

**College code:5113**

E. Chanikya-(511321104013) – Team Head

[chanikyaeddula@gmail.com](mailto:chanikyaeddula@gmail.com)

E. Manoj-(511321104053)

[manojerragopula@gmail.com](mailto:manojerragopula@gmail.com)

R.Bharath kumar-(511321104010)

[bk7428891@gmail.com](mailto:bk7428891@gmail.com)

V.Mohith Manindranath-(511321104055)

[v.mohitmanindranath162003@gmail.com](file:///E:\v.mohitmanindranath162003@gmail.com)

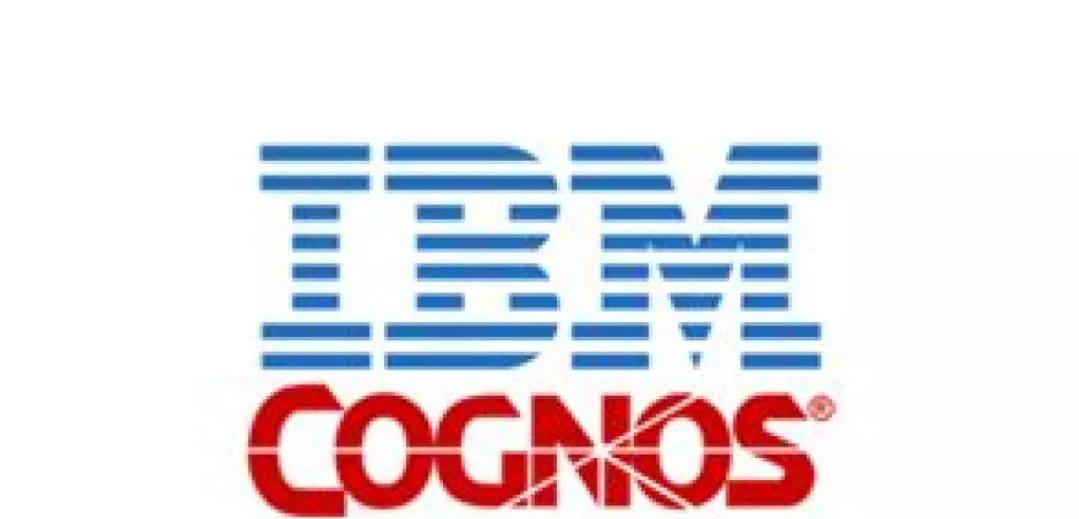
**Data Analytics with Cognos**

PHASE 5

PROJECT:

**Product Sales Analysis**

IBM Cognos logo:



Visualisation:



OBJECTIVES:

Analyzing product sales data is crucial for businesses to make informed decisions, optimize strategies, and drive growth. Here is a structured approach with steps for conducting a product sales analysis project:

**Step 1: Define Objectives and Scope**

Clearly define the objectives of your product sales analysis. Identify what specific aspects of sales you want to analyze (e.g., overall sales performance, product-specific performance, market segmentation, or sales forecasting).

**Step 2: Data Collection and Preparation**

Collect relevant sales data, which may include transaction records, product details, customer information, and market data. Ensure data quality by cleaning and preprocessing the data:

* Handle missing data by imputing values or removing incomplete records.
* Standardize and clean product and customer names for consistency.
* Convert data types and formats as needed.

**Step 3: Data Exploration and Descriptive Analysis**

Conduct exploratory data analysis (EDA) to understand the dataset's characteristics:

* Calculate basic statistics like mean, median, and standard deviation.
* Create visualizations (e.g., histograms, bar charts, scatter plots) to identify trends and patterns in sales data.
* Segment data by time (e.g., monthly, quarterly, annually) to analyze seasonality and trends.

**Step 4: Product Performance Analysis**

Analyze product-level performance to identify top-selling products, slow-moving items, and underperforming products:

* Calculate metrics like total revenue, quantity sold, profit margins, and growth rates for each product.
* Identify product categories or SKUs that contribute significantly to overall sales.

**Step 5: Customer Segmentation**

Segment customers based on various criteria such as demographics, purchase behavior, or customer lifetime value:

* Identify high-value customers, returning customers, and potential target segments.
* Analyze customer purchase patterns and preferences.

**Step 6: Market Analysis**

Analyze market data to understand external factors that influence sales:

* Study market trends, economic indicators, and competitor performance.
* Assess the impact of marketing campaigns, promotions, and pricing strategies.

**Step 7: Time Series Analysis**

Perform time series analysis to understand sales patterns over time:

* Apply forecasting models (e.g., moving averages, exponential smoothing, ARIMA) to predict future sales.
* Evaluate forecast accuracy using appropriate metrics (e.g., Mean Absolute Error, Root Mean Squared Error).

**Step 8: Sales Funnel Analysis (if applicable)**

For businesses with multi-step sales processes (e.g., e-commerce sites), analyze the sales funnel:

* Monitor conversion rates at each stage of the funnel (e.g., website visits, product views, cart additions, checkout).
* Identify bottlenecks and areas for optimization.

**Step 9: Root Cause Analysis**

Investigate the factors contributing to fluctuations or changes in sales:

* Use statistical methods or hypothesis testing to identify the root causes of sales variations.
* Assess the impact of internal and external factors (e.g., product launches, economic downturns) on sales.

**Step 10: Visualization and Reporting**

Create visual reports and dashboards to communicate insights effectively to stakeholders:

* Use tools like Tableau, Power BI, or Python libraries (e.g., Matplotlib, Seaborn) to visualize data.
* Prepare a comprehensive report summarizing key findings, trends, and recommendations.

**Step 11: Recommendations and Action Plan**

Based on the analysis, provide actionable recommendations to improve sales performance:

* Propose pricing adjustments, marketing strategies, or product enhancements.
* Create a prioritized action plan with clear objectives and timelines.

**Step 12: Implementation and Monitoring**

Implement the recommended actions and closely monitor their impact on sales:

* Track sales performance after implementing changes.
* Adjust strategies as needed and continue monitoring over time.

**Step 13: Documentation and Knowledge Sharing**

Document the entire analysis process, methodologies, and results for future reference and knowledge sharing within the organization.

**Dataset:**

The dataset is comprised of hundreds of thousands of electronics store purchases broken down by product type, prices, order date, purchase address, etc., corresponding to the following coloumns:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **Column** | **Description** | | --- | --- | | **Order ID** | Unique IDs that are used to track orders. | | **Product** | Names of the products. | | **Quantity Ordered** | Total quantity ordered of a particular item. | | **Price Each** | Prices of the products ordered. | | **Order Date** | Dates and time at which a customer made an order. | | **Purchase Address** | Addresses the orders were delivered to. | |

[**https://www.kaggle.com/datasets/beekiran/sales-data-analysis**](https://www.kaggle.com/datasets/beekiran/sales-data-analysis)

**Project:**

The study will make use of finding best month for sales, how much earned on that month. This complete evaluation will provide a clear picture of the analysis of the sales based on order ID, Quality ordered, price each, order date, purchase address, Month, Sales, City, Hours which involves in the best sales month, forecasting profit, and the demand of the product.

**Importing the required libraries:**

In this step we are going to import the required python libraries and modules to work with our data and perform various data processing and machine learning tasks.

import pandas as pd

import pathlib

import numpy as np

import matplotlib as mpl

import matplotlib.pyplot as plt

import warnings

warnings.simplefilter("ignore")

**Loading the dataset:**

This step involves loading our dataset into memory. We use libraries like pandas to read data from a CSV file or other formats.

all\_data = pd.read\_csv("/kaggle/input/sales-data-analysis/Sales Data.csv")

**Preprocessing the data:**

Preprocessing data in air quality analysis is a crucial step to ensure that the data is clean, reliable, and ready for in-depth analysis.

**Data Cleaning:**

**Missing data handling:**

Identify and address missing data points, which can result from sensor malfunctions or communication issues. Options include imputing missing values or removing affected data points if necessary.

**Outlier Detection:**

Detect and handle outliers, which can skew the analysis. Outliers may result from equipment malfunction or unusual events. You can choose to filter out extreme values or apply statistical techniques like Z-score analysis to identify them.

This step is vital for accurate comparisons and correlations between different datasets.

**Data Transformation:**

**Feature Scaling:**

Normalize or standardize numerical features to bring them to a similar scale. This is important for algorithms sensitive to feature scales.

**Feature Encoding:**

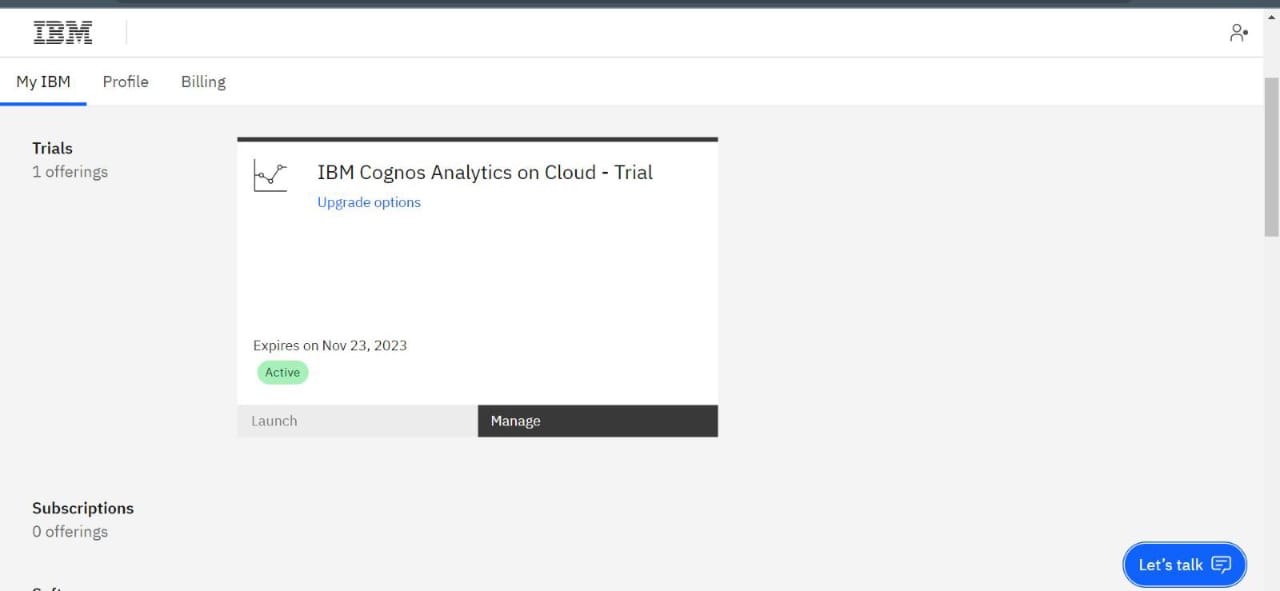
Convert categorical variables into a numerical format using techniques like one-hot encoding or label encoding.

**Feature Engineering:**

Create new features or modify existing ones to capture relevant information and patterns in the data.

**Binning:**

Group continuous data into bins or categories to simplify analysis.

**Log Transformation:**

Apply logarithmic transformations to features when necessary to make their distribution more normal.

**Data Reduction:**

**Dimensionality Reduction:**

Reduce the number of features, often using techniques like Principal Component Analysis (PCA) or feature selection to select the most relevant variables.

**Outlier Detection and Handling:**

Identify and deal with outliers, which can distort analysis and modeling results.

**Data Integration:**

Merge data from multiple sources or datasets to create a consolidated dataset for analysis.

**Exploratory Data Analysis:**

It focuses on Exploratory Data Analysis (EDA). It involves exploring and visualizing the data to gain insights. In this example, a simple time series plot is created using Matplotlib to visualize the sales.

**Question 1: What was the best month for sales? How much was earned that month?**

To answer this question, first, we need to extract only the months from the 'Order Date' coloumn and store each separately in a new coloumn, 'Months'. Second, we need to get the total sales amounts per order by multiplying the quantity ordered with the price of each individual product, and creating and storing the results in a 'Sales' coloumn. Finally, I will group the data by month, calculate the total sum of sales per month, and, lastly, visualize the data to get a better view of how sales changed from one month to the next.

[Months\_col](C:\\Users\\Bhanu\\Downloads\\www) **[=](C:\\Users\\Bhanu\\Downloads\\www)** [pd](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[to\_datetime(df['Order Date'])](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[dt](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[month\_name()](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[str[:3]](C:\\Users\\Bhanu\\Downloads\\www)

[df](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[insert(loc](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**[5, column](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**['Months', value](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**[Months\_col)](C:\\Users\\Bhanu\\Downloads\\www)

[Sales\_col](C:\\Users\\Bhanu\\Downloads\\www) **[=](C:\\Users\\Bhanu\\Downloads\\www)** [df['Quantity Ordered']](C:\\Users\\Bhanu\\Downloads\\www) **[\*](C:\\Users\\Bhanu\\Downloads\\www)** [df['Price Each']](C:\\Users\\Bhanu\\Downloads\\www)

[df](C:\\Users\\Bhanu\\Downloads\\www)**[.](C:\\Users\\Bhanu\\Downloads\\www)**[insert(loc](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**[4, column](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**['Sales', value](C:\\Users\\Bhanu\\Downloads\\www)**[=](C:\\Users\\Bhanu\\Downloads\\www)**[Sales\_col)](C:\\Users\\Bhanu\\Downloads\\www)

[sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [df](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[groupby(['Months'])](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[sum()[['Quantity Ordered', 'Sales']]](C:\\Users\\Bhanu\\Downloads\\s)

[sort\_order](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']](C:\\Users\\Bhanu\\Downloads\\s)

[sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[reindex(sort\_order)](C:\\Users\\Bhanu\\Downloads\\s)

[sales\_per\_month\_USD](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[copy()](C:\\Users\\Bhanu\\Downloads\\s)

[sales\_per\_month\_USD['Sales']](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month\_USD['Sales']](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[apply(](C:\\Users\\Bhanu\\Downloads\\s)**[lambda](C:\\Users\\Bhanu\\Downloads\\s)** [sale: '$](C:\\Users\\Bhanu\\Downloads\\s)**[{:,.2f}](C:\\Users\\Bhanu\\Downloads\\s)**['](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[format(sale))](C:\\Users\\Bhanu\\Downloads\\s)

[print('The following table displays the total sales amount (and quantities ordered) for each month:')](C:\\Users\\Bhanu\\Downloads\\s)

[sales\_per\_month\_USDsales\_per\_month\_sorted](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[sort\_values(by](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**['Sales', ascending](C:\\Users\\Bhanu\\Downloads\\s)**[=False](C:\\Users\\Bhanu\\Downloads\\s)**[)](C:\\Users\\Bhanu\\Downloads\\s)

[best\_month](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month\_sorted](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[index[0]](C:\\Users\\Bhanu\\Downloads\\s)

[best\_month](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [pd](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[to\_datetime(best\_month, format](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**['%b')](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[month\_name()](C:\\Users\\Bhanu\\Downloads\\s)

[print('The best month for sales was:', best\_month)](C:\\Users\\Bhanu\\Downloads\\s)

[maxsale](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month\_sorted['Sales']](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[iloc[0]](C:\\Users\\Bhanu\\Downloads\\s)

[print('The total sales amount earned that month was: $](C:\\Users\\Bhanu\\Downloads\\s)**[{:,.2f}](C:\\Users\\Bhanu\\Downloads\\s)**['](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[format(maxsale))](C:\\Users\\Bhanu\\Downloads\\s)

[months](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[index](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[values](C:\\Users\\Bhanu\\Downloads\\s)

[sales](C:\\Users\\Bhanu\\Downloads\\s) **[=](C:\\Users\\Bhanu\\Downloads\\s)** [sales\_per\_month['Sales']](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[figure(figsize](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[(12,7))](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[bar(months, sales,color](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**['#407bbf', linewidth](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[1,edgecolor](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**['k')](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[title('Sales Amount Per Month', fontsize](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[1](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[xlabel('Month', fontsize](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[13)](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[ylabel('Sales Amount in USD ($)', fontsize](C:\\Users\\Bhanu\\Downloads\\s)**[=](C:\\Users\\Bhanu\\Downloads\\s)**[13)](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[gcf()](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[axes[0]](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[yaxis](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[get\_major\_formatter()](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[set\_scientific(](C:\\Users\\Bhanu\\Downloads\\s)**[False](C:\\Users\\Bhanu\\Downloads\\s)**[)](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[gcf()](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[axes[0]](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[yaxis](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[set\_major\_formatter(mpl](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[ticker](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[StrMethodFormatter('$](C:\\Users\\Bhanu\\Downloads\\s)**[{x:,.0f}](C:\\Users\\Bhanu\\Downloads\\s)**['))](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[tight\_layout()](C:\\Users\\Bhanu\\Downloads\\s)

[plt](C:\\Users\\Bhanu\\Downloads\\s)**[.](C:\\Users\\Bhanu\\Downloads\\s)**[show()](C:\\Users\\Bhanu\\Downloads\\s)

**Question 2: Which city sold the most products?**

To compare cities, first we'll have to extract the city corresponding to each order from the 'Purchase Address' coloumn and store them in a  
separate coloumn 'City'. Thereafter we can group the data by city and calculate the total sum of sales for each city separately.

**[def](C:\\Users\\Bhanu\\Downloads\\x)** [get\_city\_state(address):](C:\\Users\\Bhanu\\Downloads\\x)

[address](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [address](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[split(', ')](C:\\Users\\Bhanu\\Downloads\\x)

[city](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [address[1]](C:\\Users\\Bhanu\\Downloads\\x)

[state](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [address[2][0:2]](C:\\Users\\Bhanu\\Downloads\\x)

**[return](C:\\Users\\Bhanu\\Downloads\\x)** ["](C:\\Users\\Bhanu\\Downloads\\x)**[{}](C:\\Users\\Bhanu\\Downloads\\x)** [(](C:\\Users\\Bhanu\\Downloads\\x)**[{}](C:\\Users\\Bhanu\\Downloads\\x)**[)"](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[format(city, state)](C:\\Users\\Bhanu\\Downloads\\x)

[city\_col](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [df['Purchase Address']](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[apply(](C:\\Users\\Bhanu\\Downloads\\x)**[lambda](C:\\Users\\Bhanu\\Downloads\\x)** [add: get\_city\_state(add))](C:\\Users\\Bhanu\\Downloads\\x)

[df](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[insert(loc](C:\\Users\\Bhanu\\Downloads\\x)**[=](C:\\Users\\Bhanu\\Downloads\\x)**[8, column](C:\\Users\\Bhanu\\Downloads\\x)**[=](C:\\Users\\Bhanu\\Downloads\\x)**['City', value](C:\\Users\\Bhanu\\Downloads\\x)**[=](C:\\Users\\Bhanu\\Downloads\\x)**[city\_col)](C:\\Users\\Bhanu\\Downloads\\x)

[sales\_per\_city](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [df](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[groupby(['City'])](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[sum()['Sales']](C:\\Users\\Bhanu\\Downloads\\x)

[sales\_per\_city\_USD](C:\\Users\\Bhanu\\Downloads\\x) **[=](C:\\Users\\Bhanu\\Downloads\\x)** [sales\_per\_city](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[apply(](C:\\Users\\Bhanu\\Downloads\\x)**[lambda](C:\\Users\\Bhanu\\Downloads\\x)** [sale: '$](C:\\Users\\Bhanu\\Downloads\\x)**[{:,.2f}](C:\\Users\\Bhanu\\Downloads\\x)**['](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[format(sale))](C:\\Users\\Bhanu\\Downloads\\x)**[.](C:\\Users\\Bhanu\\Downloads\\x)**[to\_frame(name](C:\\Users\\Bhanu\\Downloads\\x)**[=](C:\\Users\\Bhanu\\Downloads\\x)**['Total Sales Amount')](C:\\Users\\Bhanu\\Downloads\\x)

[print('The following table displays the total sales amount for each city:')](C:\\Users\\Bhanu\\Downloads\\x)

[sales\_per\_city\_USD](C:\\Users\\Bhanu\\Downloads\\x)

sales\_per\_city\_sorted **=** sales\_per\_city**.**sort\_values(ascending**=False**)

best\_city **=** sales\_per\_city\_sorted**.**index[0]

print('The city that sold the most products is:', best\_city)

cities **=** sales\_per\_city**.**index**.**values

plt**.**figure(figsize**=**(10,7))

plt**.**bar(cities,sales\_per\_city,color**=**'#44749d',width**=**0.6,linewidth**=**1,edgecolor**=**'k')

plt**.**title('Sales Amount Per City', fontsize**=**15)

plt**.**xlabel('City', fontsize**=**13)

plt**.**ylabel('Sales Amount in USD ($)', fontsize**=**13)

plt**.**xticks(rotation**=**60)

plt**.**gcf()**.**axes[0]**.**yaxis**.**get\_major\_formatter()**.**set\_scientific(**False**)

plt**.**gcf()**.**axes[0]**.**yaxis**.**set\_major\_formatter(mpl**.**ticker**.**StrMethodFormatter('$**{x:,.0f}**'))

plt**.**tight\_layout()

plt.show()

**Question 3: Which product sold the most? And why do you think it sold the most?**

To answer this question, we would have to group the data based on product purchases and then calculate the total amount of quantities ordered for each product to determine which one sold the most amount of quantities.

[products\_sold](C:\\Users\\Bhanu\\Downloads\\p) **[=](C:\\Users\\Bhanu\\Downloads\\p)** [df](C:\\Users\\Bhanu\\Downloads\\p)**[.](C:\\Users\\Bhanu\\Downloads\\p)**[groupby(['Product'])](C:\\Users\\Bhanu\\Downloads\\p)**[.](C:\\Users\\Bhanu\\Downloads\\p)**[sum()['Quantity Ordered']](C:\\Users\\Bhanu\\Downloads\\p)

[products\_sold](C:\\Users\\Bhanu\\Downloads\\p) **[=](C:\\Users\\Bhanu\\Downloads\\p)** [products\_sold](C:\\Users\\Bhanu\\Downloads\\p)**[.](C:\\Users\\Bhanu\\Downloads\\p)**[sort\_values(ascending](C:\\Users\\Bhanu\\Downloads\\p)**[=False](C:\\Users\\Bhanu\\Downloads\\p)**[)](C:\\Users\\Bhanu\\Downloads\\p)

[most\_sold\_pro`duct](C:\\Users\\Bhanu\\Downloads\\p) **[=](C:\\Users\\Bhanu\\Downloads\\p)** [products\_sold](C:\\Users\\Bhanu\\Downloads\\p)**[.](C:\\Users\\Bhanu\\Downloads\\p)**[index[0]](C:\\Users\\Bhanu\\Downloads\\p)

[print('The product that was sold the most is:', most\_sold\_product)](C:\\Users\\Bhanu\\Downloads\\p)

**Question 4: Is there a relationship between how much a product costs and the quantity sold?**

One way to answer this question is to create a dual-axis line chart displaying the prices of each product and the quantity sold in order to compare them.  
First, we will have to create two groups, the first representing the prices of each product, the second representing the total quantity sold for each product.

[products\_quantity](C:\\Users\\Bhanu\\Downloads\\n) **[=](C:\\Users\\Bhanu\\Downloads\\n)** [df](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[groupby(['Product'])](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[sum()['Quantity Ordered']](C:\\Users\\Bhanu\\Downloads\\n)

[products](C:\\Users\\Bhanu\\Downloads\\n) **[=](C:\\Users\\Bhanu\\Downloads\\n)** [products\_quantity](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[index](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[values](C:\\Users\\Bhanu\\Downloads\\n)

[products\_prices](C:\\Users\\Bhanu\\Downloads\\n) **[=](C:\\Users\\Bhanu\\Downloads\\n)** [df](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[groupby(['Product'])['Price Each']](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[apply(](C:\\Users\\Bhanu\\Downloads\\n)**[lambda](C:\\Users\\Bhanu\\Downloads\\n)** [price: float(np](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[unique(price)))](C:\\Users\\Bhanu\\Downloads\\n)

[fig, ax1](C:\\Users\\Bhanu\\Downloads\\n) **[=](C:\\Users\\Bhanu\\Downloads\\n)** [plt](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[subplots(figsize](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[(12,7))](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[plot(products,products\_quantity,marker](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['o',c](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['#407bbf',lw](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[2,](C:\\Users\\Bhanu\\Downloads\\n)

[label](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['Quantities')](C:\\Users\\Bhanu\\Downloads\\n)

[ax2](C:\\Users\\Bhanu\\Downloads\\n) **[=](C:\\Users\\Bhanu\\Downloads\\n)** [ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[twinx()](C:\\Users\\Bhanu\\Downloads\\n)

[ax2](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[plot(products, products\_prices,marker](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['o',c](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['#bf4040',lw](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[2,label](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['Prices')](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_title('The Relationship Between Product Price and Quantity Sold',fontsize](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[15)](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_xlabel('Product Name', fontsize](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[13)](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_ylabel('Total Quantity Sold', fontsize](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[12, color](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['#407bbf')](C:\\Users\\Bhanu\\Downloads\\n)

[ax2](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_ylabel('Prices in USD ($)', fontsize](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**[12, color](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['#cc3333')](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_xticklabels(products, rotation](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['vertical')](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[yaxis](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_major\_formatter(mpl](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[ticker](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[StrMethodFormatter('](C:\\Users\\Bhanu\\Downloads\\n)**[{x:,.0f}](C:\\Users\\Bhanu\\Downloads\\n)**['))](C:\\Users\\Bhanu\\Downloads\\n)

[ax2](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[yaxis](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[set\_major\_formatter(mpl](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[ticker](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[StrMethodFormatter('$](C:\\Users\\Bhanu\\Downloads\\n)**[{x:,.0f}](C:\\Users\\Bhanu\\Downloads\\n)**['))](C:\\Users\\Bhanu\\Downloads\\n)

[ax1](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[legend(loc](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['upper left')](C:\\Users\\Bhanu\\Downloads\\n)

[x2](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[legend(loc](C:\\Users\\Bhanu\\Downloads\\n)**[=](C:\\Users\\Bhanu\\Downloads\\n)**['upper right')](C:\\Users\\Bhanu\\Downloads\\n)

[plt](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[grid()](C:\\Users\\Bhanu\\Downloads\\n)

[plt](C:\\Users\\Bhanu\\Downloads\\n)**[.](C:\\Users\\Bhanu\\Downloads\\n)**[show()](C:\\Users\\Bhanu\\Downloads\\n)

**Question 5: Which products are most often sold together?**

For starters, we can filter data based on whether there are duplicates in the 'Order ID' coloumn, indicating that the same person made multiple product purchases, and then join the multiple products sold together and count the instances of particular products being sold together in order to extract those that most often ordered together.

[order\_filter](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [df['Order ID']](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[duplicated(keep](C:\\Users\\Bhanu\\Downloads\\a)**[=False](C:\\Users\\Bhanu\\Downloads\\a)**[)](C:\\Users\\Bhanu\\Downloads\\a)

[df\_multiple\_orders](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [df[order\_filter][['Order ID', 'Product']]](C:\\Users\\Bhanu\\Downloads\\a)

[orders\_per\_person](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [df\_multiple\_orders](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[groupby(['Order ID'])['Product']](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[transform(](C:\\Users\\Bhanu\\Downloads\\a)**[lambda](C:\\Users\\Bhanu\\Downloads\\a)** [product: ", "](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[join(product))](C:\\Users\\Bhanu\\Downloads\\a)

[df\_orders\_per\_person](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [orders\_per\_person](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[to\_frame(name](C:\\Users\\Bhanu\\Downloads\\a)**[=](C:\\Users\\Bhanu\\Downloads\\a)**['Products Sold Together')](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[reset\_index(drop](C:\\Users\\Bhanu\\Downloads\\a)**[=True](C:\\Users\\Bhanu\\Downloads\\a)**[)](C:\\Users\\Bhanu\\Downloads\\a)

[orders\_frequency](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [orders\_per\_person](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[value\_counts()](C:\\Users\\Bhanu\\Downloads\\a)

[df\_orders\_frequency](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [orders\_frequency](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[to\_frame(name](C:\\Users\\Bhanu\\Downloads\\a)**[=](C:\\Users\\Bhanu\\Downloads\\a)**['Frequency of products sold together')](C:\\Users\\Bhanu\\Downloads\\a)

[df\_orders\_frequency](C:\\Users\\Bhanu\\Downloads\\a)

[most\_sold\_together](C:\\Users\\Bhanu\\Downloads\\a) **[=](C:\\Users\\Bhanu\\Downloads\\a)** [orders\_frequency](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[index[0]](C:\\Users\\Bhanu\\Downloads\\a)

[print('The two products sold together the most often are:](C:\\Users\\Bhanu\\Downloads\\a) **[{}](C:\\Users\\Bhanu\\Downloads\\a)**['](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[format(' and '](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[join(most\_sold\_together](C:\\Users\\Bhanu\\Downloads\\a)**[.](C:\\Users\\Bhanu\\Downloads\\a)**[split(', '))))](C:\\Users\\Bhanu\\Downloads\\a)

**Question 6: Which time of the day should we display advertisments to maximize the likelihood of customer's purchasing products?**

One way to answer this question is to extract the time of the day from the 'Order Date' coloumn, and then grouping the data based on the time of  
the day (hour) in which a product was purchased to determine which times are associated with the most product purchases.

[Time\_col](E:\\a) **[=](E:\\a)** [pd](E:\\a)**[.](E:\\a)**[to\_datetime(df['Order Date'], format](E:\\a)**[=](E:\\a)**['](E:\\a)**[%d](E:\\a)**[/%m/%y %H:%M')](E:\\a)**[.](E:\\a)**[dt](E:\\a)**[.](E:\\a)**[strftime('%I %p')](E:\\a)

[df](E:\\a)**[.](E:\\a)**[insert(loc](E:\\a)**[=](E:\\a)**[6, column](E:\\a)**[=](E:\\a)**['Time of Purchase', value](E:\\a)**[=](E:\\a)**[Time\_col)](E:\\a)

[purchases\_per\_hour](E:\\a) **[=](E:\\a)** [df](E:\\a)**[.](E:\\a)**[groupby(['Time of Purchase'])](E:\\a)**[.](E:\\a)**[sum()['Quantity Ordered']](E:\\a)

[time\_sort\_order](E:\\a) **[=](E:\\a)** [['12 AM', '01 AM', '02 AM', '03 AM', '04 AM', '05 AM', '06 AM', '07 AM', '08 AM', '09 AM', '10 AM', '11 AM', '12 PM', '01 PM', '02 PM', '03 PM', '04 PM', '05 PM', '06 PM', '07 PM', '08 PM', '09 PM', '10 PM', '11 PM']](E:\\a)

[purchases\_per\_hour](E:\\a) **[=](E:\\a)** [purchases\_per\_hour](E:\\a)**[.](E:\\a)**[reindex(time\_sort\_order)](E:\\a)

[df\_purchases\_per\_hour](E:\\a) **[=](E:\\a)** [purchases\_per\_hour](E:\\a)**[.](E:\\a)**[to\_frame(name](E:\\a)**[=](E:\\a)**['Total Quantity Sold')](E:\\a)

[print('The following table displays the total sum of quantities ordered for each hour of the day:')](E:\\a)

[df\_purchases\_per\_hour](E:\\a)

[purchases\_per\_hour\_sorted](E:\\a) **[=](E:\\a)** [purchases\_per\_hour](E:\\a)**[.](E:\\a)**[sort\_values(ascending](E:\\a)**[=False](E:\\a)**[)](E:\\a)

[best\_hour](E:\\a) **[=](E:\\a)** [purchases\_per\_hour\_sorted](E:\\a)**[.](E:\\a)**[index[0]](E:\\a)

[print('The best time of day for displaying advertisements is:', best\_hour)](E:\\a)

[time\_of\_purchase](E:\\a) **[=](E:\\a)** [purchases\_per\_hour](E:\\a)**[.](E:\\a)**[index](E:\\a)**[.](E:\\a)**[values](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[figure(figsize](E:\\a)**[=](E:\\a)**[(12,7))](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[bar(time\_of\_purchase,purchases\_per\_hour,color](E:\\a)**[=](E:\\a)**['#4169e1',linewidth](E:\\a)**[=](E:\\a)**[1,](E:\\a)

[edgecolor](E:\\a)**[=](E:\\a)**['k')](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[title('Quantities Sold Per Hour', fontsize](E:\\a)**[=](E:\\a)**[15)](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[xlabel('Time of Day', fontsize](E:\\a)**[=](E:\\a)**[13)](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[ylabel('Amount of Quantities Sold', fontsize](E:\\a)**[=](E:\\a)**[13)](E:\\a)

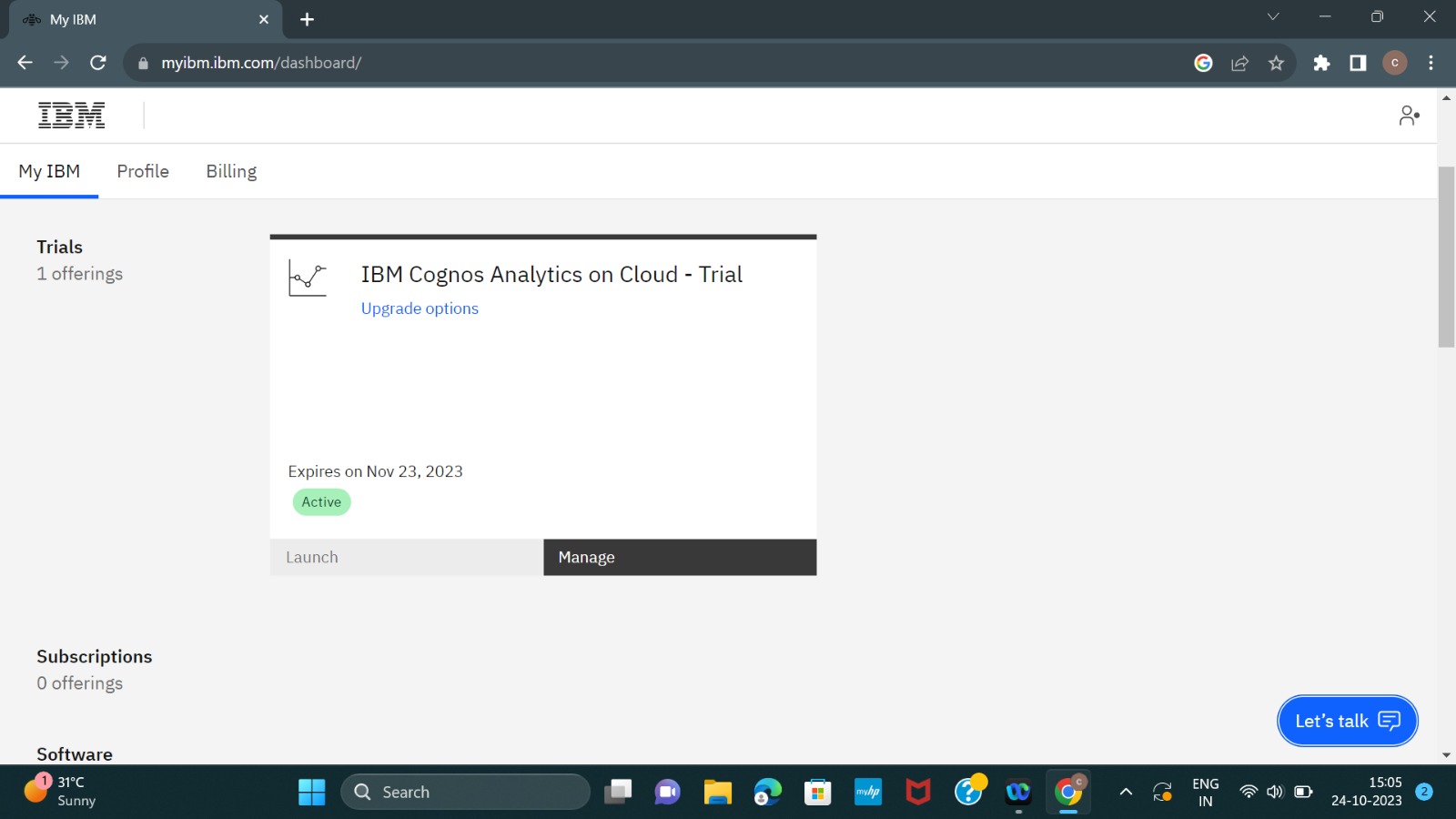
[plt](E:\\a)**[.](E:\\a)**[xticks(rotation](E:\\a)**[=](E:\\a)**[90)](E:\\a)

[plt](E:\\a)**[.](E:\\a)**[gcf()](E:\\a)**[.](E:\\a)**[axes[0]](E:\\a)**[.](E:\\a)**[yaxis](E:\\a)**[.](E:\\a)**[set\_major\_formatter(mpl](E:\\a)**[.](E:\\a)**[ticker](E:\\a)**[.](E:\\a)**[StrMethodFormatter('](E:\\a)**[{x:,.0f}](E:\\a)**['))](E:\\a)

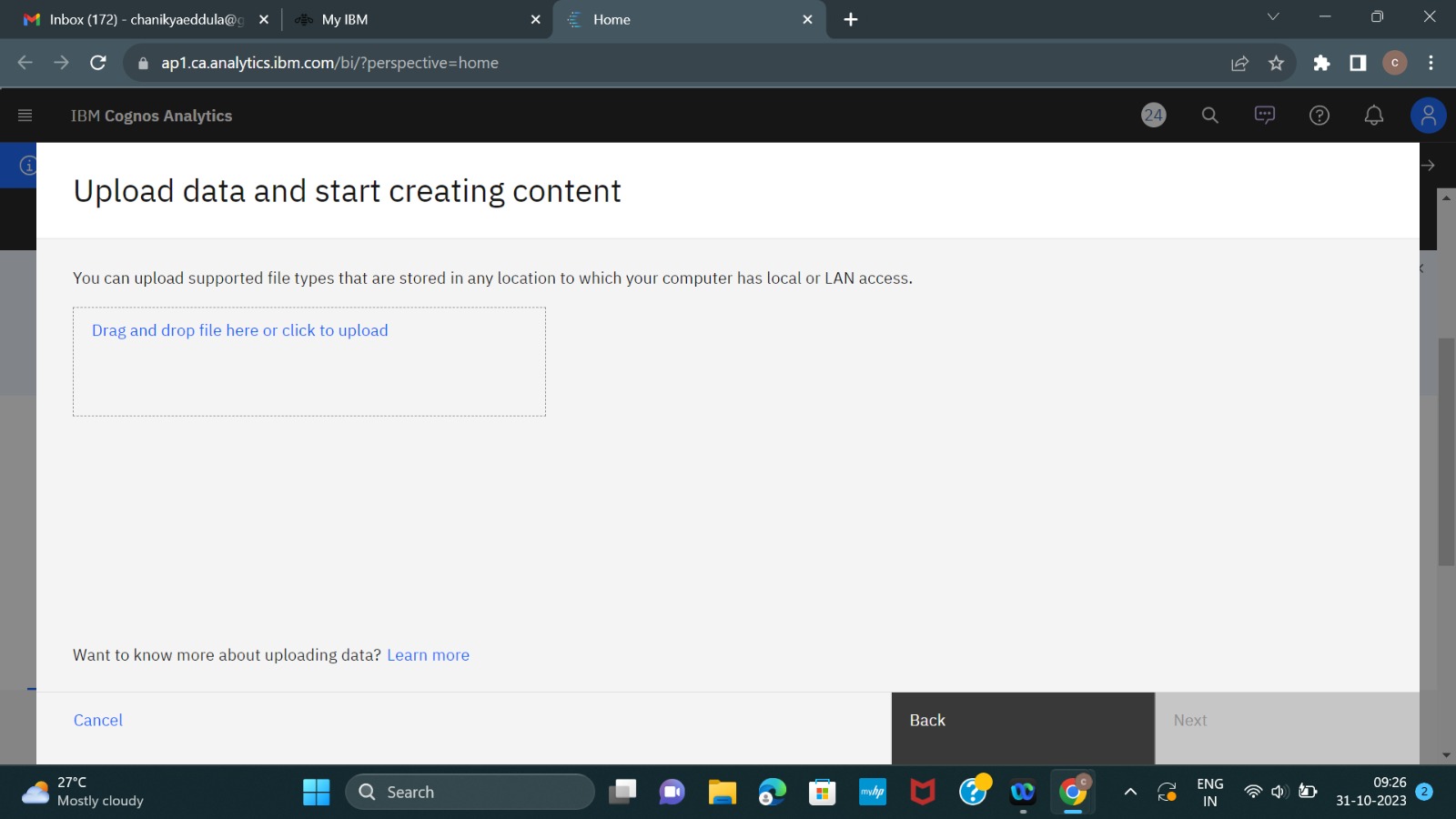
[plt](E:\\a)**[.](E:\\a)**[tight\_layout()](E:\\a)

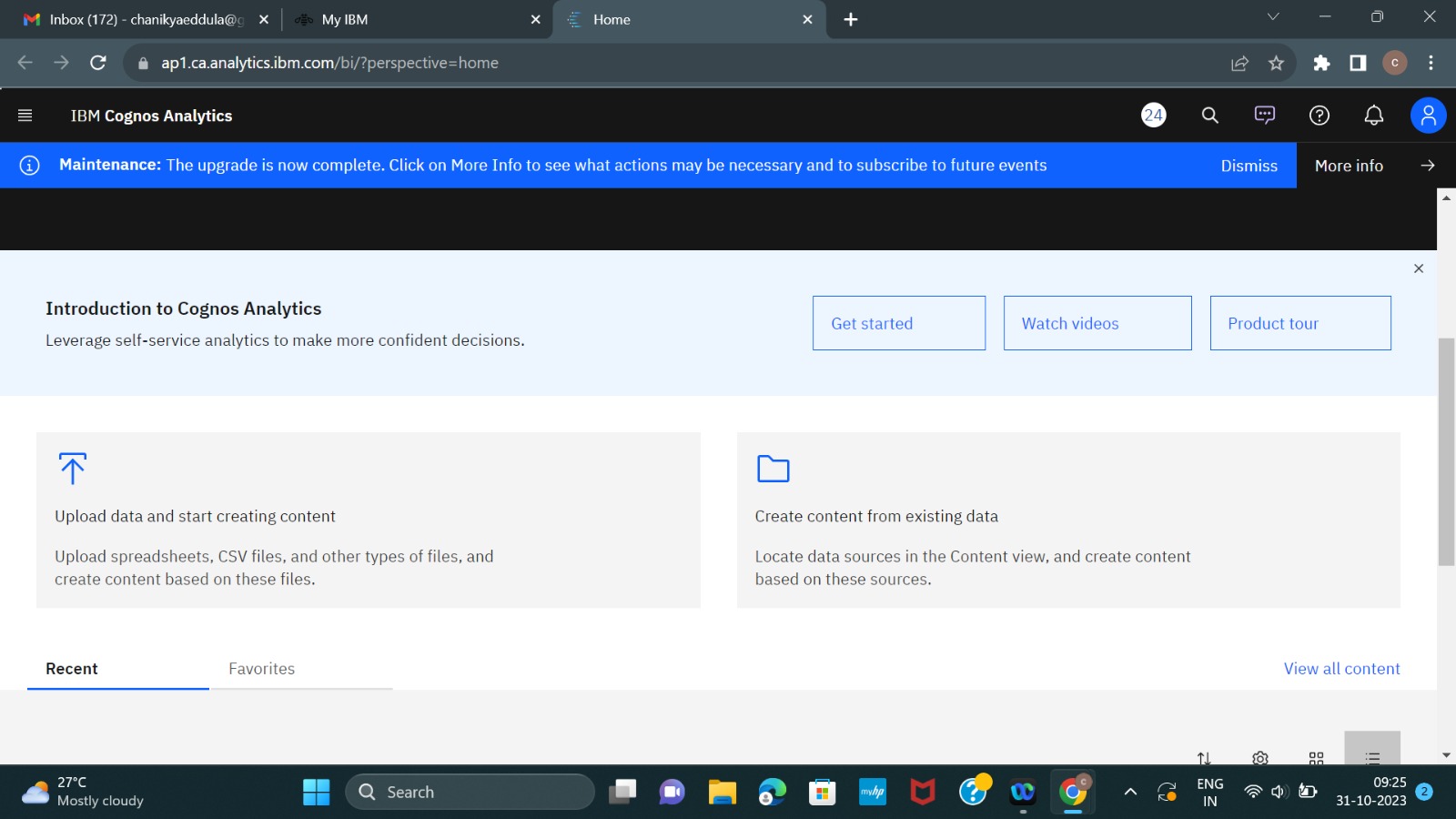
[plt](E:\\a)**[.](E:\\a)**[show()](E:\\a)

**Step1: login IBM cognos account on cloud**

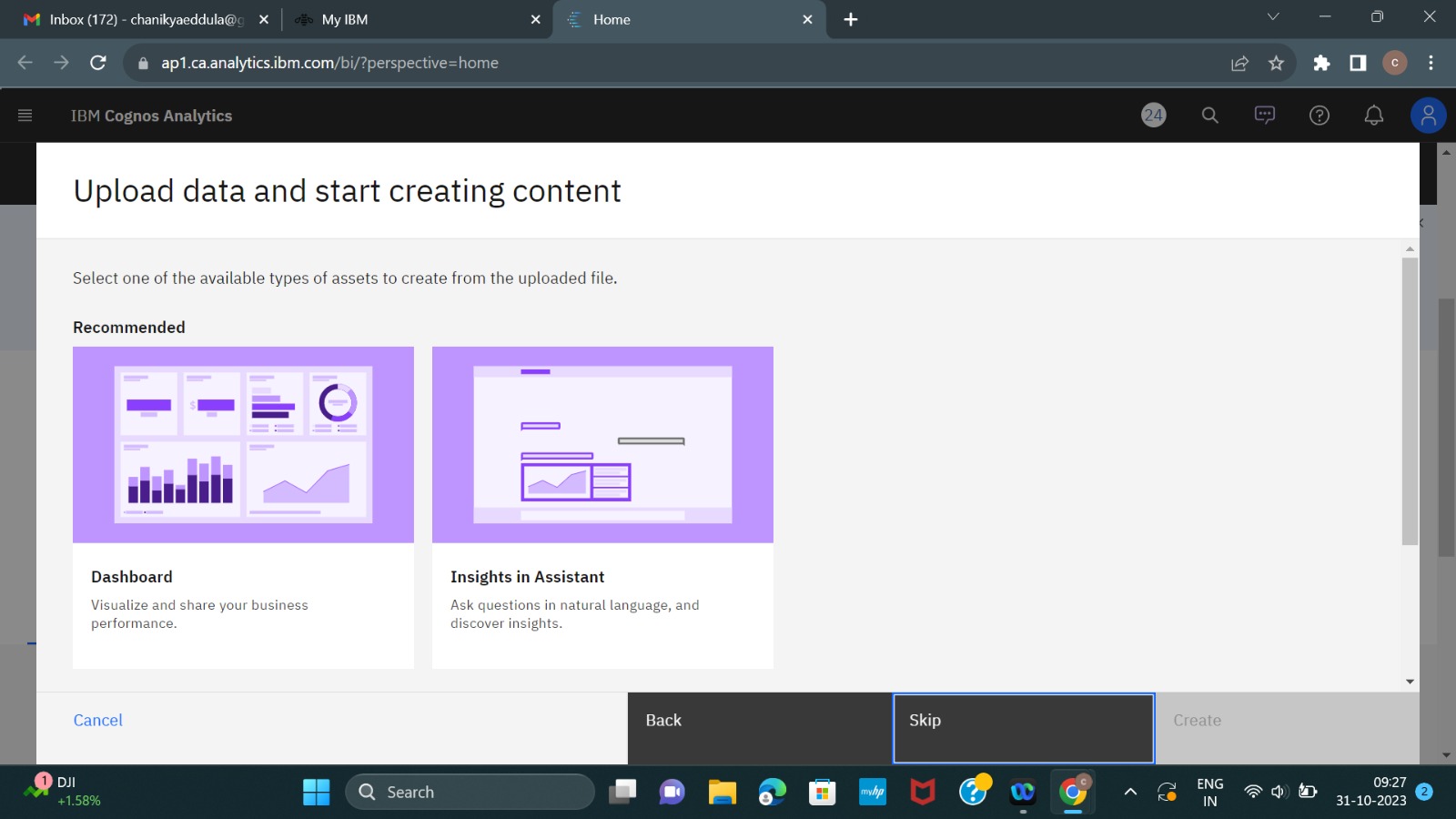


**Step 2: Upload data in csv file**

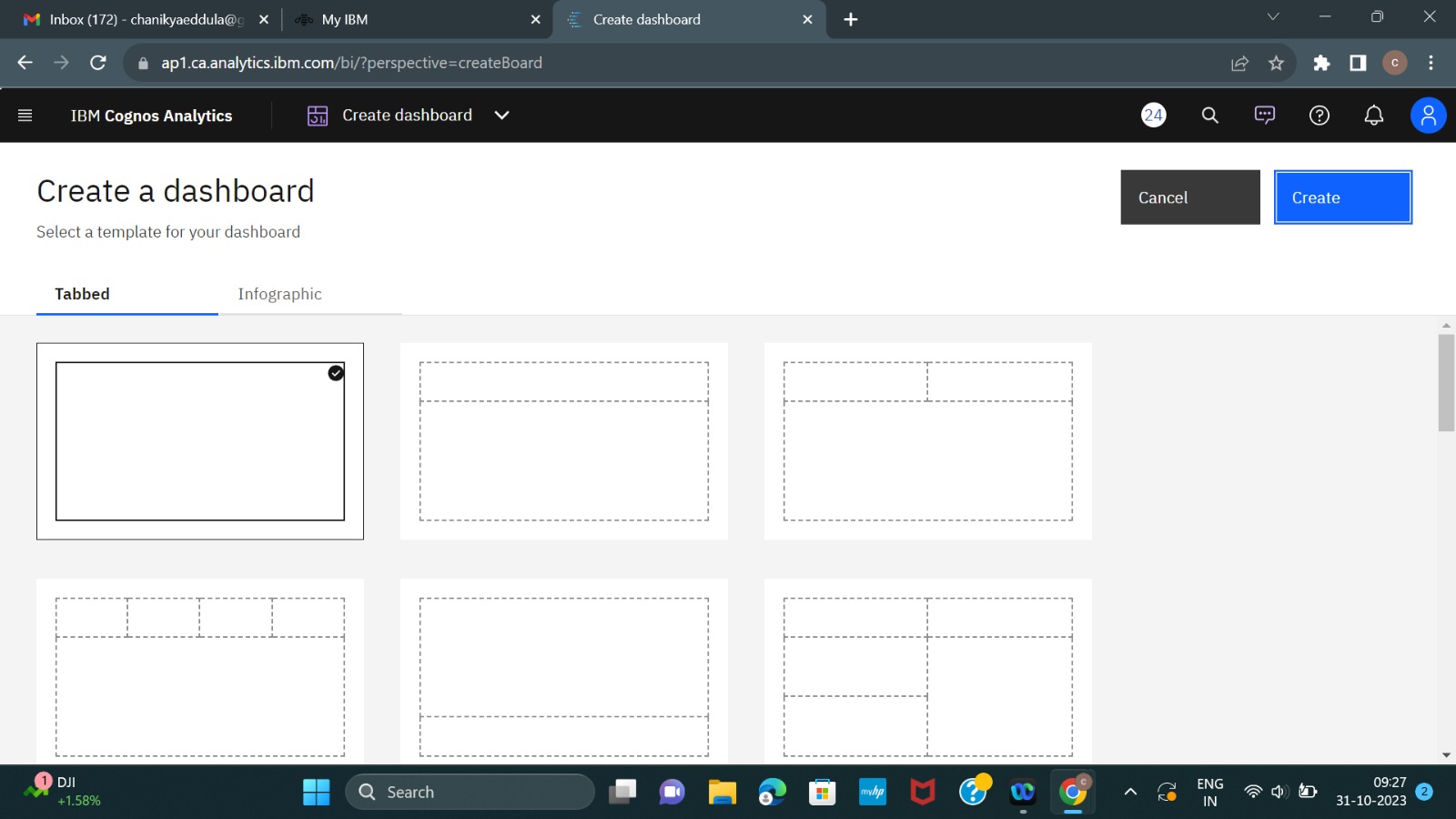
****

****

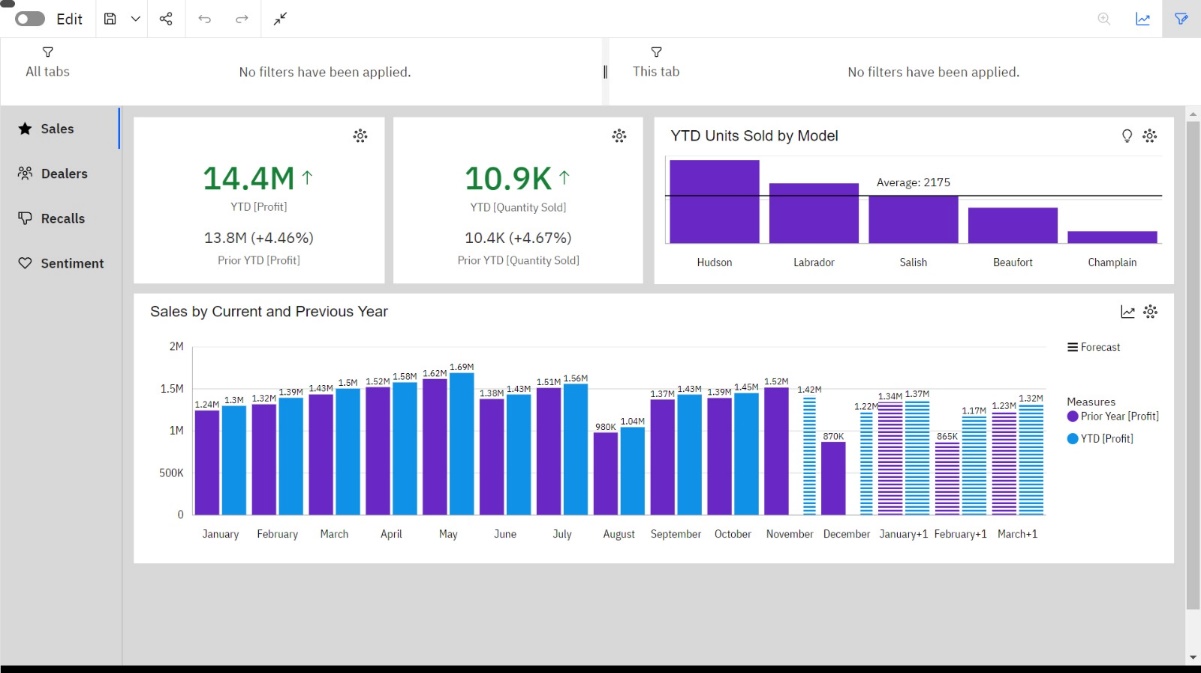
**Step 3:drag and drop the file**

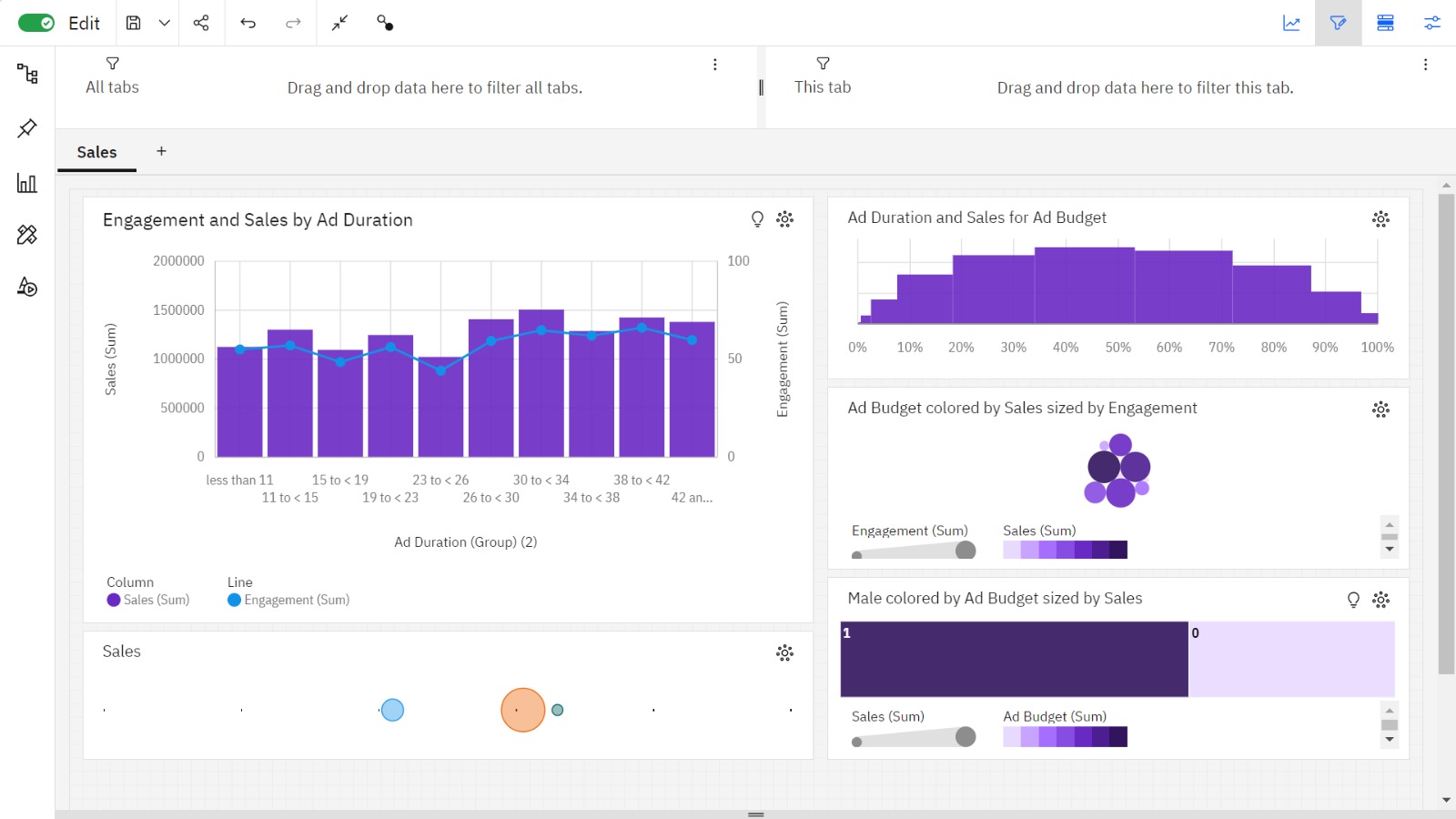
****

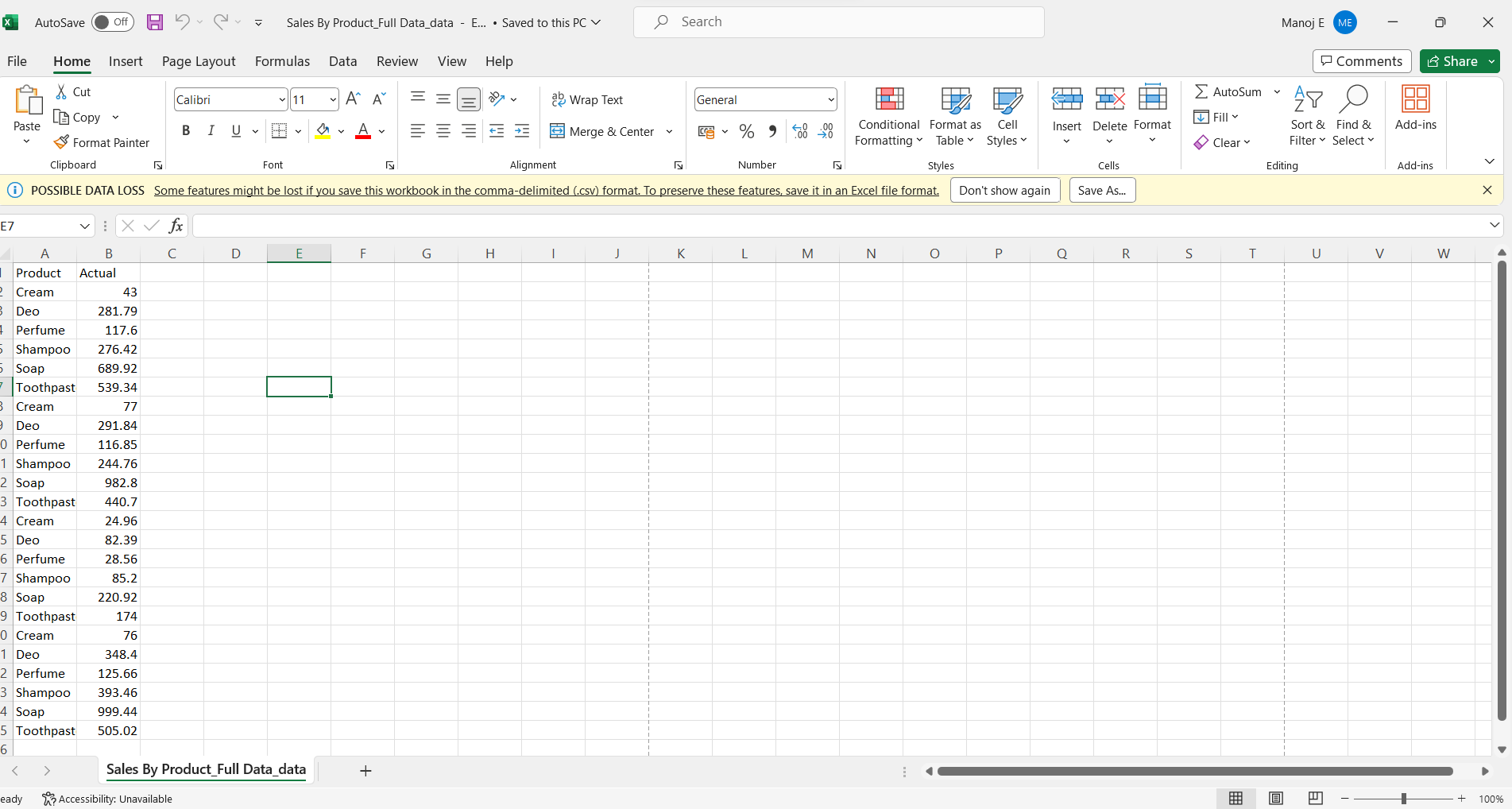
**step 4: creat a dash board template**

****

**Dash board of sales analysis of product**

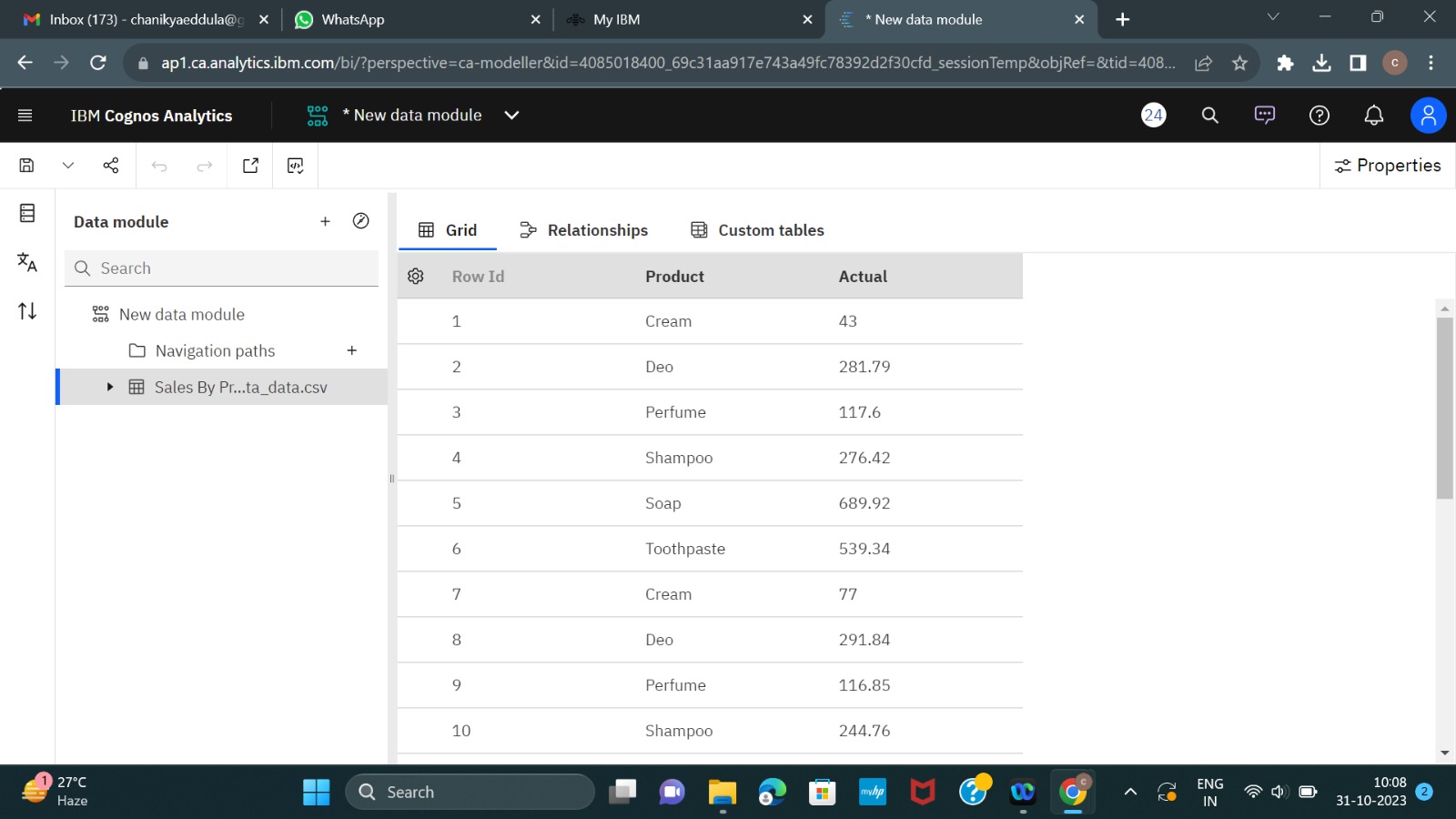
[blob](blob:https://web.whatsapp.com/9a385624-d42a-4d71-aa27-db5acd2e39f2)****

****

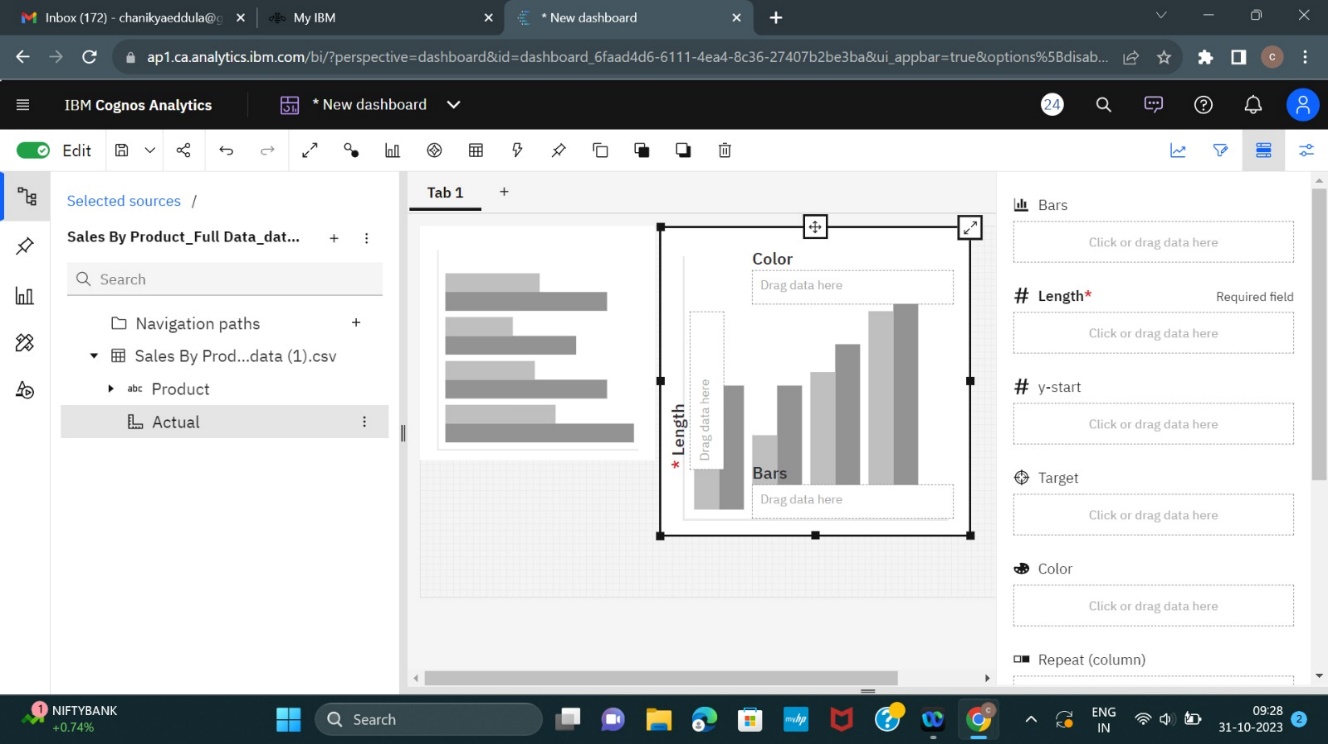
****

**step 5:here we to create visualizations of sales**

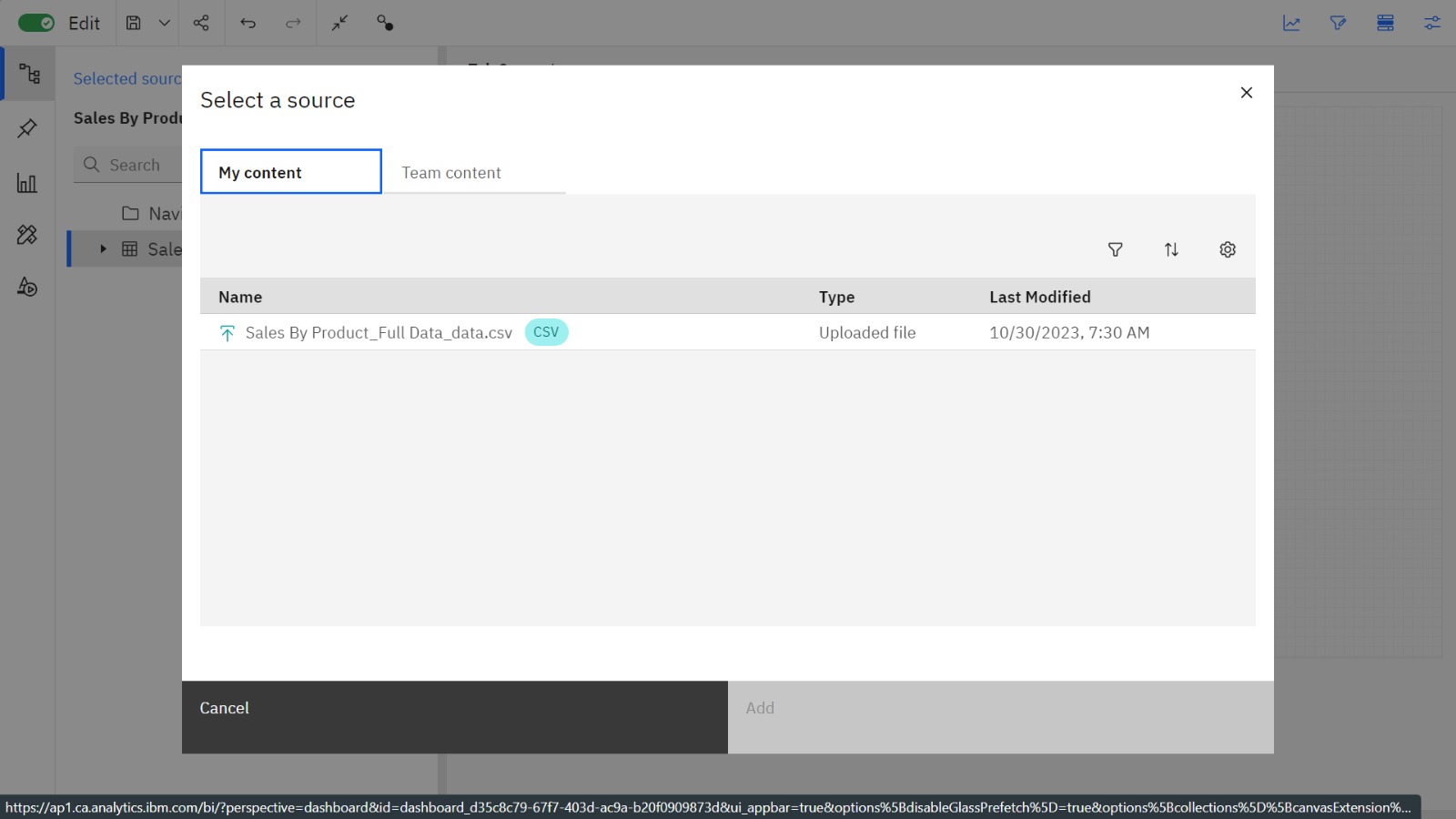
**Visualization of bar and column**

****

**Step 6:to explore data module of sales analysis**

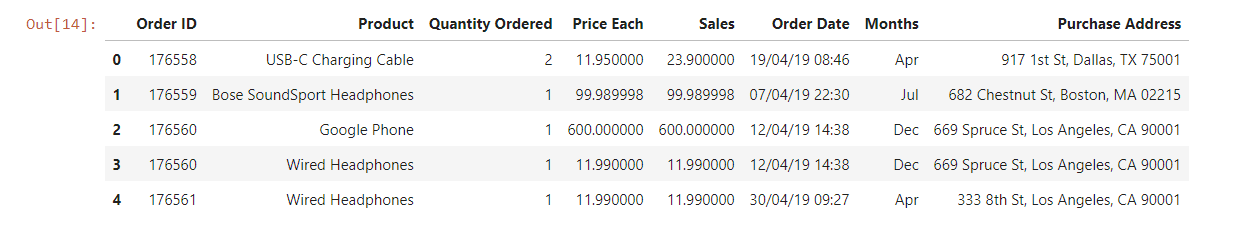
****

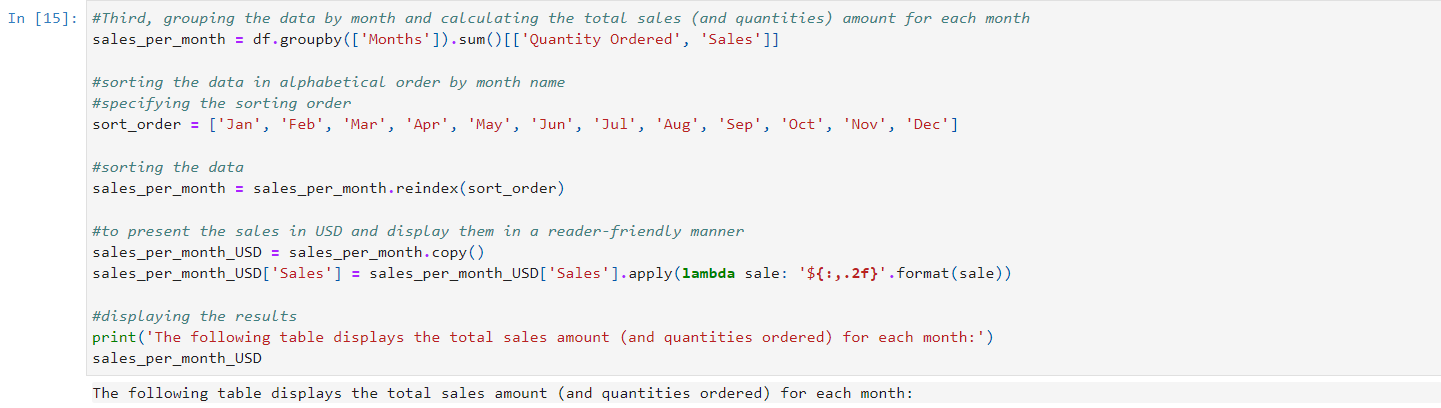
**after add the file in my contact source**

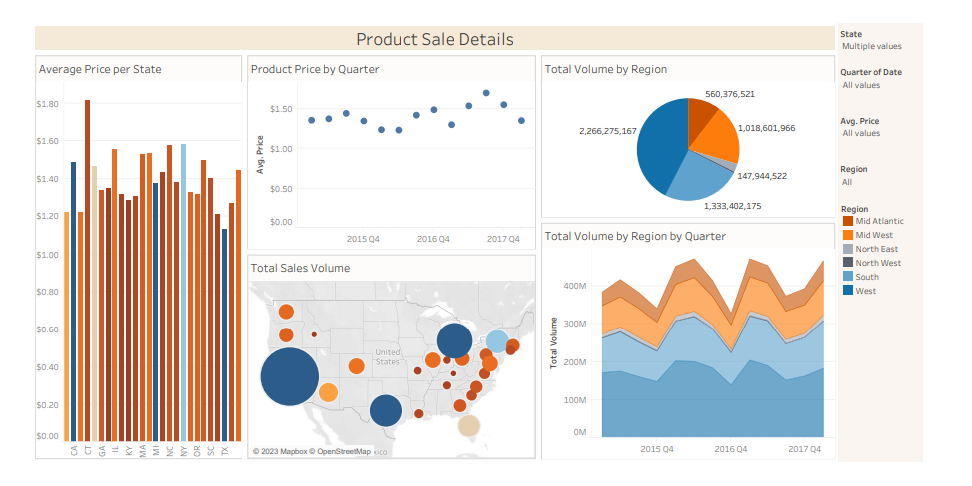
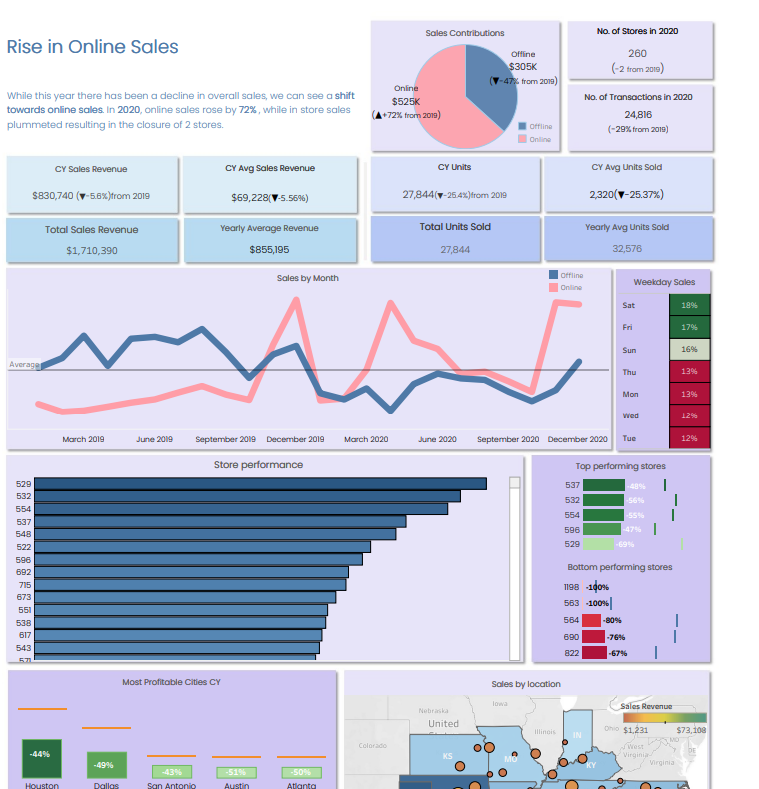
one month to the next. ****

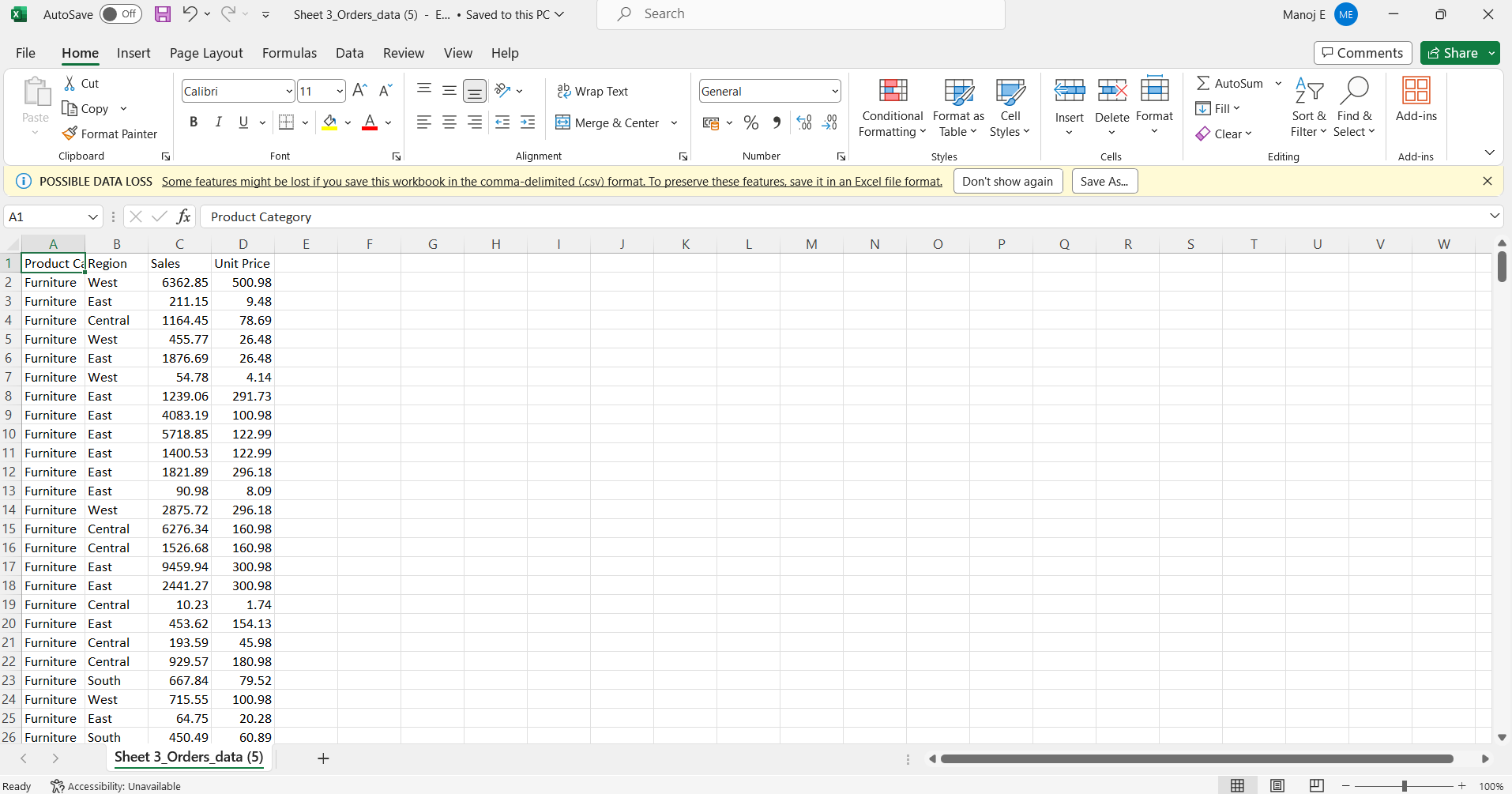
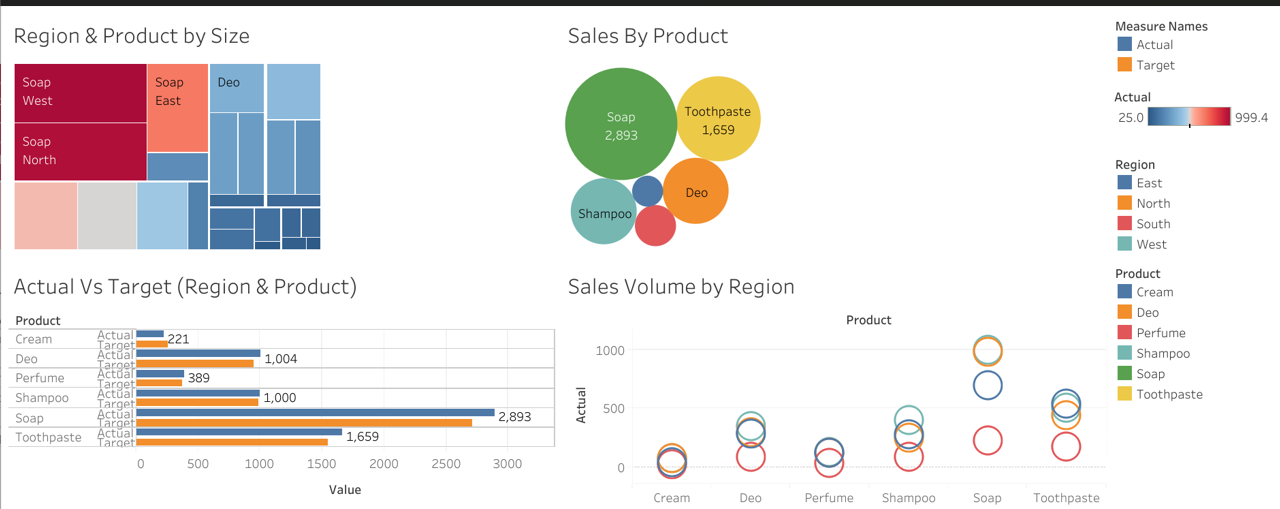
**Exploring the Data**

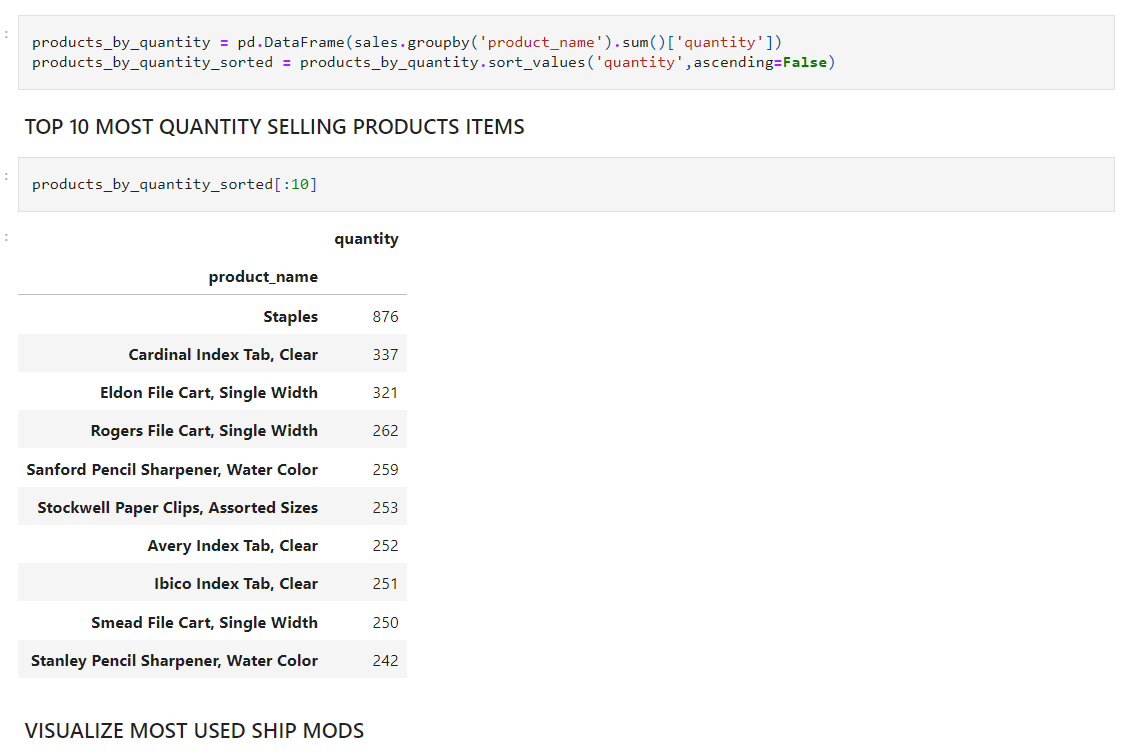
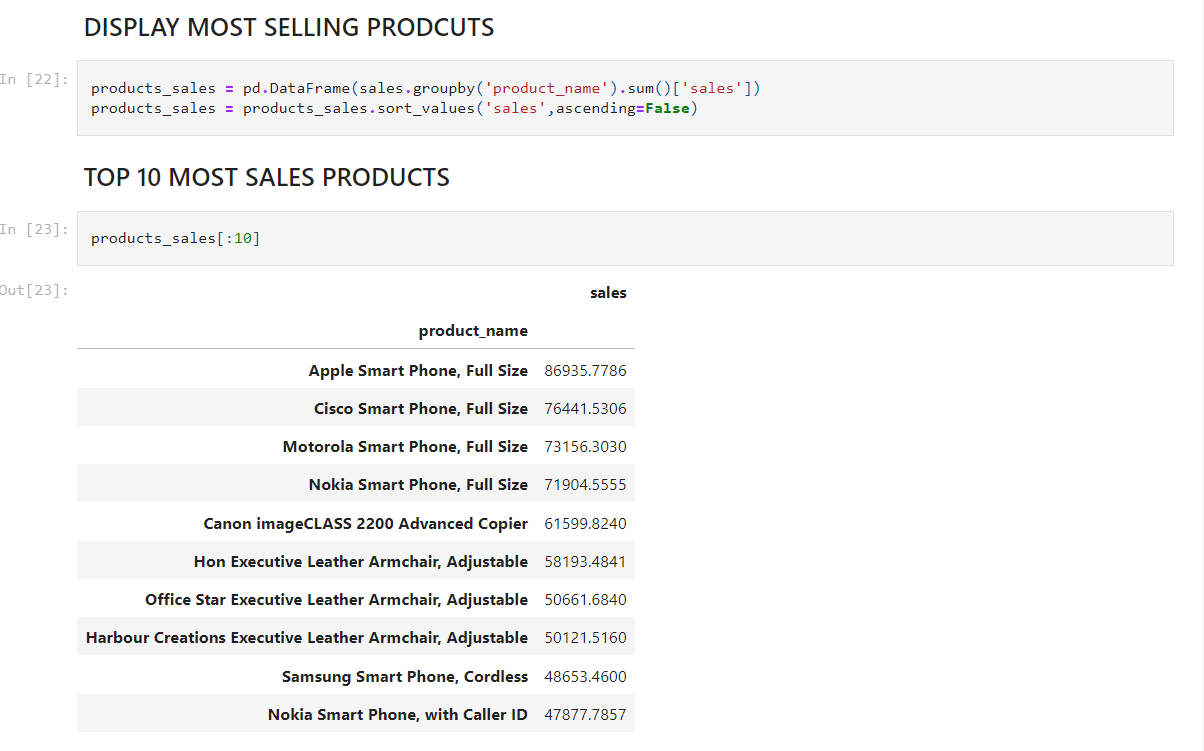
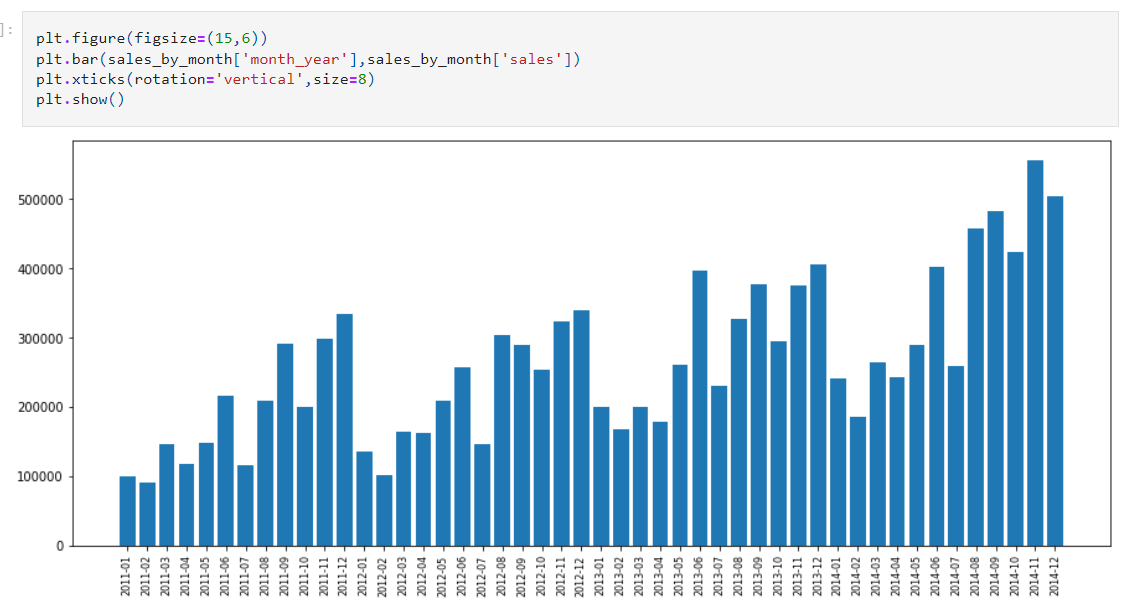
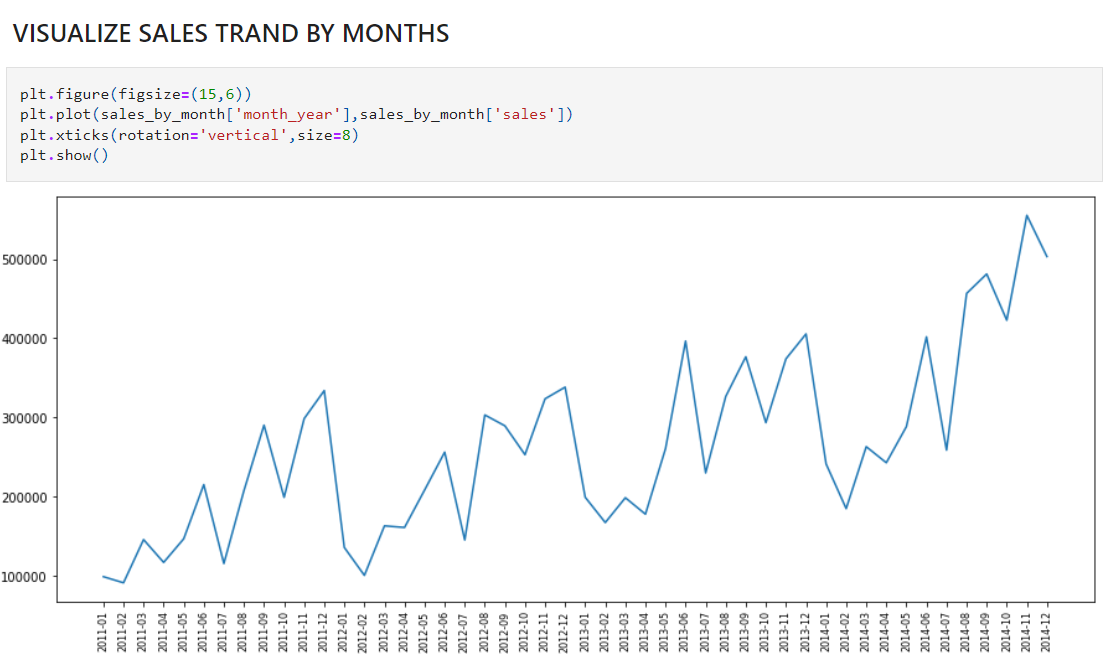
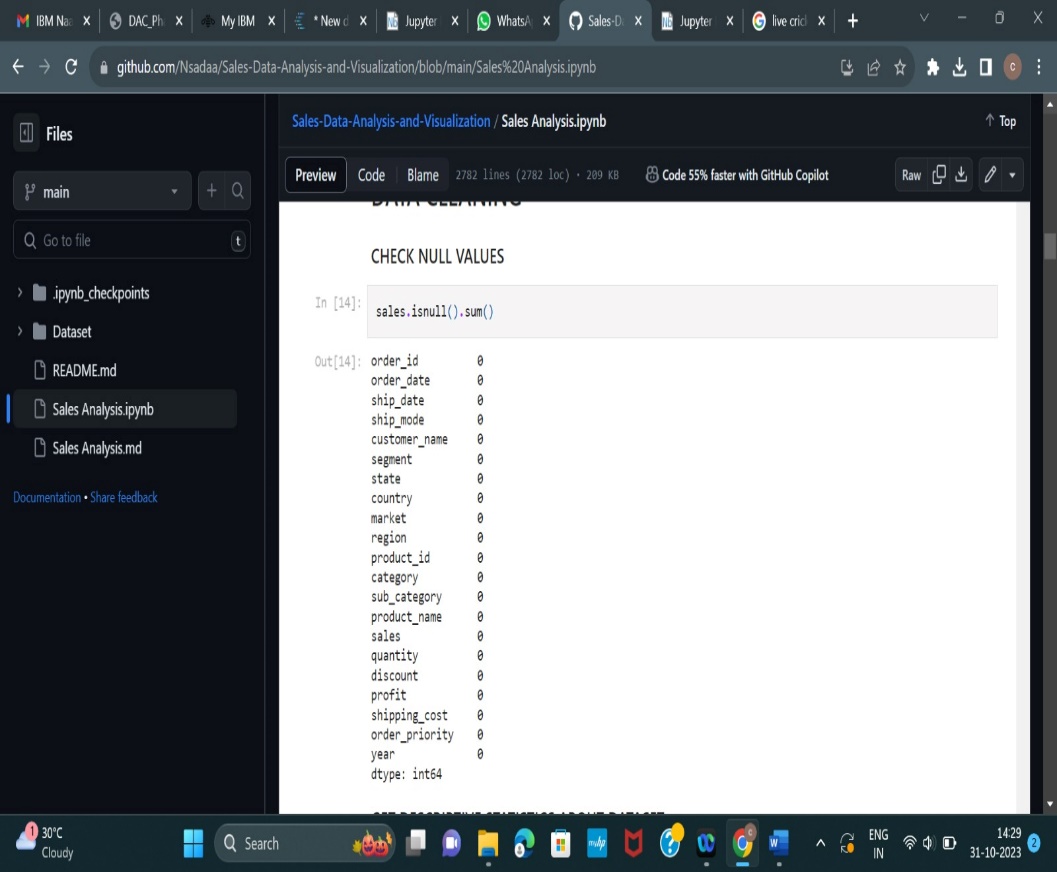
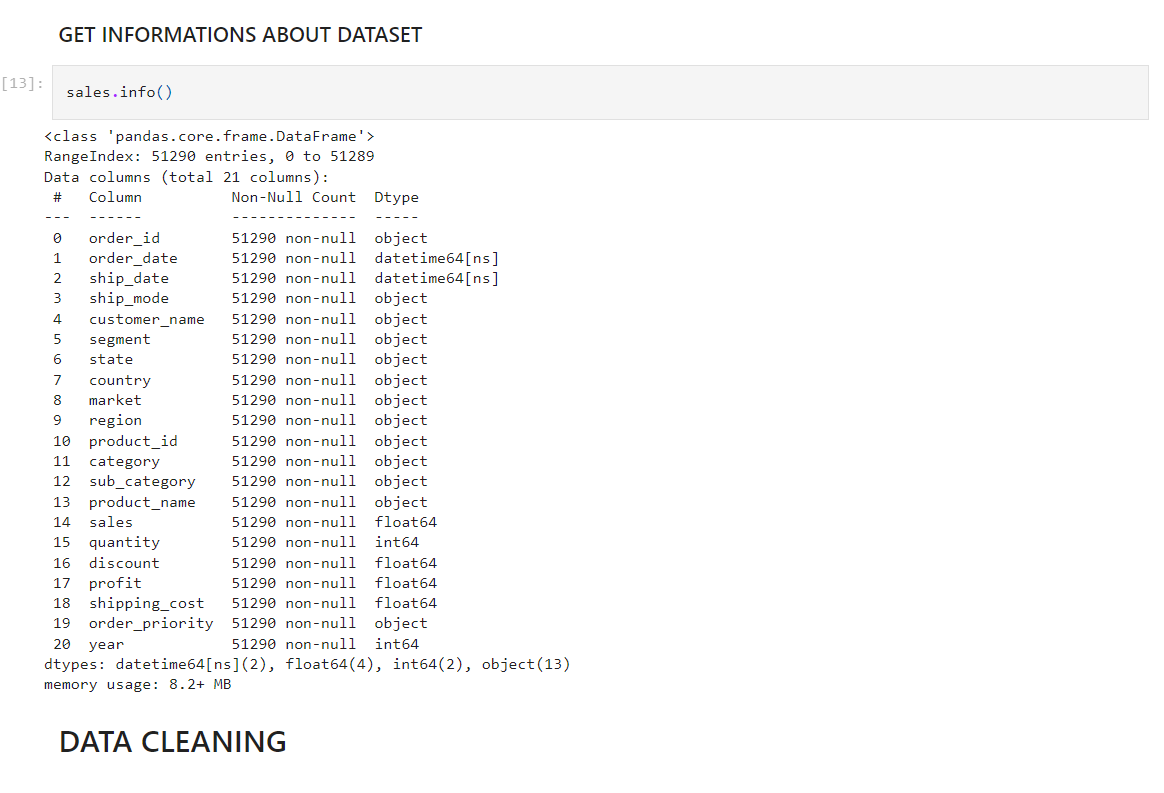
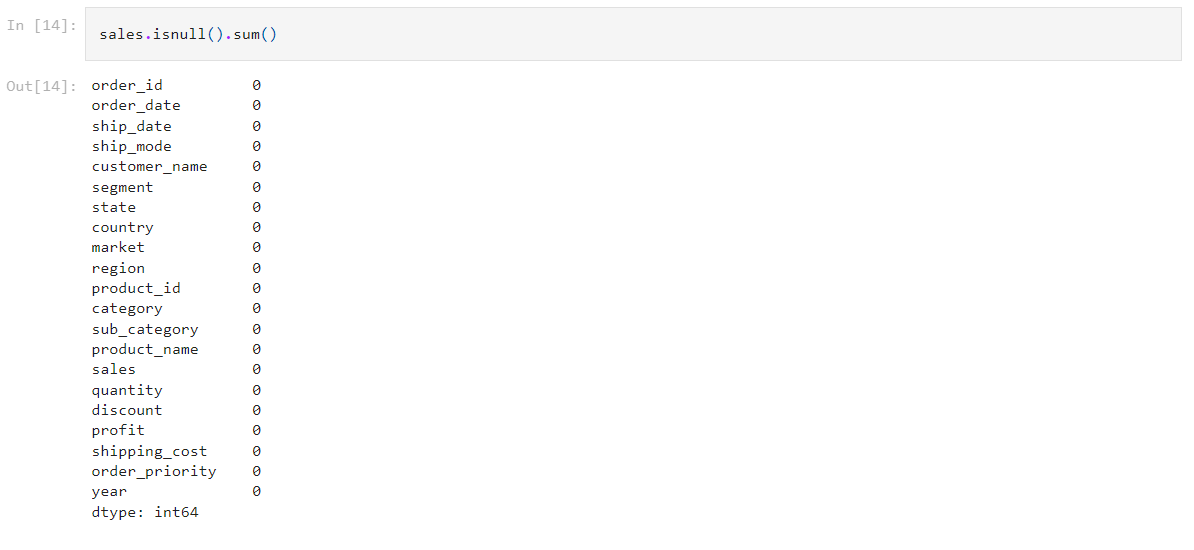
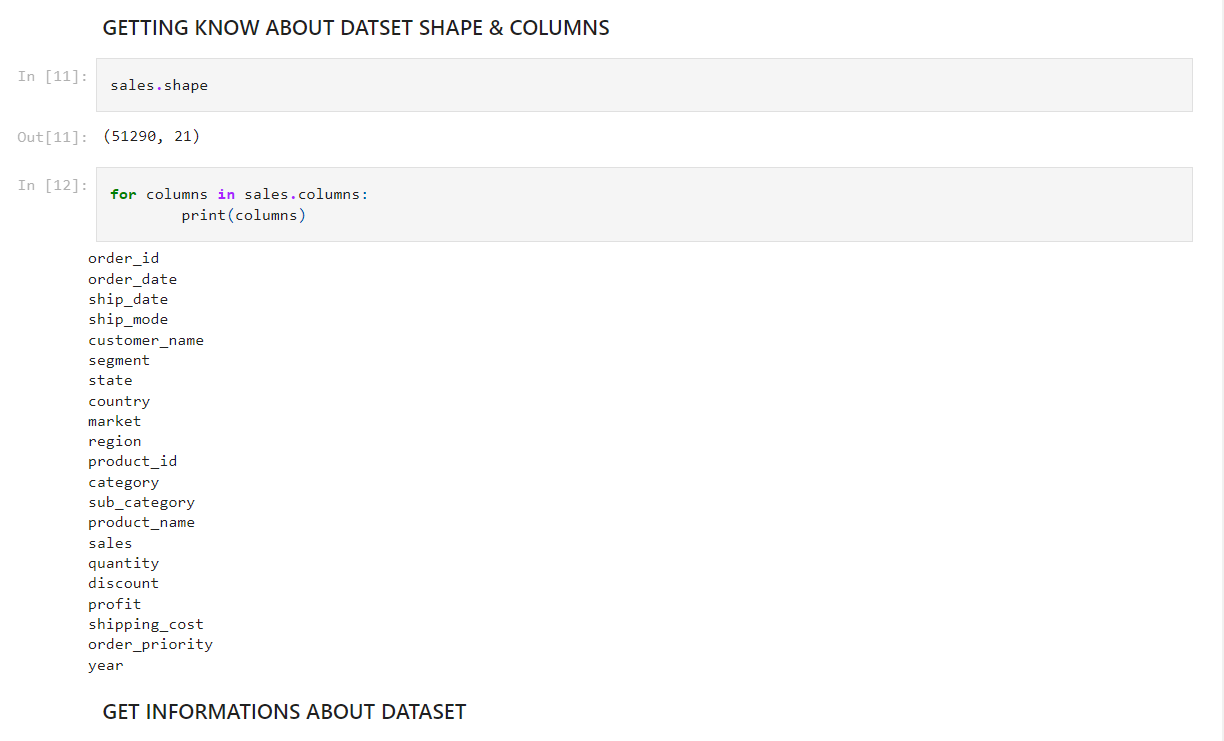
**Question 1: What was the best month for sales? How much was earned that month?**



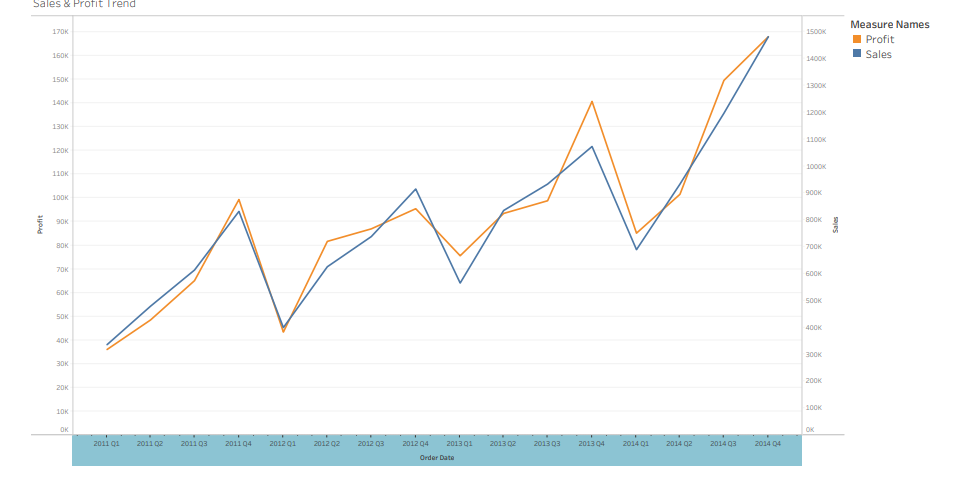


****



****

**Sales and profit trend**

****

**Conclusion:**

In conclusion, the utilization of IBM cognos for visualizing product sales data has been instrumental in providing a historic view of sales performance.The insights derived from these visualizations are pivotal for formulating data driven strategies.These conclusions can serve as a foundation for future business decisions and actions aimed at driving sales growth.These document shows how to login IBM cognos and how to create visualizations for sales and dervie for insights