R.D.&S.H NATIONAL COLLEGE & S.WA. SCIENCE COLLEGE
Bandra, Mumbai - 400050

DEPARTMENT OF COMPUTER SCIENCE

M.Sc. Computer Science – Semester III

Data Visualization

JOURNAL 2024-2025

Seat No. _____

R.D. & S.H. NATIONAL COLLEGE & S. W.A. SCIENCE COLLEGE,

Bandra, Mumbai – 400050.

Department of Computer Science

CERTIFICATE

This is to certify that **Mrs. Gulam Gaus Balasaheb Thakor** of **M.Sc Part II (Sem III)** class has satisfactorily completed **IX** Practicals in the subject of **Data Visualization** as a part of M.Sc. Degree Course in Computer Science during the academic year 2024 – 2025.

Date of Submission:

Faculty Incharge

Co-ordinator,
Department of Computer Science

Signature of External Examiner

Data-Visualization Journal
INDEX

SR NO.	AIM	DATE	PAGE NO.	SIGN
1.	Creating Charts and Reports in Power BI			
2.	Time Intelligence and data analysis Functions with DAX			
3.	Create one-dimensional data using series and perform various operations on it			
4.	Perform Reshaping of the hierarchical data and pivoting data frame data			
5.	Connecting and extracting with various data resources in tableau and Perform calculations and creating parameters in Tableau.			
6.	Designing Tableau Dashboards for different displays and devices			
7.	Create a Trend model using data, Analyse-it and use it for forecasting			
8.	Practical: Creating Geospatial feature maps in Tableau using Geospatial Data.			
9.	Create Dashboard and Storytelling using tableau.			

Practical 1

Practical: Creating Charts and Reports in Power BI

Objective: Learn to create basic charts and reports in Power BI by connecting to a dataset, transforming data, and building simple visuals.

Dataset: Sample Sales Data

Provide students with a sample dataset (like an Excel file or .csv) with the following columns:

- **Date**: Transaction date
- **Product Category**: Category of products (e.g., Electronics, Apparel)
- **Product Name**: Name of the product
- **Region**: Region of sales (e.g., North, South)
- **Sales Amount**: Total sales amount for the transaction
- **Units Sold**: Number of units sold

Step 1: Import Data into Power BI

1. **Open Power BI Desktop**.
2. **Get Data**:
    - Go to **Home** > **Get Data** > **Excel** (or **CSV**).
    - Browse and select the sample dataset file, then click **Load**.
3. **Preview the Data**:
    - Verify that the data is loaded correctly by navigating to the **Data** view in Power BI.

Step 2: Data Cleaning (Optional)

- Check if the data needs cleaning. For example, look for any missing values or errors.
- If required, go to **Transform Data** to remove any unnecessary rows or columns or rename columns if needed.

Step 3: Create Simple Visuals

Chart 1: Sales by Product Category (Column Chart)

1. Go to the **Report** view.
2. In the **Fields** pane, check **Sales Amount** and **Product Category**.
3. Power BI will automatically create a bar chart. Change it to a column chart by selecting the **Clustered Column Chart** icon in the **Visualizations** pane.
4. Customize the chart:
    - Drag **Sales Amount** to the **Y-axis** and **Product Category** to the **X-axis**.
    - In the **Visualizations** pane, adjust formatting options like **Data Labels** for easy viewing of values.

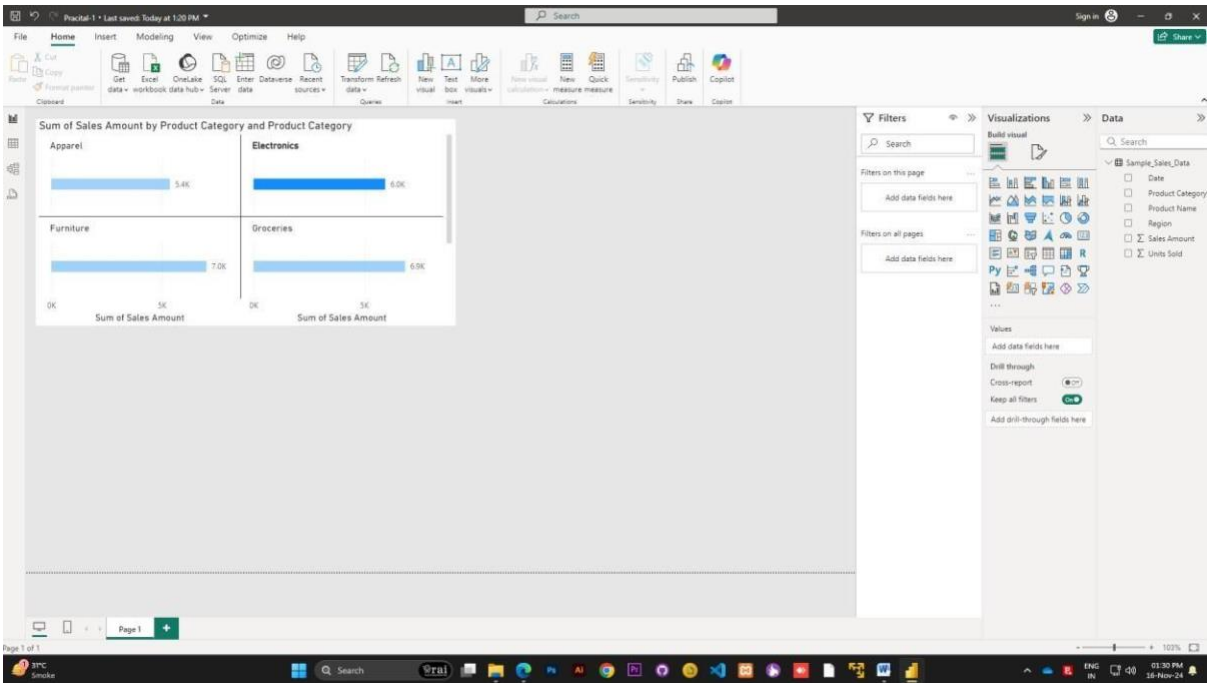


Chart 2: Sales Trend Over Time (Line Chart)

1. In the **Fields** pane, check **Date** and **Sales Amount**.
2. In the **Visualizations** pane, select the **Line Chart** icon.
3. Customize the chart:
    - Drag **Date** to the **X-axis** and **Sales Amount** to the **Y-axis**.
    - Under **Formatting**, enable **Data Labels** and format the chart as needed.

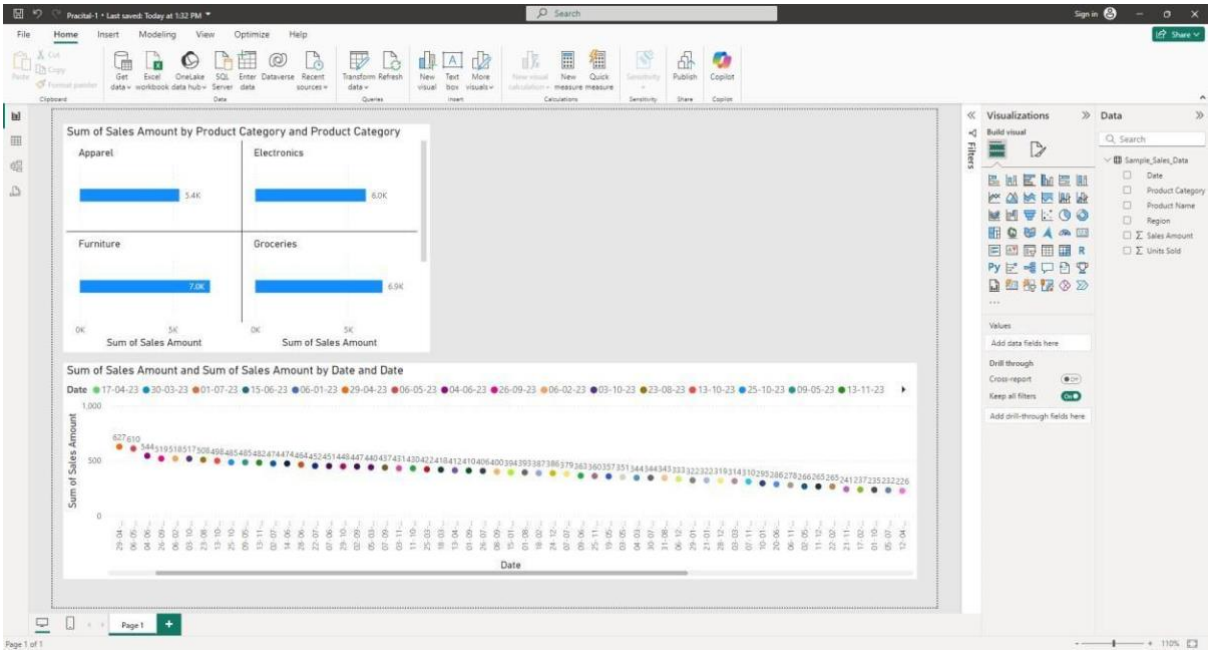
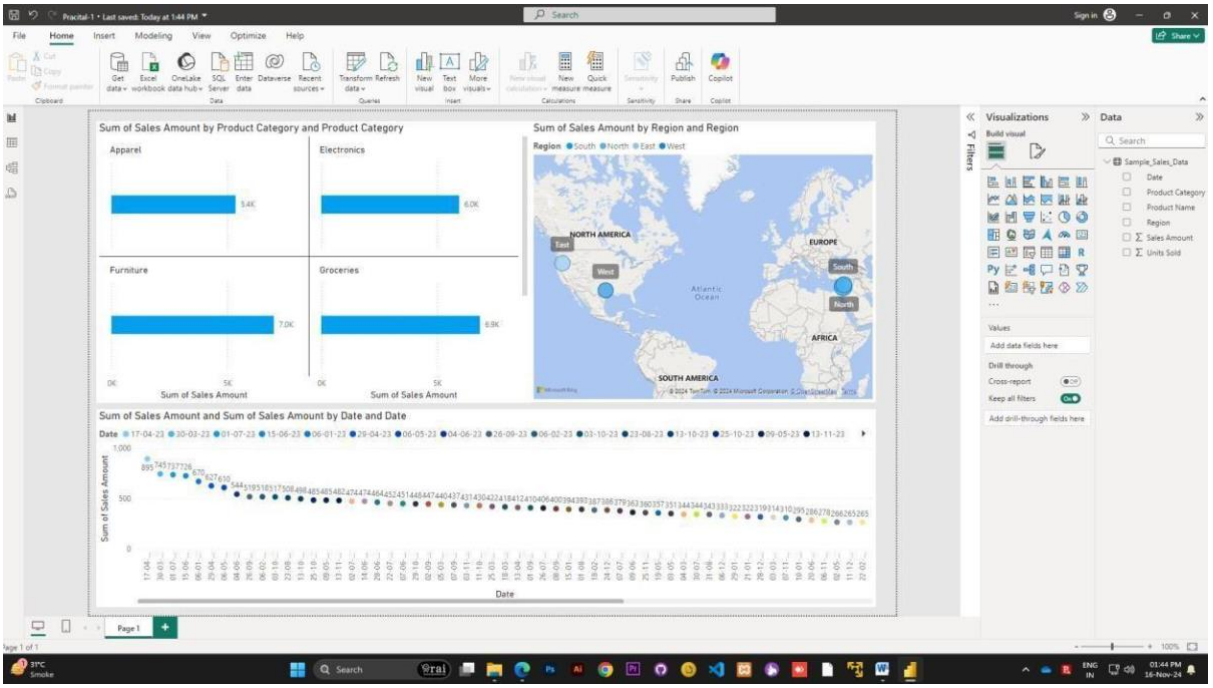


Chart 3: Regional Sales Distribution (Map)

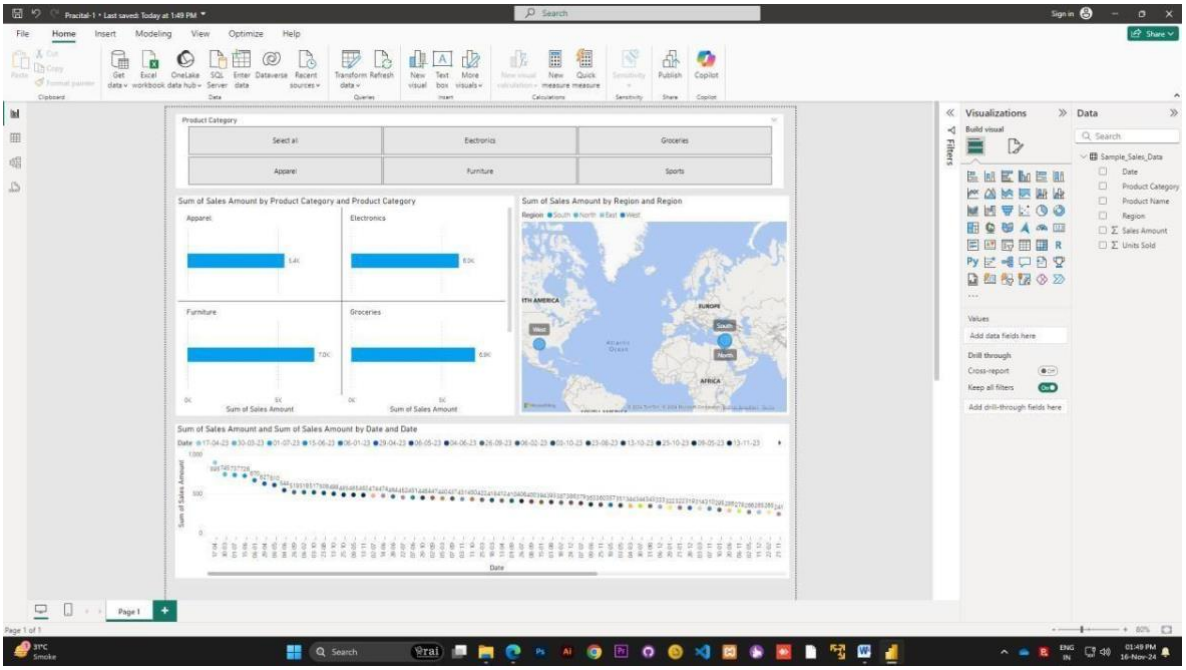
1. Check **Region** and **Sales Amount** in the **Fields** pane.
2. Select the **Map** visualization icon in the **Visualizations** pane.

3. Customize the map:
    - Drag **Region** to the **Location** field and **Sales Amount** to **Size**.
    - Enable **Location Data** in the settings to let Power BI recognize the regions.



Step 4: Create a Slicer (Filter)

1. Select the **Slicer** visualization from the **Visualizations** pane.
2. Drag the **Product Category** field to the **Field** section of the slicer.
3. Place the slicer above the charts. This allows users to filter the report based on the selected product category.



Step 5: Design the Report Layout

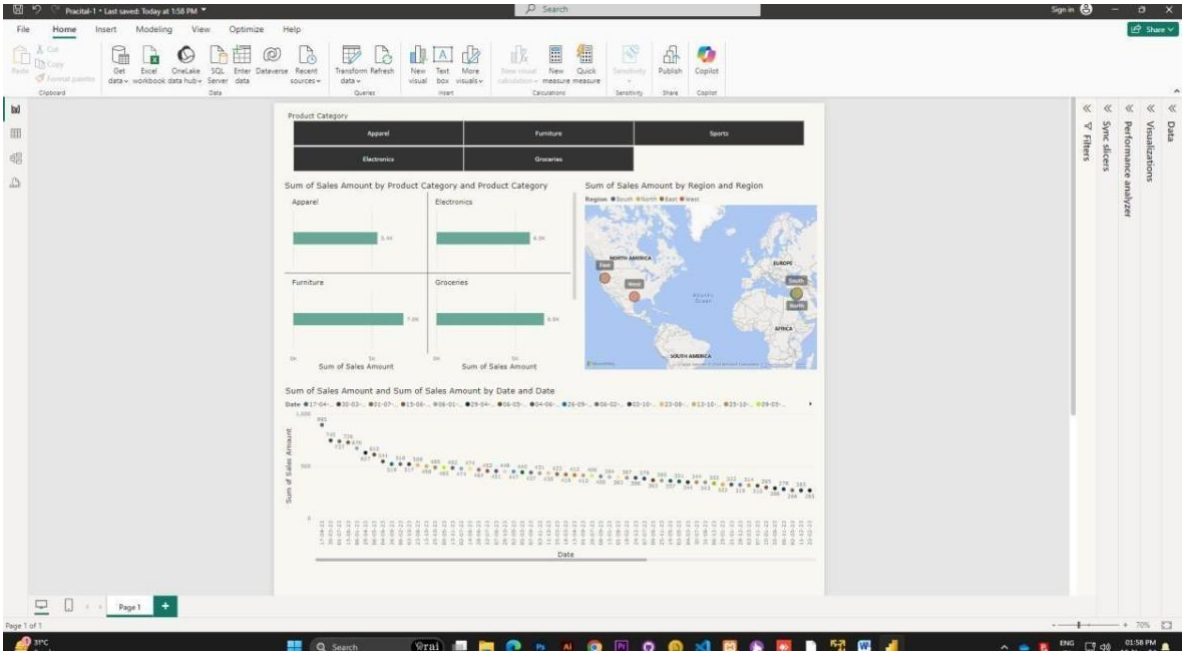
1. Arrange visuals neatly on the report canvas for a clean, organized look.
2. Customize titles, background colors, and font sizes under **Formatting** to match your desired style.

Step 6: Add a Summary Card (Optional)

1. Select the **Card** visualization from the **Visualizations** pane.
2. Drag **Sales Amount** into the card to show total sales.
3. Customize the card with a descriptive title, such as **Total**

Sales. **Step 7: Publish and Share the Report**

1. **Save** your Power BI report locally.
2. Click on **Publish** in the top-right corner to share it to Power BI Service (cloud).
3. Choose your workspace, then click **Select**.
4. In Power BI Service, you can share the report link or embed it in websites, if needed.



Practical 2

Practical: Time Intelligence and data analysis Functions with DAX

Step: Importing Data Set

1. Load Dataset into Power BI

- Open Power BI Desktop.
- Click on Get Data > Excel and load your dataset.

Step 2: Creating a Date Table

Time Intelligence functions require a Date table. If not already available, create a Date table in Power BI:

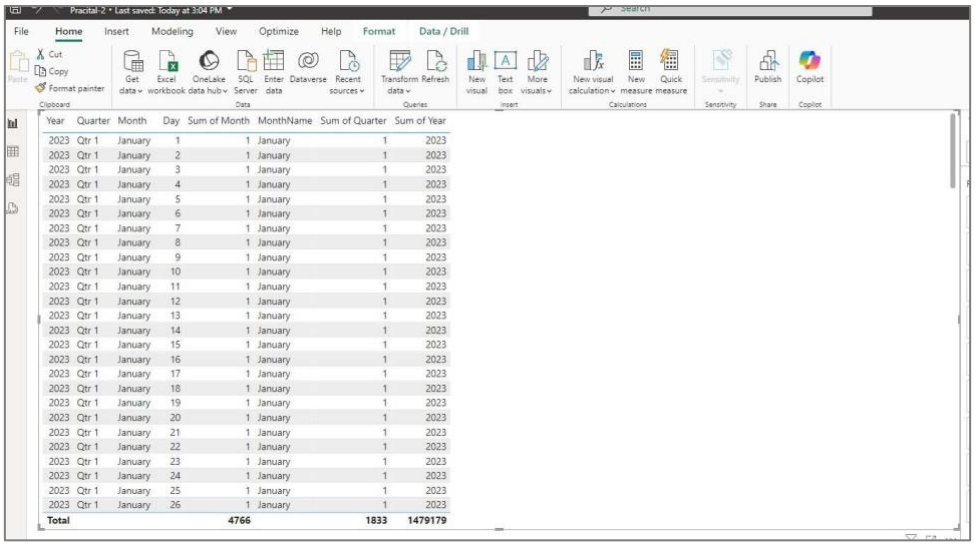
1. Go to **Modeling** > **New Table**.
2. Enter the DAX formula below to create a Date table with date ranges.

DAX

```
DateTable =
  CALENDAR(DATE(2023, 1, 1), DATE(2024, 12, 31))
```

3. Add calculated columns for **Year**, **Month**, **Quarter**, and **Month Name** for grouping data by these time periods:

```
DateTable =
  ADDCOLUMNS
  (
    CALENDAR(DATE(2023, 1, 1), DATE(2024, 12,
    31)), "Year", YEAR([Date]),
    "Month", MONTH([Date]),
    "Quarter", QUARTER([Date]),
    "MonthName", FORMAT([Date], "MMMM")
  )
```



Year	Quarter	Month	Day	Sum of Month	MonthName	Sum of Quarter	Sum of Year
2023	Qtr 1	January	1	1	January	1	2023
2023	Qtr 1	January	2	1	January	1	2023
2023	Qtr 1	January	3	1	January	1	2023
2023	Qtr 1	January	4	1	January	1	2023
2023	Qtr 1	January	5	1	January	1	2023
2023	Qtr 1	January	6	1	January	1	2023
2023	Qtr 1	January	7	1	January	1	2023
2023	Qtr 1	January	8	1	January	1	2023
2023	Qtr 1	January	9	1	January	1	2023
2023	Qtr 1	January	10	1	January	1	2023
2023	Qtr 1	January	11	1	January	1	2023
2023	Qtr 1	January	12	1	January	1	2023
2023	Qtr 1	January	13	1	January	1	2023
2023	Qtr 1	January	14	1	January	1	2023
2023	Qtr 1	January	15	1	January	1	2023
2023	Qtr 1	January	16	1	January	1	2023
2023	Qtr 1	January	17	1	January	1	2023
2023	Qtr 1	January	18	1	January	1	2023
2023	Qtr 1	January	19	1	January	1	2023
2023	Qtr 1	January	20	1	January	1	2023
2023	Qtr 1	January	21	1	January	1	2023
2023	Qtr 1	January	22	1	January	1	2023
2023	Qtr 1	January	23	1	January	1	2023
2023	Qtr 1	January	24	1	January	1	2023
2023	Qtr 1	January	25	1	January	1	2023
2023	Qtr 1	January	26	1	January	1	2023
Total				**4766**		**1833**	**1479179**

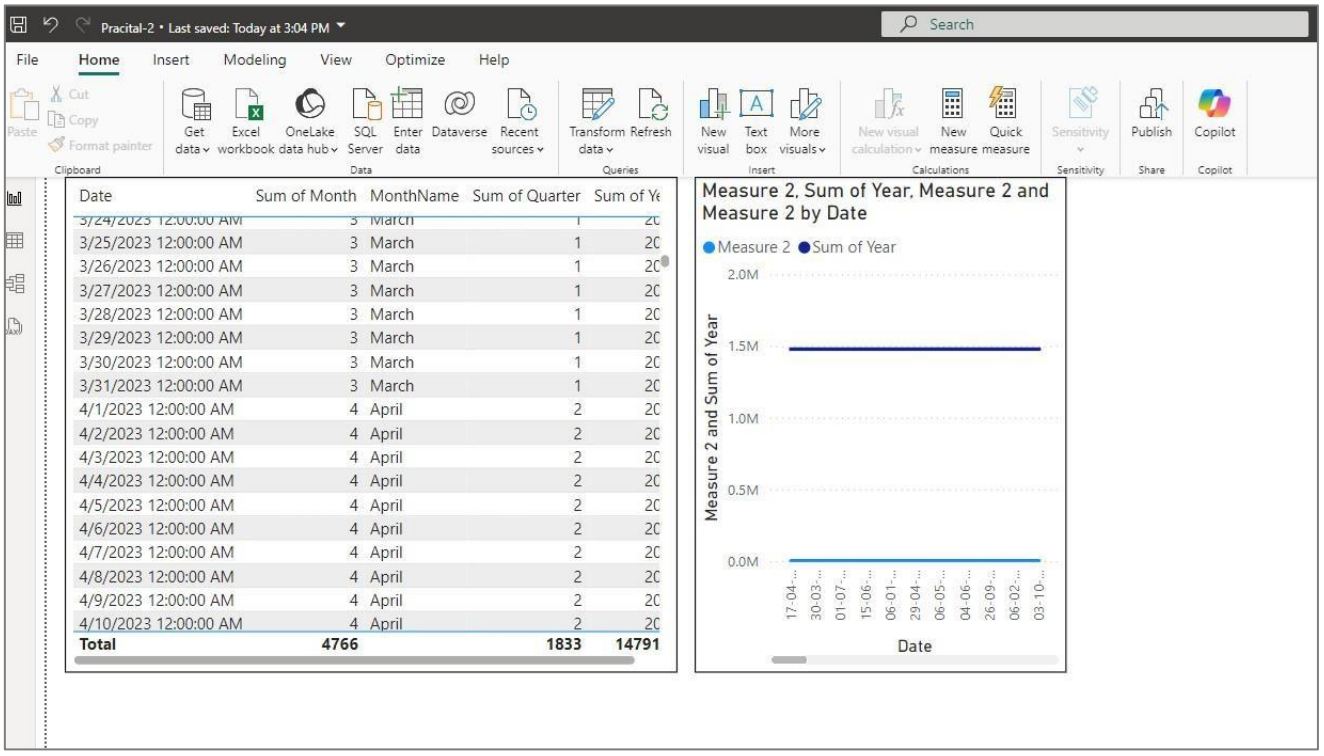
4. Mark this table as the Date Table by selecting it in **Modeling** > **Mark as Date Table**. **Step 3: Calculating Year-To-Date Sales**

1. Go to the **Modeling** tab > **New Measure**.
2. Enter the following DAX formula to calculate Year-To-Date (YTD) sales: DAX

Measure 2 =

TOTALYTD(SUM(Sample_Sales_Data[Sales Amount]), DateTable[Date])

3. Use **Sales YTD** in a visual, such as a line chart or table, to see cumulative sales over the year.



Date	Sum of Month	MonthName	Sum of Quarter	Sum of Ye
3/24/2023 12:00:00 AM	3	March	1	20
3/25/2023 12:00:00 AM	3	March	1	20
3/26/2023 12:00:00 AM	3	March	1	20
3/27/2023 12:00:00 AM	3	March	1	20
3/28/2023 12:00:00 AM	3	March	1	20
3/29/2023 12:00:00 AM	3	March	1	20
3/30/2023 12:00:00 AM	3	March	1	20
3/31/2023 12:00:00 AM	3	March	1	20
4/1/2023 12:00:00 AM	4	April	2	20
4/2/2023 12:00:00 AM	4	April	2	20
4/3/2023 12:00:00 AM	4	April	2	20
4/4/2023 12:00:00 AM	4	April	2	20
4/5/2023 12:00:00 AM	4	April	2	20
4/6/2023 12:00:00 AM	4	April	2	20
4/7/2023 12:00:00 AM	4	April	2	20
4/8/2023 12:00:00 AM	4	April	2	20
4/9/2023 12:00:00 AM	4	April	2	20
4/10/2023 12:00:00 AM	4	April	2	20
Total	**4766**		**1833**	**14791**

Step 4: Calculating Month-To-Date and Quarter-To-Date Sales

1. Create two more measures to calculate MTD and QTD sales. DAX

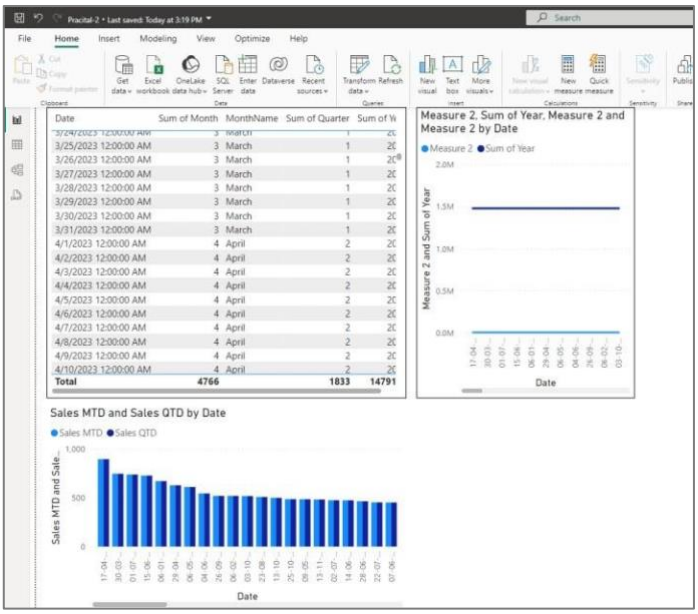
Sales MTD =

TOTALMTD(SUM(Sample_Sales_Data[Sales Amount]), DateTable[Date])

Sales QTD =

TOTALQTD(SUM(Sample_Sales_Data[Sales Amount]), DateTable[Date])

2. Use these measures to see how sales accumulate over months and quarters.



Step 5: Year-Over-Year (YoY) Comparison

1. Create a measure to calculate the prior year's sales for comparison. DAX

Sales Previous Year =

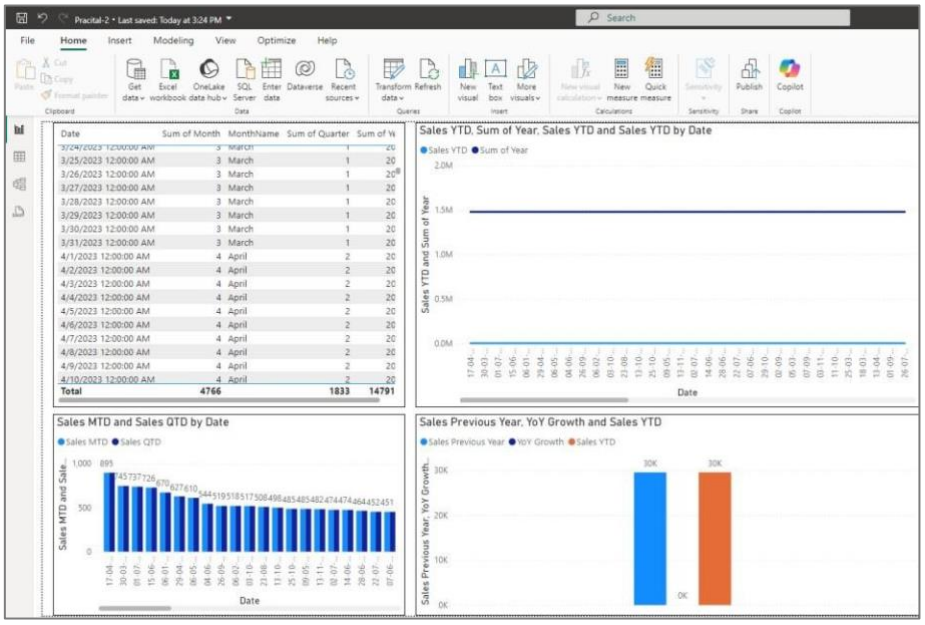
CALCULATE(SUM(Sample_Sales_Data[Sales Amount]), SAMEPERIODLASTYEAR(DateTable[Date]))

2. Create a Year-over-Year growth measure. DAX

YoY Growth =

DIVIDE([Sales YTD] - [Sales Previous Year], [Sales Previous Year], 0)

3. Add YoY Growth to a visual alongside Sales YTD and Sales Previous Year to show growth percentages.



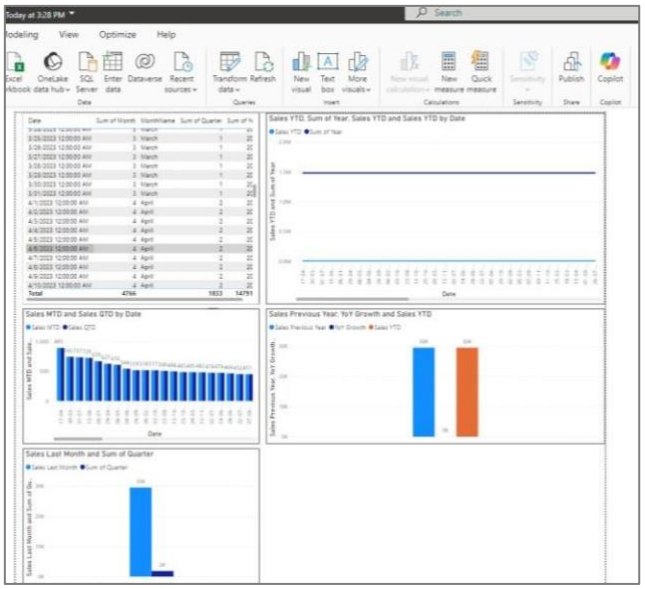
Step 6: Using DATEADD for Custom Time Shifts

1. The DATEADD function is flexible for shifting dates by any period (days, months, quarters, years). DAX

Sales Last Month =

CALCULATE(SUM(Sales[SalesAmount]), DATEADD(DateTable[Date], -1, MONTH))

2. This formula shifts the date back one month to calculate sales for the previous month.



Step 7: ParallelPeriod for Period Offset Comparisons

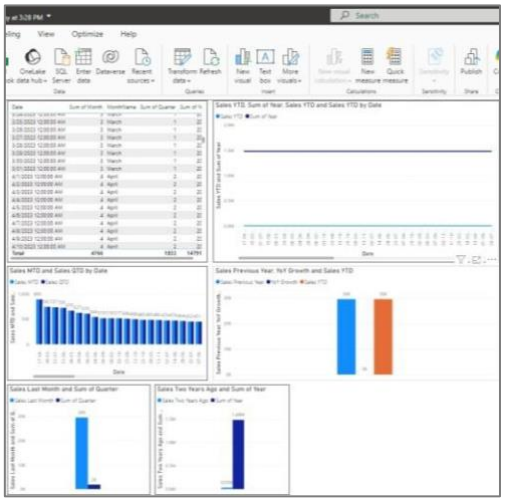
1. PARALLELPERIOD allows offsets similar to DATEADD, often used to compare the same period across different years.

DAX

Sales Two Years Ago =

CALCULATE(SUM(Sales[SalesAmount]), PARALLELPERIOD(DateTable[Date], -2, YEAR))

2. This formula shifts the date back by two years to calculate sales from two years ago.



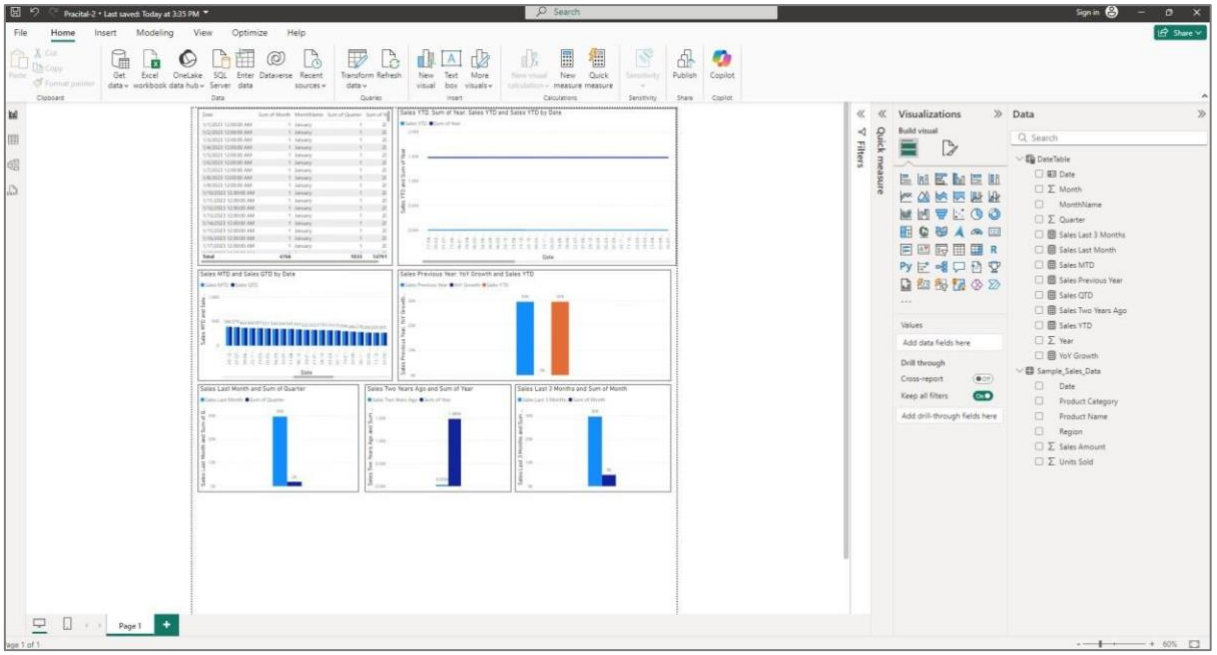
Step 8: Custom Period Aggregations with CALCULATE and FILTER

Sometimes, custom calculations don't fit predefined DAX functions. Use CALCULATE and FILTER to create custom aggregations.

1. **Rolling 3-Month Sales**: Calculate the last three months' sales

dynamically. DAX

```
Sales Last 3 Months =
  CALCULATE(SUM(Sales[SalesAmount])
  ,
    DATESINPERIOD(DateTable[Date],
    LASTDATE(DateTable[Date]), -3, MONTH))
```



Step 9: Adding These Measures to a Power BI Report

1. Visualize each measure in Power BI:
    - **Sales YTD, MTD, QTD**: Use line charts for trend analysis.
    - **Sales Previous Year and YoY Growth**: Use bar or line and bar combo charts.
    - **Rolling 3-Month Sales**: A line chart or KPI card for insights on recent sales trends.
2. Encourage students to customize visuals by adding slicers for Year, Quarter, and Month.

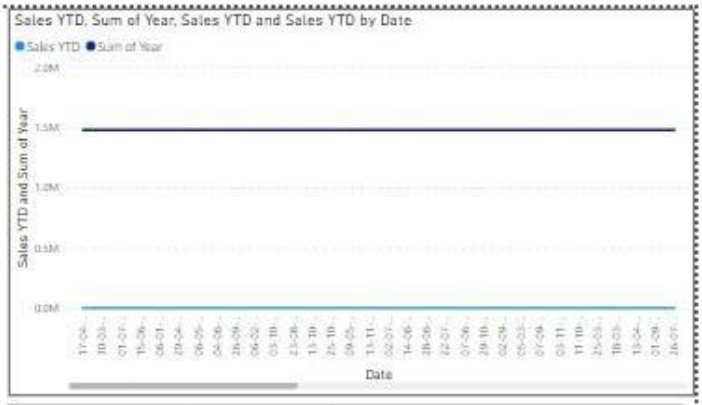
Step 10: Recap and Analysis

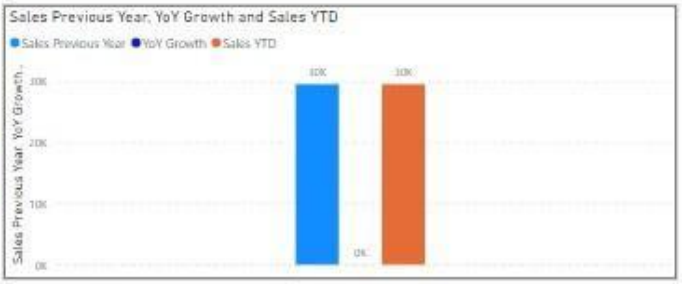
Encourage students to use their DAX measures to analyze:

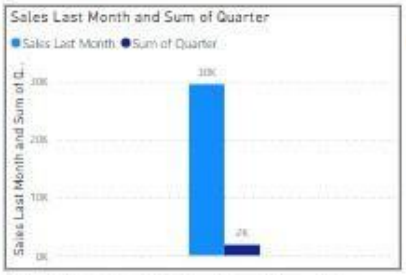
- Seasonal trends, such as quarterly and monthly performance.
- Long-term growth by examining YoY Growth.

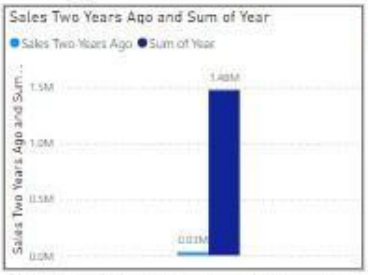
- Recent changes in performance with rolling averages.

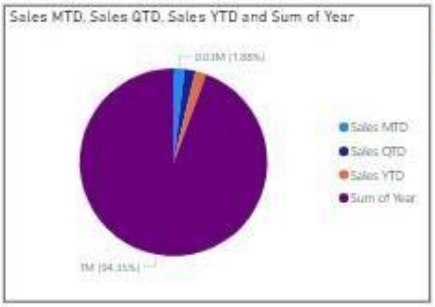
Date	Sum of Month	MonthName	Sum of Quarter	Sum of Y
1/1/2023 12:00:00 AM	1	January	1	20
1/2/2023 12:00:00 AM	1	January	1	20
1/3/2023 12:00:00 AM	1	January	1	20
1/4/2023 12:00:00 AM	1	January	1	20
1/5/2023 12:00:00 AM	1	January	1	20
1/6/2023 12:00:00 AM	1	January	1	20
1/7/2023 12:00:00 AM	1	January	1	20
1/8/2023 12:00:00 AM	1	January	1	20
1/9/2023 12:00:00 AM	1	January	1	20
1/10/2023 12:00:00 AM	1	January	1	20
1/11/2023 12:00:00 AM	1	January	1	20
1/12/2023 12:00:00 AM	1	January	1	20
1/13/2023 12:00:00 AM	1	January	1	20
1/14/2023 12:00:00 AM	1	January	1	20
1/15/2023 12:00:00 AM	1	January	1	20
1/16/2023 12:00:00 AM	1	January	1	20
1/17/2023 12:00:00 AM	1	January	1	20
Total	4766		1833	14791











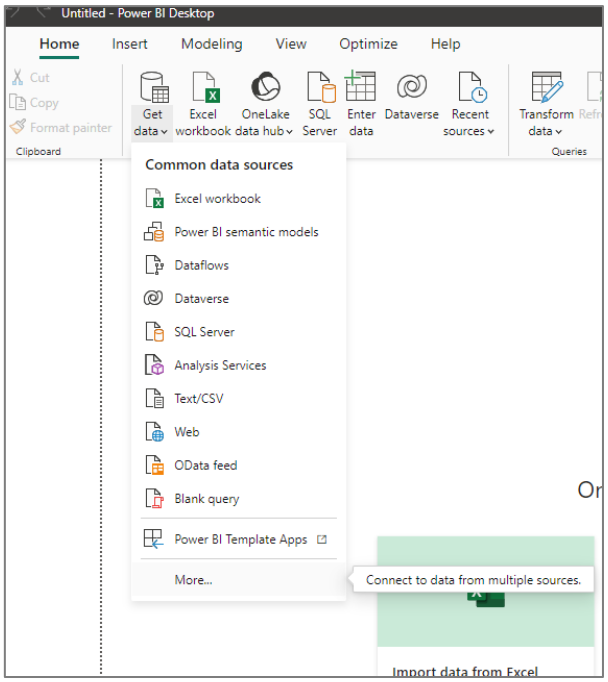
Quarter

1	3
2	4

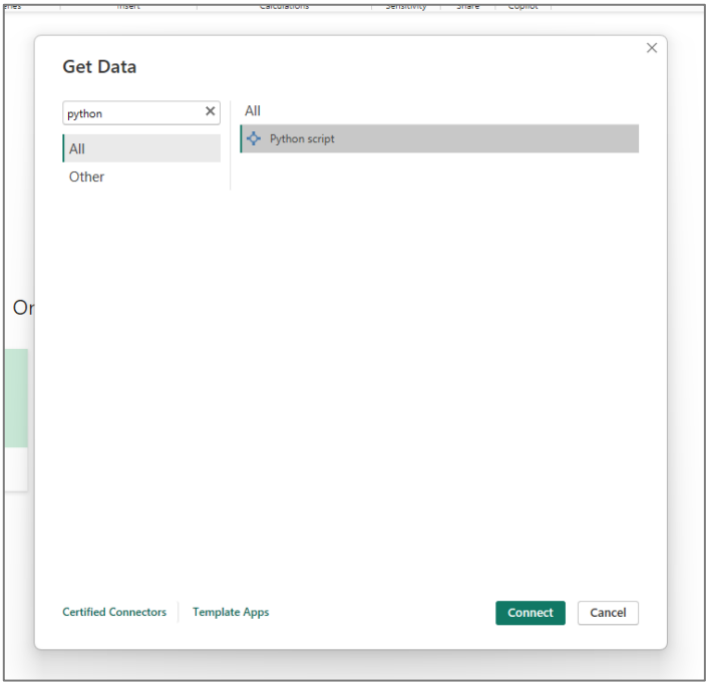
Practical 3

Practical: Create one-dimensional data using series and perform various operations on it

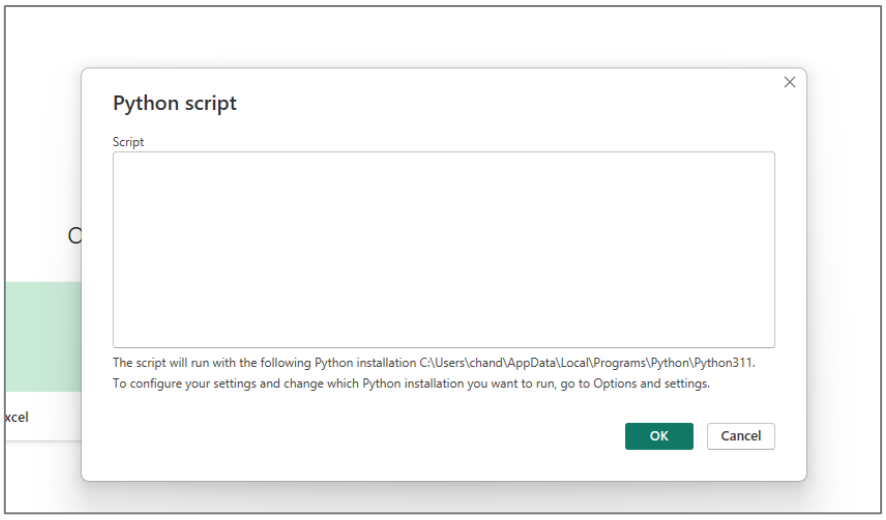
Step 1: Open Power BI go to Get data and click more



Step 2: Search python and click connect



Step 3: Python script dialogue box will open type your python code here (Make sure you have python installed in your system)

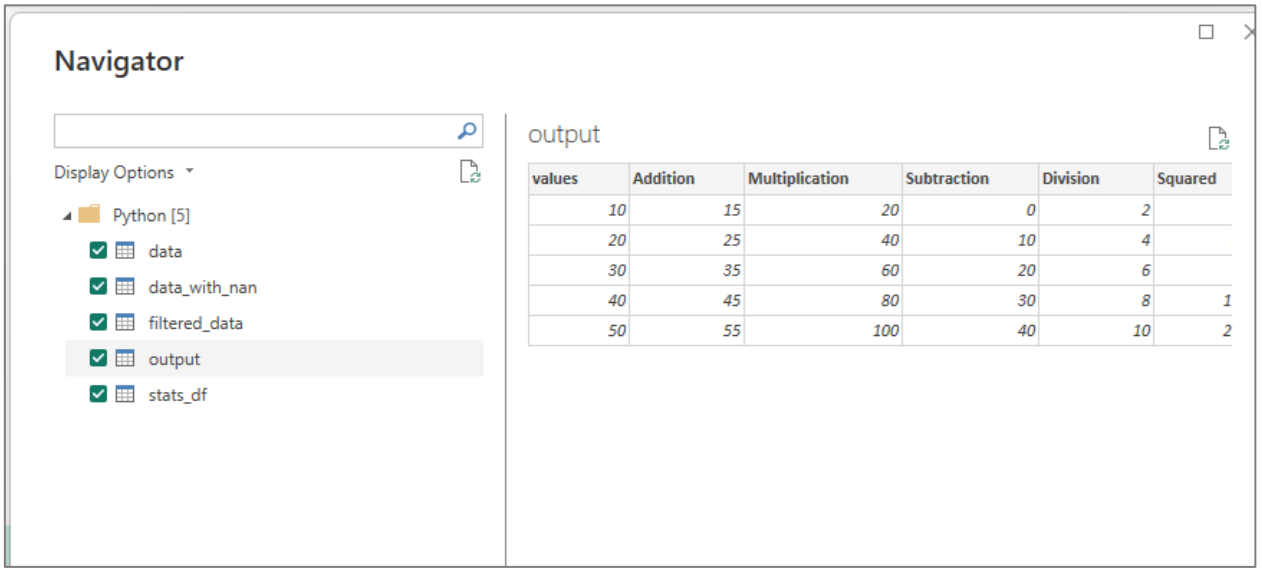


Step 4 code:

```
import pandas as pd

import numpy as np

data = pd.DataFrame({'values': [10, 20, 30, 40, 50]})

data['Addition'] = data['values'] + 5

data['Multiplication'] = data['values'] * 2
```

```
data['Subtraction'] = data['values'] - 10
data['Division'] = data['values'] / 5
def square(x):
    return x * x
data['Squared'] =
data['values'].apply(square) statistics = {
    'Sum': [data['values'].sum()],
    'Mean': [data['values'].mean()],
    'Max': [data['values'].max()],
    'Min': [data['values'].min()]
}
stats_df = pd.DataFrame(statistics)
filtered_data = data[data['values'] > 25]
data_with_nan = pd.DataFrame({'values': [10, 20, np.nan, 40, 50]})
data_with_nan['Is_NaN'] = data_with_nan['values'].isna()
data_with_nan['Filled'] = data_with_nan['values'].fillna(0)
data['Sorted_Ascending'] =
data['values'].sort_values().reset_index(drop=True)
data['Sorted_Descending'] =
data['values'].sort_values(ascending=False).reset_index(drop=True) output = pd.concat([data,
stats_df], axis=1)
```

output:

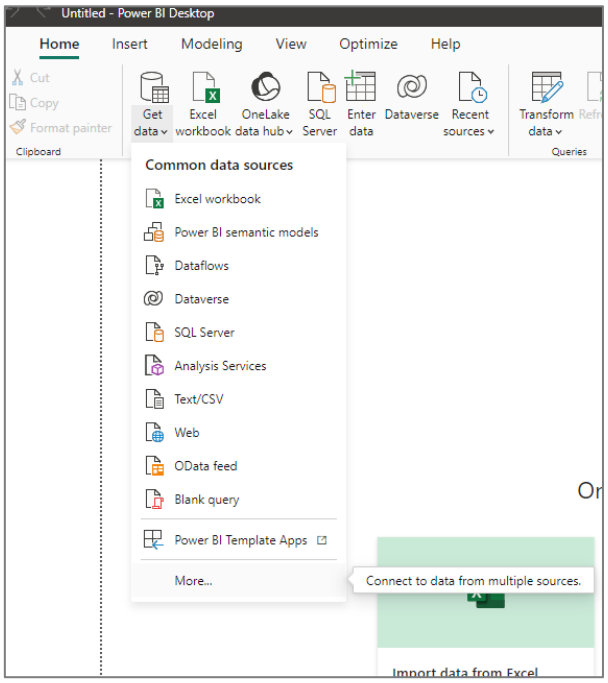


values	Addition	Multiplication	Subtraction	Division	Squared
10	15	20	0	2	
20	25	40	10	4	
30	35	60	20	6	
40	45	80	30	8	1
50	55	100	40	10	2

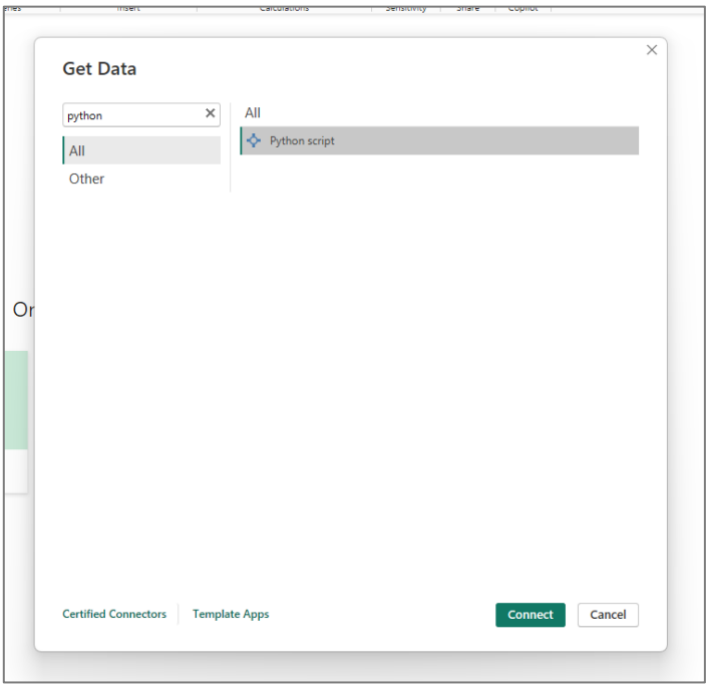
Practical 4

Practical: Perform Reshaping of the hierarchical data and pivoting data frame data

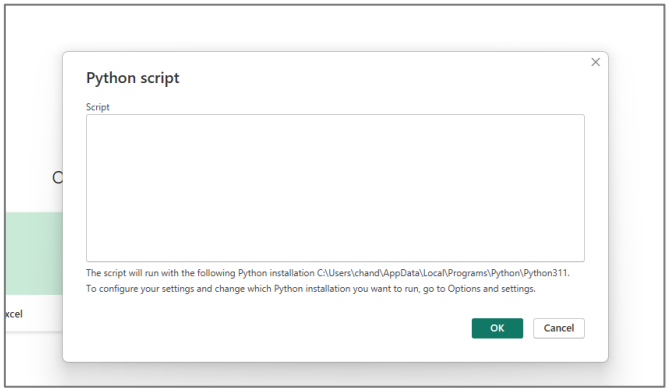
Step 1: Open Power BI go to Get data and click more



Step 2: Search python and click connect



Step 3: Python script dialogue box will open type your python code here (Make sure you have python installed in your system)



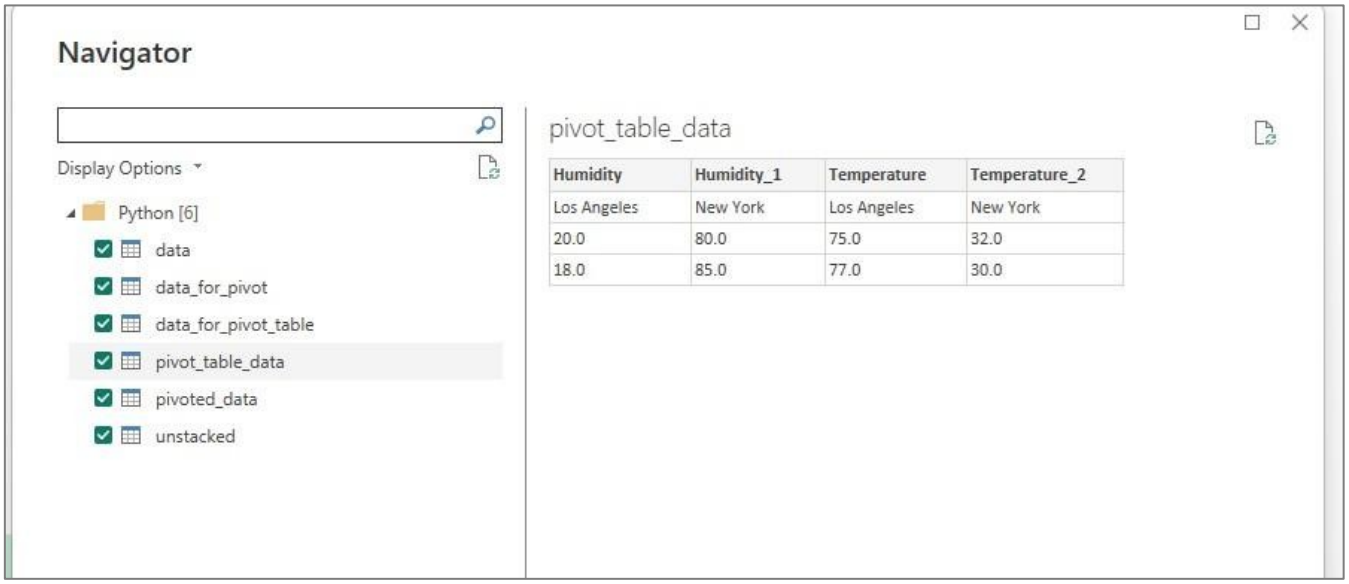
Step 4 code:

```
import pandas as pd
import numpy as np

# Create a MultiIndex DataFrame (Hierarchical Data)
arrays = [
    ['A', 'A', 'A', 'B', 'B', 'B'],
    ['X', 'Y', 'Z', 'X', 'Y', 'Z']
]
index = pd.MultiIndex.from_arrays(arrays, names=('Letter', 'Symbol'))
data = pd.DataFrame({
    'Value1': [10, 20, 30, 40, 50, 60],
```

```
    'Value2': [15, 25, 35, 45, 55, 65]
}, index=index)

print("Original DataFrame (Hierarchical
Data):") print(data)
print("\n")

#---------------- Stack Operation ----------------

# Stack the DataFrame (Convert Columns to Rows)
stacked = data.stack()
print("Stacked DataFrame:")
print(stacked)
print("\n")

#---------------- Unstack Operation ----------------

# Unstack the DataFrame (Convert Rows to Columns)
unstacked = stacked.unstack()
print("Unstacked DataFrame:")
print(unstacked)
print("\n")

#---------------- Pivoting Data ----------------

# Create DataFrame for Pivoting
data_for_pivot = pd.DataFrame({
    'Date': ['2024-01-01', '2024-01-01', '2024-01-02', '2024-01-02'],
    'City': ['New York', 'Los Angeles', 'New York', 'Los
    Angeles'], 'Temperature': [32, 75, 30, 77]
})

print("Original Data for Pivoting:")
print(data_for_pivot)
print("\n")

# Pivot the DataFrame to make cities as columns and dates as rows
pivoted_data = data_for_pivot.pivot(index='Date', columns='City',
values='Temperature') print("Pivoted DataFrame (Cities as Columns):")
print(pivoted_data
) print("\n")

#---------------- Pivot Table Data----------------

# Create DataFrame for Pivot Table
data_for_pivot_table =
pd.DataFrame({
    'Date': ['2024-01-01', '2024-01-01', '2024-01-02', '2024-01-02'],
    'City': ['New York', 'Los Angeles', 'New York', 'Los
    Angeles'], 'Temperature': [32, 75, 30, 77],
    'Humidity': [80, 20, 85, 18]
})
```

```
print("Original Data for Pivot
Table:") print(data_for_pivot_table)
print("\n")

# Pivot Table to calculate the average temperature and humidity per city and date
pivot_table_data = data_for_pivot_table.pivot_table(
    index='Date',
    columns='City'
    ,
    values=['Temperature', 'Humidity'],
    aggfunc=np.mean
)

print("Pivot Table DataFrame (Average Temperature and
Humidity):") print(pivot_table_data)
```

Output:



Humidity	Humidity_1	Temperature	Temperature_2
Los Angeles	New York	Los Angeles	New York
20.0	80.0	75.0	32.0
18.0	85.0	77.0	30.0

Practical 5

Practical: Connecting and extracting with various data resources in tableau and Perform calculations and creating parameters in Tableau.

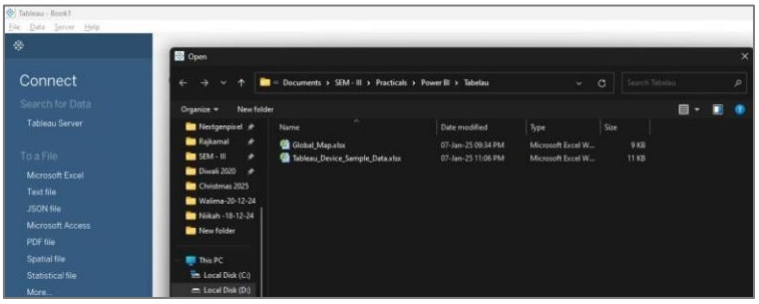
1. Connecting to Data

Sources Steps:

- Open Tableau:
- On the "Connect" pane, select your data source (e.g., Excel, CSV, SQL Server, etc.).

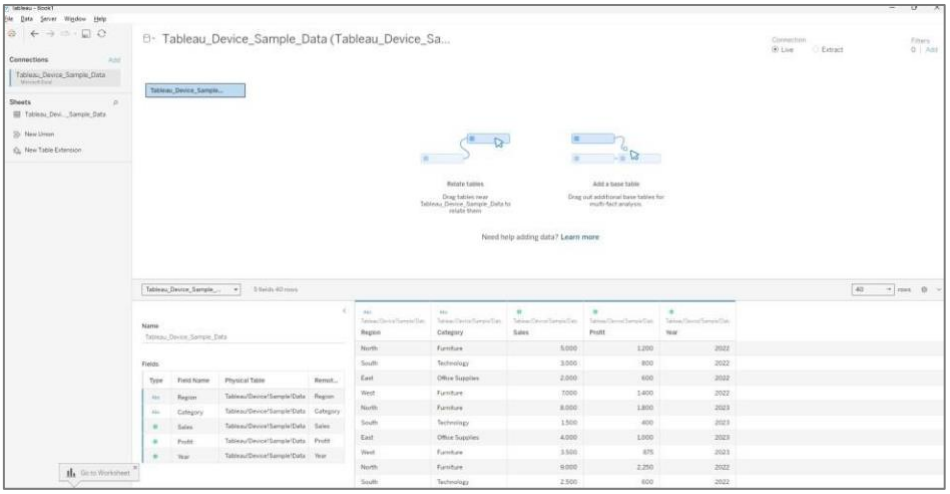
2. Connect to a File:

- Choose Microsoft Excel for an example.
- Browse to the desired file and click Open.



3. Preview the Data:

- The data preview will appear. Drag the sheet to the canvas.



4. Extract Data:
   - In the top-right corner of the data source screen, choose Extract.

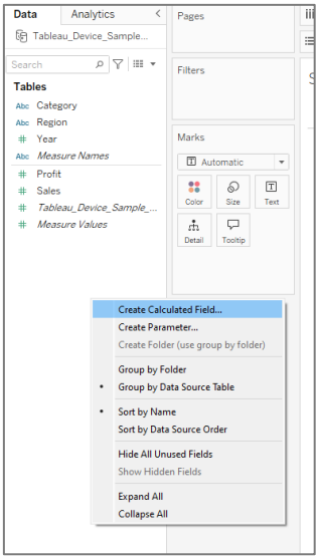
   - Click on Sheet 1 to proceed.

2. Creating Calculations

Calculations in Tableau are used for custom computations or field transformations.

Example 1: Calculated Field

1. Go to the Data Pane (left sidebar) and right-click.



2. Select Create Calculated Field.
3. Give it a name (e.g., "Sales Growth").
4. Enter the formula:

(SUM([Current Year Sales]) - SUM([Previous Year Sales])) / SUM([Previous Year Sales])

(SUM([Sales]) - LOOKUP(SUM([Sales]), -1)) / LOOKUP(SUM([Sales]), -1)

5. Click OK.

Example 2: Conditional Calculation

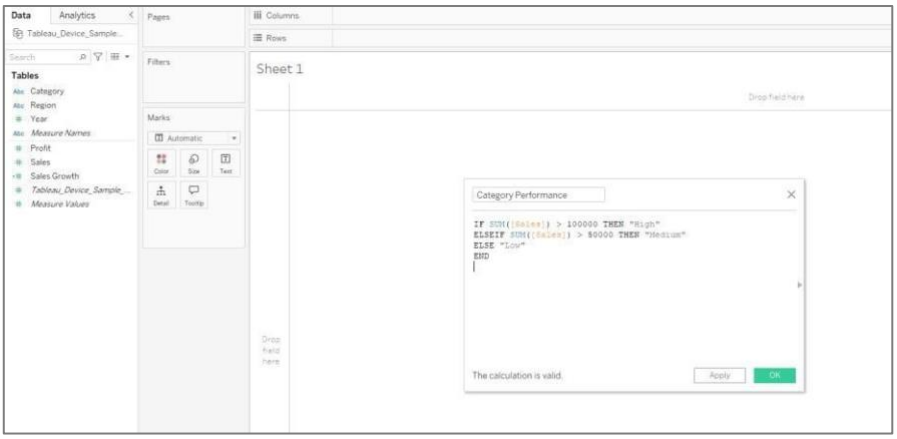
1. Create another calculated field, name it "Category Performance".
2. Use the following formula

IF SUM([Sales]) > 100000 THEN "High"

ELSEIF SUM([Sales]) > 50000 THEN

"Medium" ELSE "Low"

END

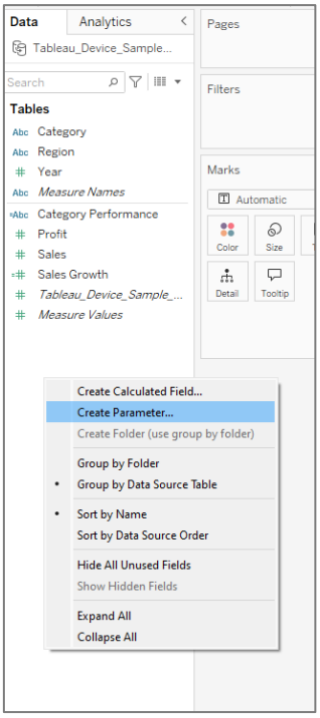


3. Creating Parameters

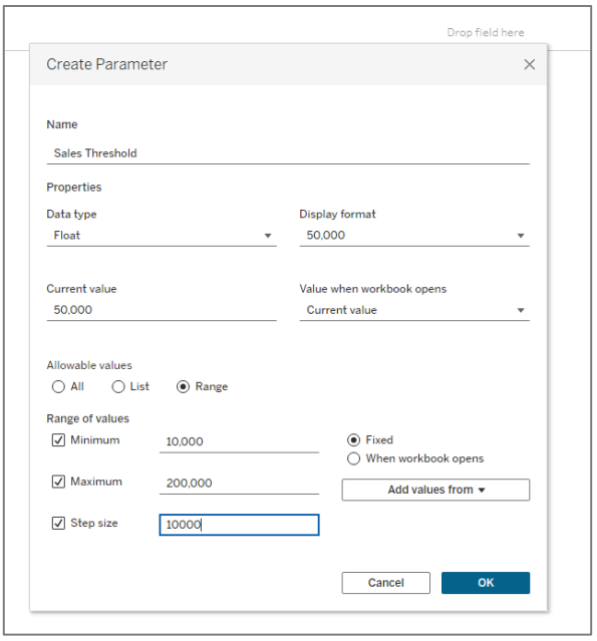
Parameters let users dynamically control values.

Example: Dynamic Sales Filter

1. Go to the **Data Pane** and right-click.
2. Choose **Create Parameter**.



3. Name the parameter (e.g., "Sales Threshold").
4. Set the following:
   - **Data Type:** Float
   - **Current Value:** 50000
   - **Allowable Values:** Range
   - **Minimum:** 10000
   - **Maximum:** 200000
   - **Step Size:** 10000

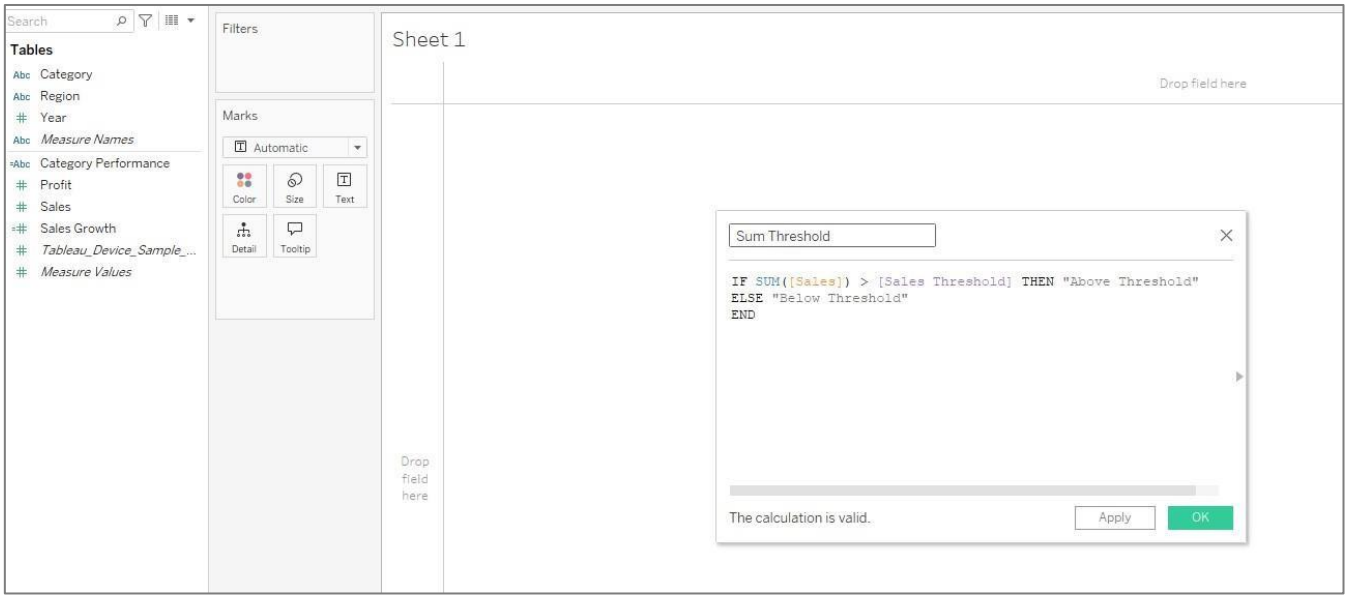


5. Click **OK**.
6. Create a calculated field using this parameter:

IF SUM([Sales]) > [Sales Threshold] THEN "Above Threshold"

ELSE "Below Threshold"

END



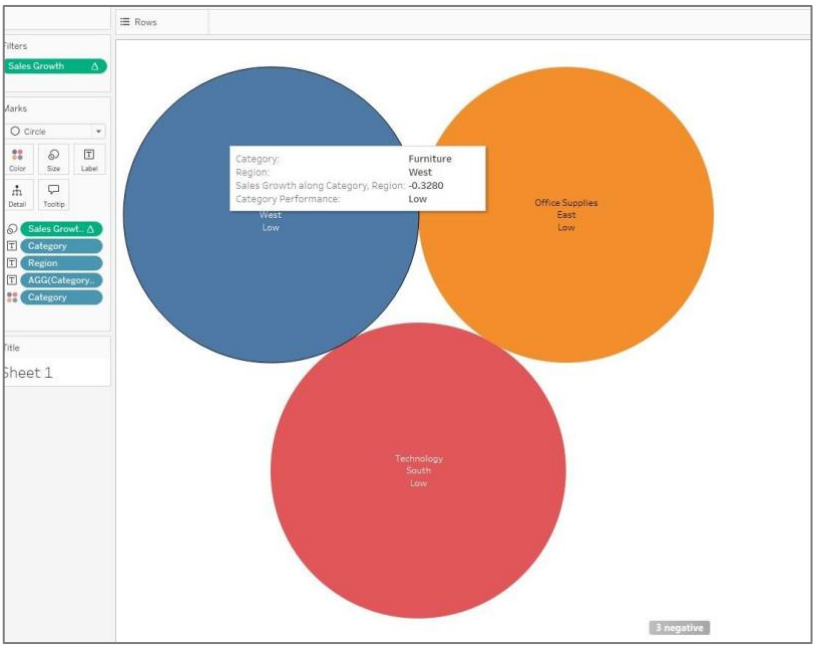
Add the parameter to your dashboard and observe the interactivity.

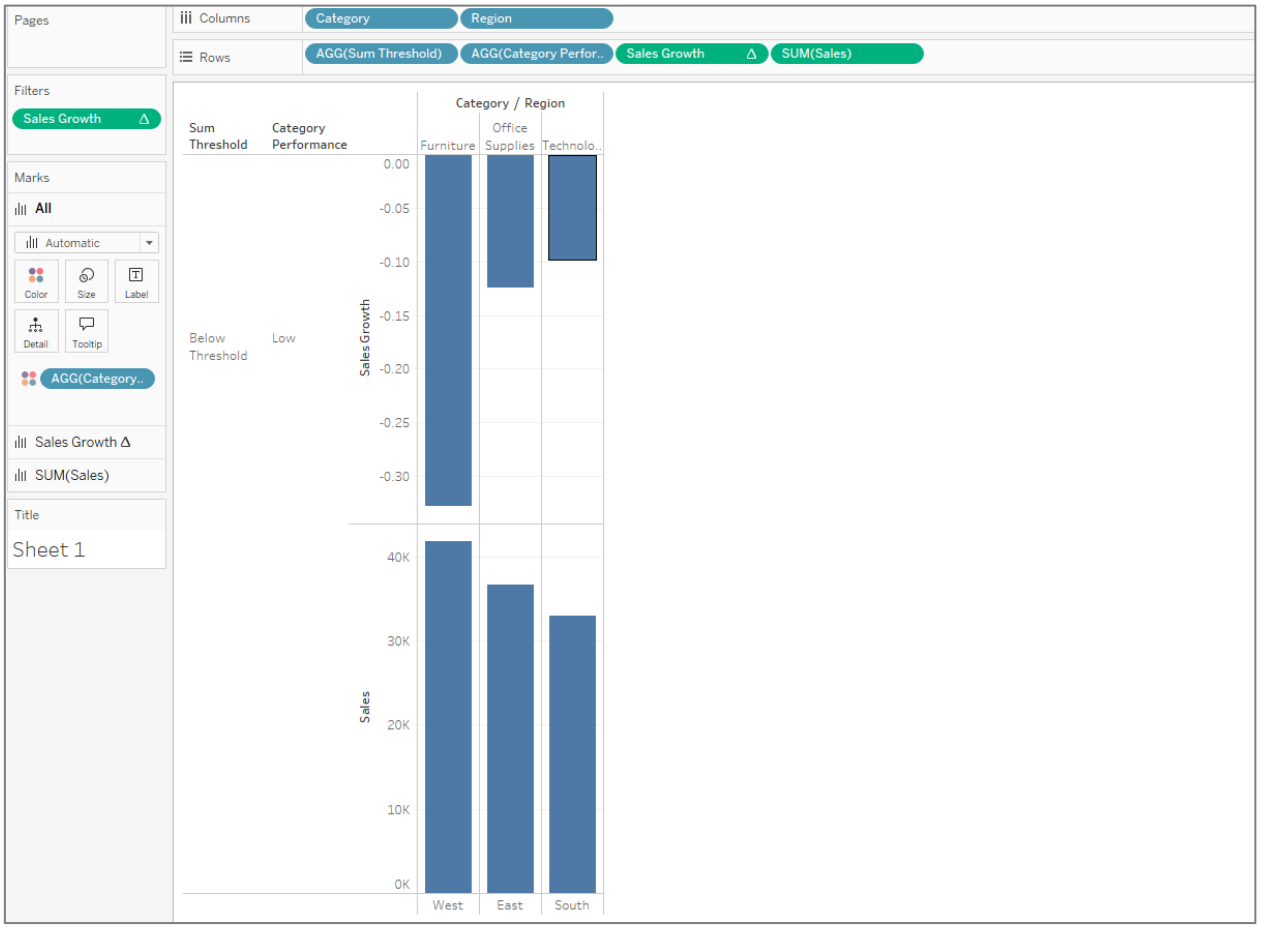
4. Visualizing Data

1. Drag dimensions and measures to Rows and Columns for visualizations.
2. Use filters, marks, and colors to enhance the

chart. Custom fields

1. Sales Growth
2. Category Performance
3. Sales Threshold

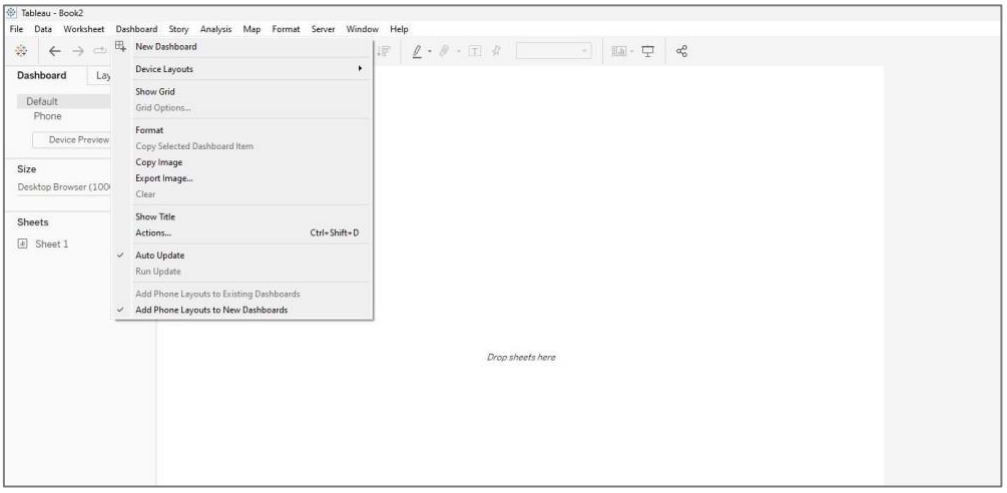




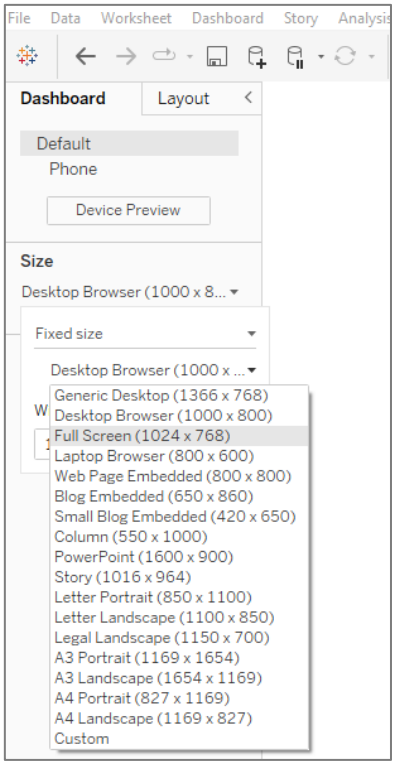
Practical 6

Practical: Designing Tableau Dashboards for different displays and devices

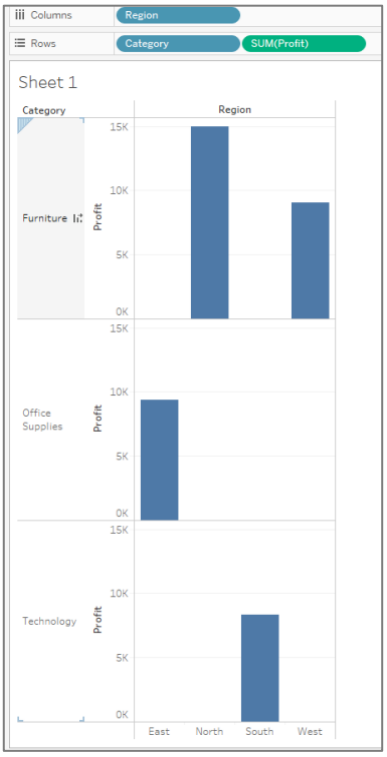
1. Dashboard Creation Basics
    1. Open Tableau Desktop and create a new workbook.
    2. Click Dashboard > New Dashboard.

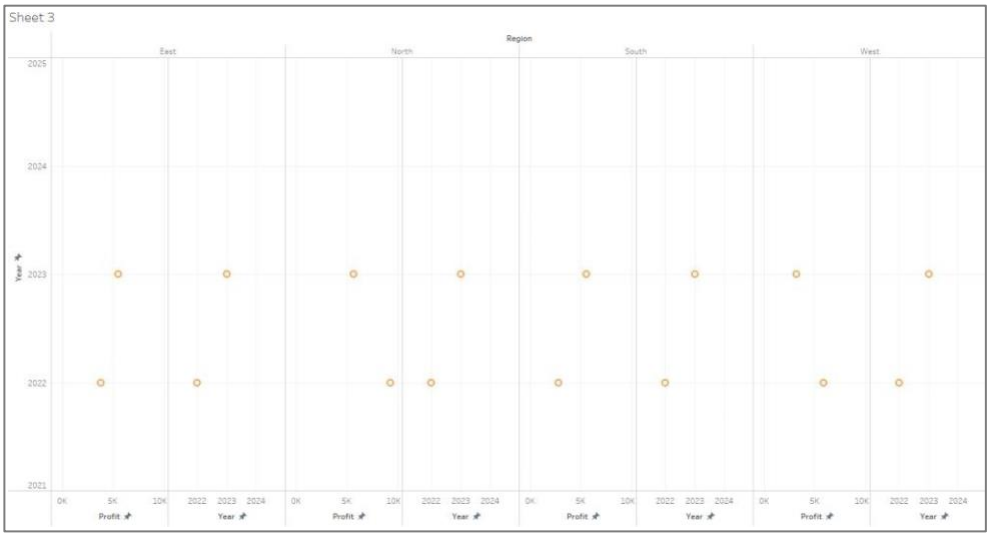


    3. In the dashboard pane:
        - Set the Size dropdown to Automatic or a specific size like Desktop (1024 x 768).



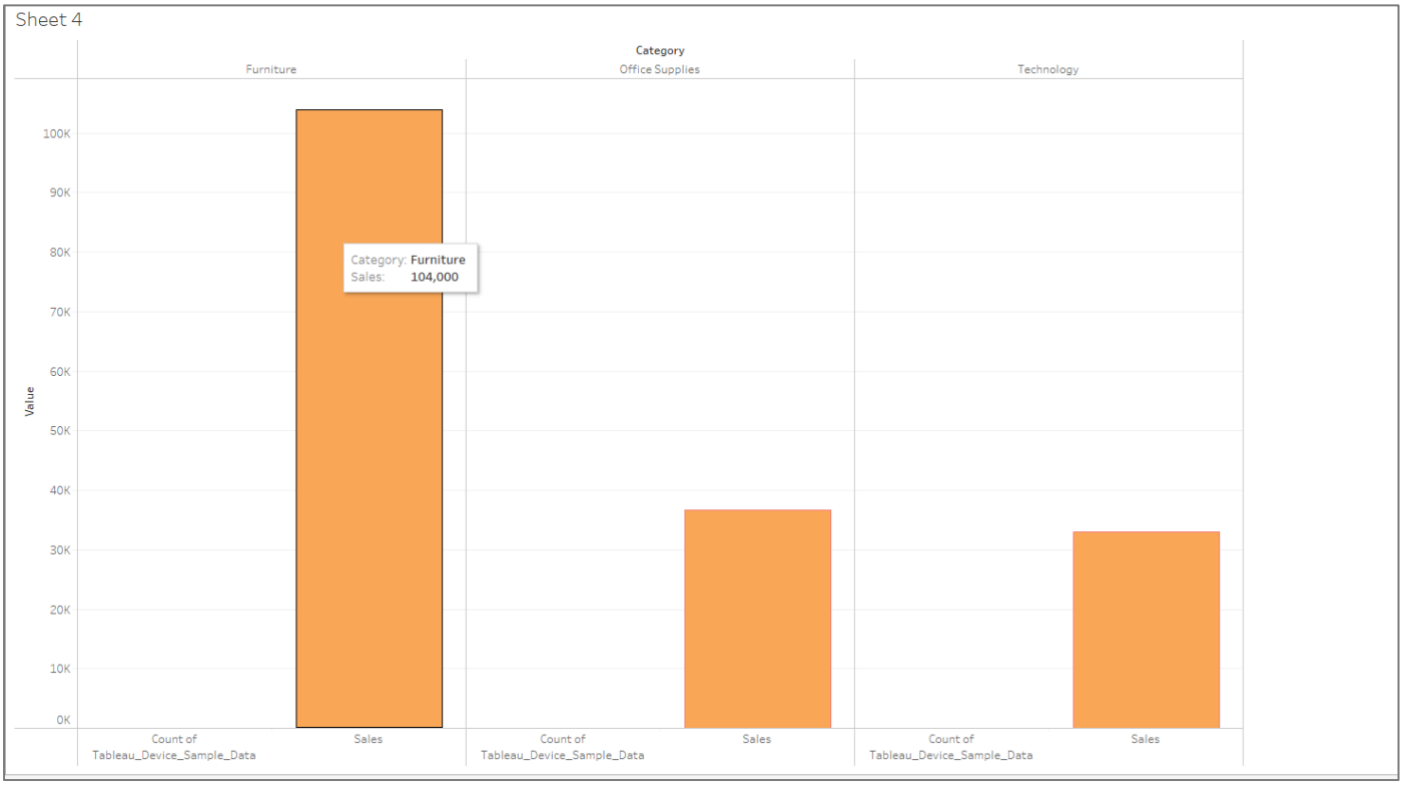
2. Adding Content to the Dashboard
    1. Drag and drop Sheets (visualizations) onto the dashboard canvas.
    2. Arrange the components (e.g., charts, filters, legends) to fit the design goals.





Sheet 2

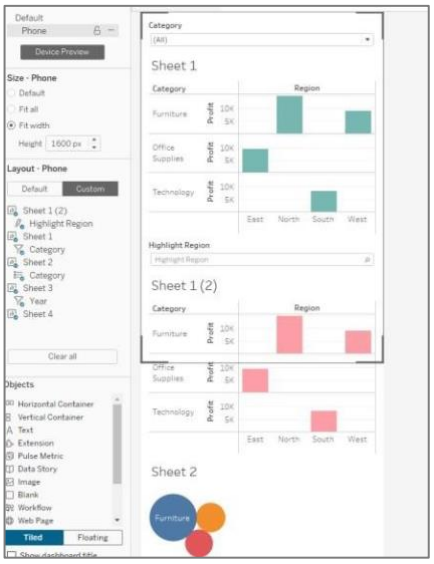
Category	
Furniture	24,000
Office Supplies	9,350
Technology	8,350



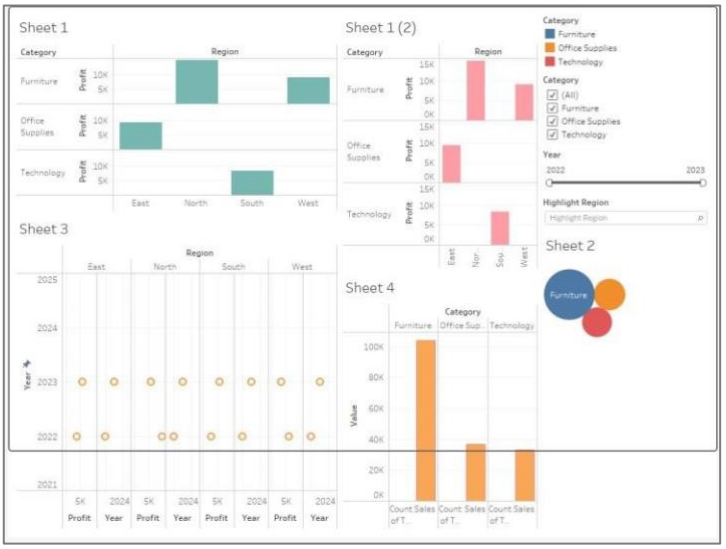
Dashboard

3. Enabling Device-Specific Dashboards

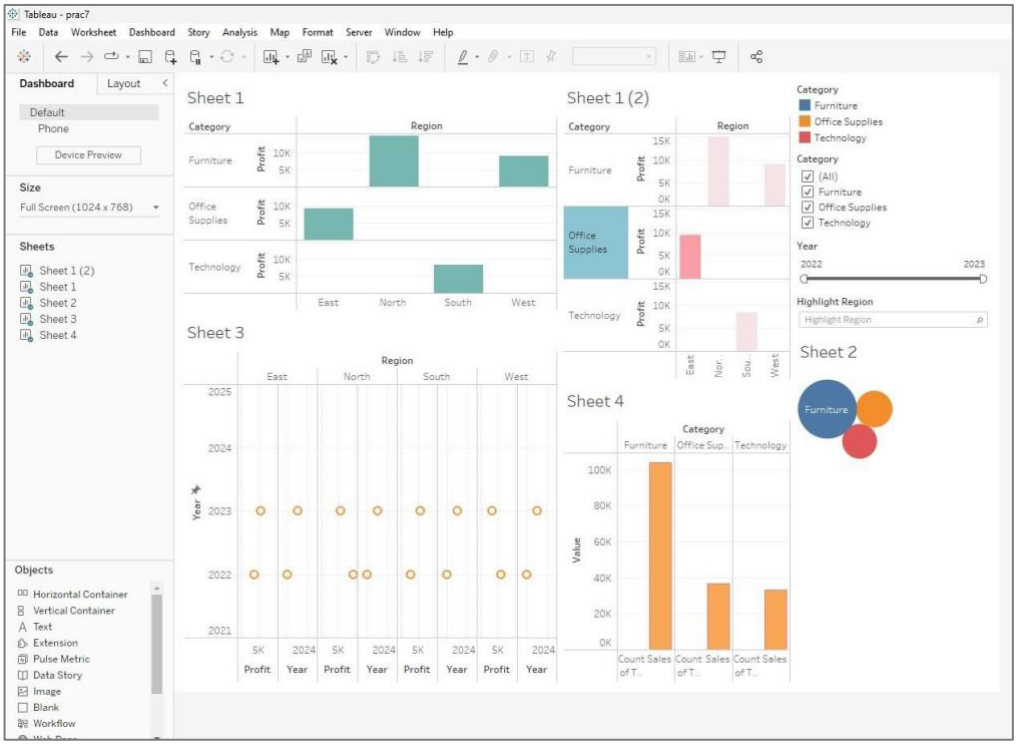
1. In the Dashboard Pane, click Device Preview.
2. Select Add a Device Layout.
3. Choose from the available devices:
   - Phone



   - Tablet



   - Desktop



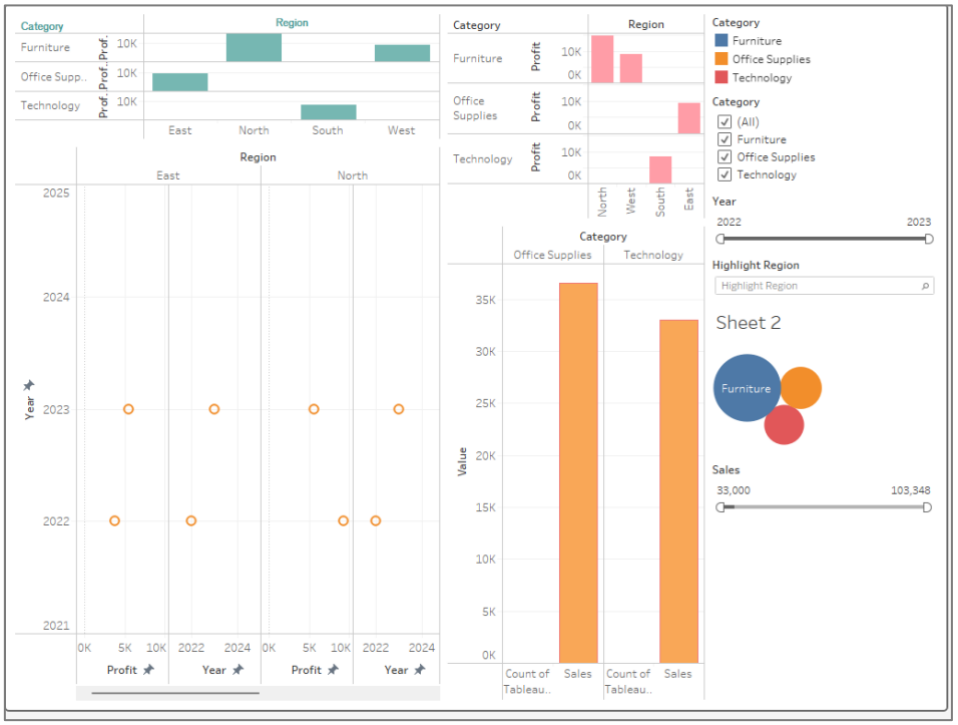
4. Customize each layout:
   - Phone Layout:

- Adjust for narrow screens by stacking charts vertically.

- Remove unnecessary elements to keep it simple.
  - Tablet Layout:
- Use moderate-sized elements.

  Balance interactivity and

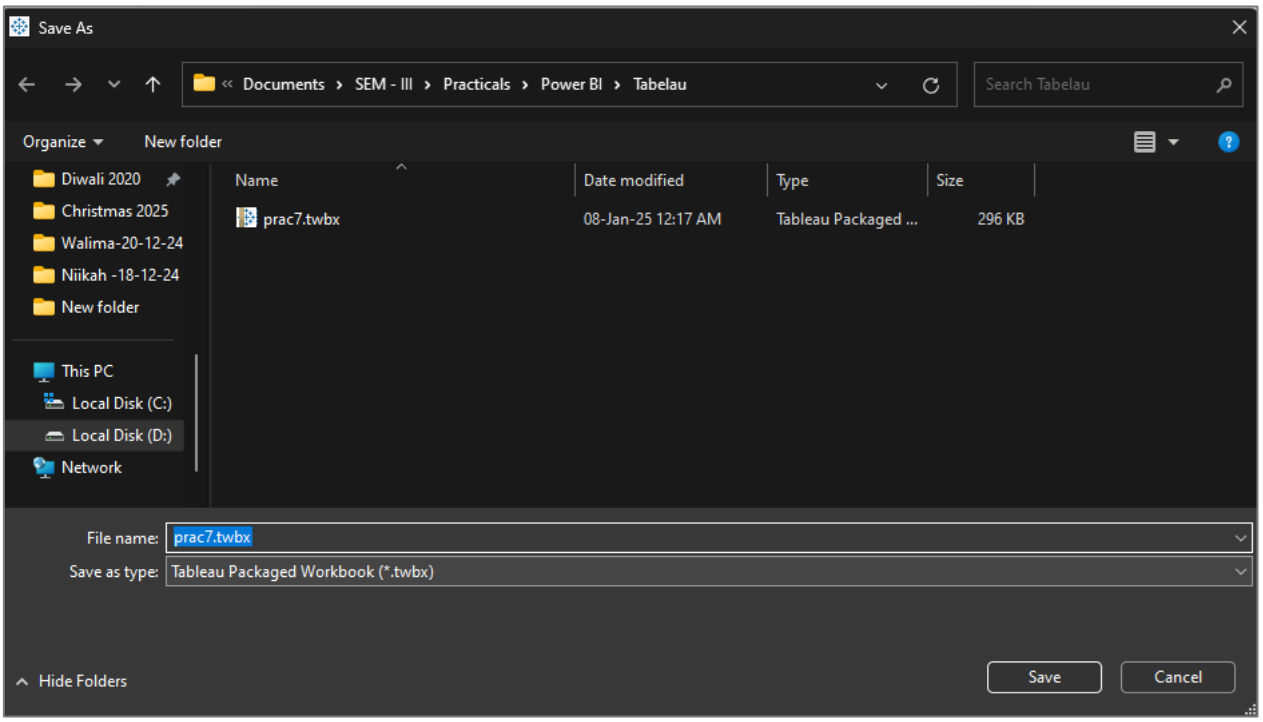
  detail.
  - Desktop Layout:
- Add full-sized visuals and interactive features.



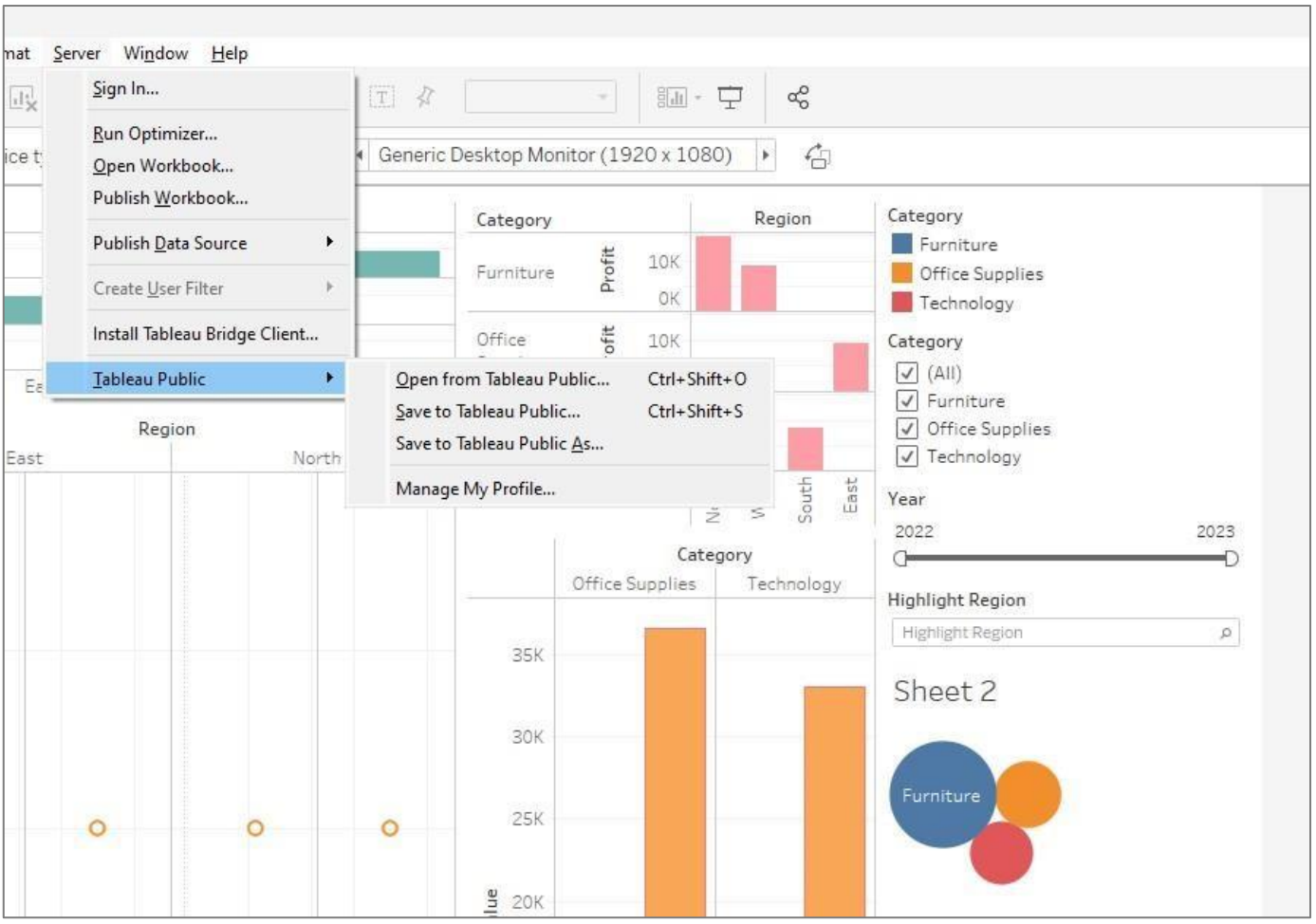
- Optimize font sizes and colors for readability across devices.
- Avoid overcrowding the dashboard by prioritizing key metrics.
- Test on actual devices to ensure usability.

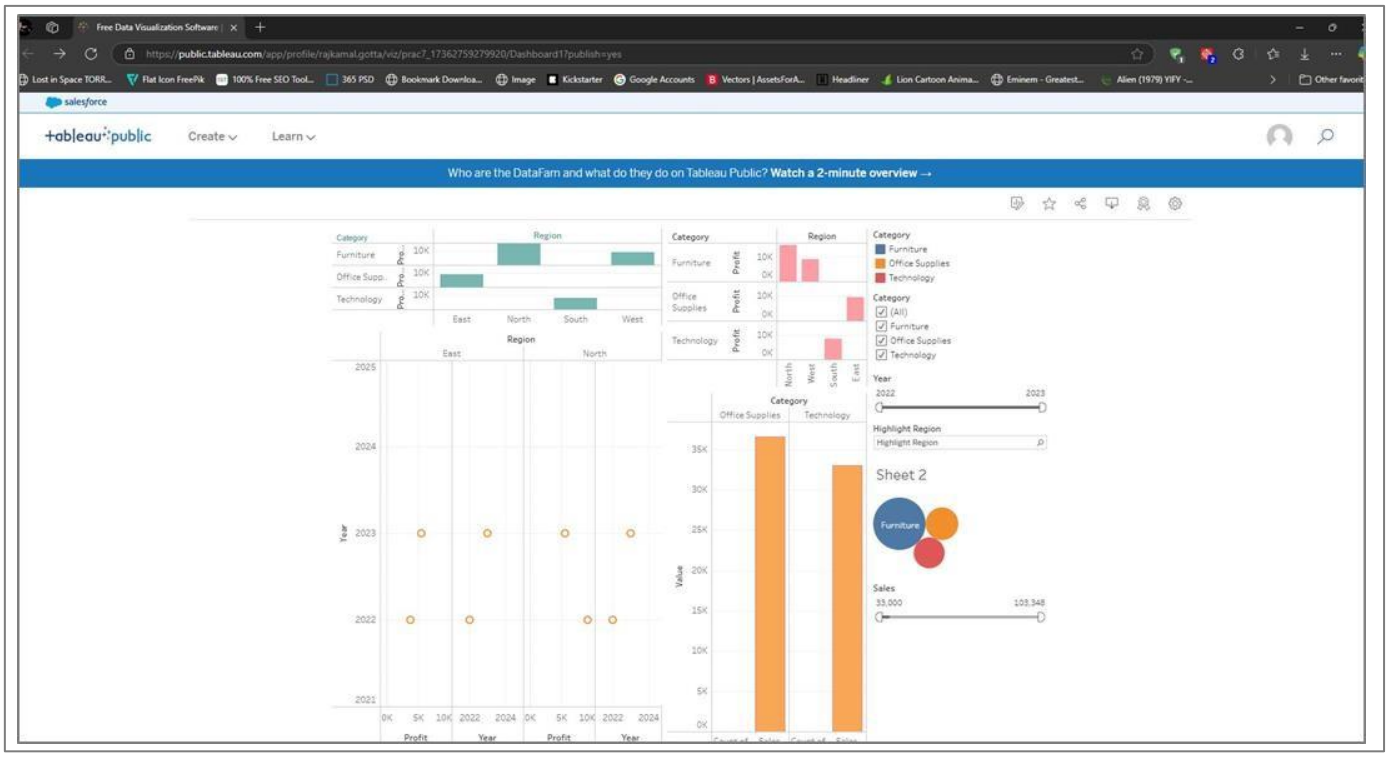
4. Exporting and Sharing

1. Save the workbook as a .twbx file to include data.



2. Publish to Tableau Server or Tableau Public for accessibility across devices.





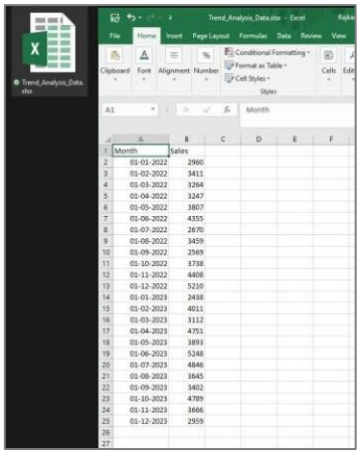
https://public.tableau.com/app/profile/mansi.gotta/viz/prac7_17362759279920/Dashboard1?publish=yes

Practical 7

Practical: Create a Trend model using data, Analyse-it and use it for forecasting.

1. Prepare the Dataset

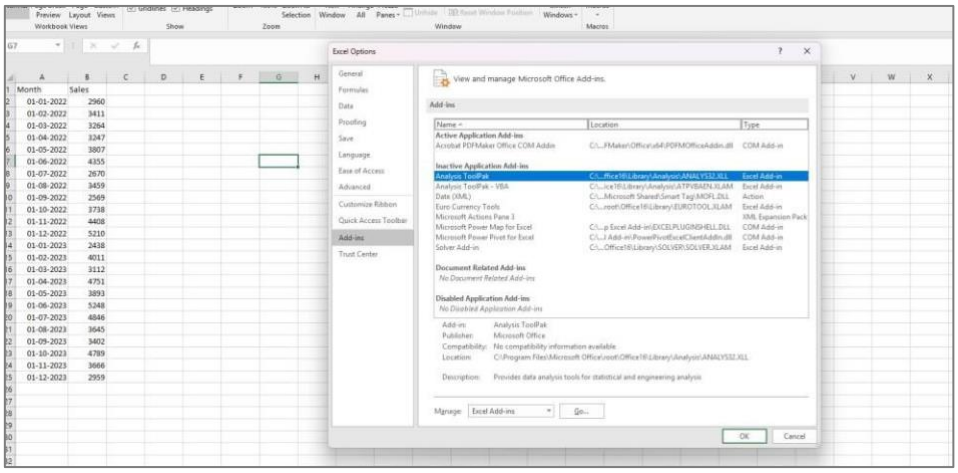
We will create a simple dataset containing monthly sales data for two years. This data will serve as the foundation for trend analysis.

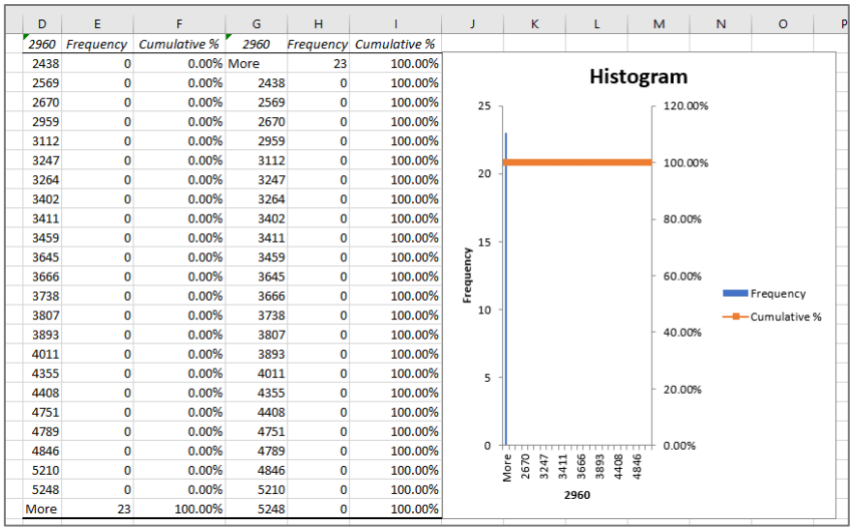


Month	Sales
01-01-2022	2960
01-02-2022	3411
01-03-2022	3264
01-04-2022	3247
01-05-2022	3807
01-06-2022	4355
01-07-2022	2670
01-08-2022	3459
01-09-2022	2569
01-10-2022	3738
01-11-2022	4408
01-12-2022	5210
01-01-2023	2438
01-02-2023	4011
01-03-2023	3112
01-04-2023	4751
01-05-2023	3893
01-06-2023	5248
01-07-2023	4846
01-08-2023	3645
01-09-2023	3402
01-10-2023	4789
01-11-2023	3666
01-12-2023	2959

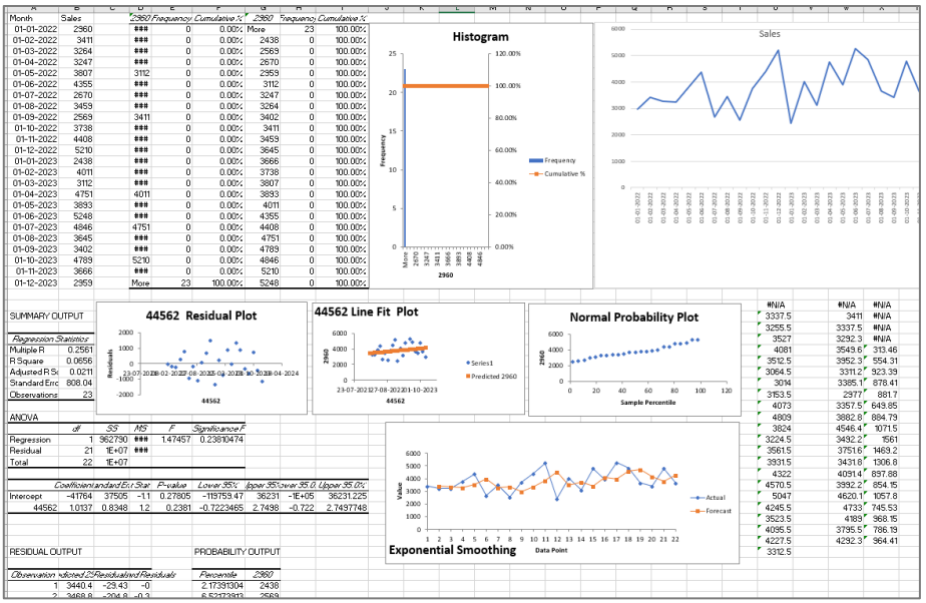
2. Load Data into the Tool

- **In Tableau**: Import the dataset by connecting to the CSV file.
- **In Analyze-it** (Excel Add-In): Open the CSV file in Excel and use Analyze-it to build the mode





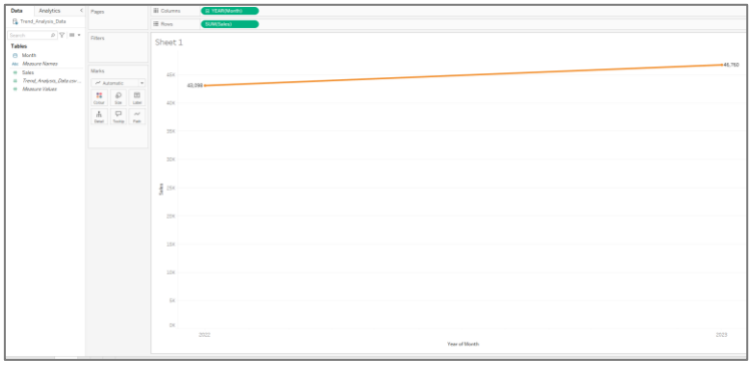
2960	Frequency	Cumulative %	2960	Frequency	Cumulative %
2438	0	0.00%	More	23	100.00%
2569	0	0.00%	2438	0	100.00%
2670	0	0.00%	2569	0	100.00%
2959	0	0.00%	2670	0	100.00%
3112	0	0.00%	2959	0	100.00%
3247	0	0.00%	3112	0	100.00%
3264	0	0.00%	3247	0	100.00%
3402	0	0.00%	3264	0	100.00%
3411	0	0.00%	3402	0	100.00%
3459	0	0.00%	3411	0	100.00%
3645	0	0.00%	3459	0	100.00%
3666	0	0.00%	3645	0	100.00%
3738	0	0.00%	3666	0	100.00%
3807	0	0.00%	3738	0	100.00%
3893	0	0.00%	3807	0	100.00%
4011	0	0.00%	3893	0	100.00%
4355	0	0.00%	4011	0	100.00%
4408	0	0.00%	4355	0	100.00%
4751	0	0.00%	4408	0	100.00%
4789	0	0.00%	4751	0	100.00%
4846	0	0.00%	4789	0	100.00%
5210	0	0.00%	4846	0	100.00%
5248	0	0.00%	5210	0	100.00%
More	23	100.00%	5248	0	100.00%



3. Visualize the

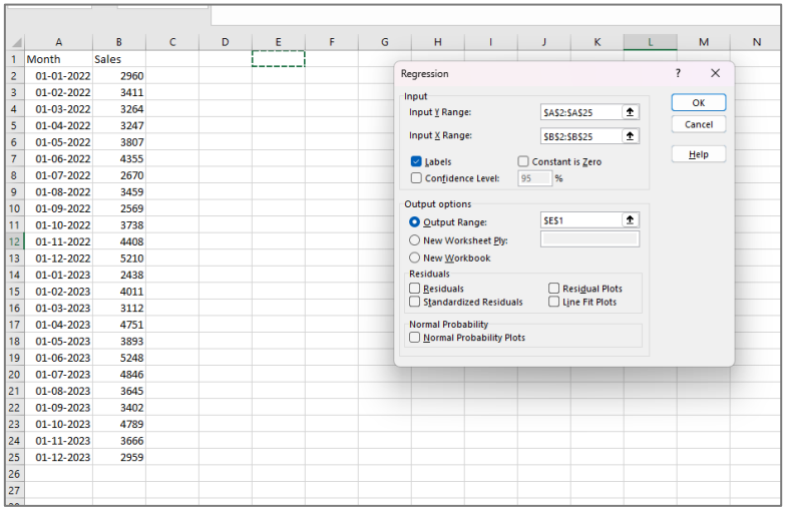
Trend In Tableau:

1. Drag **Date** to the Columns shelf (ensure it is set to "Month-Year").
2. Drag **Sales** to the Rows shelf.
3. Use a **Line Chart** to show the trend over time.



In Analyze-it:

1. Highlight the date and sales columns.
2. Use **Analyze-it** > **Regression** to analyze the trend.



Regression Statistics	
Multiple R	0.256146
R Square	0.065611
Adjusted R Square	0.021116
Standard Error	204.1759
Observations	23

ANOVA

	df	SS	MS	F	gnificance F
Regression	1	61471.74	61471.74	1.474574	0.238105
Residual	21	875443.6	41687.79		
Total	22	936915.3			

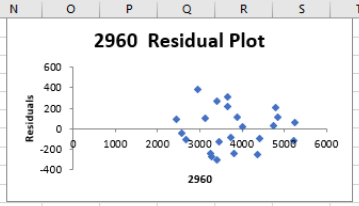
	Coefficients	andard Err	t Stat	P-value	Lower 95%	Upper 95%	ower 95.0%	pper 95.0%
Intercept	44681.64	205.8274	217.0831	1.12E-36	44253.6	45109.68	44253.6	45109.68
2960	0.064723	0.0533	1.21432	0.238105	-0.04612	0.175566	-0.04612	0.175566

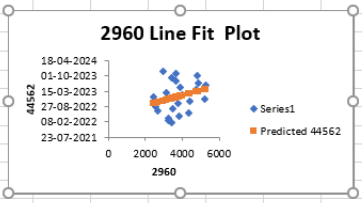
RESIDUAL OUTPUT

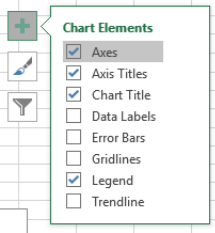
Observation	dicted 445	Residuals	dard Residuals
1	44902.41	-309.409	-1.55107
2	44892.89	-271.895	-1.36301
3	44891.79	-239.795	-1.20209
4	44928.04	-246.04	-1.2334
5	44963.51	-250.508	-1.25579
6	44854.45	-111.449	-0.5587
7	44905.52	-131.516	-0.65929
8	44847.91	-42.9124	-0.21512
9	44923.57	-88.5737	-0.44402
10	44966.94	-100.938	-0.506
11	45018.85	-122.846	-0.61583
12	44839.43	87.56637	0.43897
13	44941.24	16.75685	0.084002
14	44883.06	102.943	0.516053
15	44989.14	27.86172	0.139671
16	44933.61	113.3942	0.568444
17	45021.31	56.69431	0.284208
18	44995.29	112.713	0.56503
19	44917.55	221.4455	1.110105
20	44901.83	268.1732	1.344351

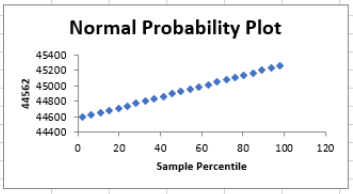
PROBABILITY OUTPUT

Percentile	44562
2.173913	44593
6.521739	44621
10.86957	44652
15.21739	44682
19.56522	44713
23.91304	44743
28.26087	44774
32.6087	44805
36.95652	44835
41.30435	44866
45.65217	44896
50	44927
54.34783	44958
58.69565	44986
63.04348	45017
67.3913	45047
71.73913	45078
76.08696	45108
80.43478	45139
84.78261	45170





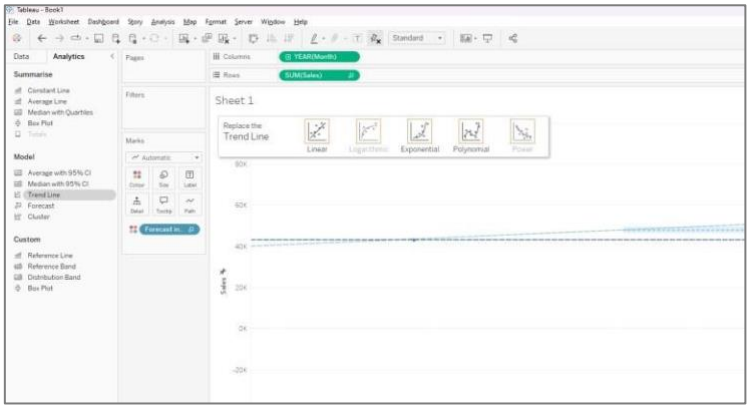


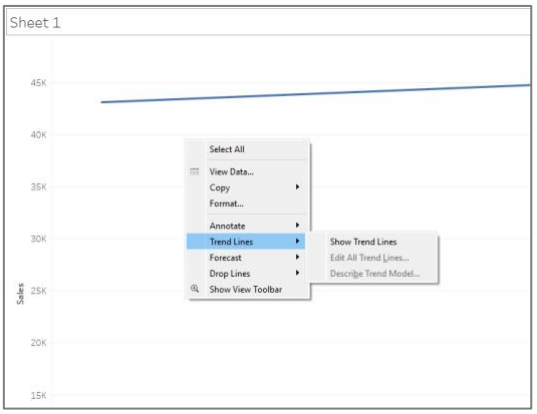


4. Add a Trend

Line In Tableau:

1. Right-click on the chart and select **Trend Line** > **Show Trend Lines**.
2. Choose the type of trend model (Linear, Logarithmic, or Polynomial).





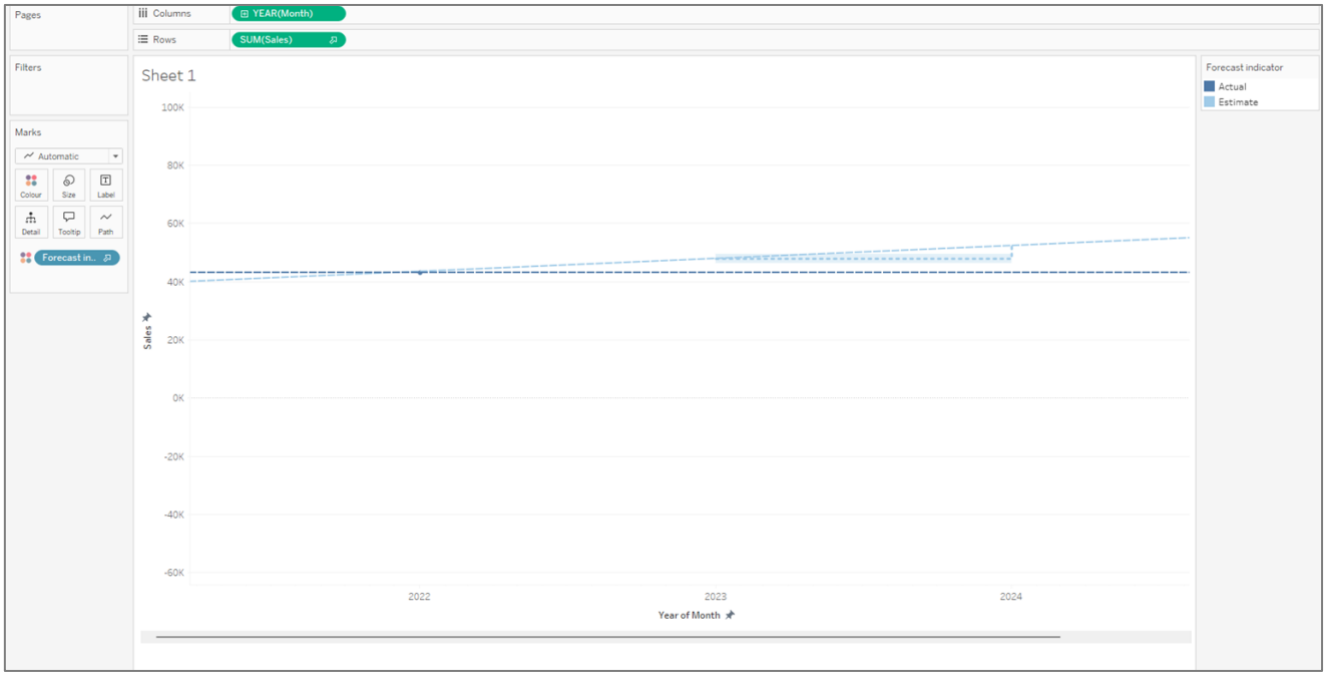
In Analyze-it:

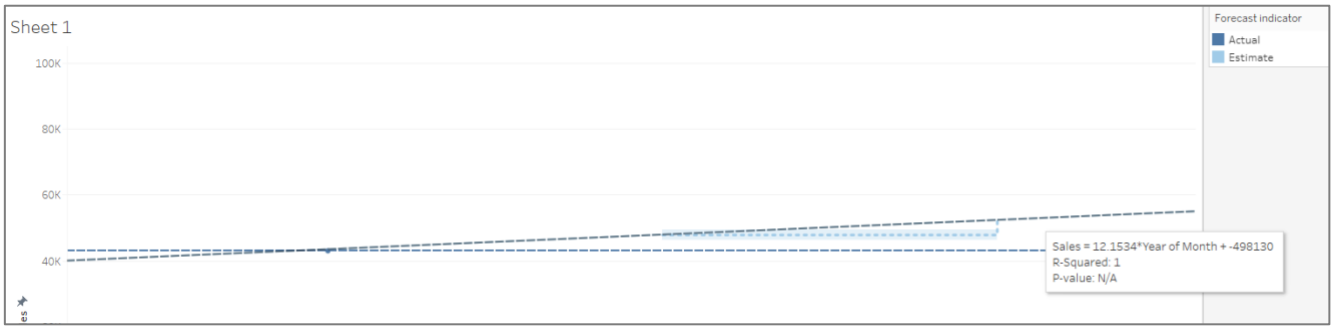
1. After running the regression, view the output for the trend equation (e.g., y = mx + c).

5. Forecast Future

Values In Tableau:

1. Right-click on the chart and choose **Forecast** > **Show Forecast**.
2. Customize the forecast length (e.g., 6 months).
3. Analyze the confidence interval and projected values.





Options Used to Create Forecasts

Time series: Year of Month

Measures: Sum of Sales

Forecast forward:

Forecast based on: January 2022 – November 2023

Ignore last: 1 month (December 2023)

Seasonal pattern: None (Not enough data to search for a seasonal pattern recurring every 12 Months)

Sum of Sales

Initial December 2023	Change From Initial December 2023 – December 2024	Seasonal Effect High	Seasonal Effect Low	Contribution Trend	Contribution Season	Quality
4,086 ± 1,551	516	None		100.0%	0.0%	OK

In Analyze-it:

1. Use the trend equation from the regression output.
2. Add future dates in Excel and compute forecasts using the equation.

	A	B
1	Month	Sales
2	01-01-2022	2960
3	01-02-2022	3411
4	01-03-2022	3264
5	01-04-2022	3247
6	01-05-2022	3807
7	01-06-2022	4355
8	01-07-2022	2670
9	01-08-2022	3459
10	01-09-2022	2569
11	01-10-2022	3738
12	01-11-2022	4408
13	01-12-2022	5210
14	01-01-2023	2438
15	01-02-2023	4011
16	01-03-2023	3112
17	01-04-2023	4751
18	01-05-2023	3893
19	01-06-2023	5248
20	01-07-2023	4846
21	01-08-2023	3645
22	01-09-2023	3402
23	01-10-2023	4789
24	01-11-2023	3666
25	01-12-2023	2959
26	01-01-2024	4186.420419
27	01-02-2024	4186.698188
28	01-03-2024	4213.962132
29	01-04-2024	4225.934118
30	01-05-2024	4226.928512
31	01-06-2024	4271.228727

RESIDUAL OUTPUT

Observation	Predicted 44562	Residuals	Standard Residuals
1	44902.40926	-309.4092603	-1.551067044
2	44892.89496	-271.8949571	-1.363008034
3	44891.79466	-239.7946635	-1.202089426
4	44928.03963	-246.0396282	-1.233395402
5	44963.50792	-250.5079151	-1.255794901
6	44854.44941	-111.4494053	-0.5586953
7	44905.51597	-131.5159716	-0.659288895
8	44847.91237	-42.91236699	-0.215119477
9	44923.57373	-88.57373079	-0.444019661
10	44966.93824	-100.9382421	-0.506002893
11	45018.84621	-122.8462094	-0.615827421
12	44839.43363	87.56636582	0.438969745
13	44941.24315	16.75684892	0.084001998
14	44883.05704	102.9429619	0.516052543
15	44989.13828	27.861717	0.139670645
16	44933.60582	113.3941808	0.56844445
17	45021.30569	56.69431083	0.284208291
18	44995.28698	112.7130176	0.565029783
19	44917.55448	221.4455223	1.110105274
20	44901.82675	268.173248	1.344351125
21	44991.59776	208.4022372	1.044719353
22	44918.91366	312.0863361	1.564487212
23	44873.1544	387.845604	1.944268035

PROBABILITY OUTPUT

Percentile	44562
2.173913	44593
6.521739	44621
10.86957	44652
15.21739	44682
19.56522	44713
23.91304	44743
28.26087	44774
32.6087	44805
36.95652	44835
41.30435	44866
45.65217	44896
50	44927
54.34783	44958
58.69565	44986
63.04348	45017
67.3913	45047
71.73913	45078
76.08696	45108
80.43478	45139
84.78261	45170
89.13043	45200
93.47826	45231
97.82609	45261

=FORECAST(x, known_y's, known_x's

6. Interpret the Results

- Observe the overall trend (e.g., increasing or decreasing).

Practical 8

Practical: Creating Geospatial feature maps in Tableau using Geospatial Data.

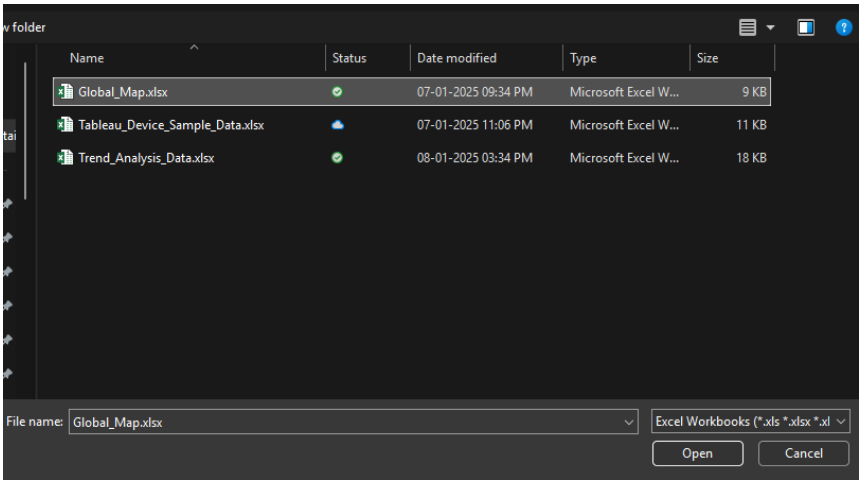
Steps to Create Geospatial Maps in Tableau

1. **Prepare Your Data**:
   - Use a dataset that includes geospatial data, such as latitude, longitude, or geographical regions (e.g., city names, states, or countries).
   - Ensure the data is clean and organized in a CSV or Excel format.

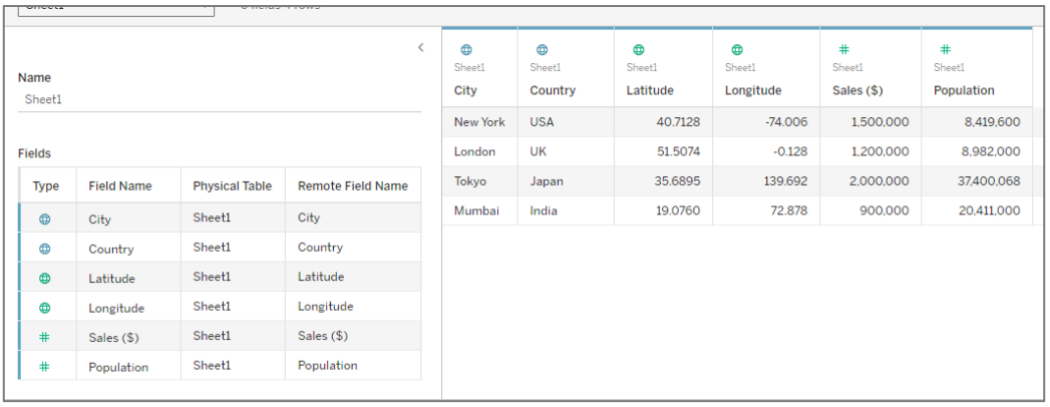
City	Country	Latitude	Longitude	Sales ($)	Population
New York	USA	40.7128	-74.006	1500000	8419600
London	UK	51.5074	-0.1278	1200000	8982000
Tokyo	Japan	35.6895	139.6917	2000000	37400068
Mumbai	India	19.076	72.8777	900000	20411000

2. **Load Data into Tableau**:
   - Open Tableau and connect to your dataset by clicking **Connect > Text File** or **Microsoft Excel**, then select your file.

3. **Assign Geographic Roles**:



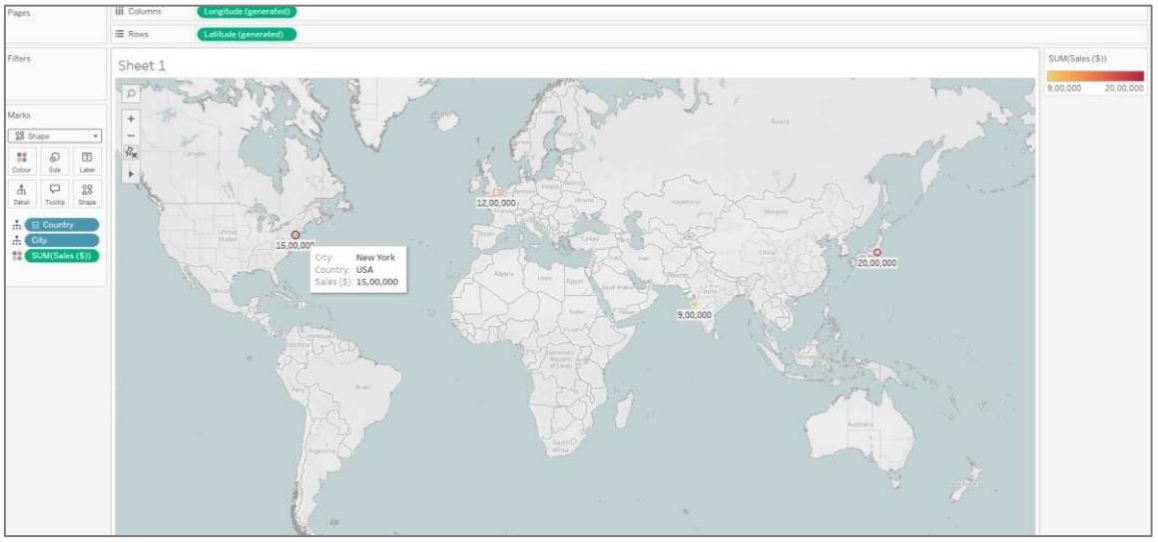
   - Tableau often automatically assigns geographic roles based on column names (e.g., "City," "Country"). If it doesn't, right-click the field in the Data pane, select **Geographic Role**, and choose the appropriate role.



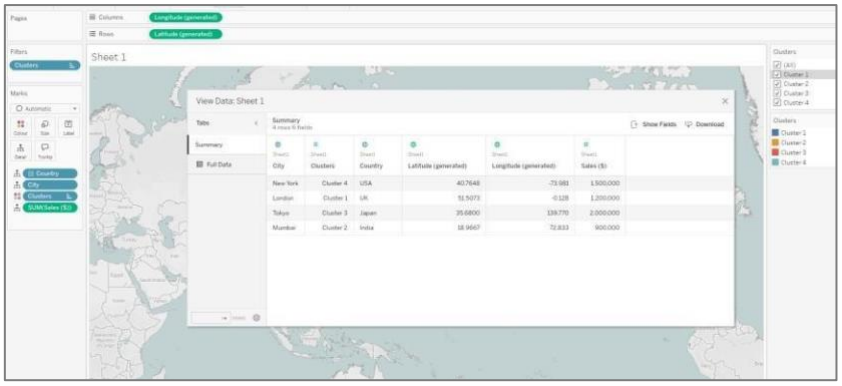
4. **Create a Map View**:
   - Drag the geographic field (e.g., City or Country) onto the **Rows** or **Columns** shelf.
   - Tableau will automatically generate a map with data points.
5. **Enhance the Map with Data**:
   - Drag a measure (e.g., Sales, Population) to **Color** or **Size** on the **Marks** card to visualize data distribution.

   - Use the **Filter** card to focus on specific regions or categories.

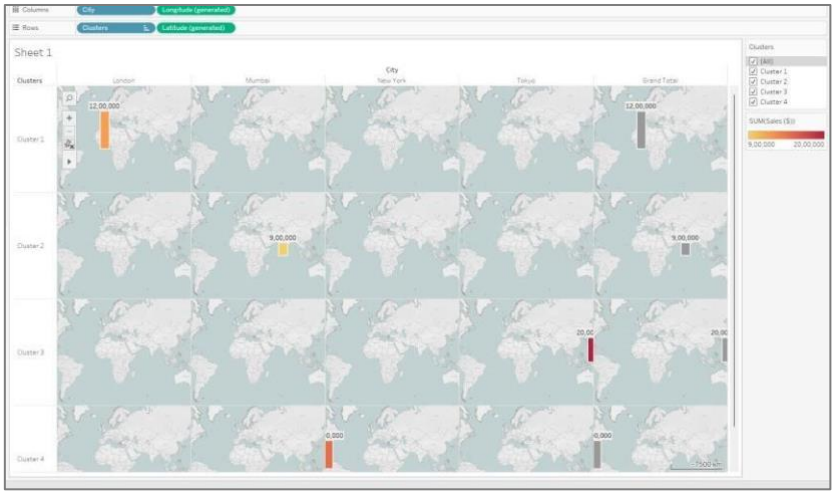
6. **Customize the Map**:
   - Click **Map** > **Map Layers** to adjust map styles, such as borders, terrain, or dark mode.
   - Add tooltips, labels, or additional measures for a richer display.



7. **Analyze Geospatial Features**:
   - Use Tableau's **Analytics pane** to overlay trends or clusters.
   - Implement filters to allow interactive exploration of the data.



8. **Save and Share**:
   - Save your workbook or export it as a packaged workbook (.twbx) to include data and visualization.



Practical 9

Practical: Create Dashboard and Storytelling using tableau.

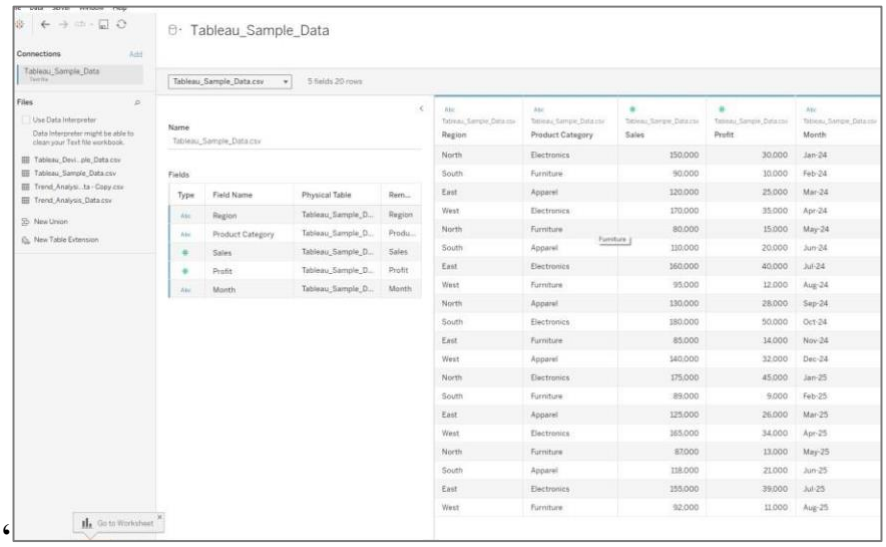
1. Prepare Your Data

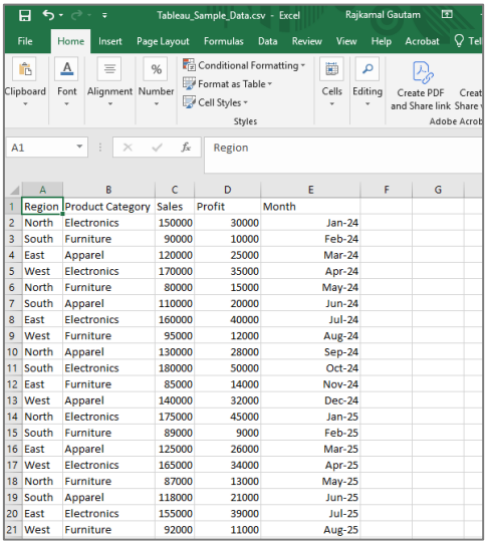
- Use a dataset with multiple dimensions and measures (e.g., sales, region, category, profit). Example dataset: Sales_Performance_Data.xlsx

Region	Product Category	Sales ($)	Profit ($)	Month
North	Electronics	150000	30000	January 2024
South	Furniture	90000	10000	February 2024
East	Apparel	120000	25000	March 2024
West	Electronics	170000	35000	April 2024

2. Load the Data

1. Open Tableau and click **Connect > Microsoft Excel** or another format.
2. Choose your file and load the data.

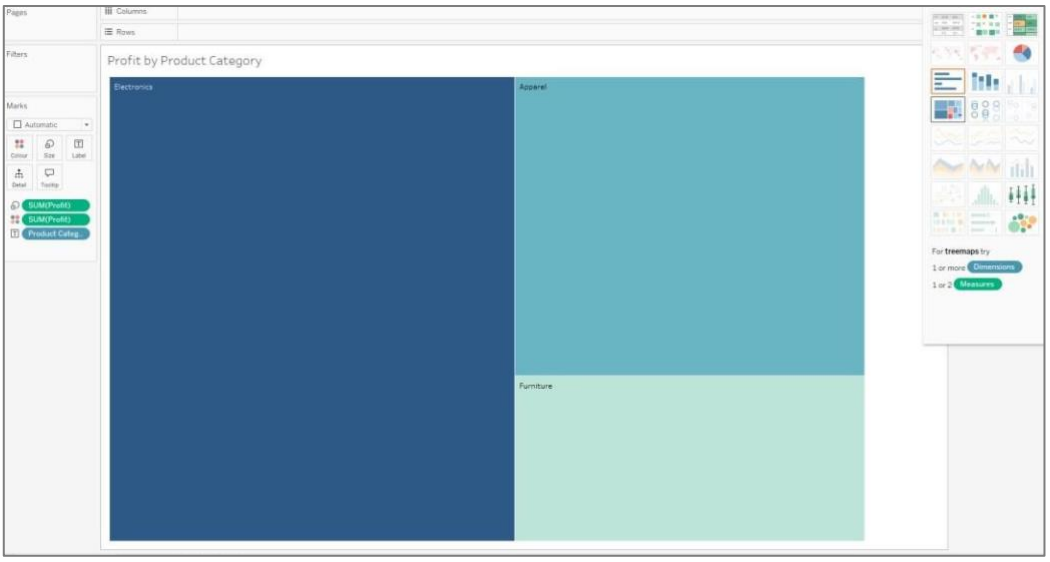


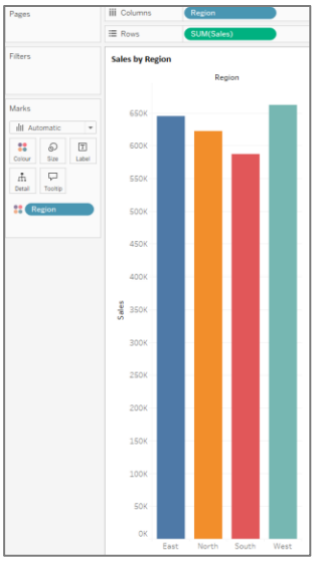


3. Create Sheets for the Dashboard

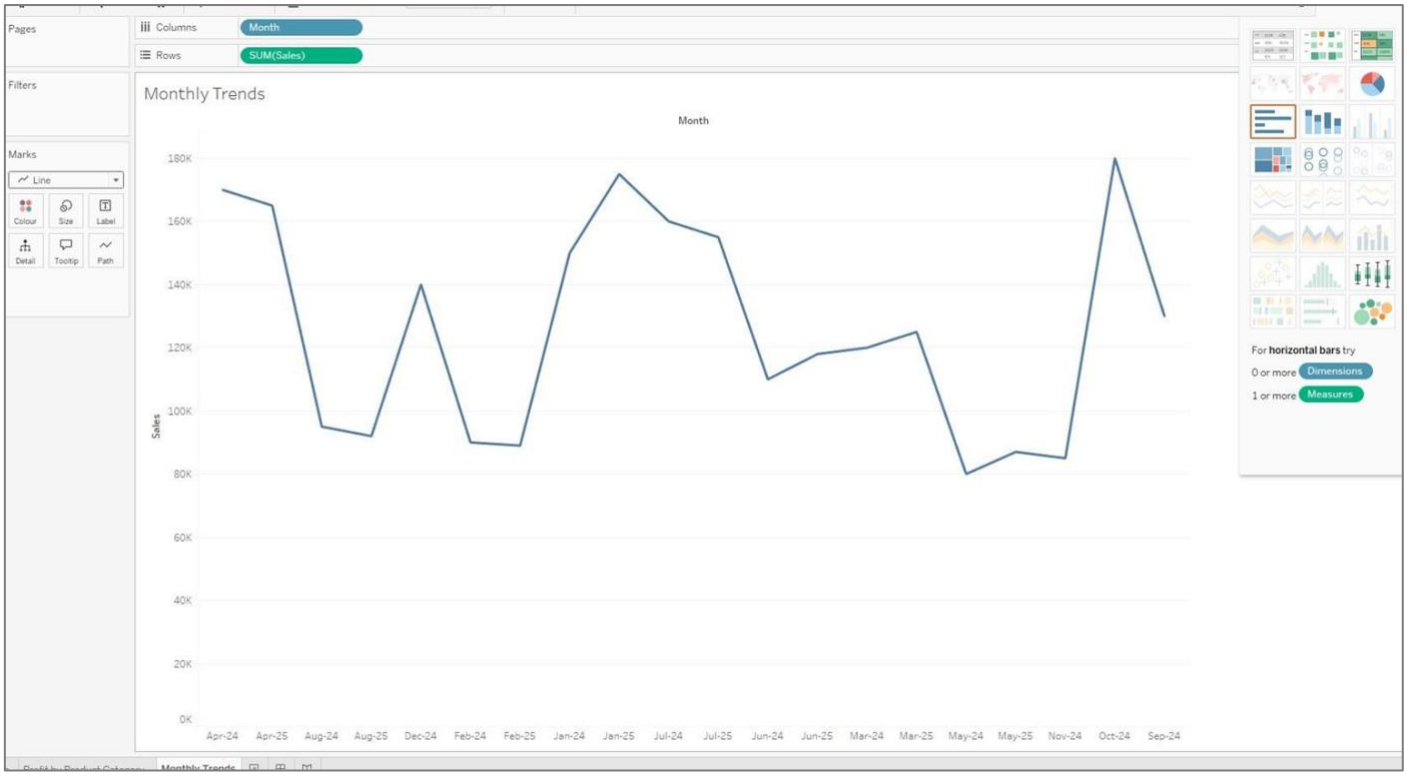
- **Sheet 1: Sales by Region**
  - Drag Region to Rows and Sales to Columns.
  - Add Sales to the **Color** shelf for a heat map effect.
  - Use a bar or pie chart.

- **Sheet 2: Profit by Product Category**
  - Drag Product Category to Rows and Profit to Columns.
  - Choose a tree map for better visualization.





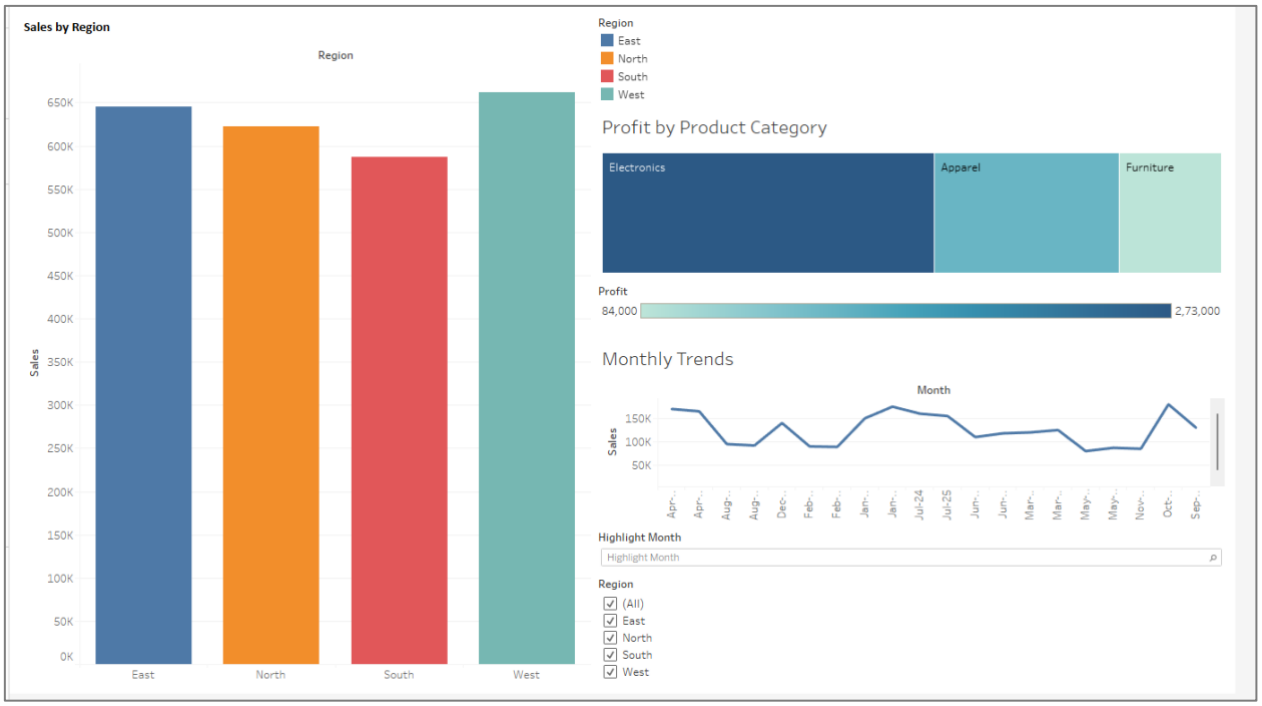
- **Sheet 3: Monthly Trends**
  - Drag Month to Columns and Sales to Rows.
  - Choose a line chart to show trends over time.



4. Build the Dashboard

1. Go to the **Dashboard** tab.
2. Drag and drop your created sheets (e.g., Sheet 1, Sheet 2) into the dashboard.
3. Add filters (e.g., Region filter) for interactivity:
   - Drag the filter to the dashboard for user input.
4. Customize:
   - Add text boxes, titles, or images.

- Adjust the layout for clarity.



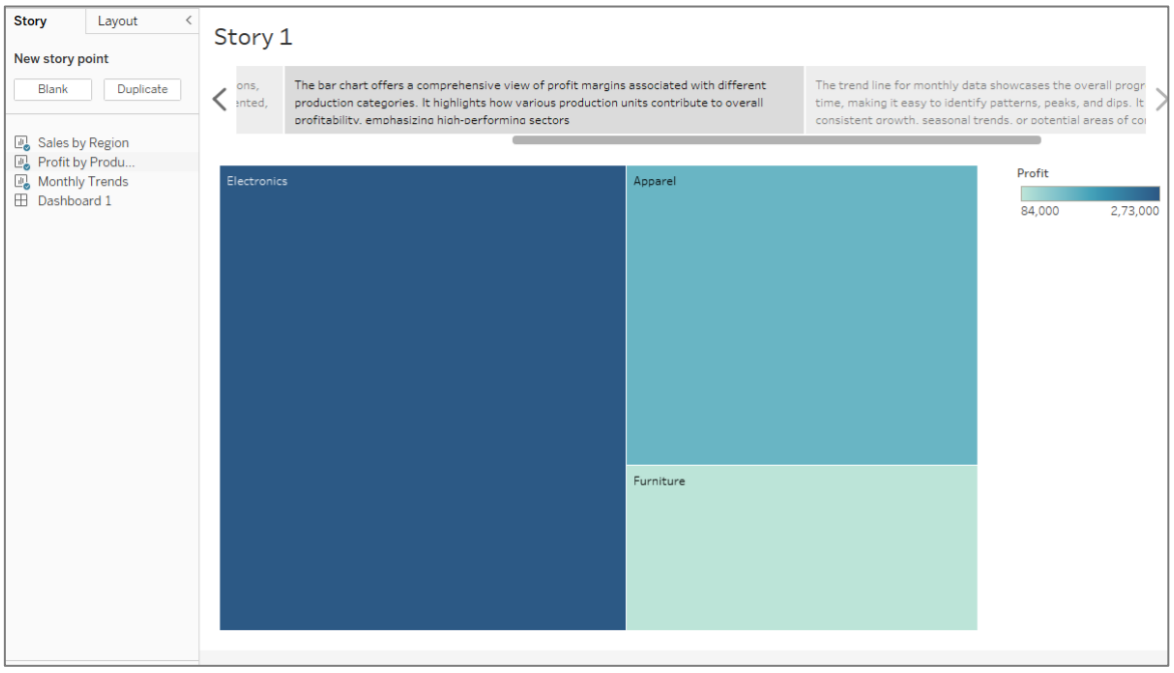
Steps to Create a Story in Tableau

1. Prepare a Story

1. Click **Story** in the Tableau toolbar.
2. Select **New Story**.

2. Add Sheets to the Story

1. Drag dashboard views or individual sheets to the story area.
2. Add captions to explain insights for each step.

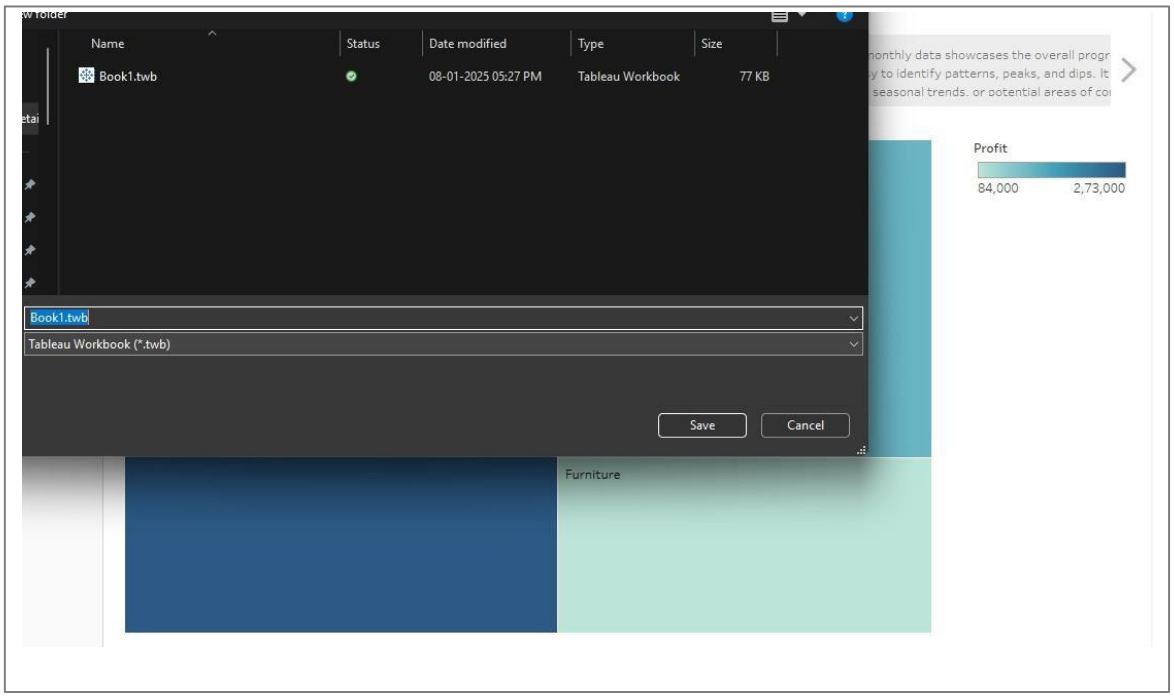
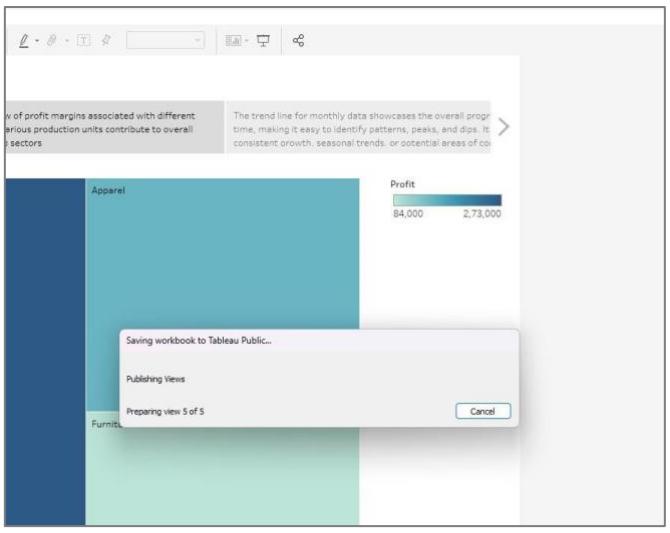
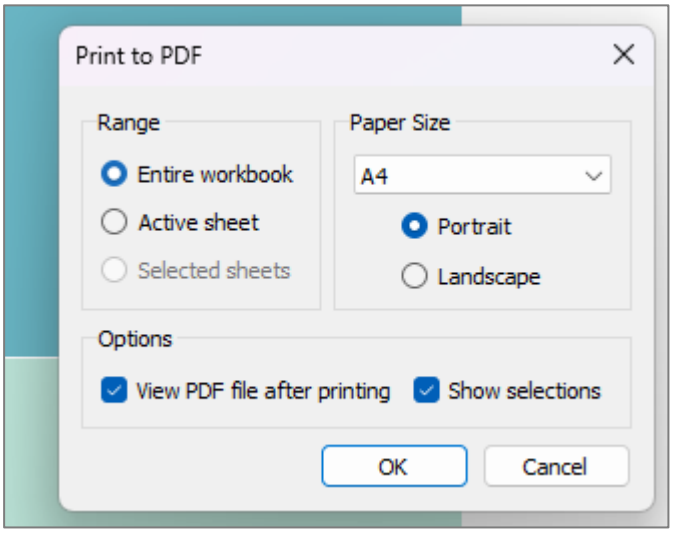


3. Customize the Story

1. Adjust the size and layout of each story point.
2. Add annotations to highlight key findings.

4. Save and Share

1. Save the workbook as .twbx to include data and visuals.



2. Export as PDF or share on Tableau Public for easy access.