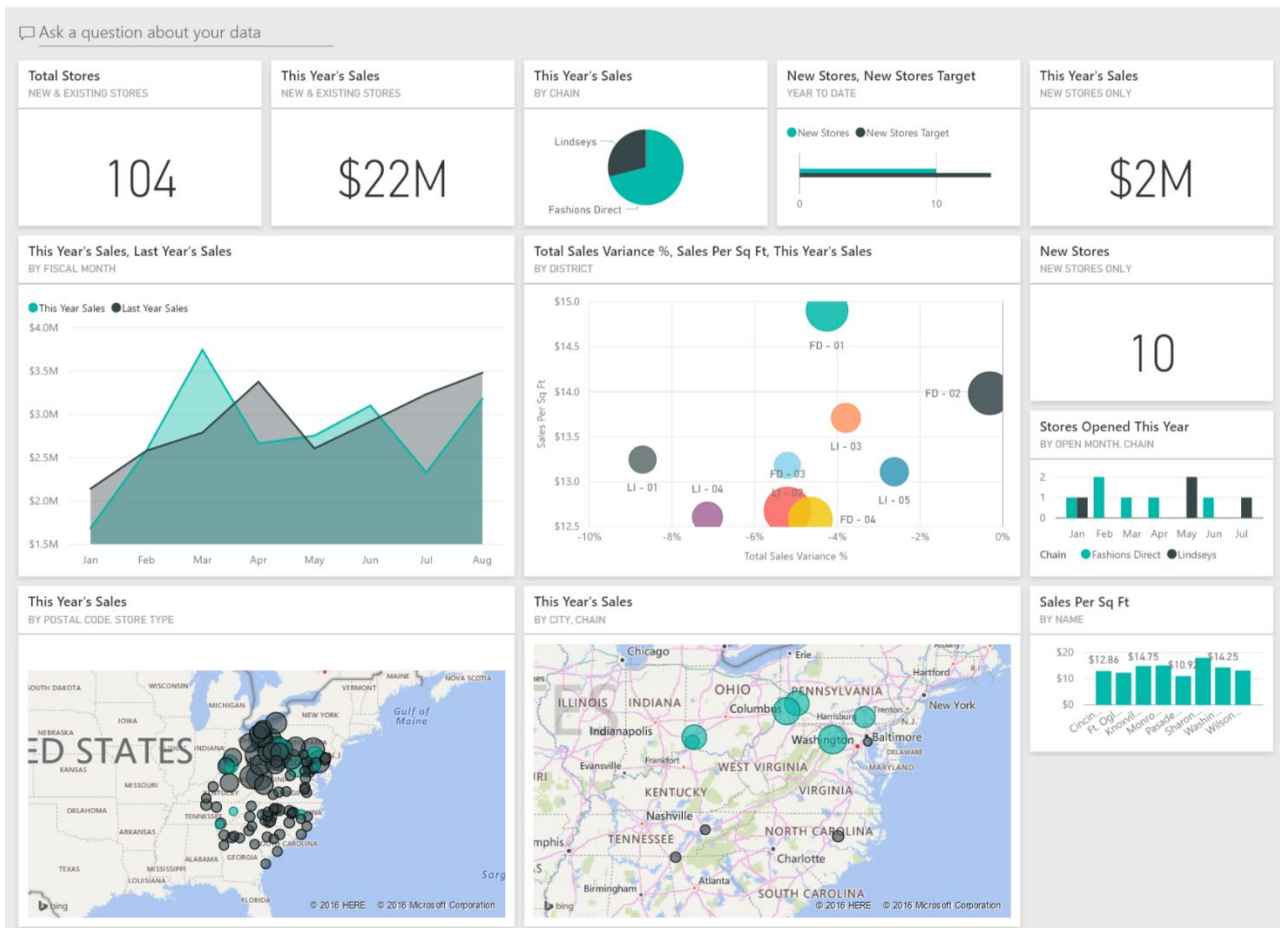
Power BI lets you connect and load data from 80+ sources that include Excel files, Text/CSV, All major databases, Web, Cloud services etc. etc.

Exploring Power BI Dashboard

A dashboard is an information management tool that visually tracks, analyzes and displays key performance indicators (KPI), metrics and key data points to monitor the health of a business, department or specific process. They are customizable to meet the specific needs of a department and company. Behind the scenes, a dashboard connects to your files, attachments, services and API's, but on the surface displays all this data in the form of tables, line charts, bar charts and gauges.

A dashboard is the most efficient way to track multiple data sources because it provides a central location for businesses to monitor and analyze performance. Real-time monitoring reduces the hours of analyzing and long line of communication that previously challenged businesses.

Retail Analysis Sample

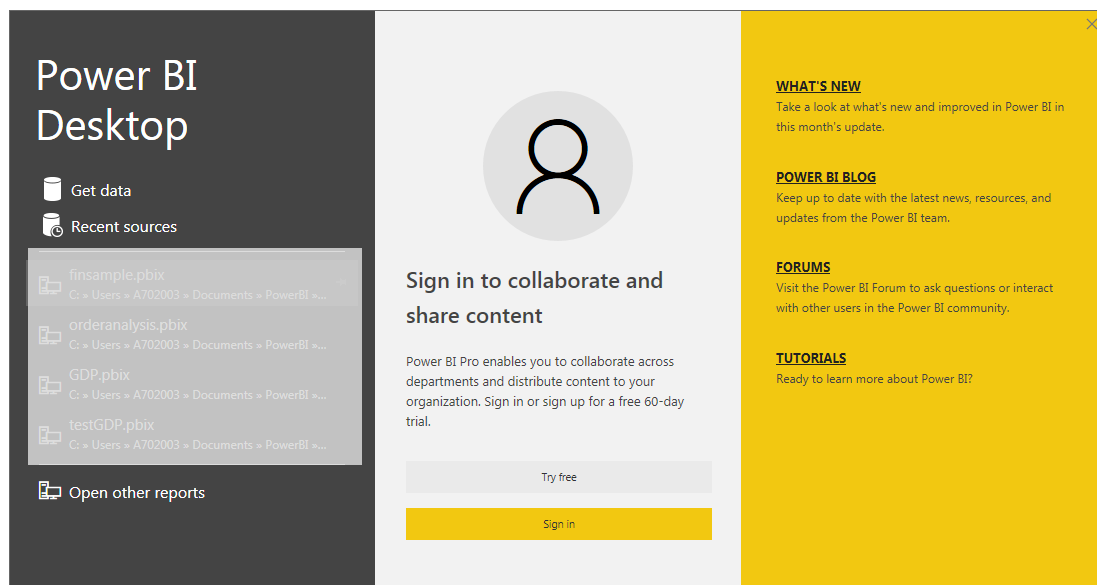


Visit the below site for more samples of Power BI dashboards...

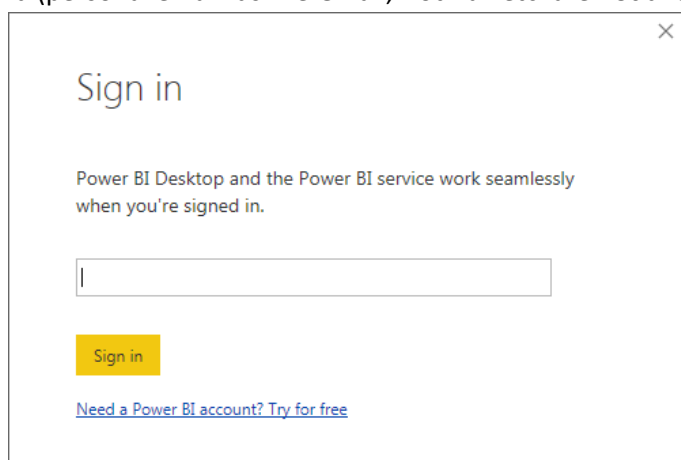
<https://community.powerbi.com/t5/Data-Stories-Gallery/bd-p/DataStoriesGallery>

Installing and starting Power BI Desktop

1. Download latest version of Power BI Desktop from Microsoft web site URL given below.
<https://powerbi.microsoft.com/en-us/desktop/>
2. Once downloaded, run the installer and follow the easy steps to complete the installation.
3. Post successful installation, a shortcut icon will be available on your desktop, clicking which will open the Power BI Desktop tool.
4. When you start Power BI Desktop very first time or when you are offline (no internet) then, you will be presented with a Sign in screen as follows:



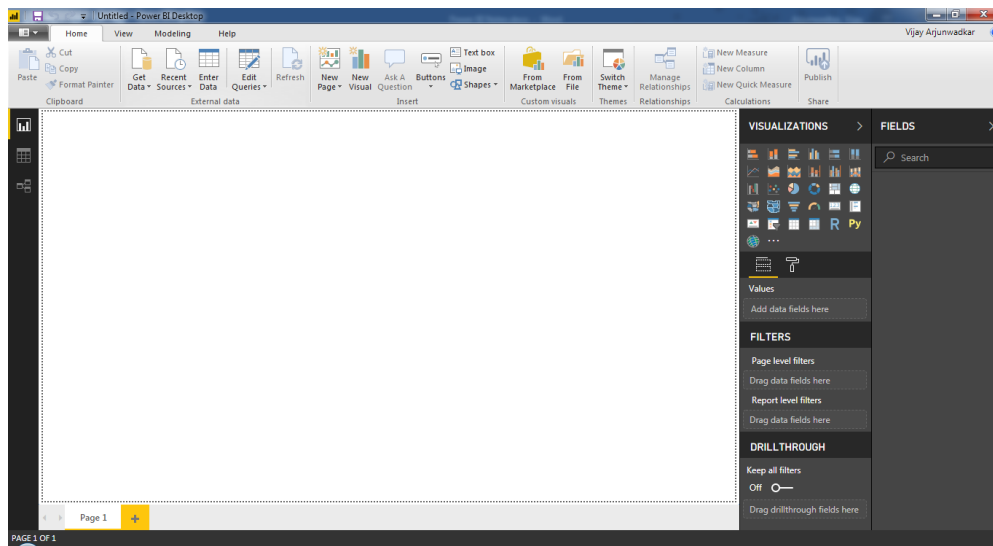
5. Click the Sign in button. You must sign in using an organizational account like your company/corporate email id (personal email ids like Gmail, Hotmail etc. are **not allowed**)



6. Once you sign in, your name will be displayed in the top right corner of the Power BI Desktop window.

Note: Signing in is optional, however it will prevent you from publishing and sharing your reports/dashboards with others in organization via Power BI Service, so it is good to always sign in.

7. Finally, Power BI main window will appear like this...



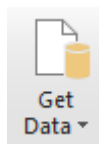
8. Now you are ready to use Power BI Desktop to connect, transform and visualize data!

Learn by Example

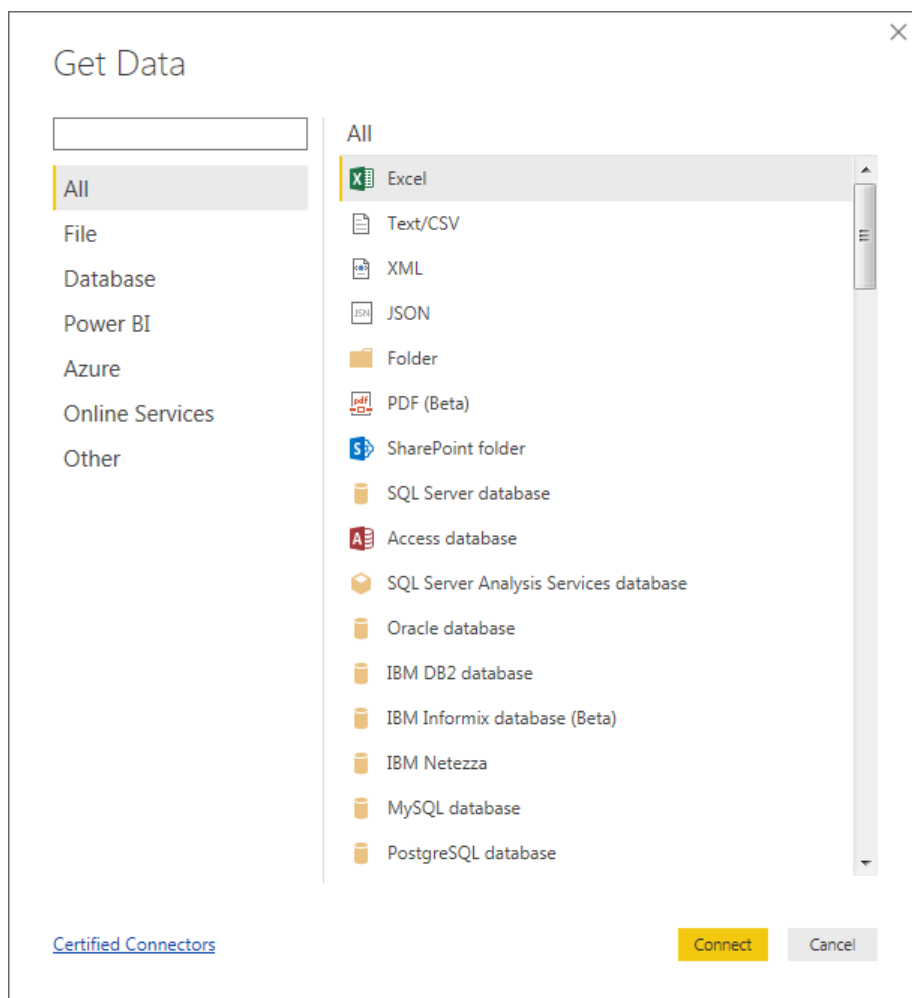
Before we get familiar with various Power BI Desktop tools and techniques, we will take a quick look at basics by an example that will cover connecting to an Excel data source, load data and then create a report with some visuals on that data.

Step by step

1. On the Power BI Desktop window, locate the Get Data button on the Home tab of ribbon.

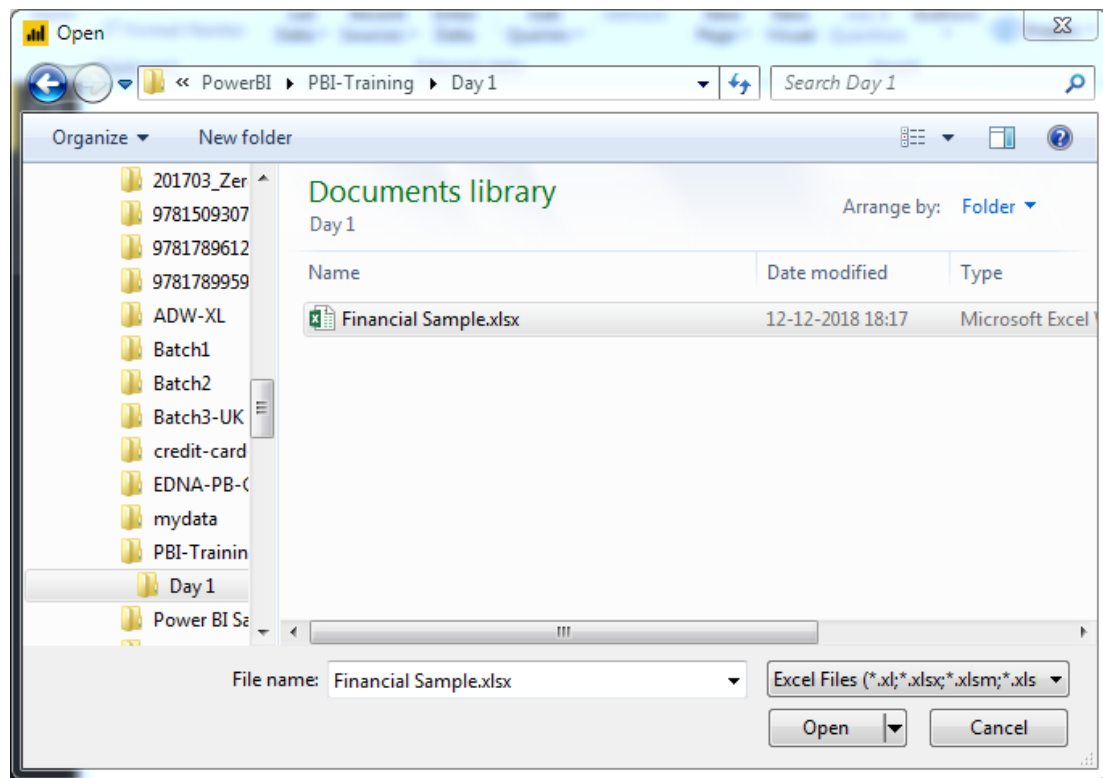


2. Click the icon on this button to open the Get Data dialog box as below...

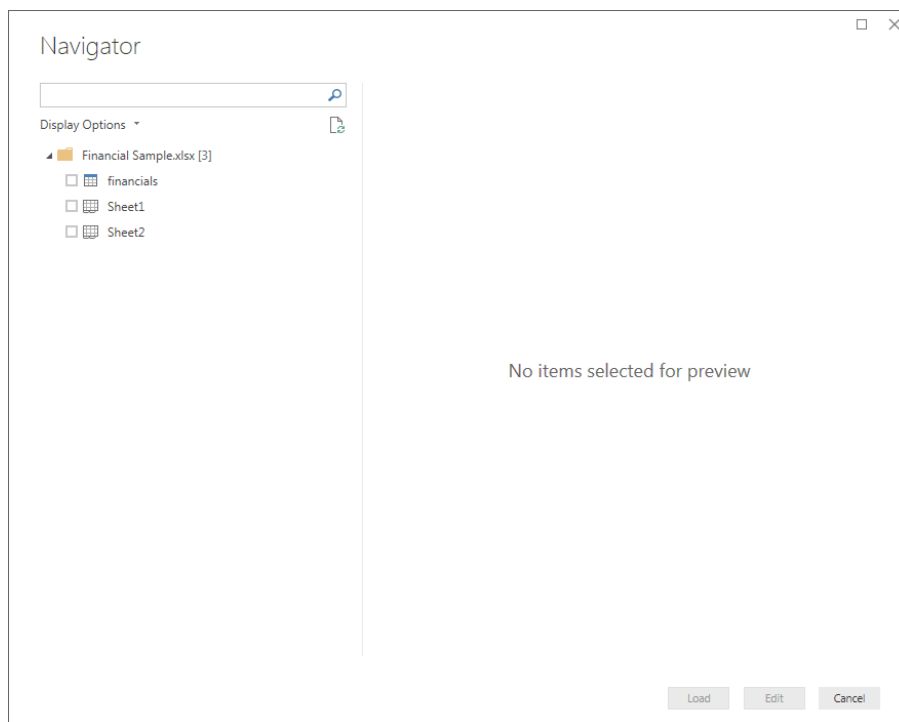


Note: We will discuss more on the various options available here later in the session, for now follow the steps ahead.

3. Ensuring that the Excel option is highlighted, click the Connect button.
4. This will open following dialog box, select the file from the path on your machine ...**\PBI-Training\Day 1**

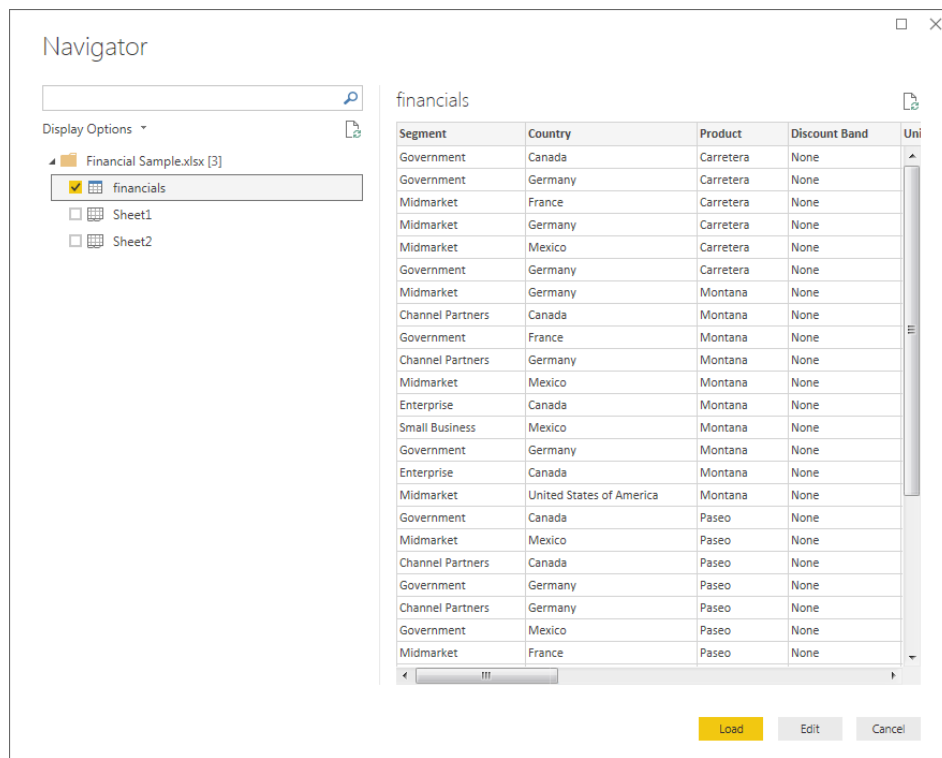


5. When you open the file, you will see following window on screen.

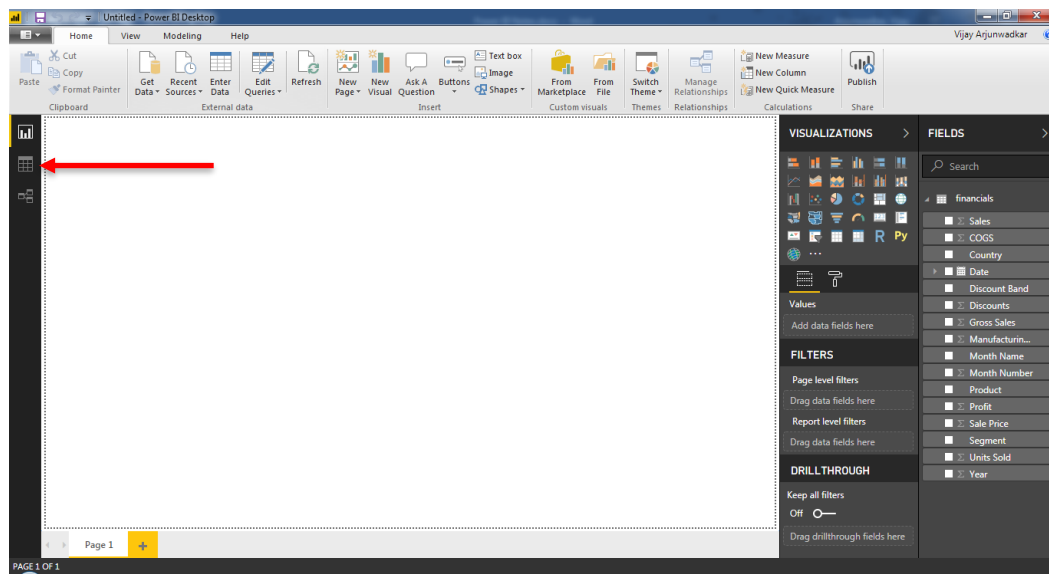


6. The Navigator window shows a list of tables and worksheets from the selected file on the left side.

7. For this example, you will select the financials table by clicking the checkbox, which results in a preview of data visible on the right side of window as shown below.



8. While you see both Load and Edit option, click the Load option for this example. We will discuss more about Edit later in the session.
9. Once you click load, a load progress window appears for a while, and then the rightmost side of your screen displays a list of fields from the loaded file, as shown below.



10. To check the loaded data, click the second button labelled Data, shown on vertical bar on the leftmost side of screen. Check the figure above where the arrow is pointing.
11. When you click the Data button, following screen appears.

Untitled - Power BI Desktop

HomeModelingHelp

Cut

Copy

Format Painter

Paste

Get Data

Recent Sources

Enter Data

Edit Queries

Refresh

New Page

New Visual

Ask A Question

Buttons

Text box

Image

Shapes

From Marketplace

From File

Switch Theme

Relationships

New Measure

New Column

New Quick Measure

Publish

ClipboardExternal dataCustom visualsThemesRelationshipsCalculationsShare

SegmentCountryProductDiscount BandUnits SoldManufacturing PriceSale PriceGross SalesDiscountsSalesCOGSProfitDate

Government	Germany	Carretera	None	1513	3	350	529550	0	529550	393380	136170	01 Decem
Government	Germany	Paseo	None	1006	10	350	352100	0	352100	261560	90540	01 Ja
Government	Canada	Paseo	None	1725	10	350	603750	0	603750	448500	155250	01 Novem
Government	Germany	Paseo	None	1513	10	350	529550	0	529550	393380	136170	01 Decem
Government	Germany	Veio	None	1006	120	350	352100	0	352100	261560	90540	01 Ja
Government	France	VTT	None	1527	250	350	534450	0	534450	397020	137430	01 Septem
Government	France	Amanilla	None	2750	260	350	962500	0	962500	715000	247500	01 Febr
Government	Mexico	Carretera	Low	1210	3	350	423500	4235	419265	314600	104665	01 Mo
Government	Mexico	Carretera	Low	1397	3	350	488950	4889.5	484060.5	365320	120940.5	01 Octo
Government	France	Carretera	Low	2155	3	350	754250	7542.5	746707.5	560300	186407.5	01 Decem
Government	France	Paseo	Low	2155	10	350	754250	7542.5	746707.5	560300	186407.5	01 Decem
Government	Canada	VTT	Low	943.5	250	350	330225	3302.25	326922.75	245310	81612.75	01 A
Government	Mexico	VTT	Low	1397	250	350	488950	4889.5	484060.5	365320	120940.5	01 Octo
Government	Canada	Carretera	Low	2852	3	350	988200	19964	978236	741520	236716	01 Decem
Government	Canada	Paseo	Low	2852	10	350	988200	19964	978236	741520	236716	01 Decem
Government	Germany	Veio	Low	2966	120	350	1038100	20762	1017338	771160	246178	01 Octo
Government	Germany	Veio	Low	2877	120	350	1006950	20139	986811	748020	238791	01 Octo
Government	Germany	VTT	Low	2877	250	350	1006950	20139	986811	748020	238791	01 Octo
Government	United States of America	VTT	Low	266	250	350	93100	1862	91238	69160	22078	01 Decem
Government	VTT	Low	1940	250	350	679000	13580	665420	504400	161020	01 Decem	
Government	Germany	Amanilla	Low	2966	260	350	1038100	20762	1017338	771160	246178	01 Octo
Government	Germany	Montana	Low	1797	5	350	628950	18868.5	610081.5	467220	142861.5	01 Septem
Government	Mexico	VTT	Low	1642	250	350	574700	17241	557459	426920	130539	01 Aug
Government	United States of America	Carretera	Low	274	3	350	95900	3836	92064	71240	20824	01 Decem

TABLE financials (700 rows)

FIELDS

Search

financials

Sales

COGS

Country

Date

Discount Band

Discounts

Gross Sales

Manufacturing Price

Month Name

Month Number

Product

Profit

Sale Price

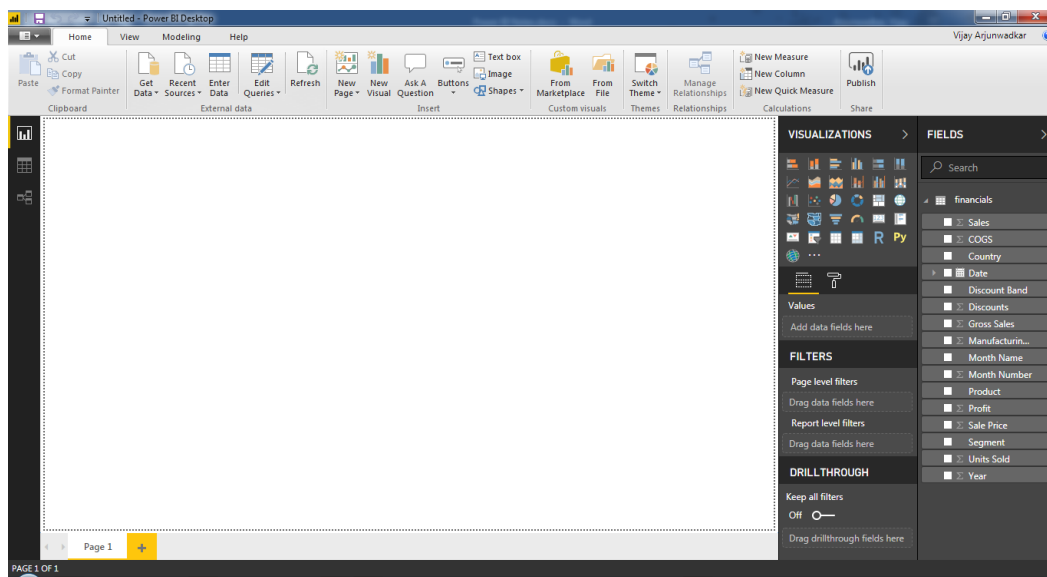
Segment

Units Sold

Year

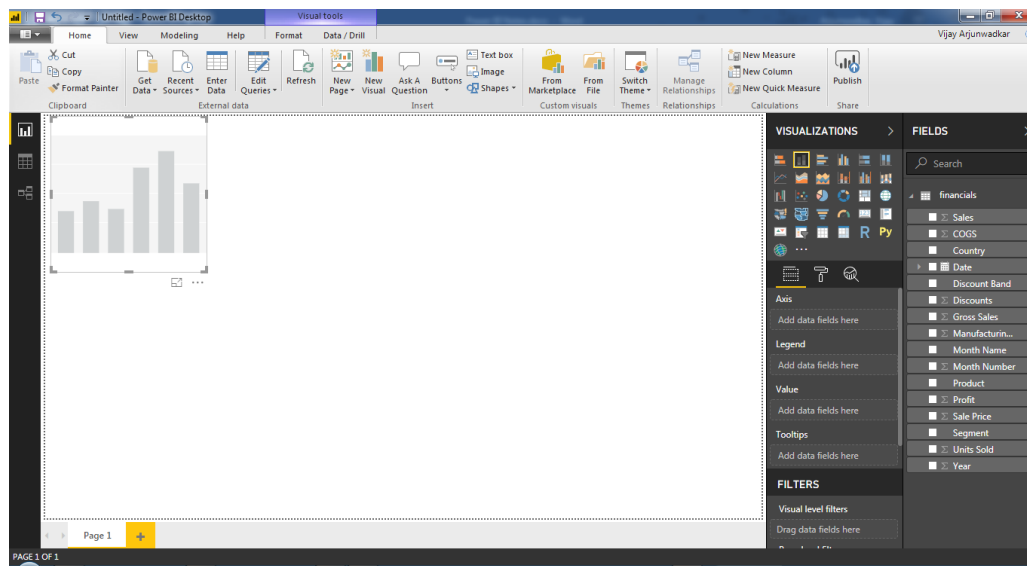
12. The Data view as this is called, shows the loaded data in an Excel style worksheet, and in fact you can perform some basic operations like sorting, filtering data much like Excel by clicking the dropdown buttons in the column headers.

13. For this example, we will get back to Report screen after viewing the data, so click the Report button just above the Data button which brings you back to a white blank screen.



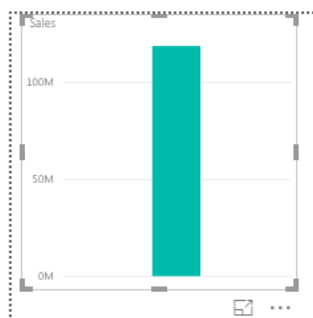
14. The Report screen is where, you create your data visualization by using the Visualizations panel on the right side just before the Fields panel. It contains all major visualization tools, which are different types of charts and other elements used to present data graphically.

15. For this example, you will begin by clicking the Stacked column chart button (second one in first row) from Visualization panel, which places a blank placeholder on the white Report portion on left as shown below.

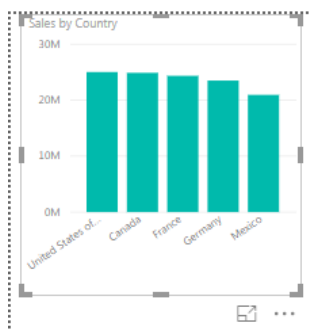


16. We will present country wise sales on this chart. Follow the sub steps below:

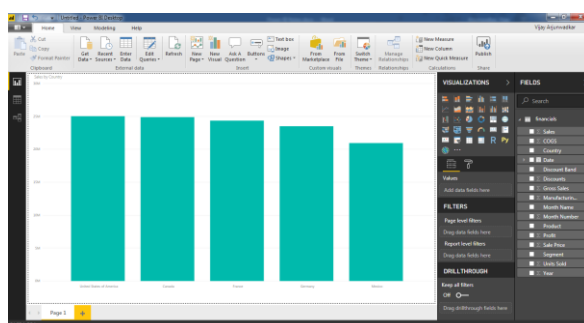
- Click the checkbox against the Sales field in the Fields list on right.
- This will display a single bar showing the entire (total) Sales.



- Next, click the checkbox against the Country field, which results in following.

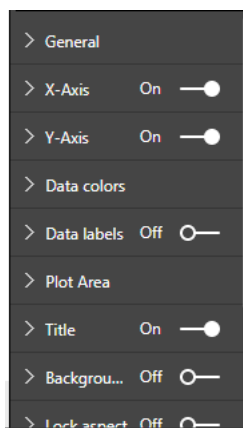


- As you see, a chart showing country wise sales is displayed in basic format.
- Now, it is time to format the chart to make it look better.
- Begin by increasing the size of the chart by dragging the bottom right handle using mouse to fill the page as shown below.

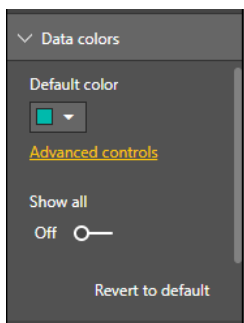




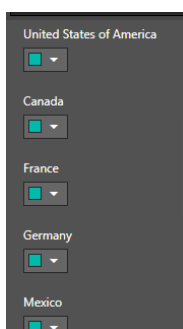
- g. Next, look at the area below the Visualization tools, where you see a format button.
- h. When you click it, you see below options in same place.



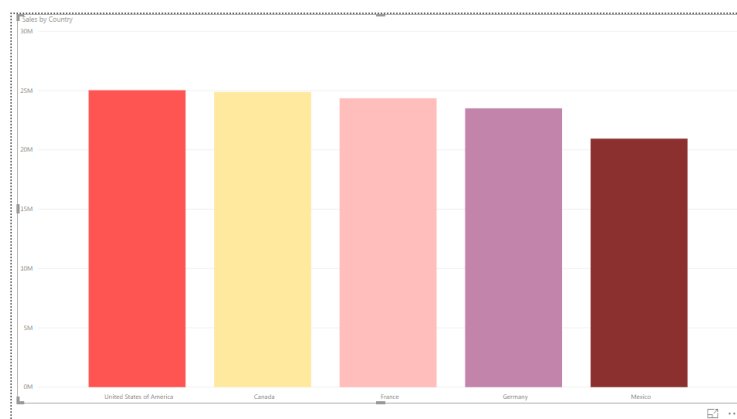
- i. Click the Data colors option to reveal the settings as below.



- j. First, turn On the Show all option, which opens further options.



- k. Now, you can set individual country bar color to distinguish it, choose a color for each by clicking the small arrow head to display the color panel.
- l. The final display should look like below. (your choice of colors can differ)

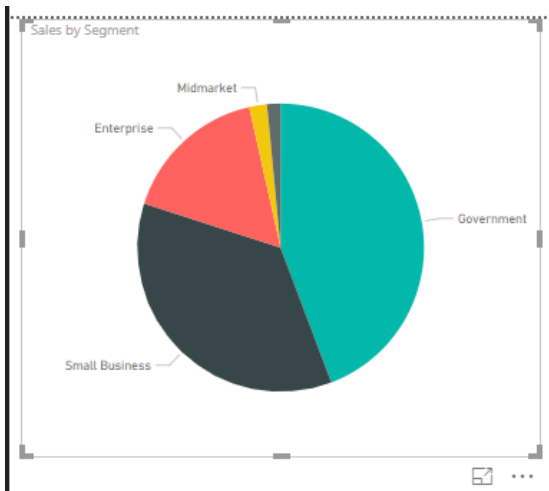


Let's create another visual on a new page in the same report. Click on the + sign next to Page 1 on the Report screen and you will get a blank page labelled Page 2.

On this page, we will create a Pie chart and a Map visual.

Click on the Pie chart icon/tool from Visualizations pane, that will place a blank pie chart on report page.

Next click the checkbox against the Sales field and then the checkbox against the Segment field. The resulting pie chart should look like below.



Next, place a Map by clicking the respective icon (appears in same row as Pie in last position), which places a blank world map. Select the checkbox against Country field, followed by Sales field, and the resulting visual should look like below one.



Note that the circles (called bubbles) appear on respective country showing the Sales figure when you point at it using mouse.

We will learn more about customising these visuals as we progress through next sections.

By now, you should have understand, how to perform common basic tasks in Power BI desktop like connecting to a data source and then design visuals in report.

Exploring Power BI

In this section, we will further explore the various components of Power BI, learning along with examples.

As you learnt in previous section, Power BI makes it easy to connect to a data source and then load and work with that data through various GUI components.

The overall workflow of Power BI is as follows.



The first two processes become, what is called as an **ETL (Extract, Transform, Load)** operation as known in business intelligence (BI) world.

The last process in this workflow, allows you to create reports and dashboards (we will differentiate them next) for which, Power BI provides you with several built-in visualization tools, as well as custom ones (those created by third party vendors/persons or even by you).

Reports v/s Dashboards

Power BI Reports contain multiple visualizations that can spread over several pages and can be created and consumed via Power BI Desktop tool as well as Power BI Service. The Desktop tool doesn't support creation of dashboards, as they can be created and consumed only via the Power BI Service available online.

A Dashboard collects several visualizations that can come from different reports in a single page and allows people (mostly top-level management) know their business status in a nutshell, thus helping them get an insight into business progress and take necessary decisions more effectively and quickly.

We will look at dashboards in detail, in a later section of this training.

Power BI comes as a bundle of various tools that combinedly allow you to explore data of all sorts from any sources, transform it by means of shaping and combining operations and then create great visuals out of it to be shared with users (consumers) of this data.

Beginning next section, we will keep learning about each of these aspects in detail, first by looking at the connections, followed by transformations and finally the visuals.

Towards the end, we will also learn about publishing our work on Power BI Service, so the end users can consume our end results for their decision making.

Connect to data sources

As Power BI helps you derive business intelligence from data, it allows you to connect to and use data from close to 100 different sources that span everything from text, Excel and other files right through databases, web and cloud services. The list is comprehensive, and every new release of Power BI keeps adding new data sources.

We will explore some of the commonly used major data sources below.

Connect to databases, files, folders

Every major business maintains a database using one of the main products, namely Oracle, SQL Server and MySQL.

In addition, there are others such as MS Access, IBM DB2, Teradata etc.

Power BI allows you to easily connect and extract data from all such databases through a wizard like interface.

Connect to Oracle Database



The screenshot shows a dialog box titled "Oracle database" with a close button (X) in the top right corner. Inside the dialog, there is a "Server" label followed by a text input field. Below this, there is a "Data Connectivity mode" section with a help icon (i). Under this section, there are two radio buttons: "Import" (which is selected) and "DirectQuery". Below the radio buttons, there is a link "Advanced options" preceded by a right-pointing triangle. At the bottom right of the dialog, there are two buttons: "OK" (highlighted in yellow) and "Cancel" (disabled).

Connect to SQL Server



The screenshot shows a dialog box titled "SQL Server database" with a close button (X) in the top right corner. Inside the dialog, there is a "Server" label followed by a text input field. Below this, there is a "Database (optional)" label followed by a text input field. Below these fields, there is a "Data Connectivity mode" section with a help icon (i). Under this section, there are two radio buttons: "Import" (which is selected) and "DirectQuery". Below the radio buttons, there is a link "Advanced options" preceded by a right-pointing triangle. At the bottom right of the dialog, there are two buttons: "OK" (highlighted in yellow) and "Cancel" (disabled).

Import from Excel

Excel is a very common source for a lot of business data and so Power BI has many features that let you work with Excel intuitively. You can connect to any Excel file and import data stored in tables (preferred way) or worksheets.

You can also create data models in Excel using its add-ins, Power Query (now named as Get Data) and Power Pivot, which can then be directly imported in Power BI for further processing and enhancements

Finally, you can also import Excel reports and visualizations into Power BI directly.

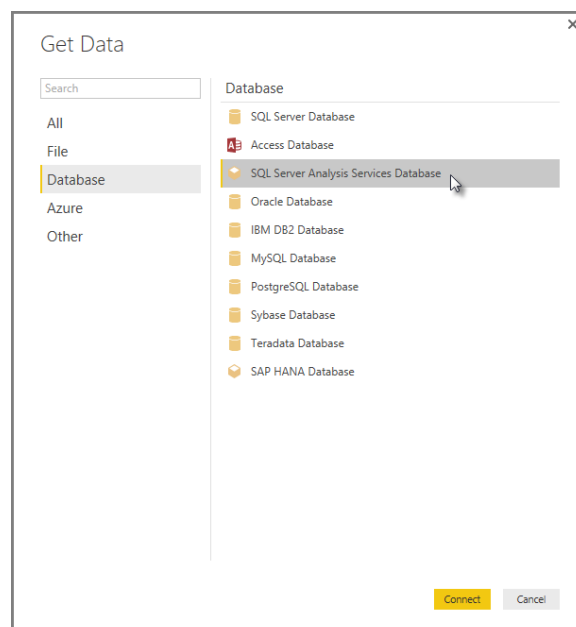
Connect to SQL Azure, Big Data, SQL Server Analysis Services (SSAS)

Business analysts can seamlessly operate on data stored in Azure Data Lake Storage taking advantage of its scale, performance, security and analytics with the self-service capabilities in Power BI, while data engineers, data scientists and other users can extend access to insights with advanced analytics and AI from complementary Azure Data Services like Azure Data Factory, Azure Databricks, and Azure Machine Learning.

Business analysts can ingest, transform, integrate and enrich big data directly in the Power BI web service – including data from a large and growing set of supported on-premises and cloud-based data sources, such as Dynamics 365, Salesforce, Azure SQL Data Warehouse, Excel and SharePoint. The ingested data can now be shared across multiple Power BI models, reports and dashboards enabling easy data reuse.

With Power BI Desktop, you can access SSAS Multidimensional models, commonly referred to as SSAS MD.

To connect to an SSAS MD database, select Get Data > Database > SQL Server Analysis Services Database as shown in the following image:



Perform transformations

Data as it gets imported from any source is not always usable as is, when we consider creating a report out of it. So, most of the time we need to transform the imported data by performing various operations on it. This typically includes, changing data types, e.g. text to number, text to date etc. calculating new columns and measures, fixing data anomalies like duplication, missing / invalid values and so on.

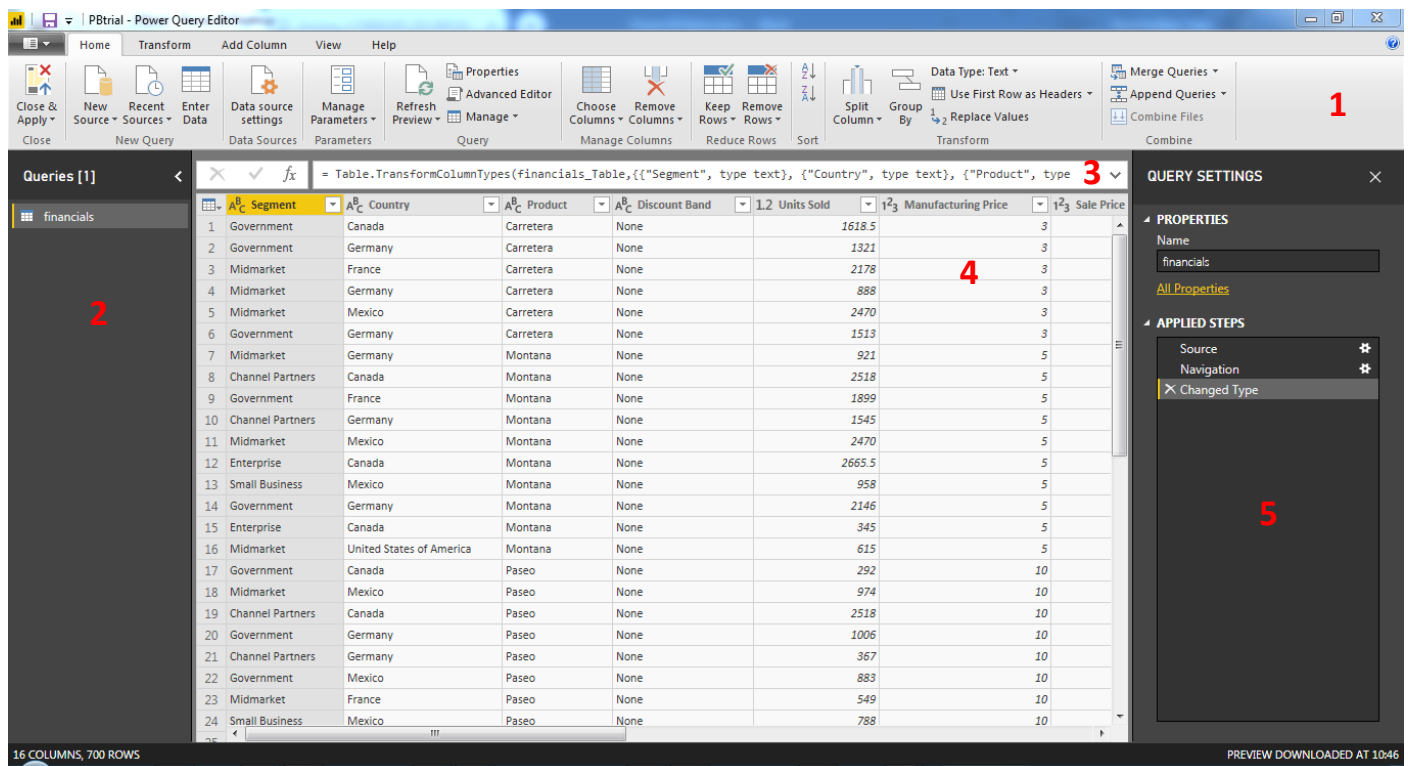
Power BI comes with lot of built-in transformation techniques, that can be used readily to transform data as described before.

Below we will see various transformation techniques with practical examples.

Basic transformations

To perform transformations, you use the Power Query editor. Let's quickly look at the Power Query interface to know what are the elements that make it up and allow us to perform the various operations.

Refer the numbers on the image and read its description below.



1. Ribbon
2. Queries pane
3. Formula bar
4. Data preview
5. Query Settings pane

The ribbon in Query Editor consists of four tabs – **Home**, **Transform**, **Add Column**, and **View**.

The **Home** tab contains the common query tasks, including the first step in any query, which is **Get Data**.

The **Transform** tab provides access to common data transformation tasks, such as adding or removing columns, changing data types, splitting columns, and other data-driven tasks.

The **Add Column** tab provides additional tasks associated with adding a column, formatting column data, and adding custom columns.

The **View** tab on the ribbon is used to toggle whether certain panes or windows are displayed. It's also used to display the Advanced Editor.

The **Queries** pane displays the number of active queries, as well as the name of the query. When you select a query from the left pane, its data is displayed in the center pane, where you can shape and transform the data to meet your needs.

The **Formula bar** displays formula in M Language (the built-in formula language for data transformation in Power BI) as you perform transformation using options from ribbon/menu. You can also type your own formula in M language if you wish so. (for advanced users)

The **Data preview** pane, data from the selected query is displayed. This is where much of the work of the Query view is accomplished.

The **Query Settings** pane is where all steps associated with a query are displayed.

Now that you are familiar with the Power BI Query Editor, it's time to fetch some data and work on different transformations.

Transform Example #1 – Use First Row as Headers

In this example, we will see how to **promote first row in a table as a header row**. Often you get data where column headings are generic, like Column1, Column2 etc. Look at the sample below.

	A	B	C	D	E	F	G
1	Column1	Column2	Column3	Column4	Column5	Column6	Column7
2	Bank Name	City	ST	CERT	Acquiring In	Closing Date	Updated Date
3	Washington	Chicago	IL	30570	Royal Saving	15-Dec-17	21-Feb-18
4	The Farmers	Argonia	KS	17719	Conway Ban	13-Oct-17	21-Feb-18
5	Fayette Cou	Saint Elmo	IL	1802	United Fide	26-May-17	26-Jul-17
6	Guaranty Ba	Milwaukee	WI	30003	First-Citizen	05-May-17	22-Mar-18
7	First NBC Ba	New Orlean	LA	58302	Whitney Bai	28-Apr-17	05-Dec-17
8	Proficio Ban	Cottonwood	UT	35495	Cache Valle	03-Mar-17	07-Mar-18
9	Seaway Ban	Chicago	IL	19328	State Bank c	27-Jan-17	18-May-17
10	Harvest Con	Pennsville	NJ	34951	First-Citizen	13-Jan-17	18-May-17
11	Allied Bank	Mulberry	AR	91	Today's Ban	23-Sep-16	25-Sep-17
12	The Woodb	Woodbury	GA	11297	United Bank	19-Aug-16	01-Jun-17
13	First Corner	King of Prus	PA	35312	First-Citizen	06-May-16	06-Sep-16
14	Trust Comp	Memphis	TN	9956	The Bank of	29-Apr-16	06-Sep-16
15	North Milw	Milwaukee	WI	20364	First-Citizen	11-Mar-16	13-Mar-17

In this file, the column headings are generic, but you can notice the next row (row no. 2) has text which are the actual headings. Follow the steps below to try this transformation.

1. Click Get Data from Home ribbon.
2. Select Excel, then navigate and select Failed Bank List.xlsx file from ...\\PBI-Training\\Day 1 folder.
3. In the Navigator window, select the table called Data, and then click Edit. When the Power Query Editor launches, you will see that the generic column headers Column1, Column2 etc. are selected by default.
4. To push the column names that are in the first row of data to header section, select the transform called Use First Row as Headers from the Home ribbon.

Transform Example #2 – Remove Columns

Often, the data sources you will connect to will include many columns that are not necessary for the solution you are designing. It is important to remove these unnecessary columns from your dataset because they will eat up space in your data model. Follow the steps below to try this transformation on the same file you opened in previous Transformation example.

1. Multi-select (Ctrl+click) the column headers of the columns you wish to keep. In this example, select the columns Bank Name, City, ST and Closing Date.
2. With these four columns selected, right-click on any of the selected columns and choose Remove Other Columns option from the displayed menu.
3. Once you finish, you will be left with only columns you wished to keep.

Another way to remove columns is by clicking the Choose Columns button on the Home ribbon, which provides a list of all the columns, from which you can choose the columns you wish to keep or exclude.

Transform Example #3 – Change type

Defining column data types properly early on in data scrubbing process can help to determine the type of values you are working with. Although, in our current example, all the data types were automatically interpreted correctly by the Power Query Editor, let's look at where you could change this if necessary:

1. Locate the data type indicator on the column header to the right of the column name.
2. Click the data type icon, and a menu will open that allows you to choose the new data type you desire.

Transform Example #4 – Add Column from Examples

Using the Add Column from Examples option, you can provide the Power Query Editor a sample of what you would like your data to look like, and it can then automatically determine which transforms are required to accomplish your goal. Continuing with the same example, let's try this feature:

1. Find and select the Add Column tab from ribbon.
2. Select the Column from Example button and, if prompted, choose From All Columns.
3. Our goal is to use this feature to combine the City and ST columns together. In the first empty cell, type **Chicago, IL** and then hit Enter.
4. Power BI will automatically apply the same change in rest of rows.

Advanced data transformation options

Often, you will find the need to go beyond basic transforms on some data. In this section, we will learn about some of common advanced transforms that you may need like, Conditional Columns, Fill down, Unpivot, Merge Queries, and Append Queries.

Transform Example #4 – Conditional Columns

Using the Conditional Columns feature, you can add new columns to your data that follow logical if/then/else statements. Let's look at a scenario, where you are provided a file of all the counties in the United States, and you must create a new column that extracts the state name from the county column and places it in its own column.

1. Connect to the file, FIPS_CountyName.txt file from ...\\PBI-Training\\Day 1 using the Text/CSV connector.
2. Launch the Power Query Editor and start by changing the data type of Column1 to Text. You will be prompted to replace an existing type conversion which you accept by clicking Replace current.
3. Now, on Column2, filter out United States from the field to remove this value from the column.
4. Remove the state abbreviation from Column2 by right-clicking on the column header and selecting Split Column | By Delimiter. It should show comma as delimiter, but if not, choose -- Custom -- for the delimiter type and type a comma followed by clicking on OK button.
5. Next, rename the column names Column1, Column2.1, and Column2.2 to County Code, County Name, and State Abbreviation respectively.
6. Now select the Conditional Column option from the ribbon.
7. Change the New column name property to State Name and implement the logic If State Abbreviation equals null Then return County Name Else return null as shown in the following screenshot. To return the value from another column, you must select the icon below the text Output, then choose Select a column. Once this is complete, click OK.

×

Add Conditional Column

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

New column name

State Name

	Column Name	Operator	Value		Output
If	State Abbreviation	equals	ABC 123 null	Then	<div><div></div>County Name</div>

Add rule

Otherwise

ABC 123

null

OK

Cancel

This results in a new column called State Name, that contains full state name appearing on rows where State Abbreviation is null.

This is just first step towards full data scrub, so we continue performing next steps to complete this exercise.

Transform Example #4 – Fill Down

Fill Down is a unique type of transform which when applied to a column, will replace all Null values below it until another non-null appears. When another non-null value is present, that value will then fill down to all Null values. Let's continue previous example to try this transform.

1. Right-click on the State Name column header and select Transform -> Capitalize Each Word. This transform is self-explanatory.
2. Next, select the State Name column and, in the Transform ribbon, select Fill -> Down. This will take the value in the State Name column and replace all non-null values until there is another State Name value that it can switch to. After performing this transform, scroll through the results to ensure that the value of Alabama switches to Alaska.
3. To complete this example, filter out any Null values that appear in the State Abbreviation column.

Transform Example #5 – Unpivot

The Unpivot transform allows you to reorganize your dataset into a more structured format for business intelligence. Consider following data for example, which represents three years of population of US states.

¹² ₃ Year	¹² ₃ Florida	¹² ₃ Pennsylvania	¹² ₃ New York	¹² ₃ California	¹² ₃ Texas
2018	21312211	12823989	19862512	39776830	28704330
2017	20984400	12805537	19849399	39536653	28304596
2016	20656589	12787085	19836286	39296476	27904862

A problem with data stored like this is you cannot very easily answer simple questions. For example, how would you answer questions like, *What was the total population for all states in the US in 2018* or *What was the average state population in 2016*? With data in this format, simple reports are made rather difficult to design. This is where Unpivot transform can help you. Using this transform you can change this data set to something acceptable for a BI project like this:

1 ² ₃ Year	A ^B _C State	1.2 Population
2018	Florida	21312211
2018	Pennsylvania	12823989
2018	New York	19862512
2018	California	39776830
2018	Texas	28704330
2017	Florida	20984400
2017	Pennsylvania	12805537
2017	New York	19849399
2017	California	39536653
2017	Texas	28304596
2016	Florida	20656589
2016	Pennsylvania	12787085
2016	New York	19836286
2016	California	39296476
2016	Texas	27904862

Data stored in this format can now easily be used to answer the questions we asked before.

There are three different methods for selecting the Unpivot transform that you should be aware of:

- **Unpivot Columns** – this turns any selected columns headers into row values and the data in those columns into a corresponding row. With this selection, any new columns that may get added to the data source will automatically be included in the Unpivot transform.
- **Unpivot Other Columns** – this turns all column headers that are not selected into row values and the data in those columns into a corresponding row. With this selection, any new columns that may get added to the data source will automatically be included in the Unpivot transform.
- **Unpivot Only Selected Columns** – turns any selected columns headers into row values and the data in those columns into a corresponding row. With this selection, any new columns that may get added to the data source will not be included in the Unpivot transform.

We will go through two examples using this transform.

1. Start a new instance of Power BI Desktop and use the Excel connector to import the workbook called Income Per Person.xlsx from ...\\PBI-Training\\Day 1 folder. In the Navigator window, select the spreadsheet called Data in it.
2. Now, make the first row of data column headers using Use First Row as Headers from the Home ribbon
3. Rename the column GDP per capita PPP, with projections to Country.
4. If you check this data, you can see that most of the column names are actually years and the values inside those columns are the income for those years. This is not the ideal way to store data as it is difficult to answer questions like: *What is the average income per person for Belgium?* To make it easier to answer this question, right-click on the Country column and select Unpivot Other Columns.
5. Rename the columns Attribute and Value to Year and Income respectively.
6. To finish this first example, rename the query as Income.

Queries [1]		QUERY SETTINGS		
		PROPERTIES		
		APPLIED STEPS		
		Source		
		Navigation		
		Changed Type		
		Promoted Headers		
		Changed Type1		
		Renamed Columns		
		Unpivoted Other Columns		
		Renamed Columns1		

This first method walked you through what can often be the fastest method for performing an Unpivot transform. In the next example you will learn how to use the Unpivot Columns method.

1. Remain in the already open Power Query Editor and select New Source from the Home Ribbon to use the Excel connector and import the workbook called Total Population.xlsx from the respective folder. Choose the Data spreadsheet from the Navigator window.
2. Like in last example, make the first row as header using the option from Home ribbon.
3. Rename the column Total population to Country.
4. This time, multi-select all the columns except Country, then right-click on one of the selected columns and choose Unpivot Columns. Do this by selecting first column and then Shift-click the last column wanted.
5. Rename the columns Attribute and Value to Year and Population respectively.
6. Finish this example rename the query as Population.

Merging Queries

A common requirement when building BI solutions is the need to join two tables together to form a new result that includes some columns from both tables in a single query. Power BI makes this task very simple with the Merge Queries feature. Using this feature requires that you select two tables and then determine which column or columns will be the basis of how the two queries are merged. After determining the appropriate columns for your join, you will select a join type.

The join types are listed here with the description that is provided within the product.

Left Outer (all from first, matching from second)

Right Outer (all from second, matching from first)

Full Outer (all rows from both)

Inner (only matching rows)

Left Anti (rows only in first)

Right Anti (rows only in second)

To examine the Merge Queries option, you will pick up from where you left off with the Unpivot examples in the previous section.

1. With the Population query selected, find and select Merge Queries | Merge Queries as New under the Home Ribbon.
2. In the Merge dialog box, select the Income query from the dropdown selection in the middle of the screen.
3. Then, multi-select the Country and Year columns under the Population query and do the same under the Income query. This defines which columns will be used to join the two queries together. Ensure that the number indicators next to the column headers match. If they don't, you could accidentally attempt to join on the incorrect columns.
4. Next, select Inner (only matching rows) for the Join Kind. This join type will return rows only when the columns you chose to join on exist in both queries. Before you click OK, confirm that your screen looks like this:

X

Merge

Select tables and matching columns to create a merged table.

Population 📄

Country	1	Year	2	Population
Afghanistan	1800			3280000
Afghanistan	1820			3280000
Afghanistan	1870			4207000
Afghanistan	1913			5730000
Afghanistan	1950			8151455

Income 📄

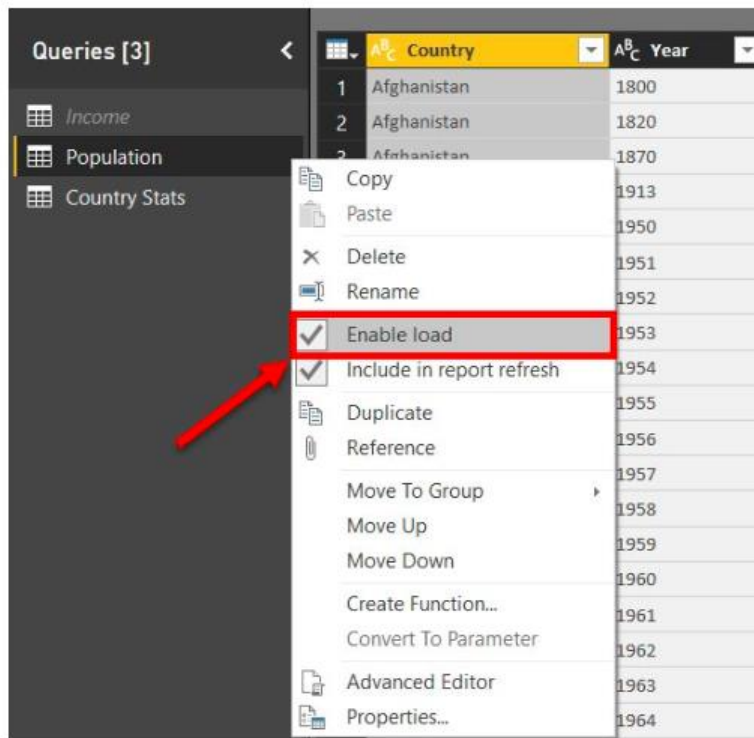
Country	1	Year	2	Income
Afghanistan	1800			472.0534996
Afghanistan	1820			472.0534996
Afghanistan	1913			638.3786419
Afghanistan	1950			757.3187954
Afghanistan	1951			766.7521974

Join Kind
 Inner (only matching rows) ▼

i The selection has matched 17221 out of the first 22127 rows.

OK
Cancel

5. Once you select OK, this will create a new query called Merge1 that combines the results of the two queries. Go ahead and rename this query Country Stats.
6. You will also notice that there is a column called Income that has a value of Table for each row. This column is representative for the entire Income query that you joined to. To choose which columns you want from this query, click the Expand button on the column header. After clicking the Expand button, uncheck Country, Year, and Use original column name as prefix then click OK.
7. Rename the column called Income .1 to Income.
8. Finally, since you chose the option Merge Queries as New in Step 1, you can disable the load option for the original queries that you started with. To do this, right-click on the Income query in the Queries pane and click Enable Load to disable it. Do the same thing for the Population query. Disabling these queries means that the only query that will be loaded into your Power BI data model is the new one, called Country Stats:



To begin using this dataset in a report, you would click Close & Apply.

In this section, you learned how the Merge Queries option is ideal for joining two queries together. In the next section, you will learn how you could solve the problem of performing a union of two or more queries.

Appending Queries

Occasionally, you will work with multiple datasets that need to be appended to each other. Here's a scenario: you work for a customer service department for a company that provides credit to customers. You are regularly provided with .xlsx files that give summaries of customer complaints regarding credit cards and student loans. You would like to be able to analyze both these data extracts at the same time but, unfortunately, the credit card and student loan complaints are provided in two separate files. In this example, you will learn how to solve this problem by performing an append operation on these two different files.

1. Launch a new instance of the Power BI Desktop, and use the Excel connector to import the workbook called Student Loan Complaints. found in the book source files. Once you select this workbook, choose the spreadsheet called Student Loan Complaints in the Navigator window, and then select Edit to launch the Power Query Editor.
2. Next, import the credit card data by selecting New Source | Text/CSV, then choose the file called Credit Card Complaints. Click OK to bring this data into the Power Query Editor.
3. With the Credit Card Complaints query selected, find and select Append Queries | Append Queries as New under the Home Ribbon.
4. Select Student Loan Complaints as the table to append to, then select OK.
5. Rename the newly created query All Complaints.
6. Like the previous example, you would want to disable the load option for the original queries that you started with. To do this, right-click on the Student Load Complaints query in the Queries pane and click Enable Load to disable it.
7. Do the same to the Credit Card Complaints query, and then select Close & Apply.

Cleanse data

Sometimes, we may have to clean the data we receive, unless you are using a data source where data quality is managed by someone. In this section, we will see how to clean data.

Manage incomplete data

Earlier you have seen how to use Fill Down feature. There are other ways in which you can add missing values to data. E.g. you may choose to replace nulls with a column average rather than values from above or below. To know this process, follow the steps below:

1. Connect to ReplaceWithAverage.csv from the respective folder and click the Edit option to open the Power Query editor
2. You can notice that, several values are missing in the table
3. To replace nulls with an average, calculate the average first. To do this right-click on the Average Price column and then select Add as New Query. This creates a new query with the column transformed into a list.
4. Note the icon of this query which is different now as it is a list and lists are not the same as tables. You can see that the Transform and Add Column tabs are inactive. However, there is a new Transform tab that is designated as List Tools, which allows you to convert this list into a table. Apart from that, you also have several other related options from which we want to select Statistics -> Average.
5. This transforms the query into a scalar value, as can be seen from its 123 icon. This value can now be used to replace nulls in original query
6. Go back to ReplaceWithAverage query and add a Custom Column, which should have following formula:

=if [Average Price] is null then #"Average Price" else [Average Price]

7. When you click OK, the nulls are replaced with the Average Price calculated before.

Meet data Quality requirements

Power Query has features that can help you with handling errors, duplicate values, and undesirable characters. Errors can be filtered out, replaced or otherwise dealt with as desired.

Handling error values

1. Connect to Target.txt file from ...**PBI-Training\Day 1** folder.
2. Rename the query to ErrorHandling.
3. Promote Headers
4. To generate an error, we can change the type of the CalendarYear column to Whole Number. Click Replace Current button on the dialog box when it appears.
5. Note that row 13 has an error now as Power Query tried to convert the text *To be confirmed into a number. There are at least three ways to deal with the error.
 - a. First you can right-click on the CalendarYear header and select Remove Errors. Alternatively, you can select the column and then click Transform -> Reduce Rows -> Remove Rows -> Remove Errors. This will remove the row completely.

- b. Second, after right-clicking on the header, you can select Replace Errors and in the opened dialog box, you will be able to specify a replacement value say null for example, to replace errors with nulls.
- c. Third, you can create a custom column that checks whether there is an error in a column. For this you need to use the “try otherwise” construct. Select Add Column -> General -> Custom Column and type in the following formula.

= try [CalendarYear] otherwise null

This formula checks whether the value in the CalendarYear column is an error, and if it is then returns a null value. If the CalendarYear is not an error, then this value is returned.