Happy Diwali
"This Diwali I pray for good health for our family, our friends and the blessings of God's lifetime"

Diwali Sales Analysis - Python ✅

liceria & co.
Happy
Diwali
Wishing you and your family a Happy Diwali filled with light, love, and laughter
DIWALI SALES ANALYSIS
PYTHON
By Sarwar Ayaan Ali

▼ 🟪 Diwali Sales Analysis - Project Description

🟨 DESCRIPTION:

A company has given us their sales dataset of previous year and wants us to analyze for all the different columns and provide them a summary that can help them to improve their customers experience by analyzing sales data. Also, that can boost their sales amount or revenue.

🟨 PROVIDE A DETAILED REPORT ANSWERING THE QUESTIONS BELOW:

💡 After downloading the dataset, use VS Code or Jupyter Notebook to answer the below questions:

Tasks to Perform

- **Task 1: Most amount spent - As per Gender**
- **Task 2: Most amount spent - As per Age Group**
- **Task 3: Most amount spent - As per Marital Status**
- **Task 4: Most amount spent - As per Occupation**
- **Task 5: Most amount spent - As per Product Category**
- **Task 6: Number of Orders - As per Product ID**
- **Task 7: Number of Orders - As per State**

Use the below Steps for EDA:

- Understanding data columns and data
- Checking for missing data
- Clubbing columns with multiple categories
- Checking for outliers
- Removing outliers
- Drawing Data Summary

🟨 HOW TO DO THIS PROJECT?

- **Download the dataset:** You are supposed to download the dataset.
- **Perform Analysis**: Use VS code or Jupyter Notebook to perform your entire analysis answering the questions asked above.
- **Submit a Report:** Make a report (PDF/PPT) to be presented to the leadership team. The report should/can contain the following details:
- **Project Description**: Give a brief about your project description i.e. what is this project about, how are you going to handle the things and what are the things that you are going to find out through the project.
- **Approach**: Write a short paragraph about your approach towards the project and how you have executed it.
- **Tech-Stack Used**: Do mention the software and the version used while making the project (For Eg. Jupyter Notebook, etc) and mention the purpose of using it.
- **Insights:** Jot down the insights and the knowledge you gained while making the project.

🟨 JUDGEMENT CRITERIA:

- **Data Analytics:** Your answers must contain relevant points related to the data analysis and should have real world case scenarios.
- **Advanced Python Understanding:** Your project must have clear explanations of the processes used, eg, Jupyter notebook, functions used, formulas explained, etc.

- **Case study completion:** All the questions present must be answered completely having correct answers.
- **Insights:** You need to use your own imagination to answer the case study while improvising it as well.
- **Plagiarism:** Project submitted should not be copied from the internet or anywhere else, it should be your own work.

▼ 🟪 Diwali Sales Analysis - My Project

🟨 DESCRIPTION:

This is **My Project on Diwali Sales Analysis - Excel.** In this project, I have tried my best to explain my thoughts and insights about user analytics. I have answered all the queries asked by our client with best of my knowledge. I tried to explain everything pointwise to make it visually appealing and easily understandable. In this case study, all the explanation is an output of my own hard work and efforts and nothing is plagiarized.

✅ PERFORM ANALYSIS

Tech Stack - VS Code, Power Point, Google Drive

I have used VS code with Jupyter Notebook extension and executed all the commands in VC Code itself by using Python Kernel Environment. Installed and imported all the libraries to get started

- **Method 1 - Jupyter Notebook - Command Prompt**

  Python (Jupyter) → Download and Install (Check Add to Path) → Check if Python Installed (Open Command Prompt)

  ```
  # Run commands in Command Prompt

  python --version
  pip install jupyter (This will install jupyter notebook)
  jupyter notebook
  ```

- **Method 2 - Jupyter Notebook - VS Code**

  VS Code → New File → `File_name.ipynb` Save → Extensions → Python, Jupyter → Install → Install ipykernel package → Execute code

  ```
  # Install Python Iibraries/Module
  %pip install numpy
  %pip install pandas
  %pip install matplotlib
  %pip install seaborn

  # Import Python Iibraries/Module
  import numpy as np                # For aarays and mathematical use
  import pandas as pd               # Dataframes or Tables
  import matplotlib.pyplot as plt   # Visualization
  %matplotlib inline
  import seaborn as sns             # Visualization
  ```

- **Cleaning the data**

  Commands Used for Data Cleaning → Removed nulls. Replaced required values. Removed duplicates.

```
# These are all the Python codes I used to clean data

df = pd.read_csv('Sarwar Ayaan Ali - Diwali Sales Data.csv', encoding= 'unicode_escape')        # Importing csv file

df.shape        # Count of rows and columns

df.head(10)        # df.head - Raw format data ; df.head() - Tabular format data ; df.head(10) - 10 rows

df.info()       # Details of rows and column with Null count and Data Type

# Dropped non-related/null columns
df.drop(['Status', 'unnamed1'], axis=1, inplace=True)      # axis=1 (It will remove entire row) ; inplace=True (Change will become permanent)

pd.isnull(df).sum()         # Count of nulls in each column

df.dropna(inplace=True)         # dropped rows containing null values

df['Amount'] = df['Amount'].astype('int')           # Changes data type from float to int

df.rename(columns= {'Marital_Status':'Married'},  inplace=True)      # Renaming a column

df.describe()           # describe() method returns description of the data in the DataFrame (i.e. count, mean, std, min, 25%, 50%, 75%, max)

# Successfully cleaned the data
```

✅ EXPLORATORY DATA ANALYSIS

Note: I have used the same formula for all tasks to achieve the desired results. Hence, wrote description of each only once in the task of Gender analysis.

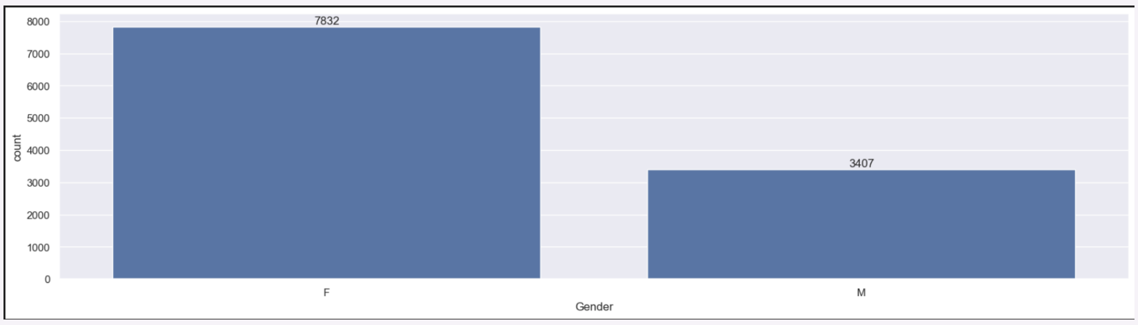
Task 1: Most amount spent - As per Gender

Created a Bar chart for the Count of Gender

```
# Bar Chart - Count of Gender

axis = sns.countplot(x = 'Gender', data = df)          # Seaborn countplot for the Count

for bars in axis.containers:                           # Bar Label using for loop
    axis.bar_label(bars)
```



Bar Chart - Gender VS Total amount

Grouped gender and amount, then sorted

Created a Bar chart using Seaborn (.sns)

Used for loop to label the bar

Also, I have converted the big numbers into **Millions** using code and divided the amount by 1000000 → `fmt='%1.1fM'`

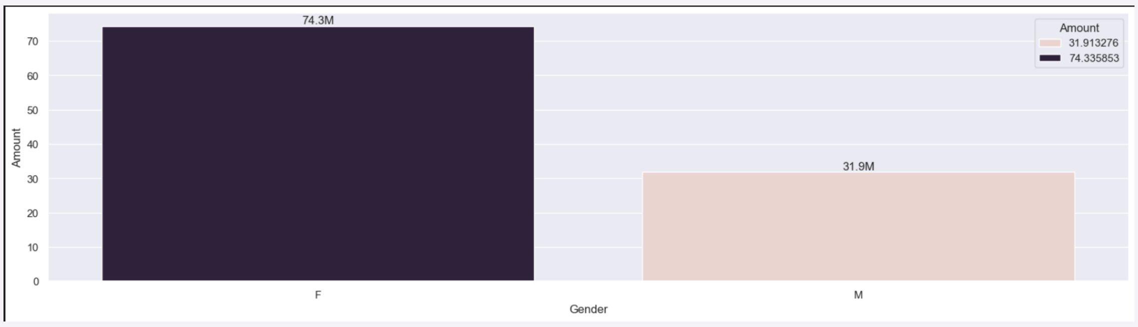
```
# Bar Chart - Gender VS Total amount

gender_amount = df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by
='Amount', ascending=False)       # Group and sort data

gender_amount['Amount'] = gender_amount['Amount'] / 1000000                # Convert
values to millions

axis = sns.barplot(x = 'Gender', y = 'Amount', data = gender_amount, hue = 'Amount')
# Bar chart

for bars in axis.containers:                                               # Bar Labe
l using for loop
    axis.bar_label(bars, fmt='%1.1fM')
```



From above graphs we can see that most of the buyers are females and even the purchasing power of females are greater than men

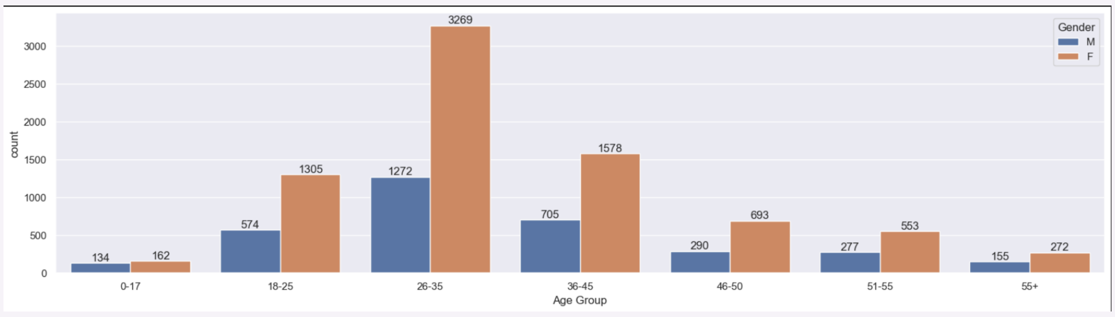
Task 2: Most amount spent - As per Age Group

Created a Bar chart for the Age Group

```
# Count of Age group.

age_gender = sns.countplot(data = df.sort_values(by='Age Group', ascending=True), x
= 'Age Group', hue = 'Gender')    # Group and sort data. Hue to categorize

for bars in age_gender.containers:                                          # Bar L
abel using for loop
    age_gender.bar_label(bars)
```



Bar Chart - Age group VS Amount

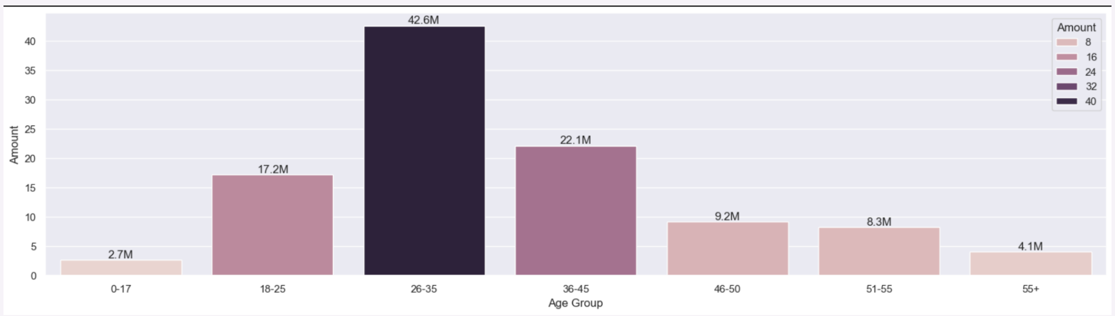
```
# Bar Chart - Age group VS Amount

age_amount = df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values
(by='Age Group', ascending=True)      # Group and sort data.

age_amount['Amount'] = age_amount['Amount'] / 1000000
# Convert values to millions

axis = sns.barplot(x = 'Age Group', y= 'Amount', data = age_amount, hue = 'Amount')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars, fmt='%1.1fM')
```



From above graphs we can see that most of the buyers are of age group between 26-35 yrs female

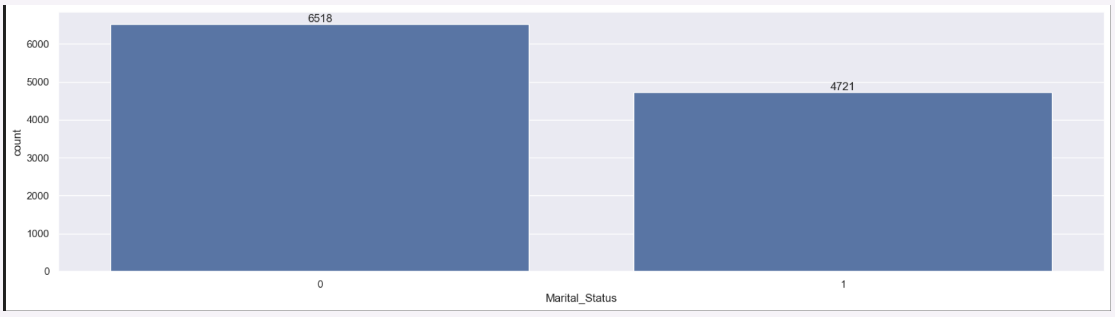
Task 3: Most amount spent - As per Marital Status

Created a Bar chart for the Marital_Status

```
# Bar Chart - Count of Marital_Status. 0 = Married ; 1 = Unmarried

axis = sns.countplot(data = df, x = 'Marital_Status')                           # B
ar chart

for bars in axis.containers:                                                    # B
ar Label using for loop
    axis.bar_label(bars)
```



Bar Chart - Marital_Status VS Amount as per Gender

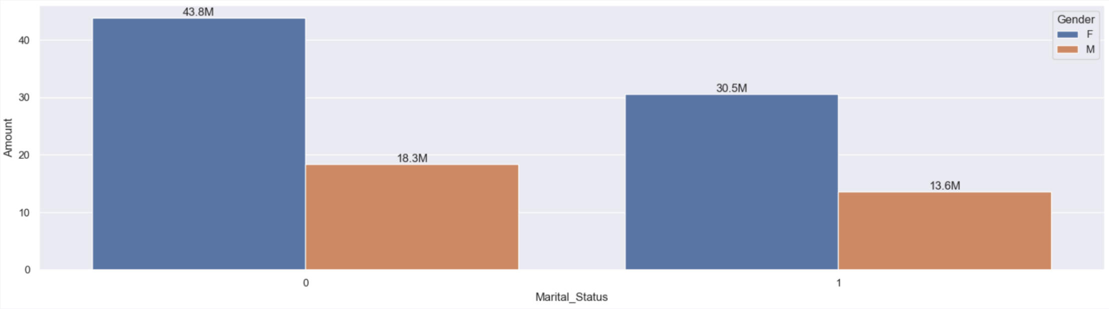
```
# Bar Chart - Marital_Status VS Amount as per Gender

marital_amount_per_gender = df.groupby(['Marital_Status', 'Gender'], as_index=Fals
e)['Amount'].sum().sort_values(by='Amount', ascending=False)         # Group and so
rt data.

marital_amount_per_gender['Amount'] = marital_amount_per_gender['Amount'] / 1000000
# Convert values to millions

axis = sns.barplot(data = marital_amount_per_gender, x = 'Marital_Status',y= 'Amoun
t', hue='Gender')          # Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars, fmt='%1.1fM')
```



From above graphs we can see that most of the buyers are married (women) and they have high purchasing power

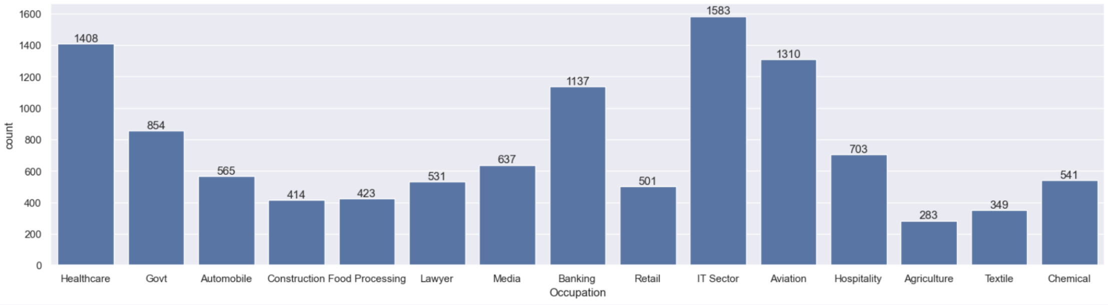
Task 4: Most amount spent - As per Occupation

Created a Bar chart for the Count of Occupation

```
# Bar Chart - Count of Occupation

axis = sns.countplot(data = df, x = 'Occupation')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars)
```



Bar Chart - Occupation VS Amount

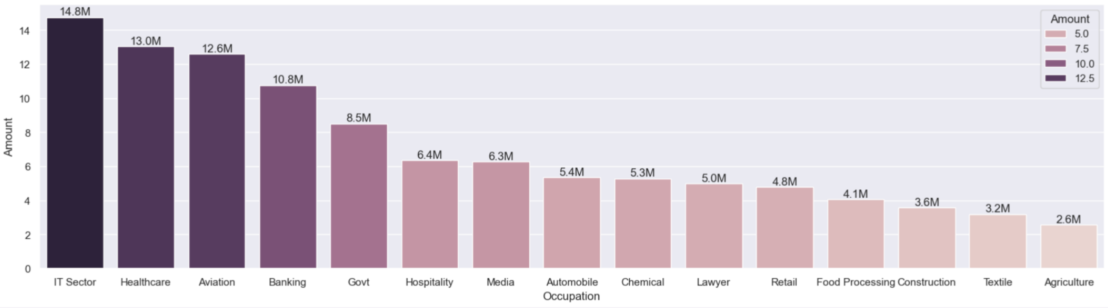
```
# Bar Chart - Occupation VS Amount

occupation_amount = df.groupby(['Occupation'], as_index=False)['Amount'].sum().sort
_values(by='Amount', ascending=False)        # Group and sort data

occupation_amount['Amount'] = occupation_amount['Amount'] / 1000000
# Convert values to millions

axis = sns.barplot(data = sales_state, x = 'Occupation',y= 'Amount',  hue='Amount')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars, fmt='%1.1fM')
```



From above graphs we can see that most of the buyers are working in IT, Healthcare and Aviation sector

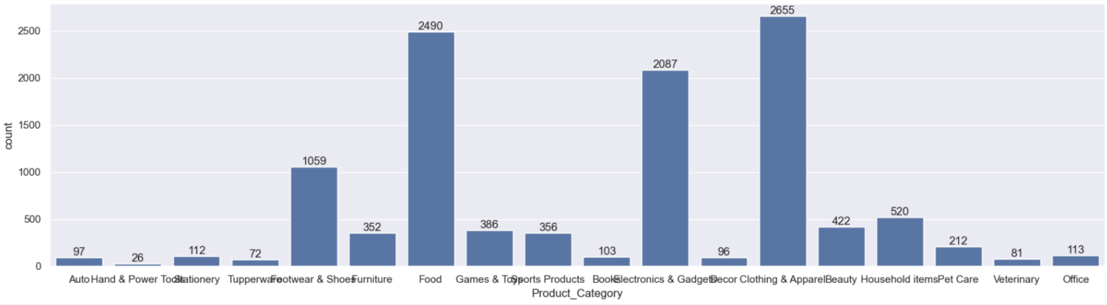
Task 5: Most amount spent - As per Product Category

Created a Bar chart for the Product_Category

```
# Bar Chart - Product_Category

axis = sns.countplot(data = df, x = 'Product_Category')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars)
```



Bar Chart - Product_Category VS Amount

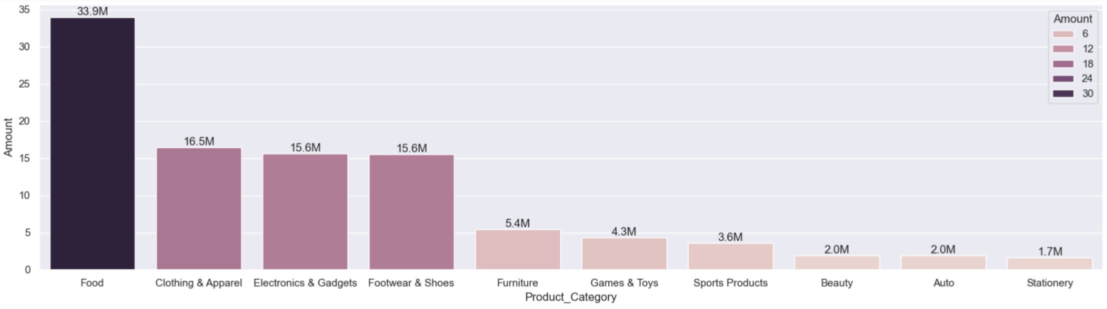
```
# Bar Chart - Product_Category VS Amount

category_amount = df.groupby(['Product_Category'], as_index=False)['Amount'].sum().
sort_values(by='Amount', ascending=False).head(10)         # Group and sort data

category_amount['Amount'] = category_amount['Amount'] / 1000000
# Convert values to millions

axis = sns.barplot(data = category_amount, x = 'Product_Category',y= 'Amount',  hue
='Amount')          # Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars, fmt='%1.1fM')
```



From above graphs we can see that most of the sold products are from Food, Clothing and Electronics category

Task 6: Number of Orders - As per Product ID

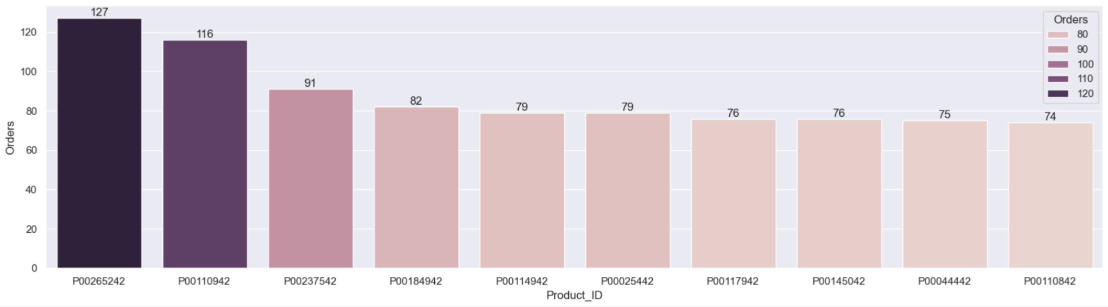
Created a Bar chart for the Product_ID VS Orders

```
# Bar Chart - Product_ID VS Orders

id_orders = df.groupby(['Product_ID'], as_index=False)['Orders'].sum().sort_values
(by='Orders', ascending=False).head(10)         # Group and sort data

axis = sns.barplot(data = id_orders, x = 'Product_ID',y= 'Orders',  hue='Orders')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars)
```



From above graphs we can see that most of the orders are for the Product ID P00265242

Task 7: Number of Orders - As per State

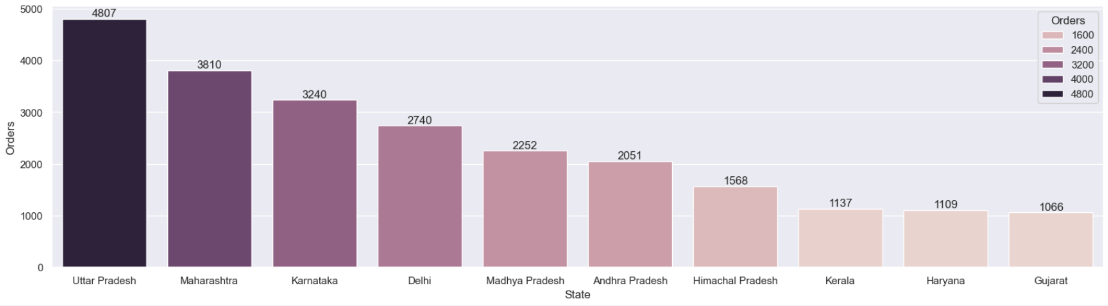
Created a Bar chart for the State VS Orders

```
# Bar Chart - State VS Orders

state_orders = df.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by
='Orders', ascending=False).head(10)         # Group and sort data

axis = sns.barplot(data = state_orders, x = 'State',y= 'Orders', hue = 'Orders')
# Bar chart

for bars in axis.containers:
# Bar Label using for loop
    axis.bar_label(bars)
```



From above graphs we can see that most of the orders & total sales/amount are from Uttar Pradesh, Maharashtra and Karnataka respectively

☑️ INSIGHTS

- Performed Data Cleaning and Manipulation
- Performed Exploratory Data Analysis (EDA) using Pandas, Matplotlib and seaborn libraries
- Improved customer experience by identifying potential customers across different states, occupation, gender and age groups
- Improved sales by identifying most selling product categories and products, which can help to plan inventory and hence meet the demands

Potential Customers

- **Female customers** are **230%** more than the male customers
- **Female customers** spent **233%** more than the male customers
- **40.38% customers** are from **26-35 age group** and **20.32%** are from **36-45 age group**
- **26-36 age group customers spent 40.11%** i.e. the most
- **58.47% married** and **41.53% unmarried** customers
- **41.21% married female customers** and **28.75% unmarried female customers**
- **IT sector customers** spent **13.89%** and **Healthcare customers** spent **12.27%** i.e. the most
- **Uttar Pradesh** has **17.30% orders** and **18.24% amount** spent i.e. the most

Potential Product

- **31.94% amount** spent on **food category**, **15.52%** on **Clothing** and **14.72%** of **Electronics** i.e. the most.
- **Product ID - P00265242** has **most number of orders**, it means it is **most selling product**

Summary

- **Married women** of **age group 26-35 years** from **UP, Maharashtra and Karnataka** working in **IT, Healthcare and Aviation** are more likely to buy products from **Food, Clothing and Electronics** category.
- **31.94% amount** spent on **food category**, **15.52%** on **Clothing** and **14.72%** of **Electronics** i.e. the most.

▼ 🟪 Diwali Sales Analysis - Project Files on Google Drive

> 👉 **Project Files on Google Drive**
>
> **Project Link**