

Context

Introduction	2
Connecting Data	3 – 4
Power Query	5 – 7
Background	7 – 8
Heading	8 – 9
Creating Cards	9 – 12
Donut Chart	12 – 13
Line Chart	13 – 14
Stacked column chart	14 – 15
Using natural Language Q&A	16 – 17
Drill Up and Drill Down	17 – 18
Create a drill through page	19 – 21
Publishing	22
Download and install Python	23 – 27
Install PYODBC	27 – 28
Install Pandas	28
Install openpyxl	29
Install matplotlib	29
Install numpy	29
Get Data	30 – 32
Bar Chart	32 – 34
Pie Chart	34 – 35
Scatter Chart	35 – 36
Area Chart	37 – 38
Create Date Table	38 – 42

Introduction

Microsoft Power BI is one of the most popular Data Visualization tools and Business Intelligence Tools, and this tutorial explains everything about it.

Power BI is a collection of connectors, apps, and software services that work together for business users to convert large junk data into more meaningful insights. Typically, an organization gets its data from Text, CSV files, Excel spreadsheets, databases, data warehouses, or the cloud.

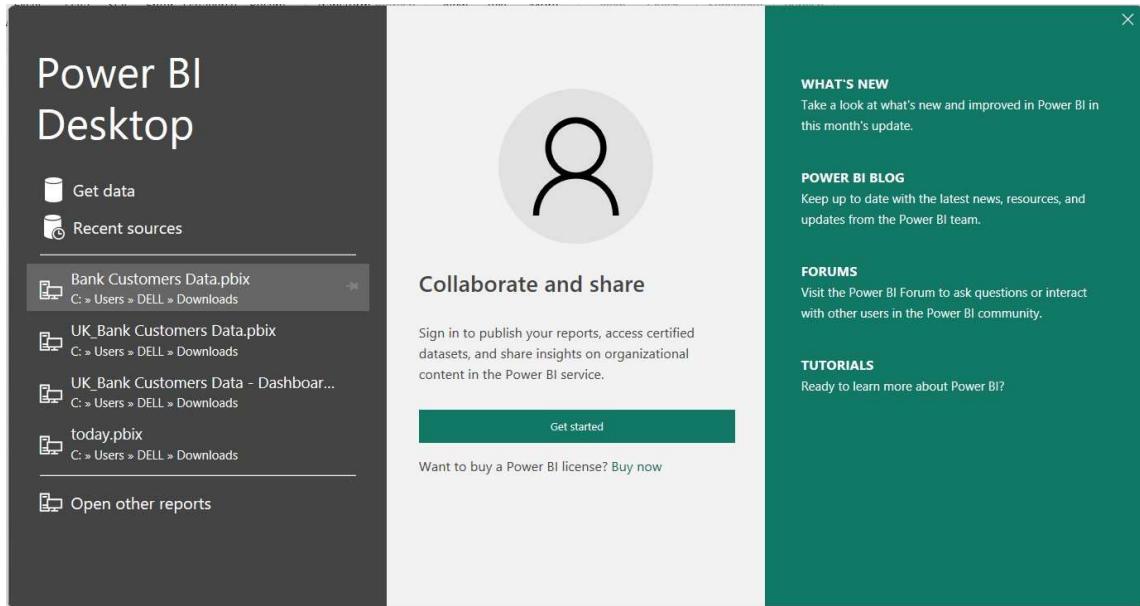
Microsoft Power BI lets you connect with all kinds of data sources to get your data. Transform data, model data, visualize (charts), and share them with anyone.

Microsoft Power BI is a cloud-based data analysis and analytics or reporting technology. It incorporates several powerful components, each having its own role in creating reports. For instance, connecting to data sources, data cleansing, analytics, calculations, sharing reports, consuming, etc.

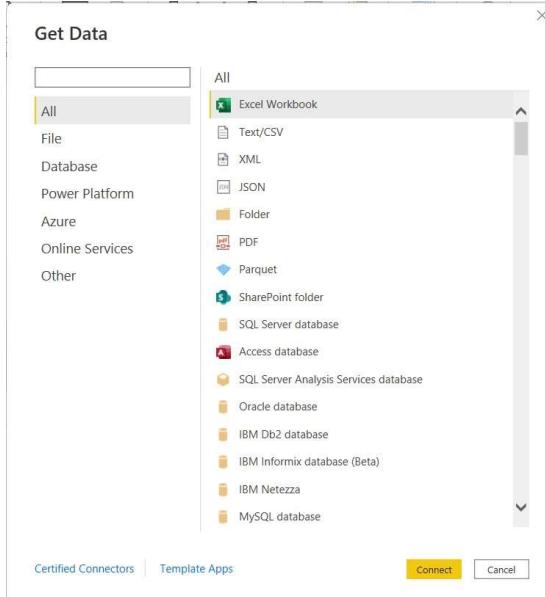
It also allows data analysts to publish reports for your organization. So that business users can access them using mobiles, tablets, etc. This Microsoft Power BI tutorial also explains these steps, so read this thoroughly to learn everything about this business intelligence tool.

Connecting Data

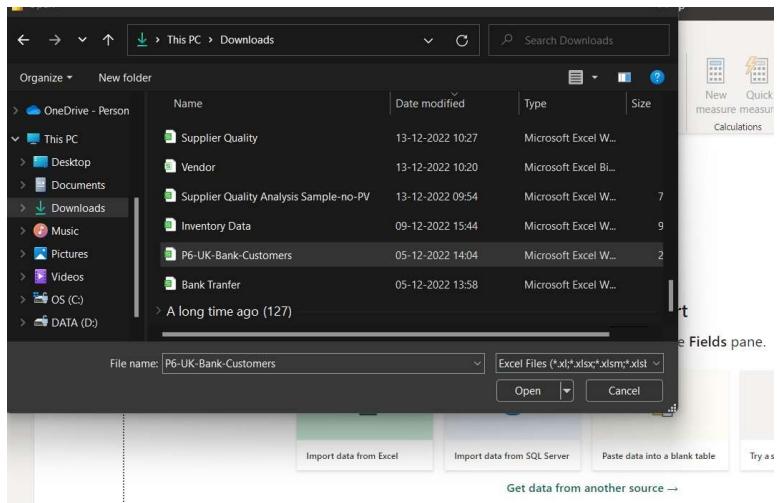
1. When you open the Power BI Desktop app you will see a window as below click Get Data option.



2. As this is an Excel file, select the **Excel Workbook** option from the drop-down list. Because our file is in Excel file format. And click connect.



3. Select the file named **P6-UK-Bank-Customers** file and click open.



4. After selecting the file and select tables in the file, data will be displayed in the below format.

Navigator

Display Options ▾

P6-UK-Bank-Customers.xlsx [1] P6-UK-Bank-Customers

Customer ID	Name	Surname	Gender	Age	Region
100000001	Simon	Walsh	Male	21	England
400000002	Jasmine	Miller	Female	34	Northern Ireland
100000003	Liam	Brown	Male	46	England
300000004	Trevor	Parr	Male	32	Wales
100000005	Deirdre	Pullman	Female	38	England
300000006	Ava	Coleman	Female	30	Wales
100000007	Dorothy	Thomson	Female	34	England
200000008	Lisa	Knox	Female	48	Scotland
300000009	Ruth	Campbell	Female	33	Wales
100000010	Dominic	Parr	Male	42	England
100000011	Dominic	Lewis	Male	40	England
100000012	Benjamin	Grant	Male	39	England
100000013	Ryan	MacDonald	Male	24	England
200000014	Thomas	Lawrence	Male	46	Scotland
300000015	Madeleine	Marshall	Female	36	Wales
100000016	Nicholas	Newman	Male	42	England
200000017	Grace	Hill	Female	31	Scotland
200000018	Samantha	Coleman	Female	42	Scotland
100000019	William	Ince	Male	40	England
100000020	Audrey	Jones	Female	46	England
300000021	Boris	Johnston	Male	37	Wales
200000022	Jason	Butler	Male	58	Scotland
300000023	Deirdre	McDonald	Female	41	Wales

Load Transform Data Cancel

5. Click on Load.

Power Query

Power Query is a powerful tool used to connect to many different data sources and transform the data into the shape you want. It will help you to find missing values, any data errors, any data type mismatch, any outliers, etc.

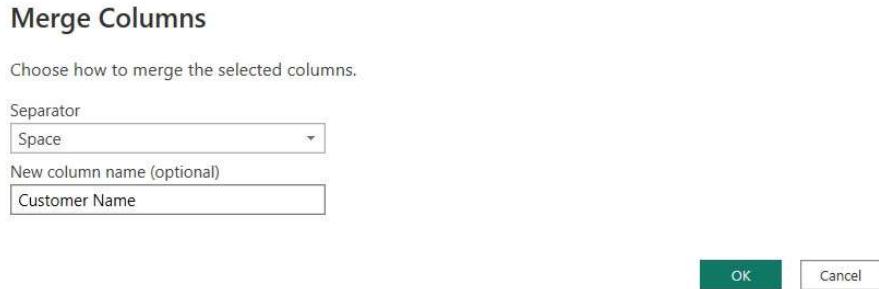
- After clicking the Transform Data. The data will appear in the Power Query Editor.

The screenshot shows the Power Query Editor interface with a table titled "P6-UK-Bank-Customers". The table contains columns: Customer ID, Name, Surname, Gender, Age, Region, and Job Classification. The "Name" and "Surname" columns are merged into a single "Name" column. The "Gender" column is converted to a whole number type. The "Age" column is converted to a date type. The "Region" and "Job Classification" columns are removed. The "Promoted Headers" step is highlighted in the "Applied Steps" pane. The "Properties" pane shows the query name as "P6-UK-Bank-Customers".

- As you can see there is Name and Surname columns, we need to combine those columns so here we need to merge these two columns.
- Select those two columns and go to Transform tab click on merge columns as shown below.

The screenshot shows the Power Query Editor interface with the "Merge Columns" step selected in the ribbon under the "Transform" tab. The table "P6-UK-Bank-Customers" is displayed with the "Name" and "Surname" columns merged into a single "Name" column. The "Gender" column is present. The "Promoted Headers" step is highlighted in the "Applied Steps" pane.

9. A small window will appear and give appropriate details for the merging as shown below.



10. Now a new column is added in to the table and the old columns are disappears as shown below.

	Customer ID	Customer Name	Gender
1	100000001	Walsh Simon	Male
2	400000002	Miller Jasmine	Female
3	100000003	Brown Liam	Male
4	300000004	Parr Trevor	Male
5	100000005	Pullman Deirdre	Female
6	300000006	Coleman Ava	Female
7	100000007	Thomson Dorothy	Female
8	200000008	Knox Lisa	Female
9	300000009	Campbell Ruth	Female
10	100000010	Parr Dominic	Male
11	100000011	Lewis Dominic	Male
12	100000012	Grant Benjamin	Male
13	100000013	MacDonald Ryan	Male
14	200000014	Lawrence Thomas	Male
15	300000015	Marshall Madeleine	Female
16	100000016	Newman Nicholas	Male

11. Now we need to create an Age Group based on the Age column as shown below.

12. Select the Age column, go to Add columns tab and click on the column from example and select the From selection.

The screenshot shows the Power Query Editor interface. The ribbon is visible with tabs for File, Home, Transform, Add Column, View, Tools, and Help. The 'Add Column' tab is currently selected. In the main area, there is a table with four columns: Customer ID, Customer Name, Gender, and a new column named 'Age'. The formula bar at the top displays the query: = Table.CombineColumns(#"Changed Type", {"Surname", "Name"}, Combiner.CombineTextByDelimiter(" ", {1,2}), 3). The 'Add Column' tab has a dropdown menu set to 'From Selection'. The Power Query ribbon also includes sections for Conditional Column, Index Column, Duplicate Column, Extract, Parse, Statistics, Standard, Scientific, Rounding, Date, Time, Duration, and Information.

13. An empty will appear. In that type 21-30 and 31-40 based on the Age column as shown below.

Add Column From Examples
Enter sample values to create a new column (Ctrl+Enter to apply).
Transform: if [Gender] = "Male" then "21-30" else if [Gender] = "Female" then "31-40" else null

<input type="checkbox"/> t ² Customer ID	<input type="checkbox"/> AB _C Customer Name	<input type="checkbox"/>	Gender	<input checked="" type="checkbox"/> t ² Age	<input type="checkbox"/> AB _C Region	<input type="checkbox"/>	Custom
100000002	Walsh Simon		Male		21	England	21-30
400000002	Miller Jasmine		Female		34	Northern Ireland	31-40
100000003	Brown Liam		Male		46	England	21-30
300000004	Parr Trevor		Male		32	Wales	21-30
100000005	Pullman Deirdre		Female		38	England	31-40
300000006	Coleman Ava		Female		30	Wales	31-40
100000007	Thomson Dorothy		Female		34	England	31-40
200000008	Knox Lisa		Female		48	Scotland	31-40
300000009	Campbell Ruth		Female		33	Wales	31-40
100000010	Parr Dominic		Male		42	England	21-30
100000011	Lewis Dominic		Male		40	England	21-30
100000012	Grant Benjamin		Male		39	England	21-30
100000013	MacDonald Ryan		Male		24	England	21-30
200000014	Lawrence Thomas		Male		46	Scotland	21-30
300000015	Marshall Madeleine		Female		36	Wales	31-40
100000016	Newman Nicholas		Male		42	England	21-30
200000017	Hill Grace		Female		31	Scotland	31-40

14. Click Ok and Rename the Column as Age Group.

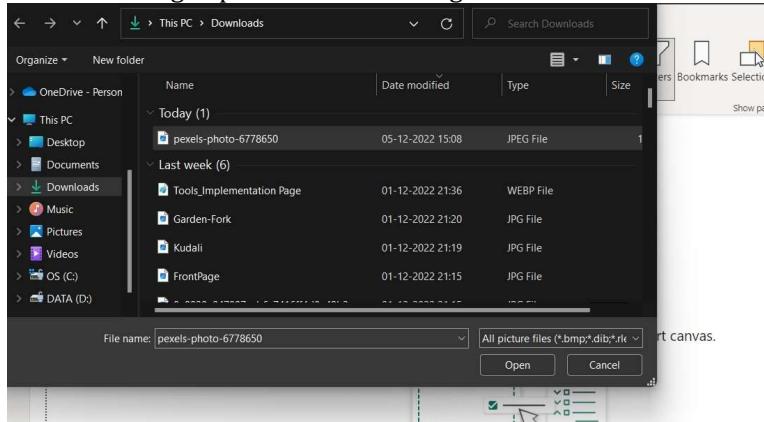
15. Click Close and Apply button.

Background

16. Now we are adding a image to our dashboard as a background.

17. Go to format page and select dropdown canvas background.

18. In the image option select the image in our device as shown below.

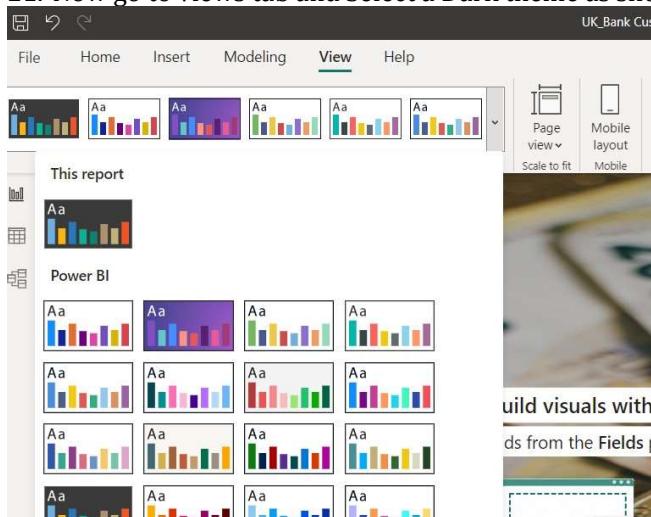


19. Select Fit in Image fit option.

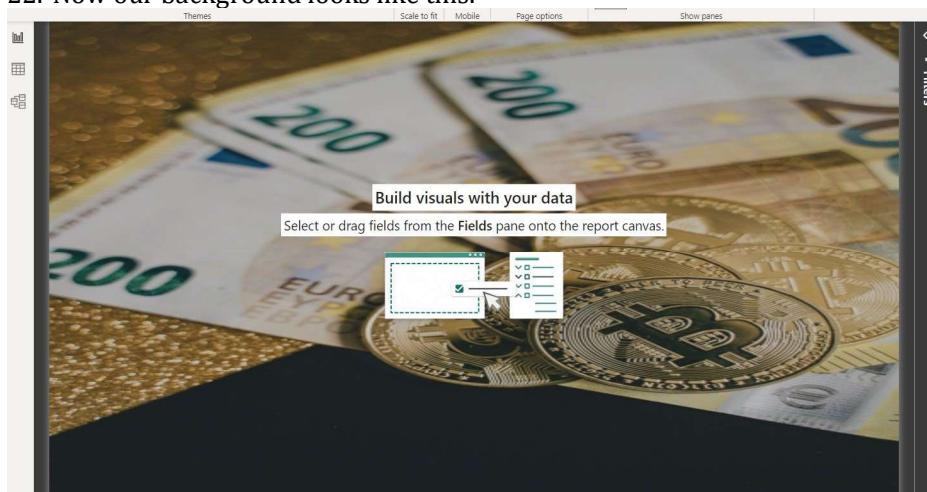
20. And Give Transparency 0% as shown below.



21. Now go to views tab and Select a Dark theme as shown below.



22. Now our background looks like this.

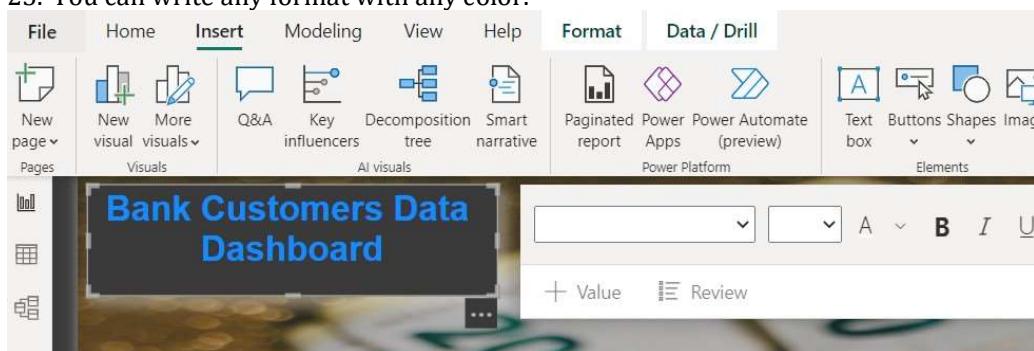


Heading

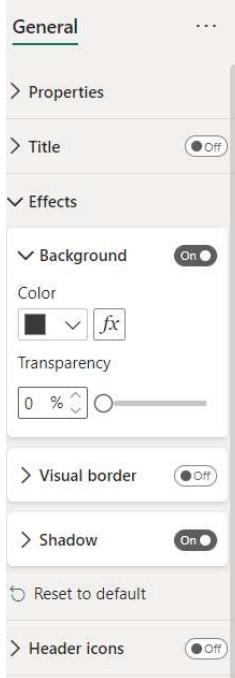
23. Now we give a heading to our dashboard.

24. Go to Insert tab select Text box option it will appear as below, and Type the Heading as shown below.

25. You can write any format with any color.



26. Follow the below visual format for the Heading as shown below.



27. Adjust the heading as shown below.



Creating Cards

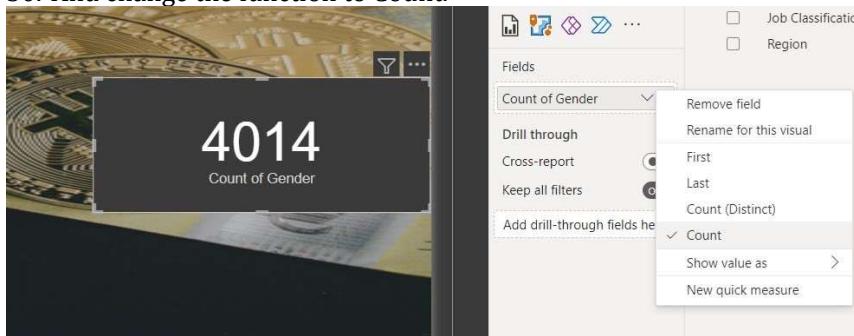
A card is useful to display a single number (or metric value). For example, If we want to track the total orders, total sales value, or total quotations we sent, then we can use this Card visualization.

28. In order to create one, first, click on the Card present in the Visualization section. It will create a Card with dummy data, as shown in the screenshot below.

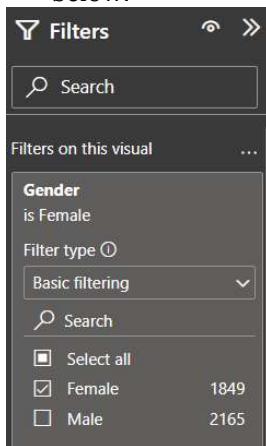


29. Let me drag the Gender into the Fields section as shown below.

30. And change the function to Count.



31. Go to Filter space and click on the drop down of gender and select Female as shown below.



32. Now create another card with same field but this time filter with Male

33. Change the following format as shown below.

34. After changing the format adjust the cards as shown below.



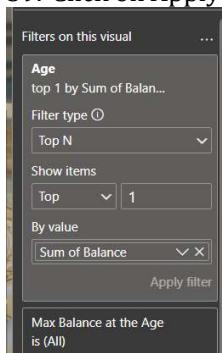
35. Create another 2 cards and use the same format.

36. For third card use Age field with the function as Maximum.

37. And in the filter drag and drop the Age field set the filter type as Top N and set the show Items as shown below.

38. In By value drag and drop Balance field as shown below.

39. Click on Apply filter.



40. For forth card use Age field with the function as Minimum.

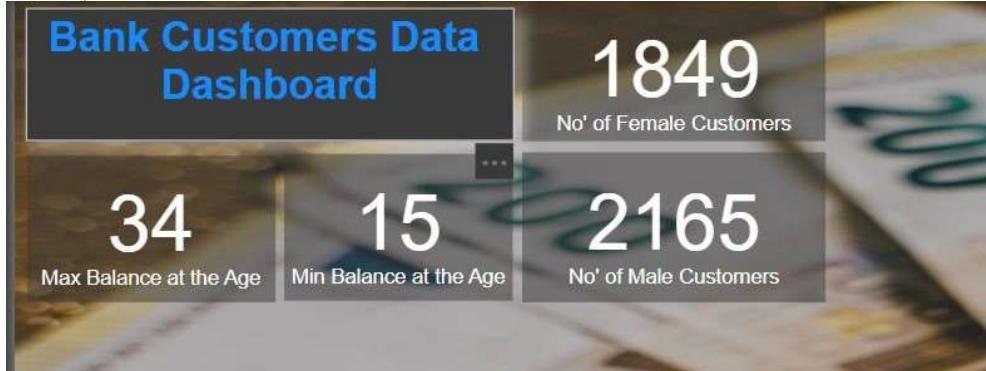
41. And in the filter drag and drop the Age field set the filter type as Top N and set the show Items as shown below.

42. In By value drag and drop Balance field as shown below.

43. Click on Apply filter.



44. Adjust the cards as shown below.



Donut Chart

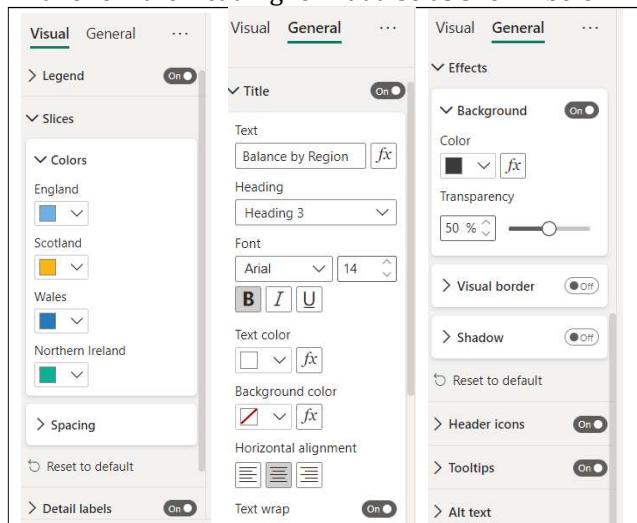
Power BI Donut Chart is similar to Pie, which is useful to visualize the higher-level data.

45. When you click on this under the Visualization section, it automatically creates a Donut Chart with dummy data.
46. Drag and Drop the Region from Fields section to Legend field.
47. And Drag and Drop the Balance from Fields section to Values field as shown below.

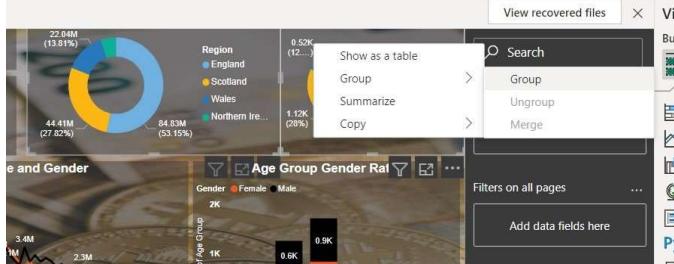


48. Change the slicer color in format visual as shown below.

49. And follow the Heading format also as shown below.



50. Do the same formatting for other donut charts also. We need to create another two chats.
51. For second donut chart, Drag and Drop the Region from Fields section to Legend field.
52. And Drag and Drop the Region from Fields section to Values field.
53. Set the Region function to Count in the values field
54. Select first and second donut chart right click on that go to group and click on group as shown below.



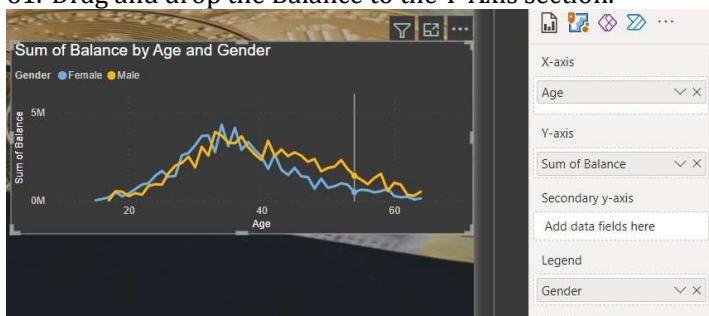
55. For third donut chart, Drag and Drop the Job Classification from Fields section to Legend field.
56. And Drag and Drop the Balance from Fields section to Values field.
57. Set the headings as shown below for each donut chart.
58. Adjust the charts as shown below.



Line Chart

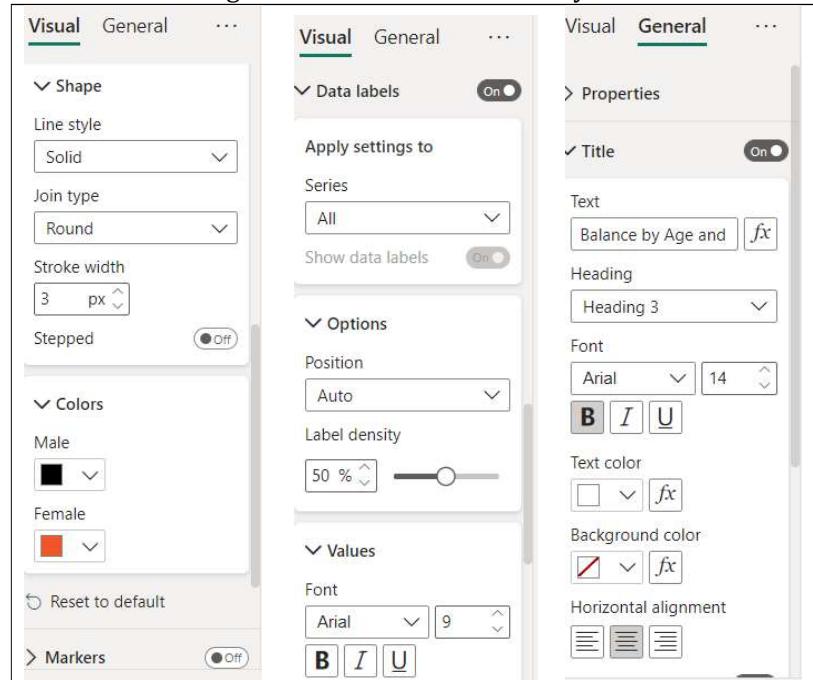
Power BI Line Chart is useful for visualizing Trends over time. For example, you can use this to create a Sales trend, Temperate Trend, etc.

59. Click on the Line Chart under the Visualization section, automatically create a Line Chart.
60. Drag and drop the Age to the X-Axis section.
61. Drag and drop the Balance to the Y-Axis section.



62. Change the following format as shown below.

63. Use the same background format above for every visual.



64. After changing format adjust the Chart as shown below.



Stacked column chart

65. Power BI Stacked Column Chart is useful to visualize multiple dimensions against a single measure.

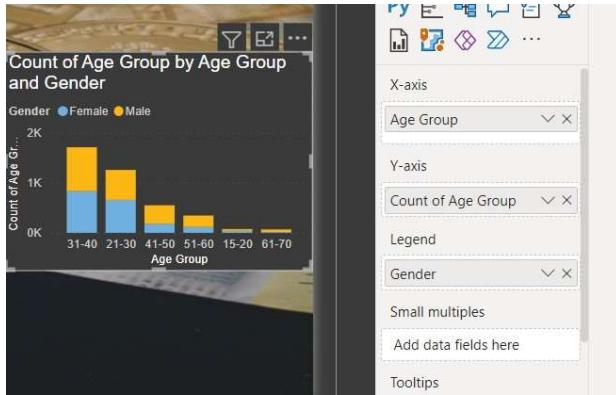
66. Click on the Stacked Column Chart under the Visualization section. It will automatically create a Stacked Column Chart with dummy data.

67. Drag and drop the Age Group from Fields section to Y-axis field.

68. Set the function as count.

69. Drag and drop the Age Group from fields section to X-axis field.

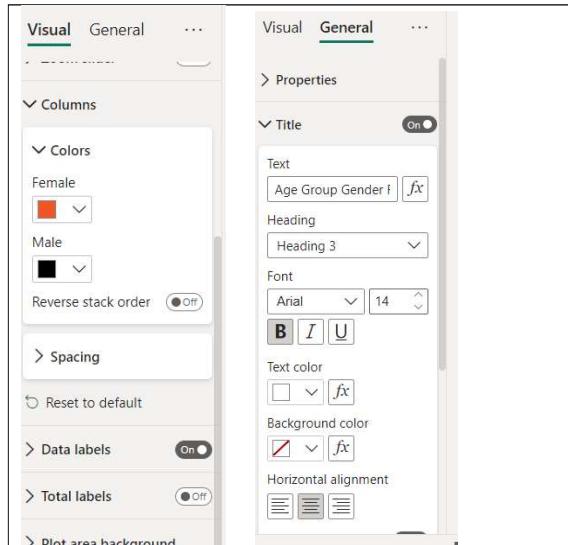
70. Drag and drop the Gender from fields section to Legend field



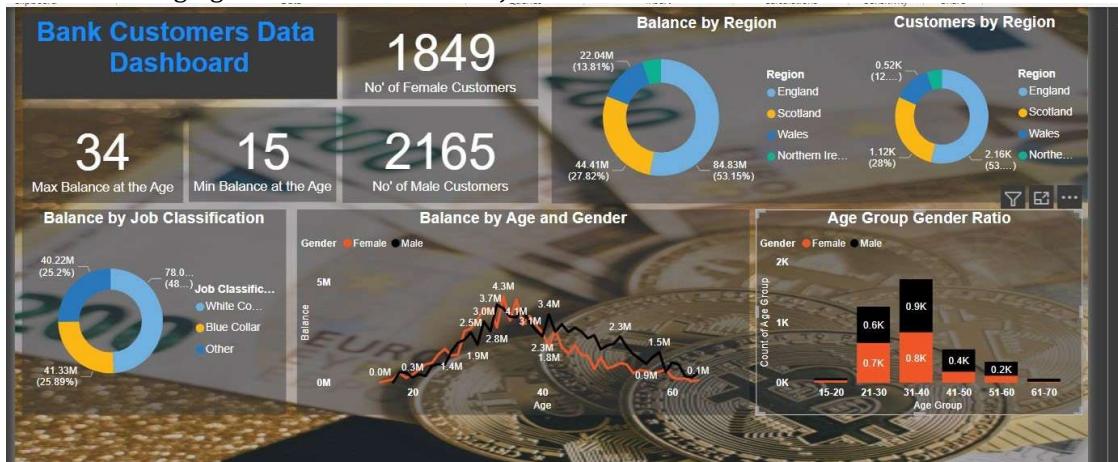
71. Change the column color as shown below.

72. And change the Title format also as shown below.

73. Use the same background format above for every visual.



74. After changing it looks like this and adjust the chart as shown below.



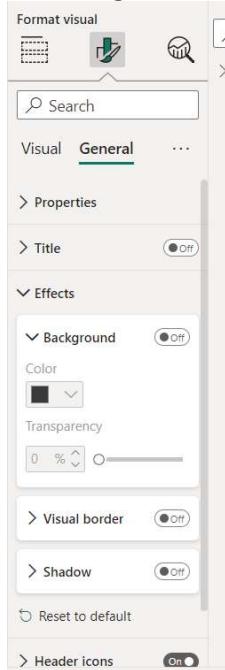
Using natural Language Q&A

Sometimes the fastest way to get an answer from your data is to ask a question using natural language. The Q&A feature in Power BI lets you explore your data in your own words.

75. Click on the Q & A under the Visualization section. It will automatically create a Q & A with dummy data.
76. Begin typing your own question and select from the dropdown suggestions.
77. As you type a question, Q&A picks the best visualization to display your answer.
78. In our case we type question as [customer by date joined](#) as shown below.



79. Change the following format for the chart as shown.



80. Adjust the chart as shown below.



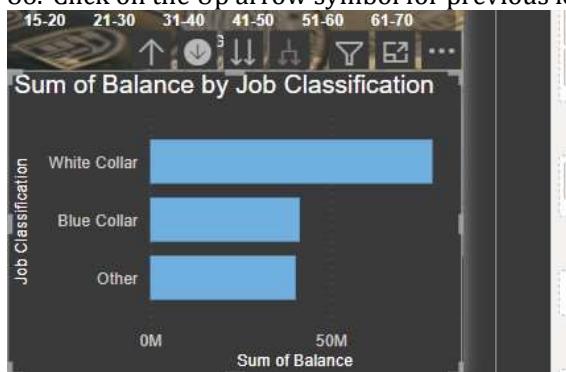
Drill Up and Drill Down

Power BI Drill Up and Drill Down Reports help you to drill down the reports to multi-levels. You can see the data at each level using this Drill Down and Up report action.

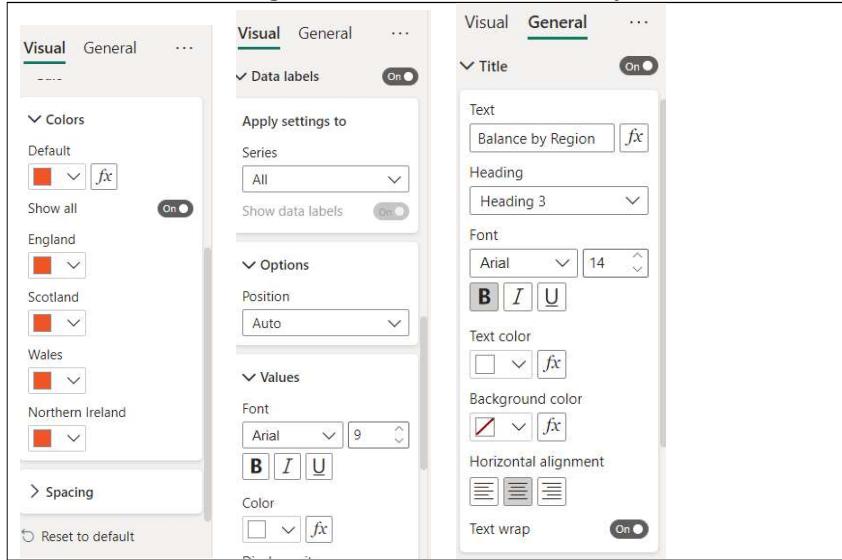
81. Click on the Clustered bar Chart under the Visualization section. It will automatically create a Clustered bar Chart with dummy data.
82. Drag and drop the Region, Job Classification and Gender from Fields section to Y-axis field.
83. Drag and drop the Balance from fields section to X-axis field.



84. Click the Down arrow symbol on the Chart to turn ON the drill down.
85. After click the two down arrow symbols to go to next level.
86. Click on the Up arrow symbol for previous level.



87. Change the column color as shown below.
 88. And change the Title format also as shown below.
 89. Use the same background format above for every visual.



90. Adjust the Chart as shown below.

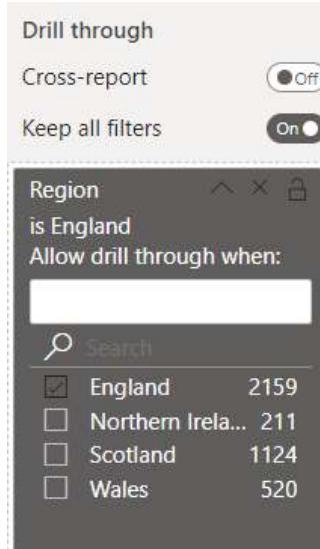


Create a drill through page

1. Add a new report page named Customer Details.
2. Right-click the Customer Details page tab, and then select Hide Page.



3. Beneath the Visualizations pane, in the Drill Through section, add the Region field to the Add Drill-Through Fields Here box.
4. To test the drill through page, in the drill through filter card, select England.



5. At the top-left of the report page, notice the arrow button.

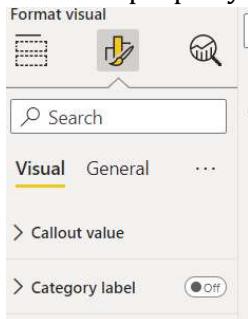


6. Add a Card visual to the page, and then resize and position it so it sits to the right of the button and fills the remaining width of the page.



7. Drag the Region field into the card visual.

8. Configure the format options for the visual, and then turn the Category Label property to Off.



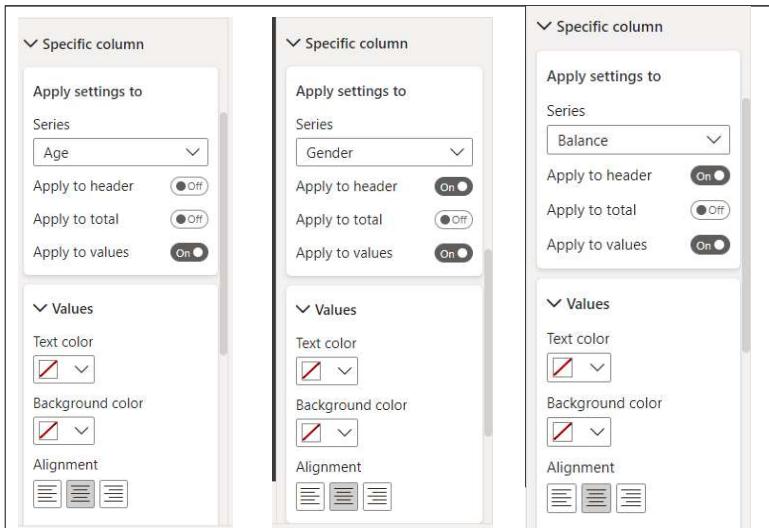
9. Add a **Table** visual to the page, and then resize and position it so it sits beneath the card visual and fills the remaining space on the page.

10. Add the following fields to the visual:

The screenshot shows the 'Columns' section of the Table visual configuration. It lists four fields: 'Customer Name', 'Age', 'Gender', and 'Balance', each with a 'X' icon to the right.

5. Change the following format for the Table as shown.

The screenshot shows the 'Visual' tab settings for the Table visual. On the left, under 'Style presets', 'Alternating rows' is selected. In the center, under 'Values', 'Customer Name' is selected for 'Apply settings to'. The 'Text color' is set to black and the 'Background color' is set to green. On the right, under 'Specific column', 'Customer Name' is selected again. The 'Text color' is set to red and the 'Background color' is set to white. Both 'Apply to header' and 'Apply to values' are turned on.



6. Adjust the Card and Table in the page as shown below.

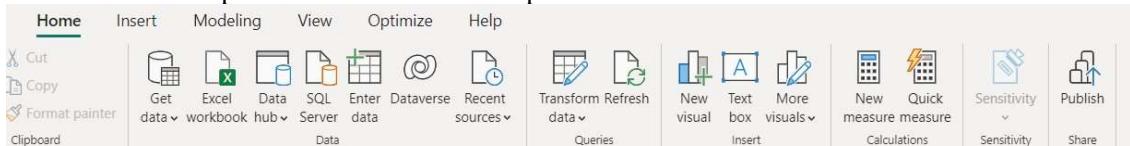
Northern Ireland

Customer Name	Age	Gender	Balance
Abraham Amy	42	Female	46.39K
Abraham Gordon	30	Male	83.55K
Alsop Emily	25	Female	29.64K
Anderson Blake	32	Male	30.92K
Avery Jan	29	Female	0.74K
Bailey Peter	41	Male	85.19K
Baker Fiona	31	Female	121.13K
Baker Jane	27	Female	18.60K
Baker Lily	23	Female	55.82K
Baker Neil	25	Male	44.58K
Ball Pippa	33	Female	0.13K
Bell Dorothy	34	Female	15.26K
Berry Audrey	31	Female	20.10K
Blake Joan	35	Female	27.87K
Blake Rose	25	Female	49.00K
Bower Amelia	22	Female	22.52K

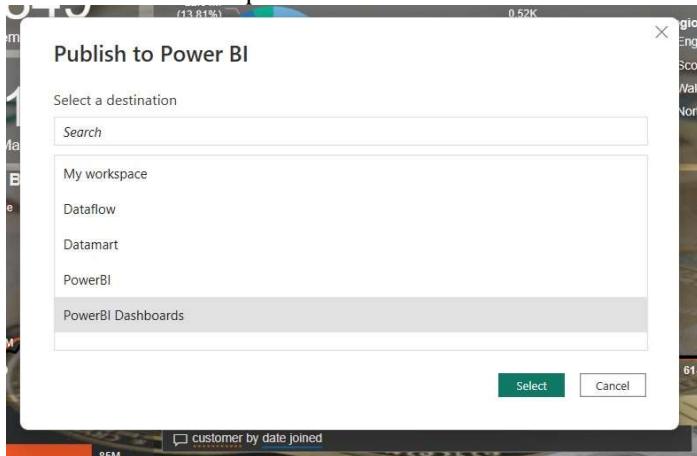
25. To test the drill through go to our main dashboard.
26. Right click on the Customer by Region Donut chart click on the Drill Through and select the Customer Details report page.
27. It will take you to the Customer Details report page.

Publishing

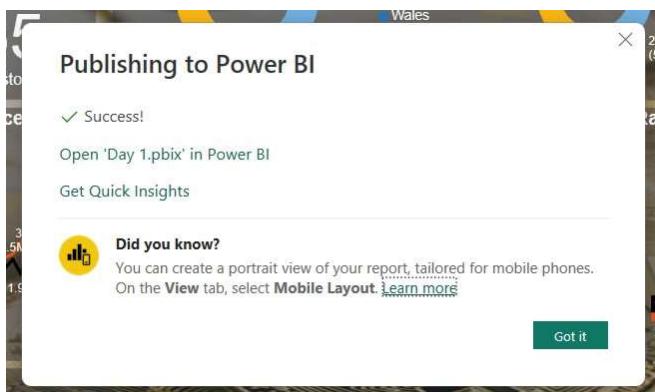
1. Publish the Report click on the Publish option as shown below.



2. Now select which workspace you want to publish the report.
3. Select the workspace and click on select.



4. After publishing you will get success full notification as shown

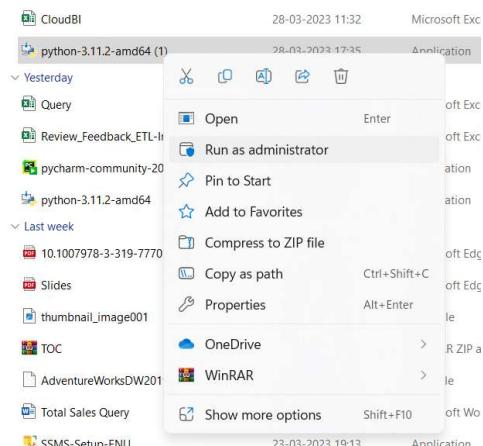


Download and install Python

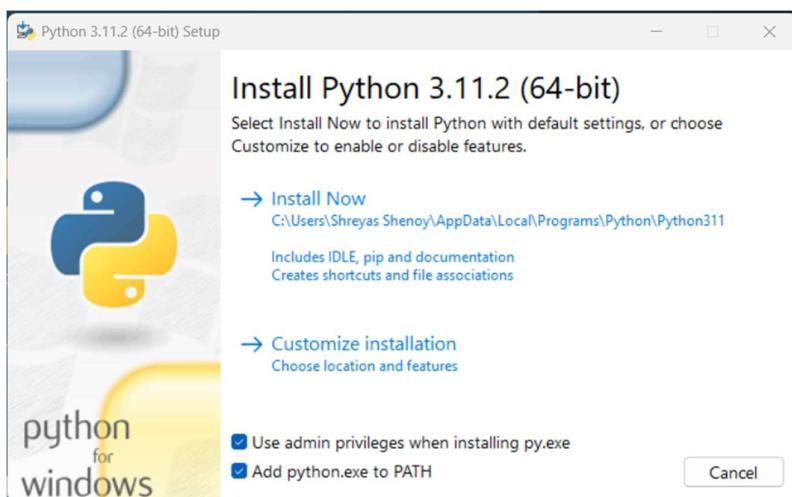
1. To download, first Go to the official website by clicking this link [Download Python | Python.org](#). It shows you the Downloads page of the official website.
2. Here, you can select the Version. For now, we are selecting the latest version.
3. Click on the download 3.11.2 button to start the process.



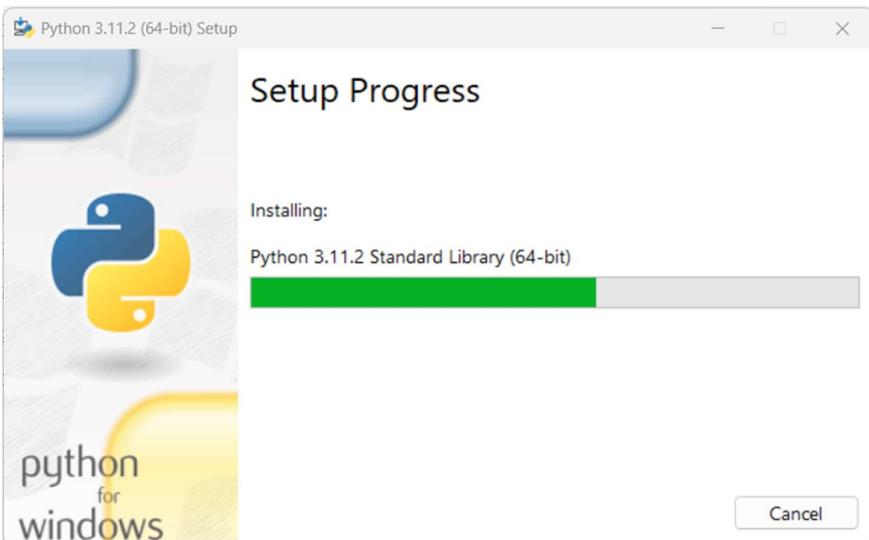
4. After downloading the file. Go to the file location right click on the file and click on Run as Administrator as shown below.



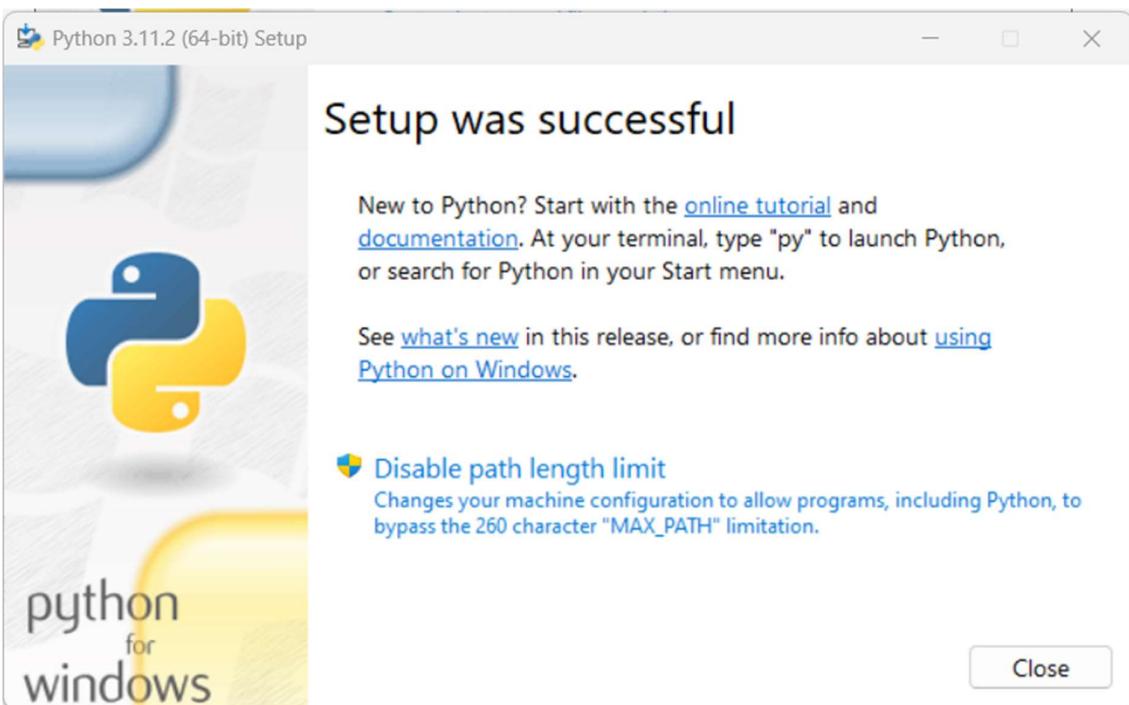
5. Now check both options below and click on the Install Now option as shown below.



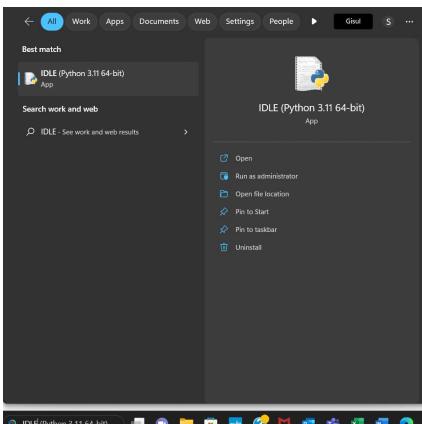
6. Here it takes a few minutes to install.



7. Now our installation is completed.

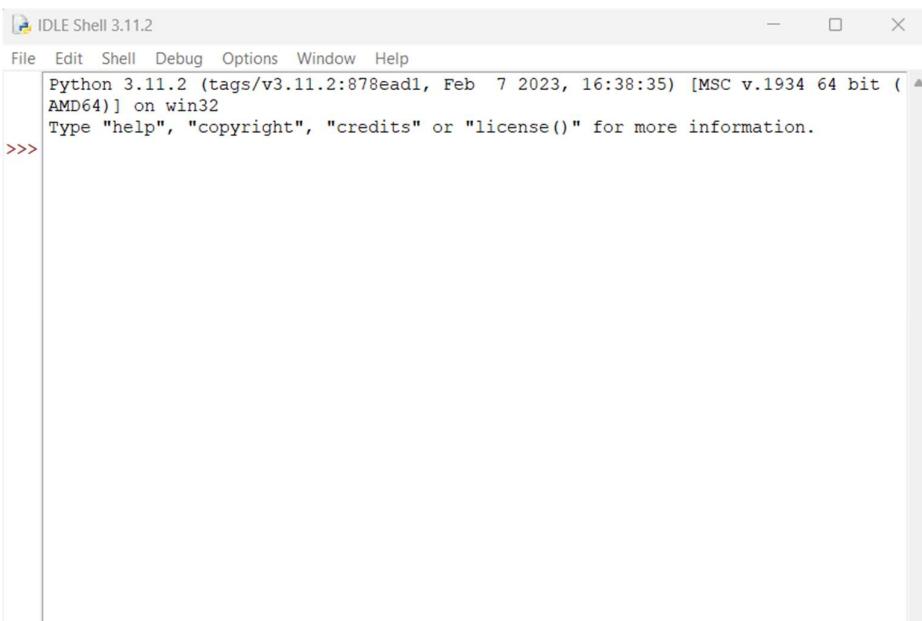


8. Next go to search box and type IDLE and open the app as shown below.

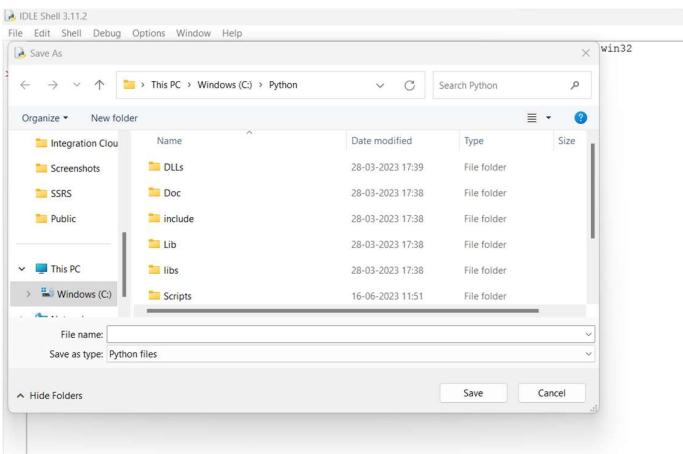


9. Now you will see a window as shown below.

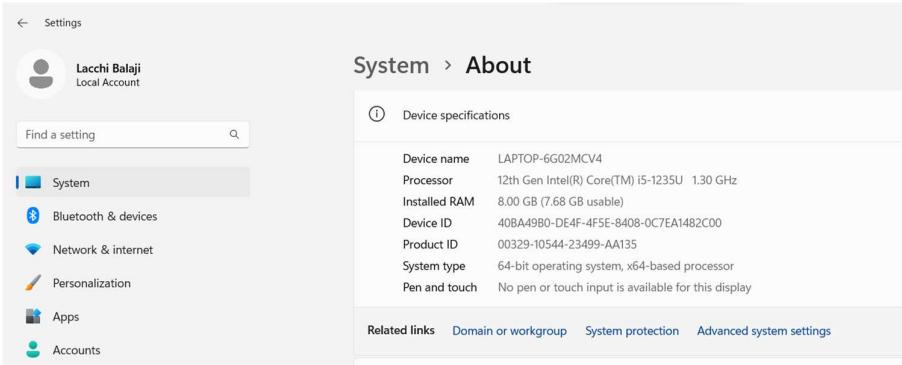
10. Now click on ctrl+s.



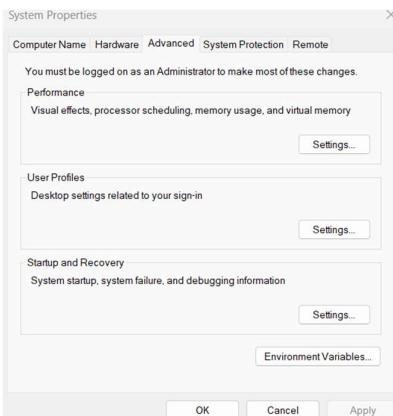
11. Copy the location address as shown below.



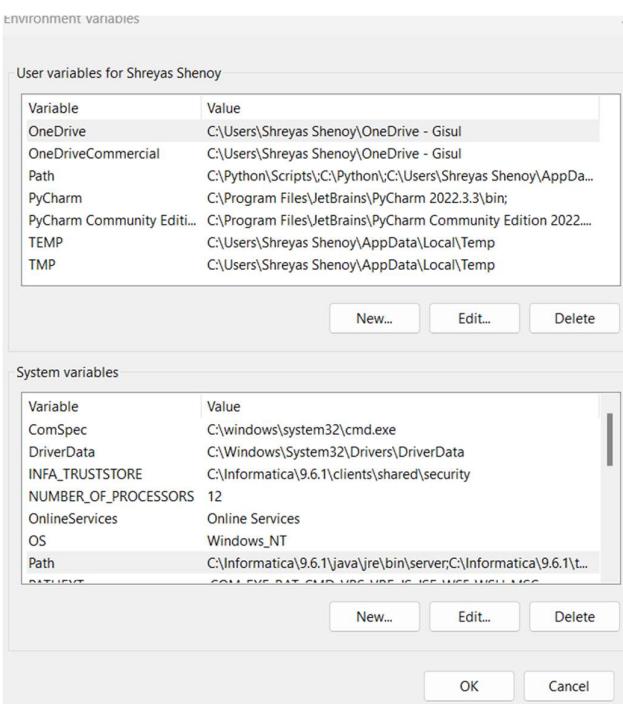
12. Next, go to settings and go to the below location then click on Advanced system settings as shown below.



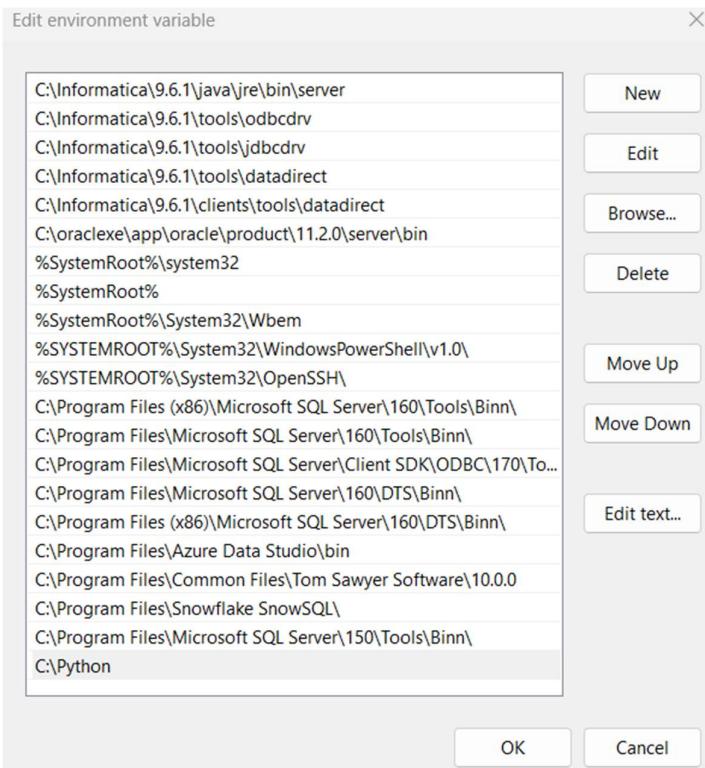
13. Click on Environment variables as shown below.



14. Under System Variables double click on Path as shown below.

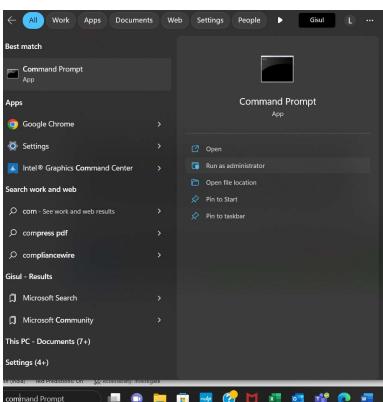


15. Now click on New and paste the location address and click ok.



Install PYODBC

16. In the search box type command prompt and click on Run as Administrator.



17. Give the below command and click enter to connect the Python.

Command: cd <paste the location address that we copied before.>

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.22621.1848]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd C:\Python

C:\Python>
```

18. Next to install the PYODBC give the below command.

Command: python -m pip install pyodbc

```
C:\Python>python -m pip install pyodbc
Requirement already satisfied: pyodbc in c:\python\lib\site-packages (4.0.35)

[notice] A new release of pip is available: 23.0.1 -> 23.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip
```

19. To update the pyodbc give the below command.

Command: python.exe -m pip install --upgrade pip

```
C:\Python>python.exe -m pip install --upgrade pip
Requirement already satisfied: pip in c:\python\lib\site-packages (23.0.1)
Collecting pip
  Downloading pip-23.1.2-py3-none-any.whl (2.1 MB)
    2.1/2.1 MB 3.1 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 23.0.1
    Uninstalling pip-23.0.1:
      Successfully uninstalled pip-23.0.1
Successfully installed pip-23.1.2

C:\Python>
```

Install Pandas

1. Now we will install Panda which will use to read our files.
2. To install pandas first we need to install a wheel and give the below command.

Command: python -m pip install wheel

```
C:\Python>python -m pip install wheel
Collecting wheel
  Downloading wheel-0.40.0-py3-none-any.whl (64 kB)
    64.5/64.5 kB 1.8 MB/s eta 0:00:00
Installing collected packages: wheel
Successfully installed wheel-0.40.0
```

3. To install a Pandas, give the below command.

Command: python -m pip install pandas

```
C:\Python>python -m pip install pandas
Collecting pandas
  Downloading pandas-2.0.2-cp311-cp311-win_amd64.whl (10.6 MB)
    10.6/10.6 MB 2.5 MB/s eta 0:00:00
Collecting python-dateutil>=2.8.2 (from pandas)
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
    247.7/247.7 kB 2.2 MB/s eta 0:00:00
Collecting pytz>=2020.1 (from pandas)
  Downloading pytz-2023.3-py2.py3-none-any.whl (502 kB)
    502.3/502.3 kB 3.2 MB/s eta 0:00:00
Collecting tzdata>=2022.1 (from pandas)
  Downloading tzdata-2023.3-py2.py3-none-any.whl (341 kB)
    341.8/341.8 kB 2.4 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.21.0 in c:\python\lib\site-packages (from pandas) (1.24.2)
Collecting six>=1.5 (from python-dateutil>=2.8.2->pandas)
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, tzdata, six, python-dateutil, pandas
Successfully installed pandas-2.0.2 python-dateutil-2.8.2 pytz-2023.3 six-1.16.0 tzdata-2023.3

C:\Python>
```

Install openpyxl

1. Openpyxl is used to read and write an Excel file in Python.
2. To install an openpyxl give the following command as shown below.

Command: python -m pip install openpyxl

```
C:\Python>python -m pip install openpyxl
Collecting openpyxl
  Downloading openpyxl-3.1.2-py2.py3-none-any.whl (249 kB)
                                             250.0/250.0 kB 2.6 MB/s eta 0:00:00
Collecting et-xmlfile (from openpyxl)
  Downloading et_xmlfile-1.1.0-py3-none-any.whl (4.7 kB)
Installing collected packages: et-xmlfile, openpyxl
Successfully installed et-xmlfile-1.1.0 openpyxl-3.1.2
```

Install matplotlib

1. Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.
2. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.
3. To install a matplotlib give the below command as shown below.

Command: python -m pip install matplotlib

```
C:\Python>python -m pip install matplotlib
Collecting matplotlib
  Downloading matplotlib-3.7.1-cp311-cp311-win_amd64.whl (7.6 MB)
                                             7.6/7.6 kB 503.6 kB/s eta 0:00:00
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.1.0-cp311-cp311-win_amd64.whl (470 kB)
                                             #70.9/470.9 kB 1.6 MB/s eta 0:00:00
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.11.0-py3-none-any.whl (6.4 kB)
Collecting fonttools>=4.22.0 (from matplotlib)
  Downloading fonttools-4.40.0-cp311-cp311-win_amd64.whl (1.9 MB)
                                             1.9/1.9 kB 1.1 MB/s eta 0:00:00
Collecting kiwisolver>=1.0.1 (from matplotlib)
  Downloading kiwisolver-1.4.4-cp311-cp311-win_amd64.whl (55 kB)
                                             55.4/55.4 kB 1.5 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.20 in c:\python\lib\site-packages (from matplotlib) (1.24.2)
Collecting packaging>=20.0 (from matplotlib)
  Downloading packaging-23.1-py3-none-any.whl (48 kB)
                                             48.9/48.9 kB 2.6 MB/s eta 0:00:00
Collecting pillow>=6.2.0 (from matplotlib)
  Downloading Pillow-9.5.0-cp311-cp311-win_amd64.whl (2.5 MB)
                                             2.5/2.5 kB 1.2 MB/s eta 0:00:00
Collecting pyparsing>=2.3.1 (from matplotlib)
  Downloading pyparsing-3.1.0-py3-none-any.whl (102 kB)
                                             102.6/102.6 kB 1.2 MB/s eta 0:00:00
Requirement already satisfied: python-dateutil>=2.7 in c:\python\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\python\lib\site-packages (from python-dateutil>=2.7>matplotlib) (1.16.0)
Installing collected packages: pyparsing, pillow, packaging, kiwisolver, fonttools, cycler, contourpy, matplotlib
Successfully installed contourpy-1.1.0 cycler-0.11.0 fonttools-4.40.0 kiwisolver-1.4.4 matplotlib-3.7.1 packaging-23.1 pillow-9.5.0 pyparsing-3.1.0
C:\Python>
```

Install Numpy

1. To install the numpy use below command as shown below.

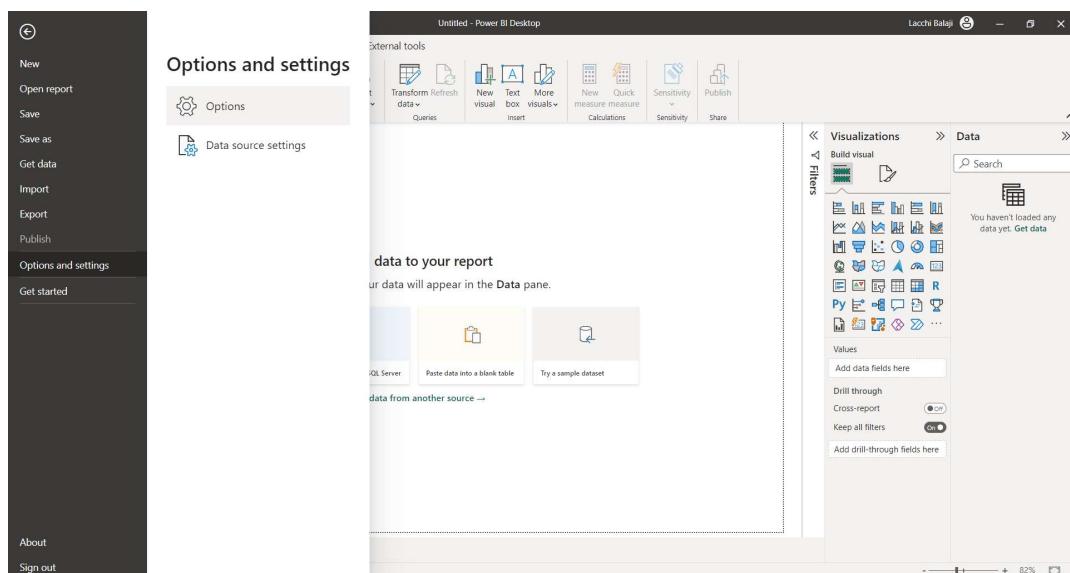
Command: python -m pip install numpy

```
C:\Python>python -m pip install numpy
Requirement already satisfied: numpy in c:\python\lib\site-packages (1.24.2)
```

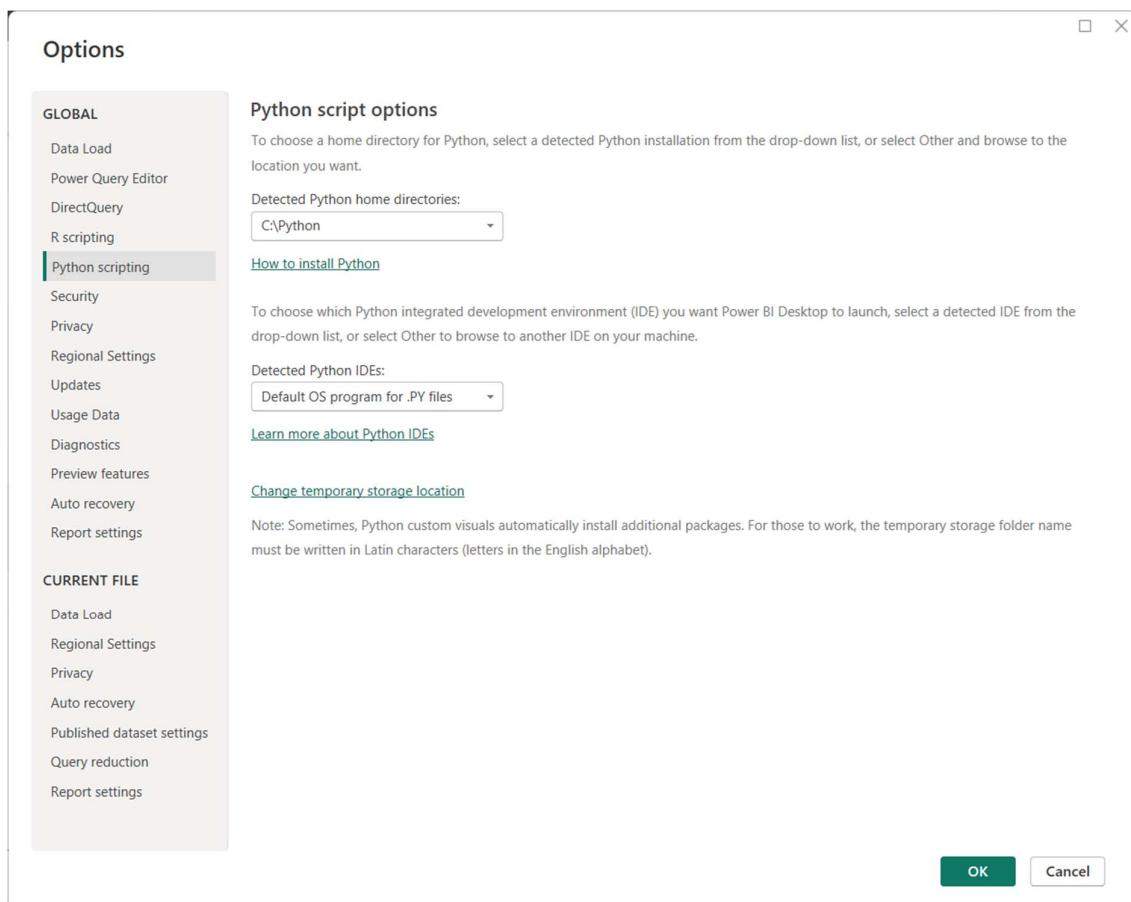
2. In my case I have already install numpy in system that's why it showing me above message.

Get Data

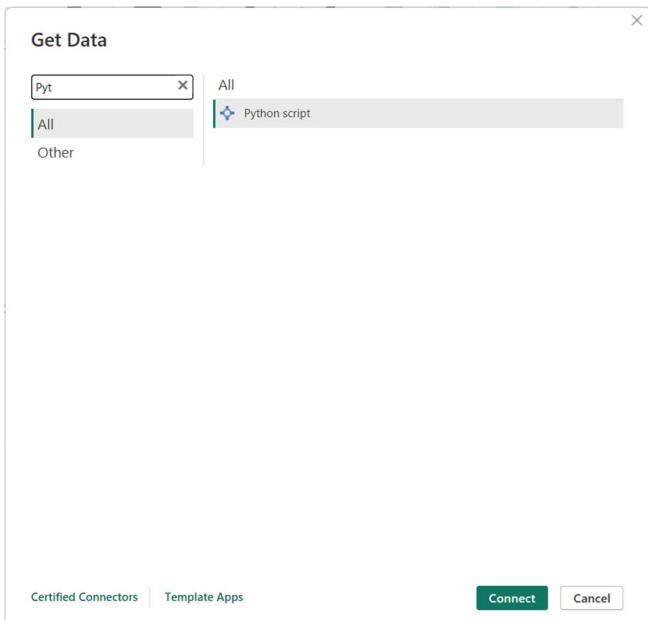
1. Open the PowerBI desktop and click on file then go to Option and settings then click on Options.



2. In the Python scripting option set the properties as shown below.

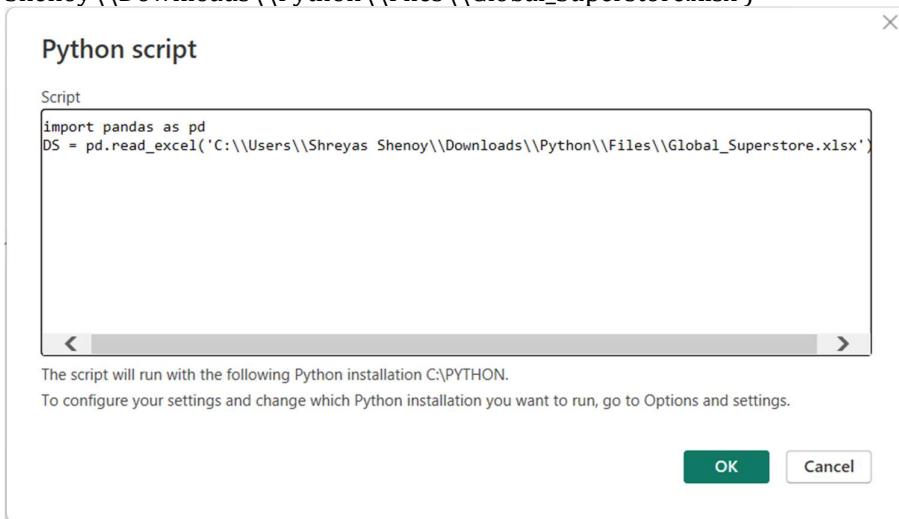


3. Now to load the data using the python script click on the get data option. Then in the search box type Python and click on Python Script then click on connect as shown below.



4. Next give the following code.
5. Give the file location where you have stored it in your local system.

Code: `import pandas as pd
DS = pd.read_excel('C:\\\\Users\\\\Shreyas
Shenoy\\\\Downloads\\\\Python\\\\Files\\\\Global_Superstore.xlsx')`



6. Next select the table and click on load as shown below.

Row ID	Order ID	Order Date	Ship Date	Ship F
32298	CA-2012-124891	7/31/2012	7/31/2012	Same
26341	IN-2013-77878	2013-05-02 00:00:00	2013-07-02 00:00:00	Secon
25330	IN-2013-71249	10/17/2013	10/18/2013	First C
13524	ES-2013-1579342	1/28/2013	1/30/2013	First C
47221	SG-2013-4320	2013-05-11 00:00:00	2013-06-11 00:00:00	Same
22732	IN-2013-42360	6/28/2013	2013-01-07 00:00:00	Secon
30570	IN-2011-81826	2011-07-11 00:00:00	2011-09-11 00:00:00	First C
31192	IN-2012-86369	4/14/2012	4/18/2012	Stand
40155	CA-2014-135909	10/14/2014	10/21/2014	Stand
40936	CA-2012-116638	1/28/2012	1/31/2012	Secon
34577	CA-2011-102988	2011-05-04 00:00:00	2011-09-04 00:00:00	Secon
28879	IN-2012-28402	4/19/2012	4/22/2012	First C
45794	SA-2011-1830	12/27/2011	12/29/2011	Secon
4132	MX-2012-130015	11/13/2012	11/13/2012	Same
27704	IN-2013-73951	2013-06-06 00:00:00	2013-08-06 00:00:00	Secon
13779	ES-2014-5099955	7/31/2014	2014-03-08 00:00:00	Secon
36178	CA-2014-143567	2014-03-11 00:00:00	2014-06-11 00:00:00	Secon
12069	ES-2014-1651774	2014-08-09 00:00:00	9/14/2014	Stand
22096	IN-2014-11763	1/31/2014	2014-01-02 00:00:00	First C
49463	TZ-2014-8190	2014-05-12 00:00:00	2014-07-12 00:00:00	Secon

7. As you can we have loaded our data into the Power BI desktop.

Bar Chart

1. To create a bar chart using Python script click on Py in the visualizations and it will ask you to enable script visuals click on Enable as shown below.

2. Now drag and drop the Region and Sales fields into the values as shown below.

The screenshot shows the Power BI desktop interface. On the left is a blank report canvas with a single rectangular placeholder. To the right is the 'Visualizations' pane, which contains various chart and report icons. Below the pane is the 'Values' section, where 'Region' and 'Sum of Sales' are listed under 'Selected'. A status bar at the bottom indicates 'from the data.' and 'dataframe and remove duplicated rows is always executed and acts as a preamble for your script:' followed by some code snippets.

3. Now in the script editor give the following code and click on the play button.

```
Code: import matplotlib.pyplot as plt
sales_data = dataset.groupby('Region')[['Sales']].sum()
```

```
colors = ['red', 'green', 'blue', 'yellow', 'black', 'cyan']
```

```
fig, ax = plt.subplots()
```

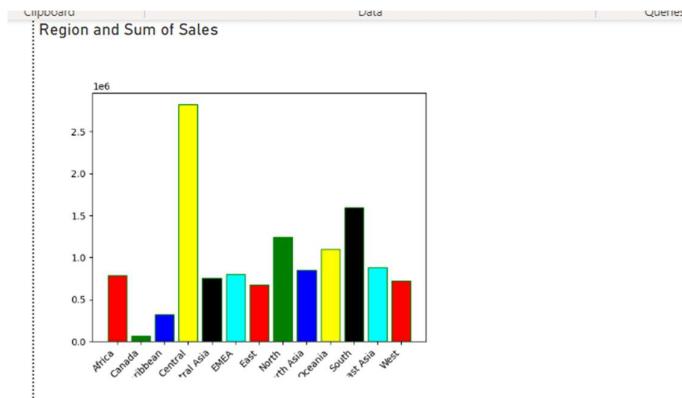
```
ax.bar(sales_data.index, sales_data['Sales'],
       color = colors, edgecolor = 'green')
```

```
labels = ax.get_xticklabels()
```

```
plt.setp(labels, rotation = 45, horizontalalignment = 'right')
plt.show()
```

The screenshot shows the Power BI desktop interface. On the left is a bar chart displayed in a report canvas. The chart has 'Region' on the x-axis and 'Sales' on the y-axis, with bars colored according to the 'colors' list in the script. To the right is the 'Python script editor' window, which contains the provided Python code. The 'Visualizations' pane is also visible on the far right.

4. As you can we have created Bar Chart using Python Script.



Pie Chart

- To create a Pie Chart using Python script click on Py and drag and drop the following fields in the values as shown below.

```
from py import *
s()
Priority, Sales
s()
Order Priority and Sum of Sales
m the data.
ataframe and remove duplicated rows is always executed and acts as a preamble for your script:
Priority, Sales
s()
2. Now in the script editor give the following code and click on the play button.
```

Code:

```
import matplotlib.pyplot as plt
sales_data = dataset.groupby('Order Priority')[['Sales']].sum()

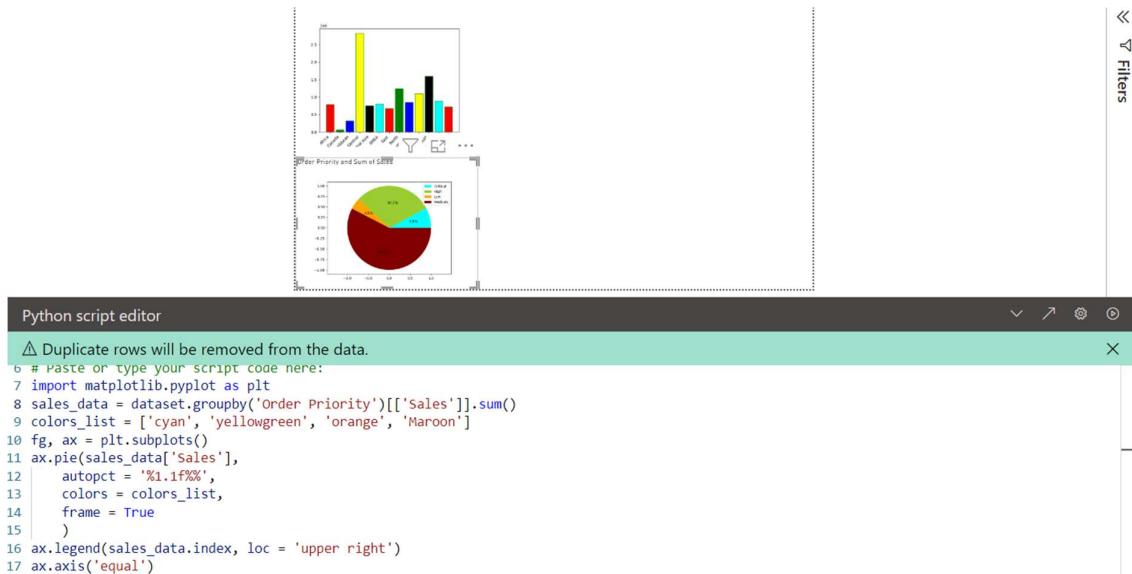
colors_list = ['cyan', 'yellowgreen', 'orange', 'Maroon']

fg, ax = plt.subplots()

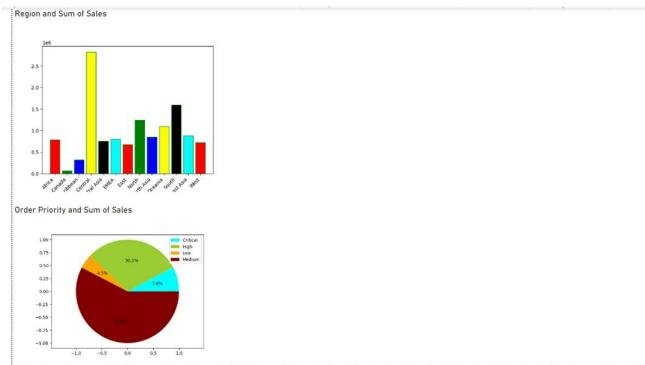
ax.pie(sales_data['Sales'],
       autopct = '%1.1f%%',
       colors = colors_list,
       frame = True
       )

ax.legend(sales_data.index, loc = 'upper right')

ax.axis('equal')
plt.show()
```

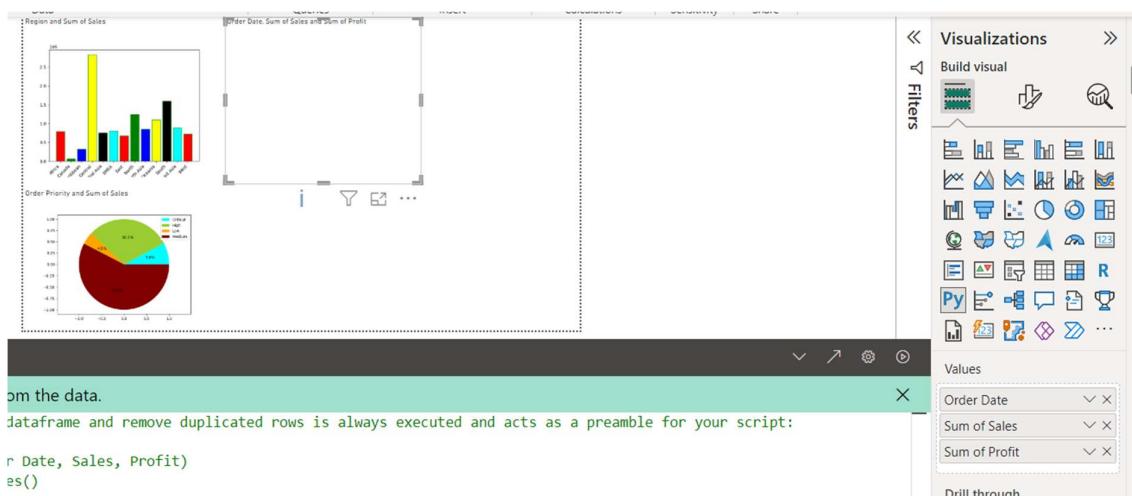


3. We have created a pie chart using Python Script.



Scatter Chart

- To create a Scatter Chart using Python script click on Py and drag and drop the following fields in the values as show below.



- Now in the script editor give the following code and click on the play button.

Code: `import matplotlib.pyplot as plt
sales_data = df.groupby('Order Date')[['Sales', 'Profit']].sum()`

```

print(sales_data.sort_values(by = ['Profit']))

plt.scatter(sales_data['Profit'], sales_data['Sales'])
plt.title('Example')
plt.xlabel('Profit')
plt.ylabel('Global Sales')
plt.show()

```

The screenshot shows a Python script editor window with the following code:

```

# Duplicate rows will be removed from the data.
4 # dataset = dataset.drop_duplicates()
5
6 # Paste or type your script code here:
7 import matplotlib.pyplot as plt
8 sales_data = dataset.groupby('Order Date')[['sales', 'Profit']].sum()
9
10 print(sales_data.sort_values(by = ['Profit']))
11
12 plt.scatter(sales_data['Profit'], sales_data['Sales'])
13 plt.xlabel('Profit')
14 plt.ylabel('Global Sales')
15 plt.show()

```

The editor has a green status bar at the bottom stating: "△ Duplicate rows will be removed from the data."

3. We have created Scatter chart using Python Script.



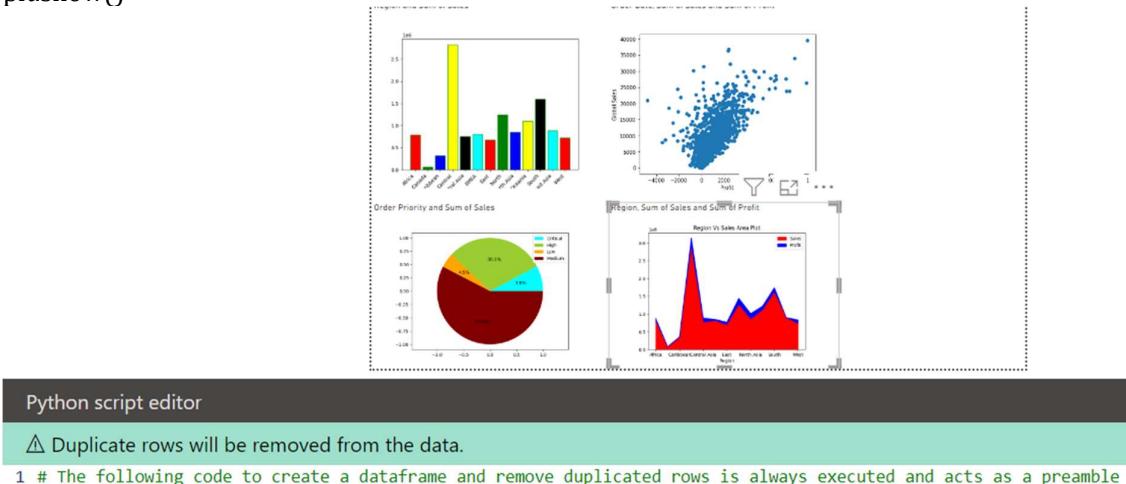
Area Chart

1. To create an Area Chart using Python script click on Py and drag and drop the following fields in the values as shown below.



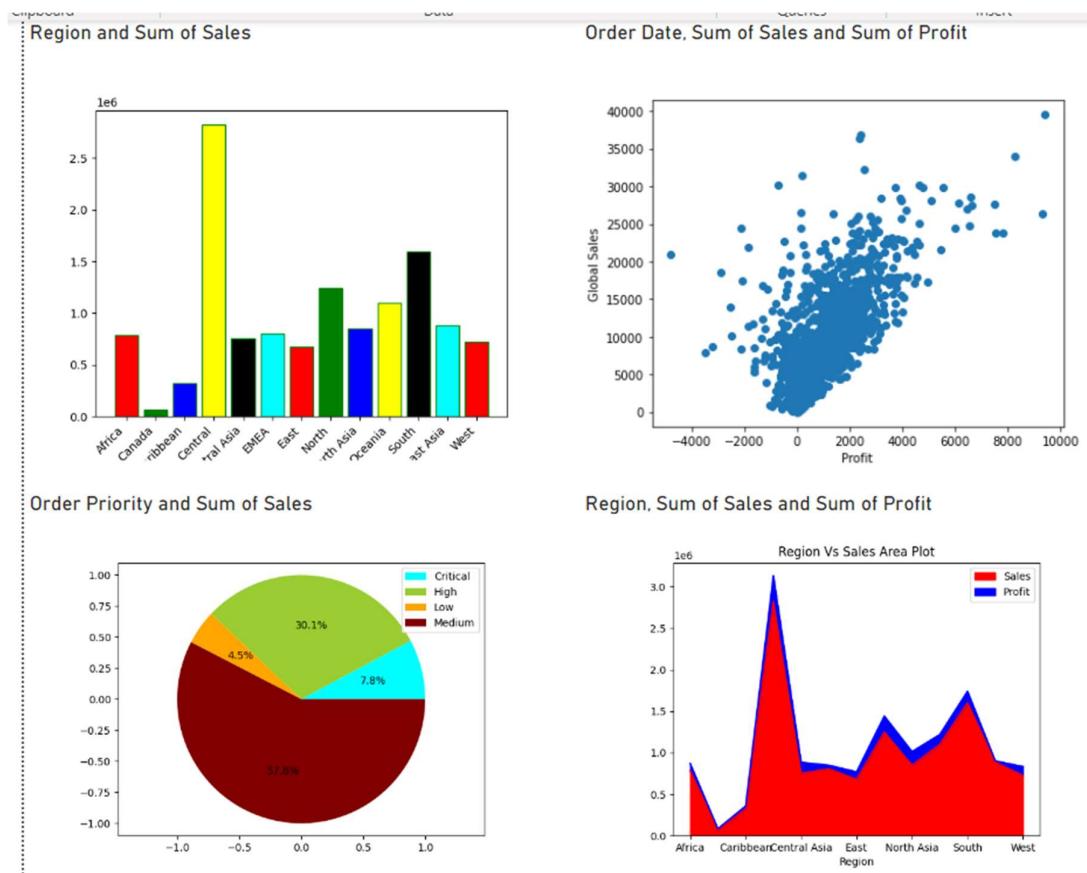
2. Now in the script editor give the following code and click on the play button.

Code: import matplotlib.pyplot as plt
groupby_Occ = dataset.groupby('Region')[['Sales', 'Profit']].sum()
groupby_Occ.plot.area(title = 'Region Vs Sales Area Plot', legend = True, color = ['r', 'b'])
plt.show()



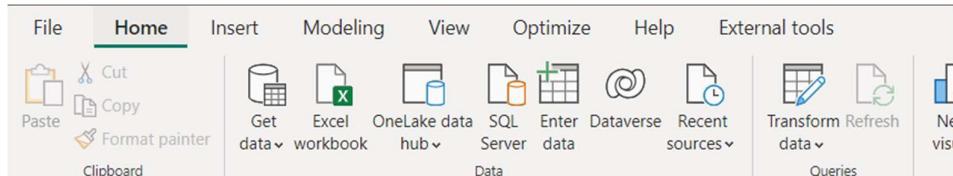
```
1 # The following code to create a dataframe and remove duplicated rows is always executed and acts as a preamble for your script:
2
3 # dataset = pandas.DataFrame(Region, Sales, Profit)
4 # dataset = dataset.drop_duplicates()
5
6 # Paste or type your script code here:
7 import matplotlib.pyplot as plt
8 groupby_Occ = dataset.groupby('Region')[['Sales', 'Profit']].sum()
9 groupby_Occ.plot.area(title = 'Region Vs Sales Area Plot', legend = True, color = ['r', 'b'])
10 plt.show()
```

3. We have created Area chart using Python Script.

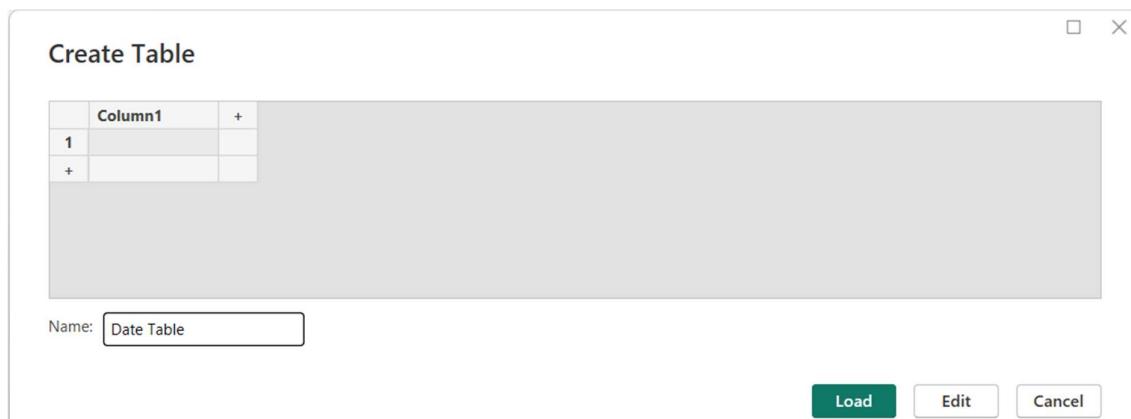


Create Date Table

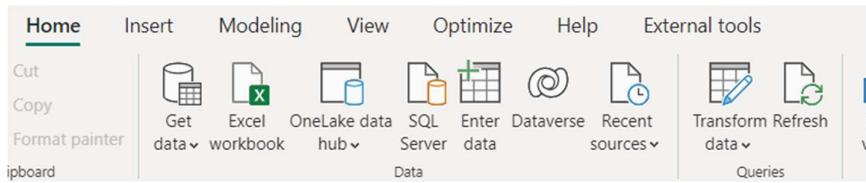
1. Click on Enter Data option.



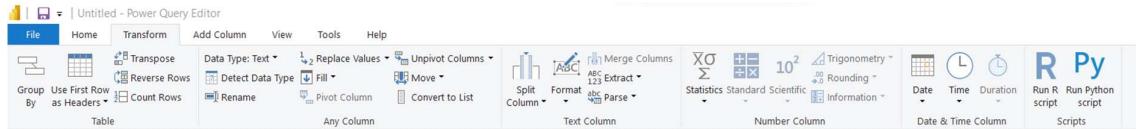
2. Give the name as Date Table and click on Load as shown below.



3. Click on the Transform Data option.

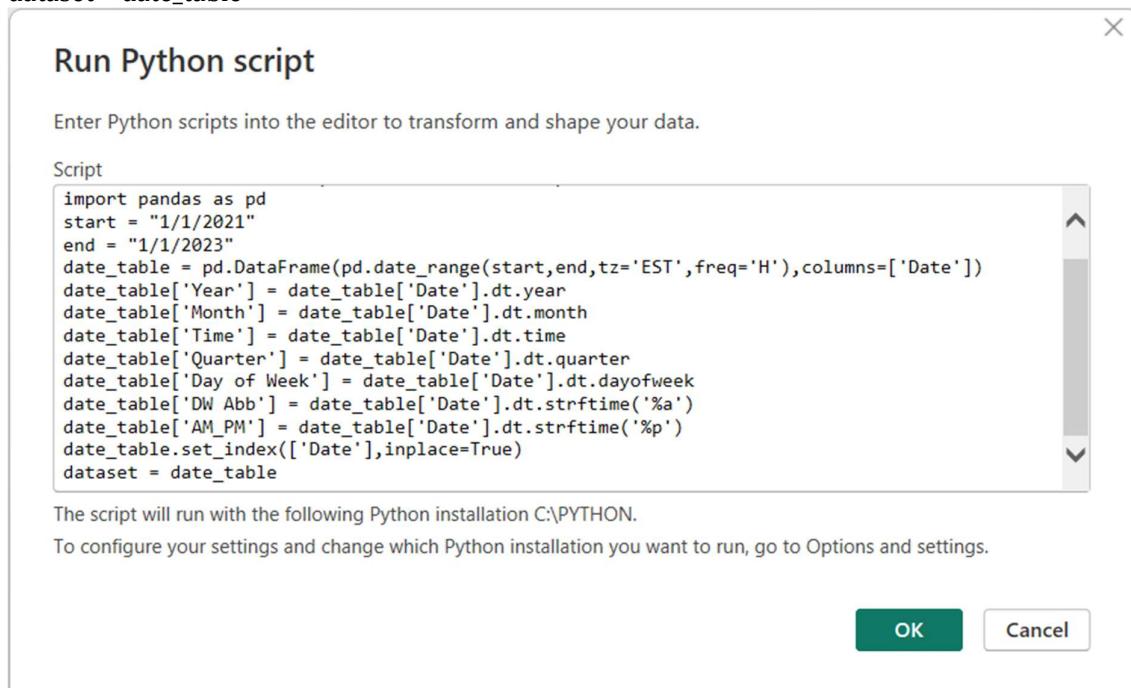


4. In the Power Query under Transformation tab click on Py as shown below.



5. In the empty space give the below code to create a date table.

```
Code: import pandas as pd
start = "1/1/2021"
end = "1/1/2023"
date_table = pd.DataFrame(pd.date_range(start,end,tz='EST',freq='H'),columns=['Date'])
date_table['Year'] = date_table['Date'].dt.year
date_table['Month'] = date_table['Date'].dt.month
date_table['Time'] = date_table['Date'].dt.time
date_table['Quarter'] = date_table['Date'].dt.quarter
date_table['Day of Week'] = date_table['Date'].dt.dayofweek
date_table['DW Abb'] = date_table['Date'].dt.strftime('%a')
date_table['AM_PM'] = date_table['Date'].dt.strftime('%p')
date_table.set_index(['Date'],inplace=True)
dataset = date_table
```



6. You'll have the dataset and the Date_Table in the first column. Click on Table beside the dataset in the first row.

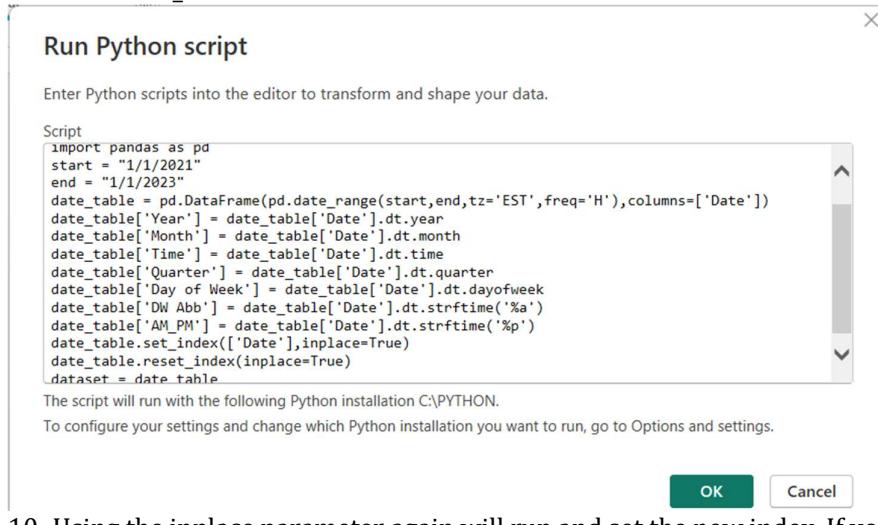
7. If you click the Table value beside dataset, it will show you a table.

8. The next thing to do is reset the index. Click the gear icon beside the Run Python Script step.

9. input date_table.reset_index(inplace=True) function will pull out the Date that was set to index.

Code: import pandas as pd

```
start = "1/1/2021"
end = "1/1/2023"
date_table = pd.DataFrame(pd.date_range(start,end,tz='EST',freq='H'),columns=['Date'])
date_table['Year'] = date_table['Date'].dt.year
date_table['Month'] = date_table['Date'].dt.month
date_table['Time'] = date_table['Date'].dt.time
date_table['Quarter'] = date_table['Date'].dt.quarter
date_table['Day of Week'] = date_table['Date'].dt.dayofweek
date_table['DW Abb'] = date_table['Date'].dt.strftime('%a')
date_table['AM_PM'] = date_table['Date'].dt.strftime('%p')
date_table.set_index(['Date'],inplace=True)
date_table.reset_index(inplace=True)
dataset = date_table
```



10. Using the inplace parameter again will run and set the new index. If you go back to the table, you'll then see the Date column.

	Date	Year	Month	Time	Quarter	Day of Week	DW Abb
1	01-01-2021 00:00:00-05:00	2021	1	00:00:00	1	4	Fri
2	01-01-2021 01:00:00-05:00	2021	1	01:00:00	1	4	Fri
3	01-01-2021 02:00:00-05:00	2021	1	02:00:00	1	4	Fri
4	01-01-2021 03:00:00-05:00	2021	1	03:00:00	1	4	Fri
5	01-01-2021 04:00:00-05:00	2021	1	04:00:00	1	4	Fri
6	01-01-2021 05:00:00-05:00	2021	1	05:00:00	1	4	Fri
7	01-01-2021 06:00:00-05:00	2021	1	06:00:00	1	4	Fri
8	01-01-2021 07:00:00-05:00	2021	1	07:00:00	1	4	Fri
9	01-01-2021 08:00:00-05:00	2021	1	08:00:00	1	4	Fri
10	01-01-2021 09:00:00-05:00	2021	1	09:00:00	1	4	Fri
11	01-01-2021 10:00:00-05:00	2021	1	10:00:00	1	4	Fri
12	01-01-2021 11:00:00-05:00	2021	1	11:00:00	1	4	Fri
13	01-01-2021 12:00:00-05:00	2021	1	12:00:00	1	4	Fri
14	01-01-2021 13:00:00-05:00	2021	1	13:00:00	1	4	Fri

11. Click close & apply to load the data into power bi desktop.

