INTERNSHIP REPORT

DS GROUP

Dharampal Satyapal Group

Created by

Hamza

Submitted to

Mr. Shivendra kaura(deputy manager)

index

Introduction about company

Since 1929 Dharampal Satyapal group is a multi business corporation.

It is one of the leading fast moving consumer goods the Group works with projects across the country to achieve objectives and also focus on area such as water livelihood and education.

Vision to be a leading quality and innovation driven global conglomerate.

Doing things in newer way with cost effective and better products.

Mission the company is striving to achieve excellence in service quality and all other in divorce to create sustainable value means

creating tangible benefit for stakeholders like consumers and investors etc.

The core value which I observed during working

empathy sympathy and compassion caring anger tolerance and listening to interns.

Honesty integrity, Ethical behaviour and financial honesty and unbiased decisions.

The DS group is trendsetter: they launch first herbal mouth freshener called "pass pass"

DS group is the first who introduced various kinds of spices in one time packaging.

Fresh spring water in bottles.

Data Analysis on HR data set

My internship started as where my mentor having 5 plus year experience in IT-department he explained me how things done in this dynamic world why we create dashboard what's the us, to who you have to present, how to extract meaningful insights from raw data & **how your analysis can you show future sale of a product**.

They also taught in meeting room 4 how to use tableau to all the interns and different data connections.

They also shared there experience with me what to do after B-tech masters or job first, what projects you can make etc.

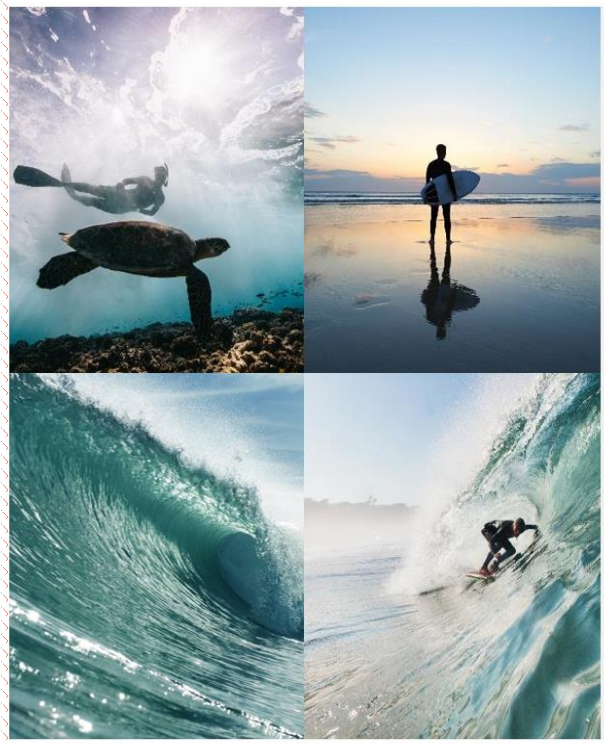
Tableau is platform were we can connect any data source to get insight.

Problem worked on it

IN some company HR department wants dashboard

- **They have raw data they wont proper interactive representation of their database do understand and represent higher authority that how many employees are leaving their company from which field and department understanding their job satisfaction levels from which age band they are and their gender.**

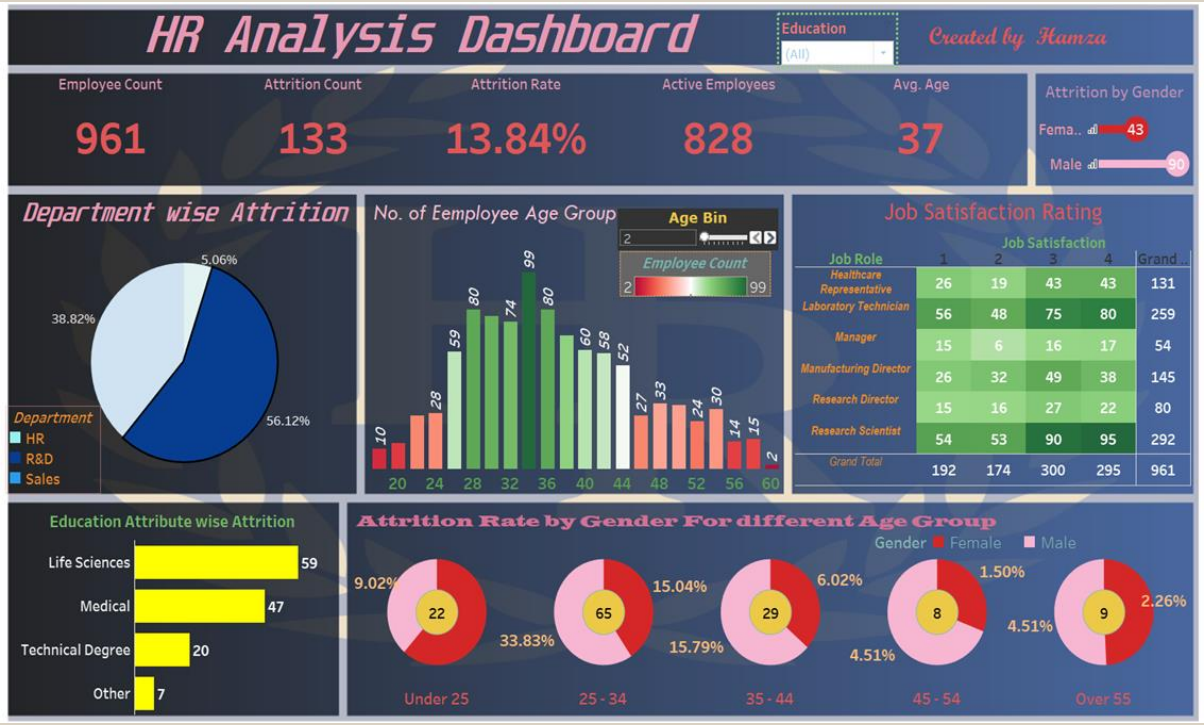
The things which I done is in following:



Agenda

Problem
introduction
KPIs and lollipop chart
pie chart
frequency chart
heat map
bar chart
donut chart
summary
live working video
thank you

Please click on following dashboard



to look at presentation.

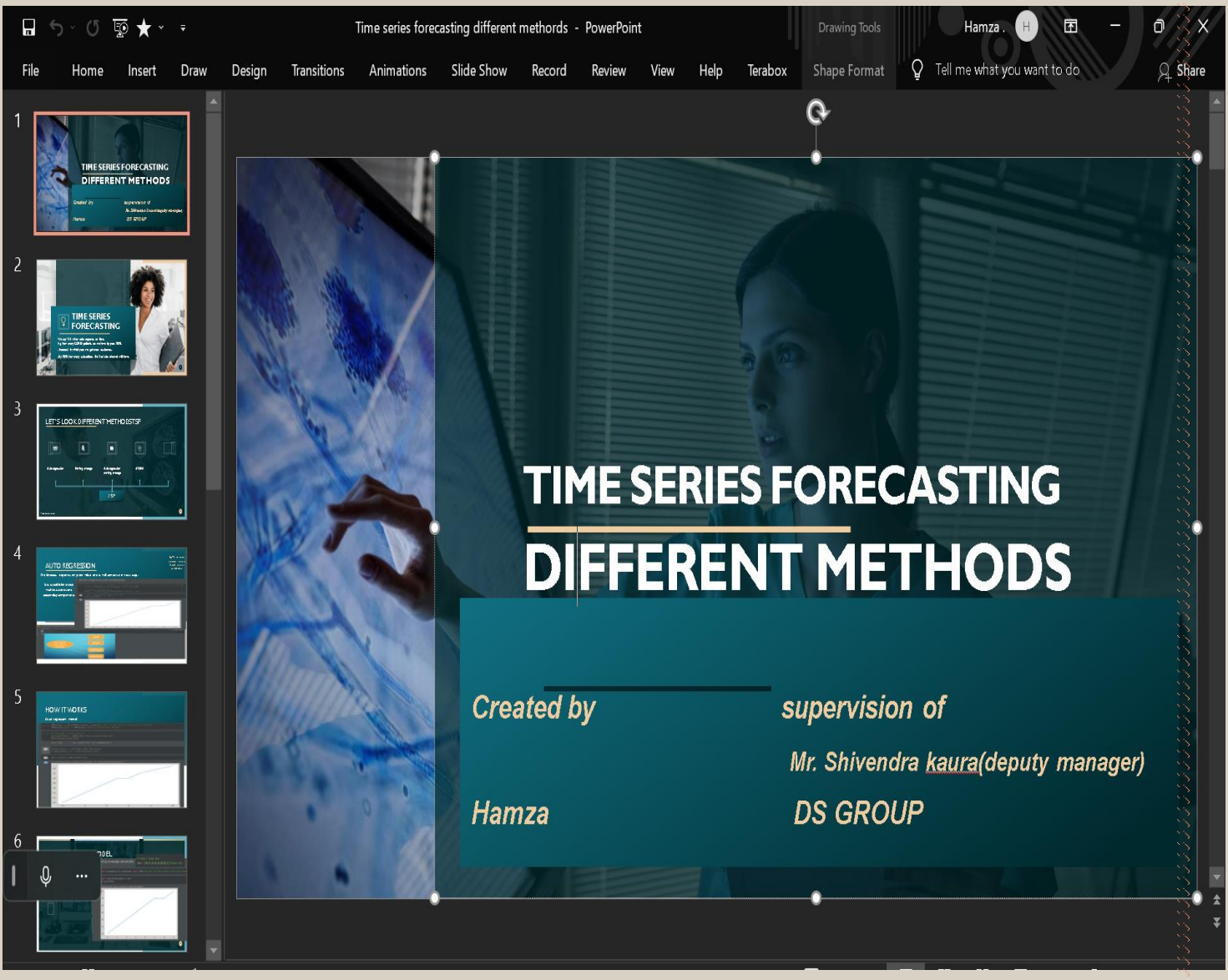
Time series forecasting

when data is depend on time.

Example: How COVID patients will increase in future.

Eg: In which month company will get more customers we use different methods.

Let's discuss **5 methods** to implement time series forecasting



click on following image to see my work on time series forecasting.

Weather prediction

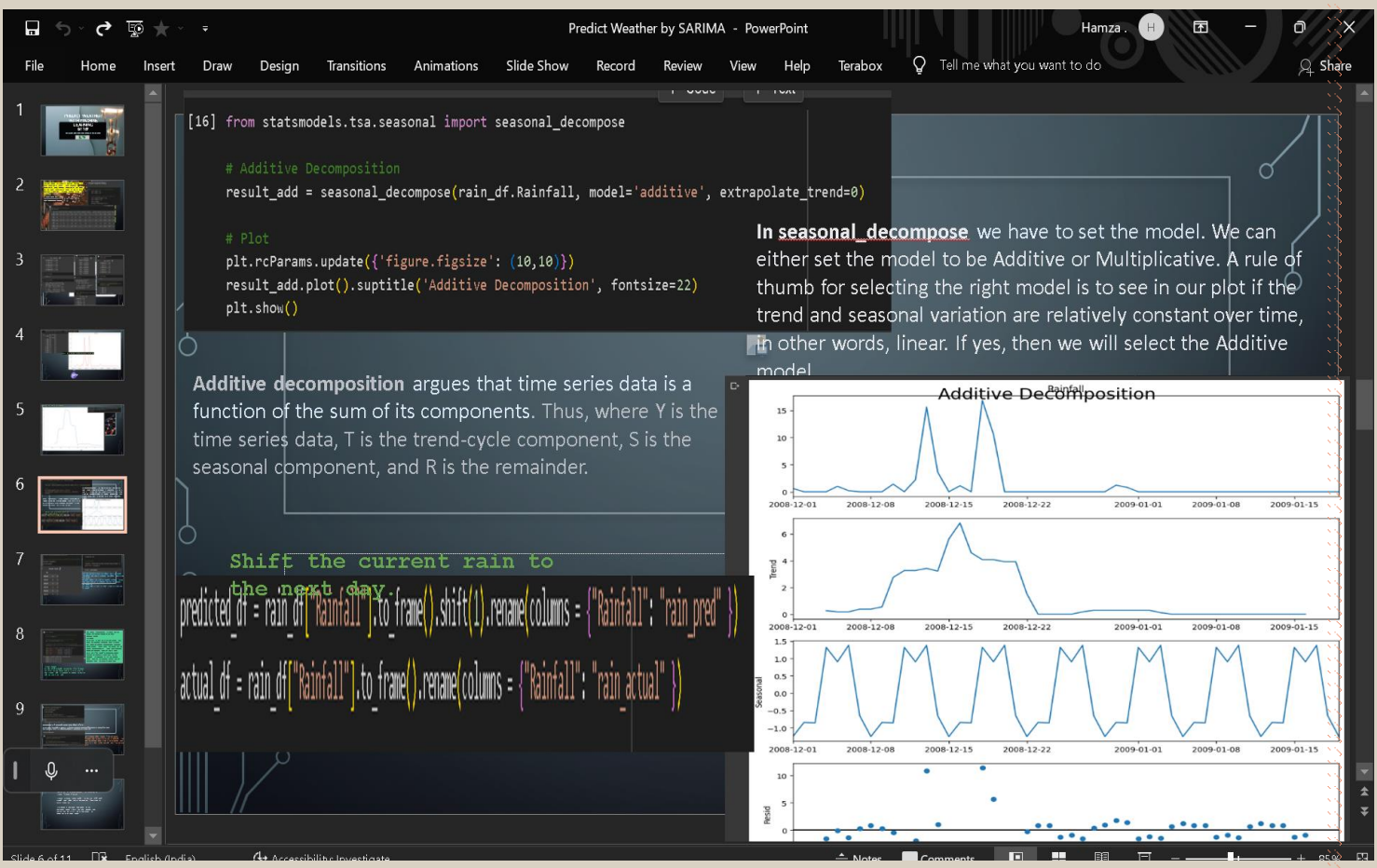
We have access to a century of historical averages of global temperatures, including global maximum temperatures, global minimum temperatures, and global land and ocean temperatures. Having all this, we know that this is a supervised regression machine learning problem.

Click on following image to see the presentation.

Using SARIMA model(**extension of ARIMA that can account for seasonal patterns in the data**)

Parameter Selection

Grid Search



We are going to apply one of the most commonly used method for time-series forecasting, known as SARIMA, which stands for **Seasonal Autoregressive Integrated Moving Average.** SARIMA models are denoted with the notation SARIMA(p,d,q)(P,D,Q,s). These three parameters account for seasonality, trend, and noise in data:

We will use a “grid search” **to iteratively explore different combinations of parameters**. For each combination of parameters, we fit a new seasonal SARIMA model with the SARIMAX() function from the statsmodels module and assess its overall quality.

Project on Exploratory Data Analysis

What is EDA?

- It is vital step before data analysis because batter you know data the better is your analysis.
- It reveals the data.
- This is the approach used to analyse the data and to discover trends patterns in data buy the graphical representation.

Importance of EDA: It helps to look data before making conclusions and assumptions.

It shows errors and anomalies presence in data set.

Shows relationship among variables.

Types of EDA

- univariate non graphical
- univariate graphical
- multivariate non graphical
- multivariate graphical and many more….

Depend on the number of fields we can divide EDA to types:

- **Univariate non graphical** it analyses just one variable describes the pattern which exists in it.
- **Univariate graphical** it shows full picture like histogram and box plots.
- **Multivariate non graphical** shows relationship for two or more variables.
- **Multivariate graphical** it use graphics to display relationship between two or more set of data.

Steps in exploratory data analysis

- **Data Collection**

  Given by the company or we have to gather and create.

- **Finding variables and understanding them-** try to get insight from variable and observe how they make impact.
- **Cleaning data set-** works on null value and irrelevant information.

- **Identify correlated variables** -helps to know particular variable is related to another.

- **Choosing statistical method** depends on size type categorical or numerical or other factors.
- **Visualising and analysing result** once analysis done now carefully interpreted it and conclude with a few lines which can increase company's profit.

Technical stacks use

Python programming language

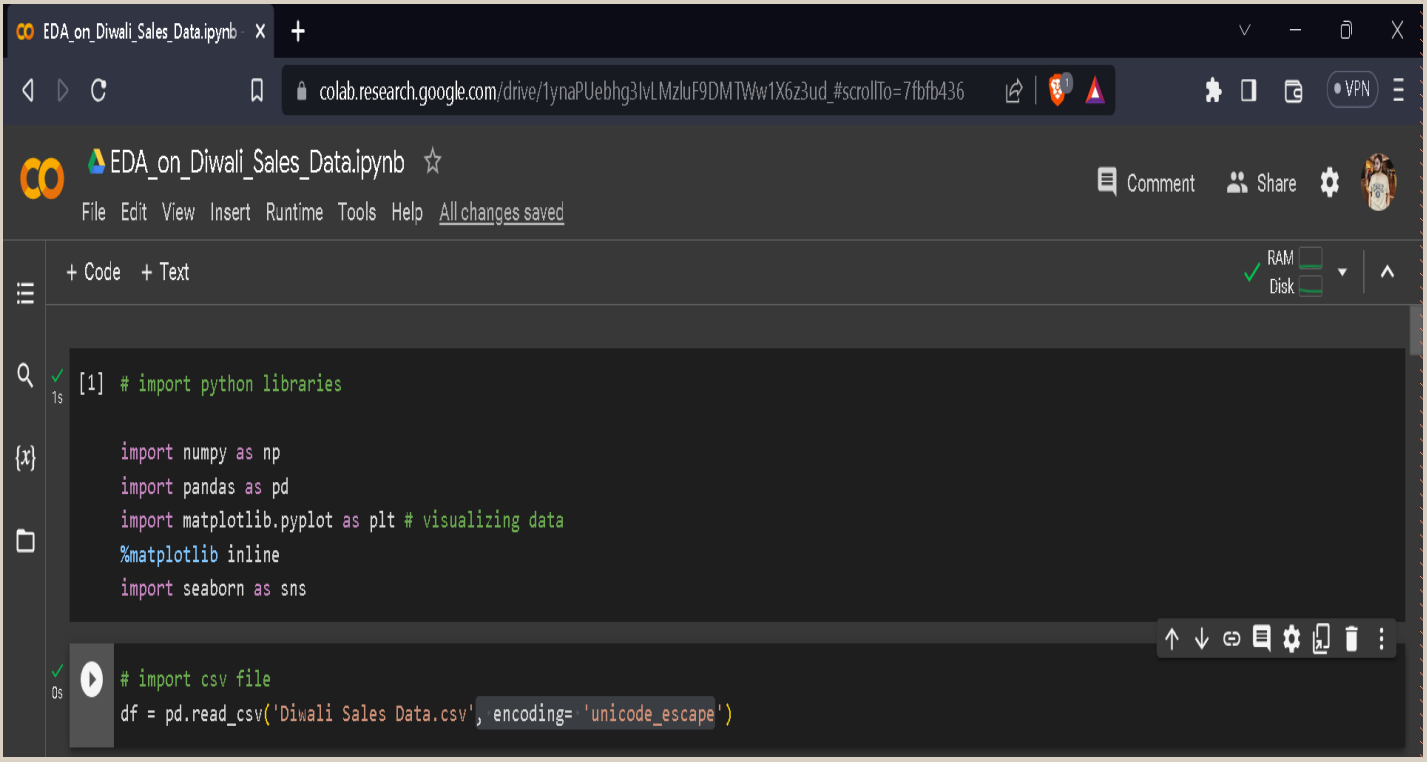
libraries

laptop with Wi-Fi

Jupiter notebook with Anaconda navigator

Data Flow Diagram

imported all required libraries and loading CSV file



NumPy used for array related work.

Pandas ***to work on data frames and rescue CSV and use other functions like drop().***

Matplotlib and Seaborn are most important libraries for this project to make chart and graph for representation.

Exploring data

```
df.shape
```

```
(11251, 15)
```

+ Code + Text

```
df.head()
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0	NaN
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0	NaN
	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0	NaN

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 15 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   User_ID           11251 non-null  int64
 1   Cust_name         11251 non-null  object
 2   Product_ID        11251 non-null  object
 3   Gender            11251 non-null  object
 4   Age Group         11251 non-null  object
 5   Age               11251 non-null  int64
 6   Marital_Status    11251 non-null  int64
 7   State             11251 non-null  object
 8   Zone              11251 non-null  object
 9   Occupation        11251 non-null  object
 10  Product_Category  11251 non-null  object
 11  Orders            11251 non-null  int64
 12  Amount            11239 non-null  float64
 13  Status            0 non-null      float64
 14  unnamed1          0 non-null      float64
dtypes: float64(3), int64(4), object(8)
memory usage: 1.3+ MB
```

Data cleaning

As we observed 2 fields having null values so we have to remove them using pandas function called drop().

```
[ ]  #drop unrelated/blank columns
     df.drop(['Status', 'unnamed1'], axis=1, inplace=True)
```

axis=1 means full vertical raw.

In place used to save.

```
[ ]  #check for null values
     pd.isnull(df).sum()
```

```
User_ID             0
Cust_name           0
Product_ID          0
Gender              0
Age Group           0
Age                 0
Marital_Status      0
State               0
Zone                0
Occupation          0
Product_Category    0
Orders              0
Amount             12
dtype: int64
```

which shows null values

```
[ ]  # drop null values
     df.dropna(inplace=True)
```

```
[ ] # drop null values
    df.dropna(inplace=True)
```

```
[ ] # change data type
    df['Amount'] = df['Amount'].astype('int')
```

```
df['Amount'].dtypes
```

```
dtype('int32')
```

Changing data type by using function.

```
[ ] df['Amount'].dtypes

    dtype('int32')
```

Using column sing all fields.

```
[ ] df.columns

    Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
           'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
           'Orders', 'Amount'],
          dtype='object')
```

```
# describe() method returns description of the data in the DataFrame (i.e. count, mean, std, etc)
df.describe()
```

	User_ID	Age	Marital_Status	Orders	Amount
count	1.123900e+04	11239.000000	11239.000000	11239.000000	11239.000000
mean	1.003004e+06	35.410357	0.420055	2.489634	9453.610553
std	1.716039e+03	12.753866	0.493589	1.114967	5222.355168
min	1.000001e+06	12.000000	0.000000	1.000000	188.000000
25%	1.001492e+06	27.000000	0.000000	2.000000	5443.000000
50%	1.003064e+06	33.000000	0.000000	2.000000	8109.000000
75%	1.004426e+06	43.000000	1.000000	3.000000	12675.000000
max	1.006040e+06	92.000000	1.000000	4.000000	23952.000000

Using describe function let's see mathematical description.

Example minimum and maximum price of product.

And different percentile 79%.

```
+ Code   + Text
# use describe() for specific columns
df[['Age', 'Orders', 'Amount']].describe()
```

	Age	Orders	Amount
count	11239.000000	11239.000000	11239.000000
mean	35.410357	2.489634	9453.610553
std	12.753866	1.114967	5222.355168
min	12.000000	1.000000	188.000000
25%	27.000000	2.000000	5443.000000
50%	33.000000	2.000000	8109.000000
75%	43.000000	3.000000	12675.000000
max	92.000000	4.000000	23952.000000

Exploratory Data Analysis

let's check attributes available in our data set to apply EDA on them.

```
[ ] df.columns

Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

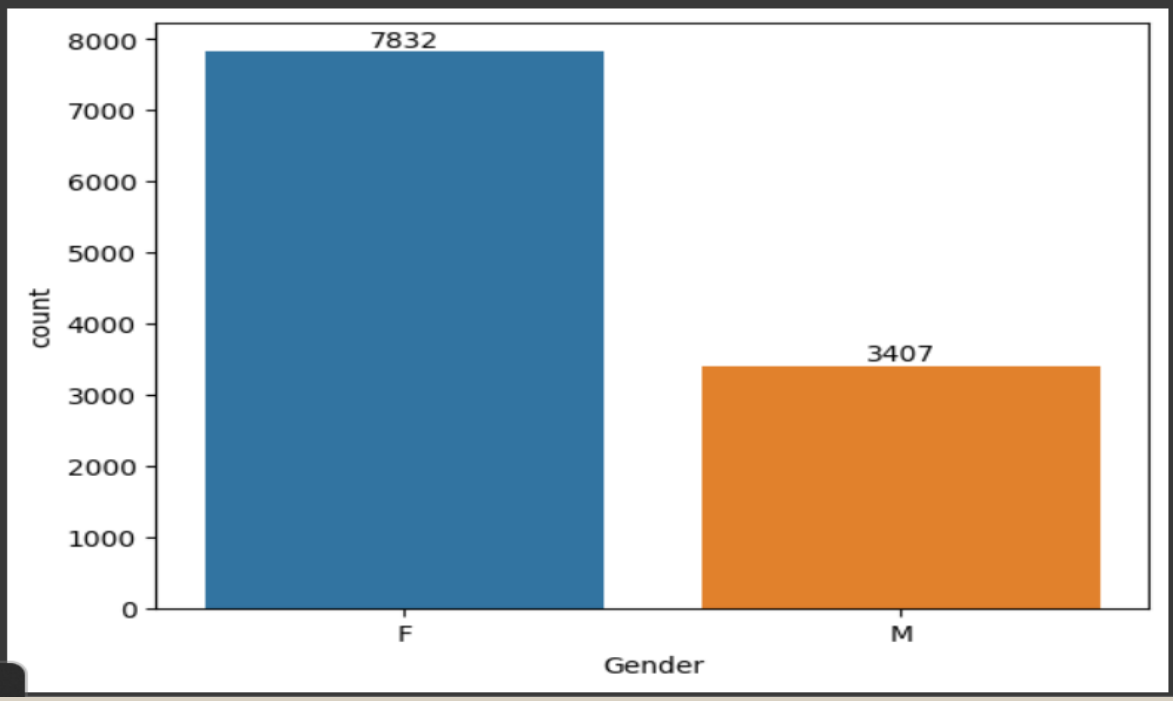
So let's apply ***EDA*** on gender field.

Gender

```
# plotting a bar chart for Gender and it's count

ax = sns.countplot(x = 'Gender',data = df)

for bars in ax.containers:
    ax.bar_label(bars)
```



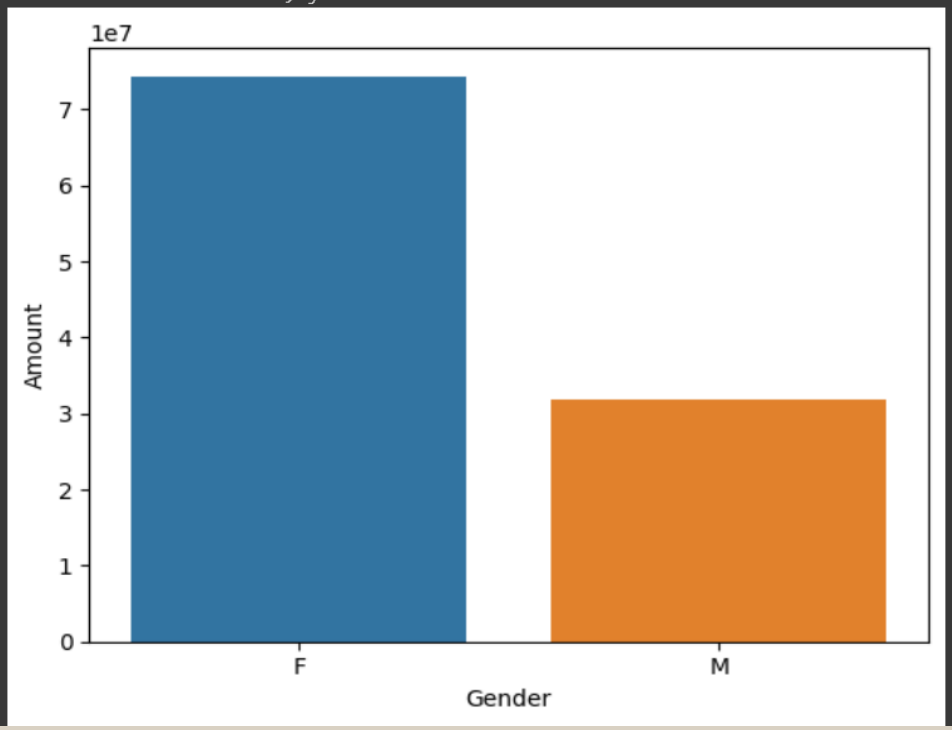
By the help of seaborne library , count plot () shows count of orders placed by females.

```
# plotting a bar chart for gender vs total amount

sales_gen = df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)

sns.barplot(x = 'Gender',y= 'Amount' ,data = sales_gen)
```

```
<Axes: xlabel='Gender', ylabel='Amount'>
```



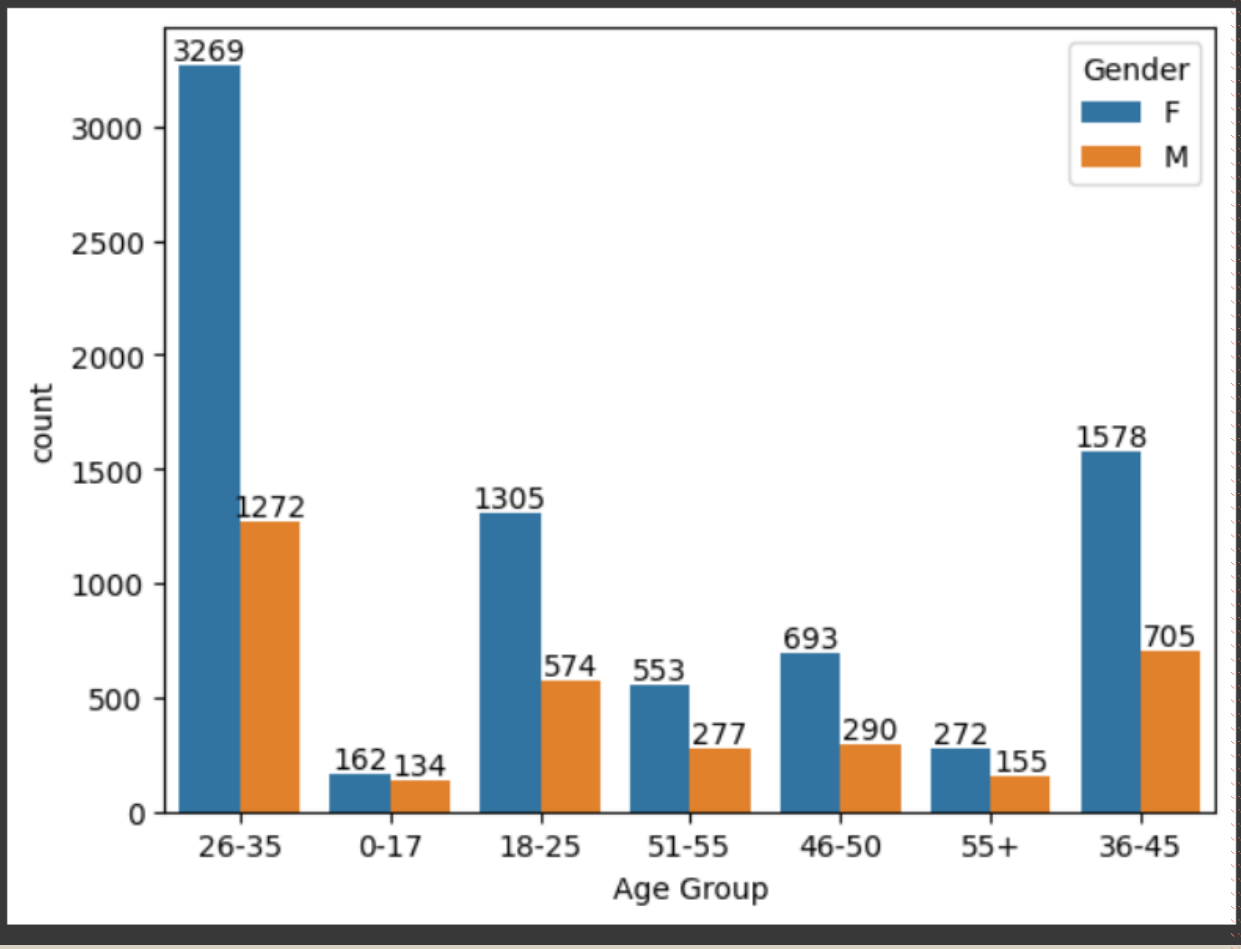
From above graphs we can see that most of the buyers are females and even the purchasing power of females are greater than men.

Now let's focus on <u>Age</u> attribute to know which age group of females are buying more products.

Age

```
ax = sns.countplot(data = df, x = 'Age Group', hue = 'Gender')

for bars in ax.containers:
    ax.bar_label(bars)
```

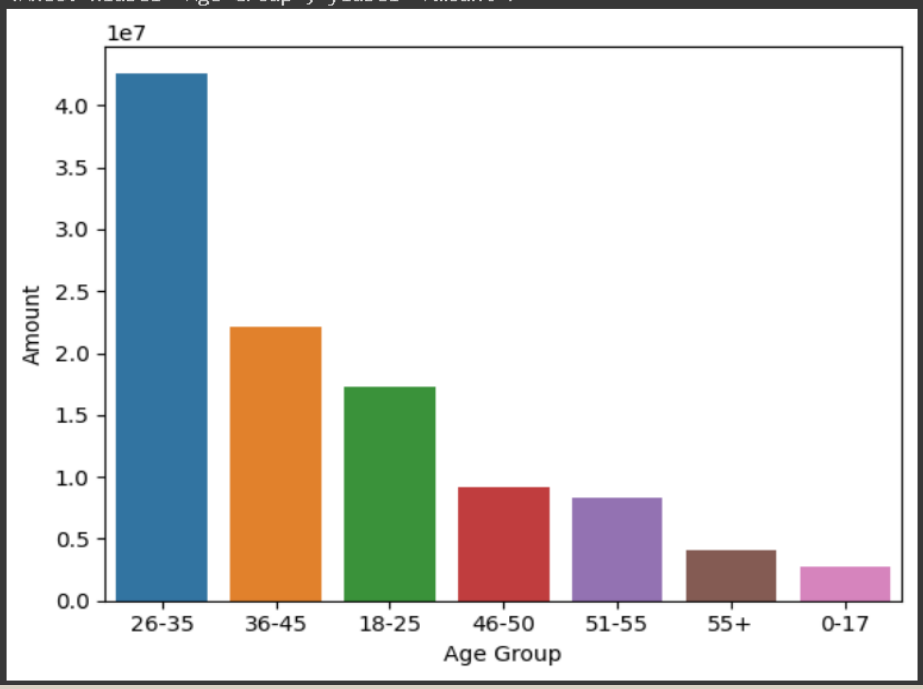


As by seeing the legend blue colour for female.

```python
# Total Amount vs Age Group
sales_age = df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)

sns.barplot(x = 'Age Group',y= 'Amount' ,data = sales_age)
```

```
<Axes: xlabel='Age Group', ylabel='Amount'>
```



we can see from bar chart that most of the buyers are of age group between 26-35 years female.

Now let's see from which **states** we are getting more orders.

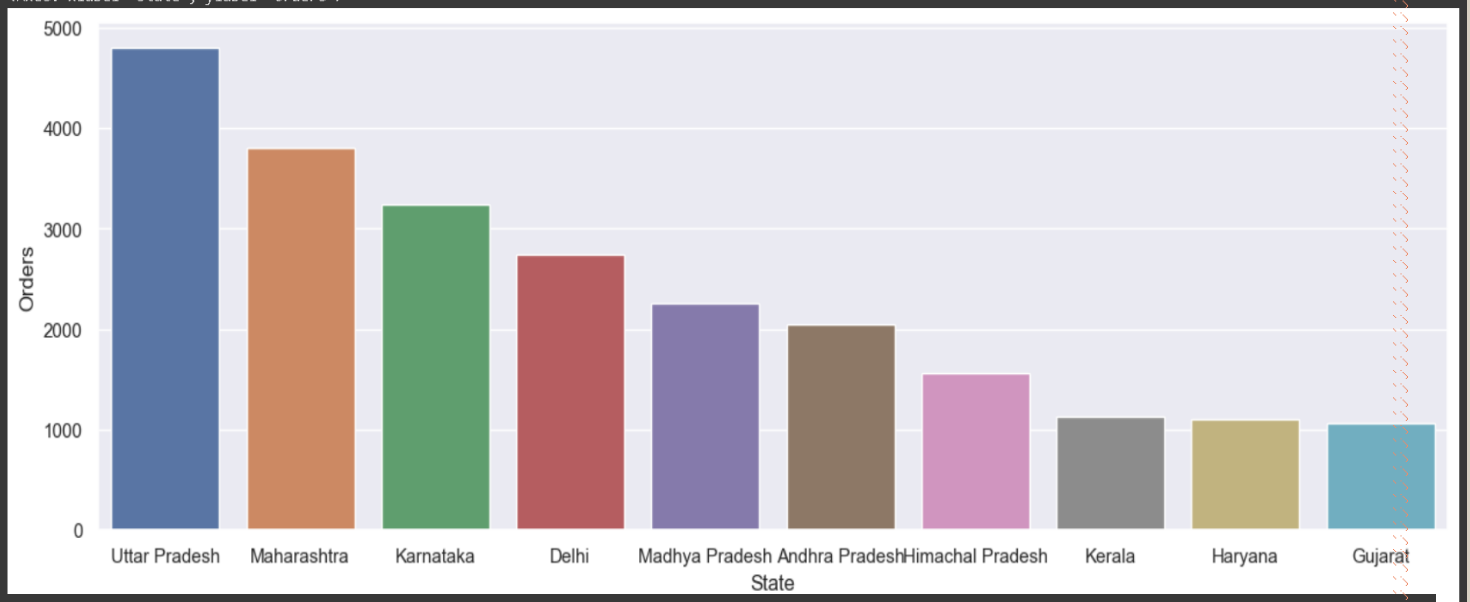
State

```python
# total number of orders from top 10 states

sales_state = df.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(data = sales_state, x = 'State',y= 'Orders')
```

```
<Axes: xlabel='State', ylabel='Orders'>
```

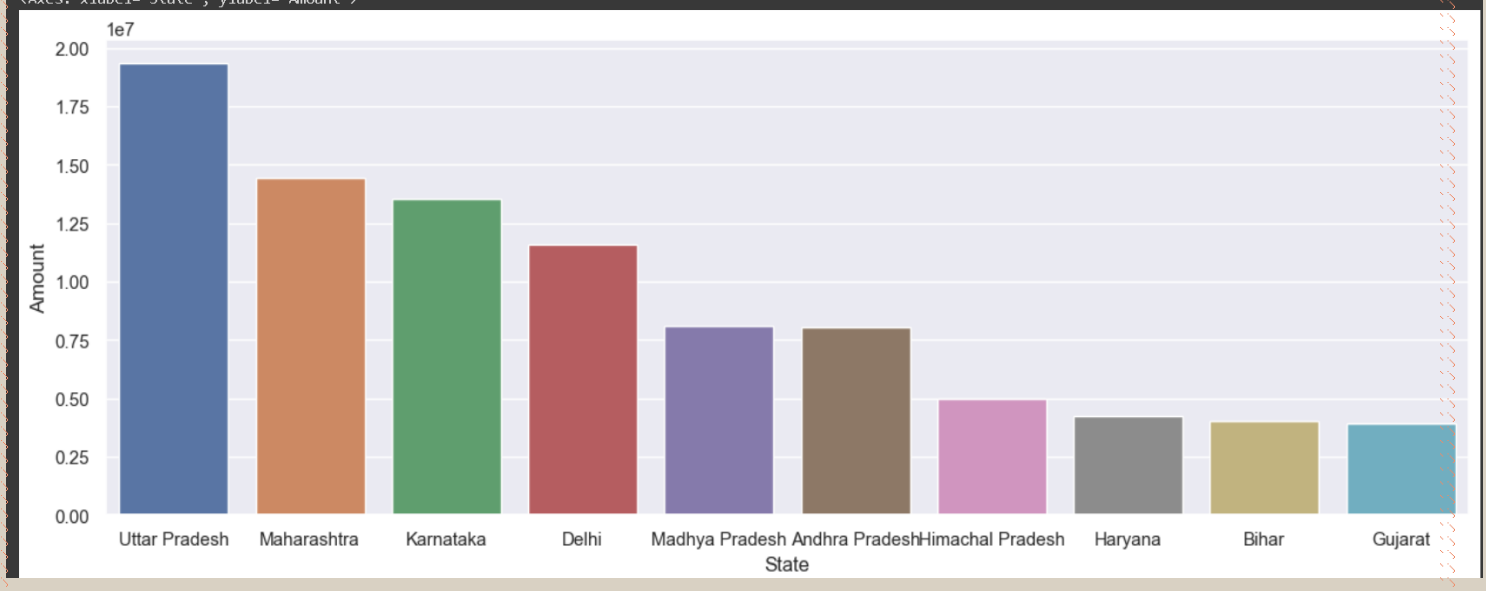



```
# total amount/sales from top 10 states

sales_state = df.groupby(['State'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False).head(10)

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(data = sales_state, x = 'State',y= 'Amount')
```

```
<Axes: xlabel='State', ylabel='Amount'>
```



Most of the orders & total sales/amount are from Uttar Pradesh, Maharashtra and Karnataka respectively

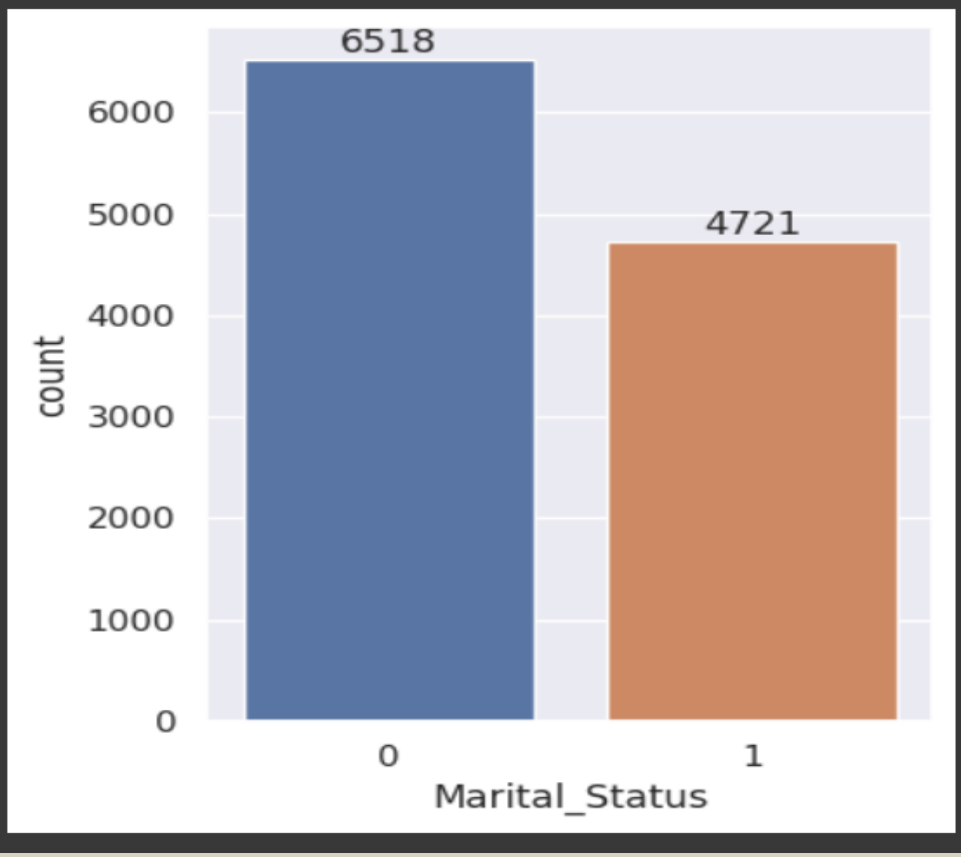
```
df.columns
```

```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

Marital Status

```
ax = sns.countplot(data = df, x = 'Marital_Status')

sns.set(rc={'figure.figsize':(7,2)})
for bars in ax.containers:
    ax.bar_label(bars)
```

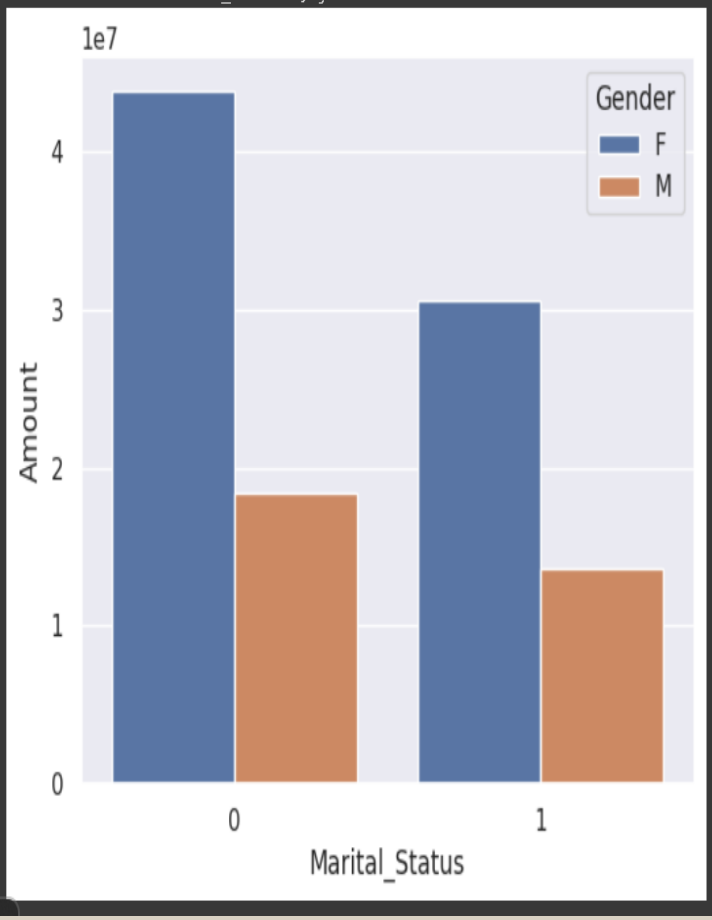


it will help us to know the married or unmarried who do more shopping.

```
sales_state = df.groupby(['Marital_Status', 'Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)

sns.set(rc={'figure.figsize':(6,5)})
sns.barplot(data = sales_state, x = 'Marital_Status',y= 'Amount', hue='Gender')
```

```
<Axes: xlabel='Marital_Status', ylabel='Amount'>
```



Subsequently we can see married woman bar chart age at peak.

Means most of the buyers are married (women) and they have high purchasing power.

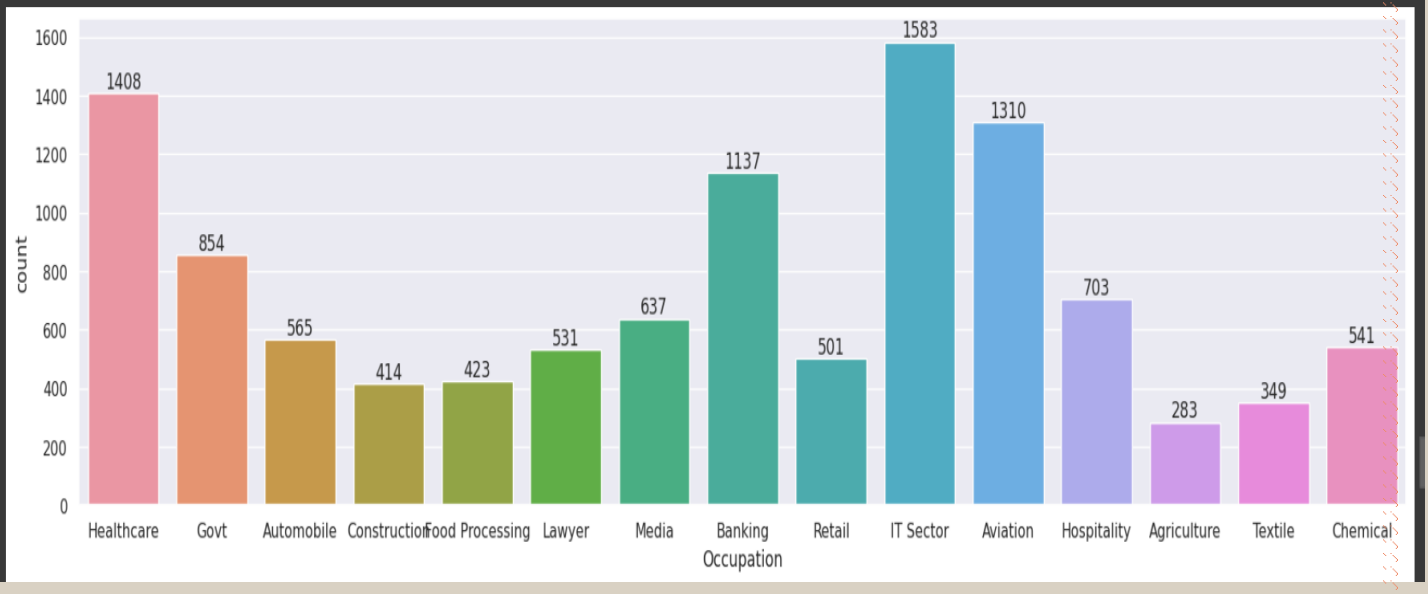
Occupation

let's try to get some insight from occupation field the buyers are from which occupation those who are purchasing.

It is showing just count.

```
sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Occupation')

for bars in ax.containers:
    ax.bar_label(bars)
```

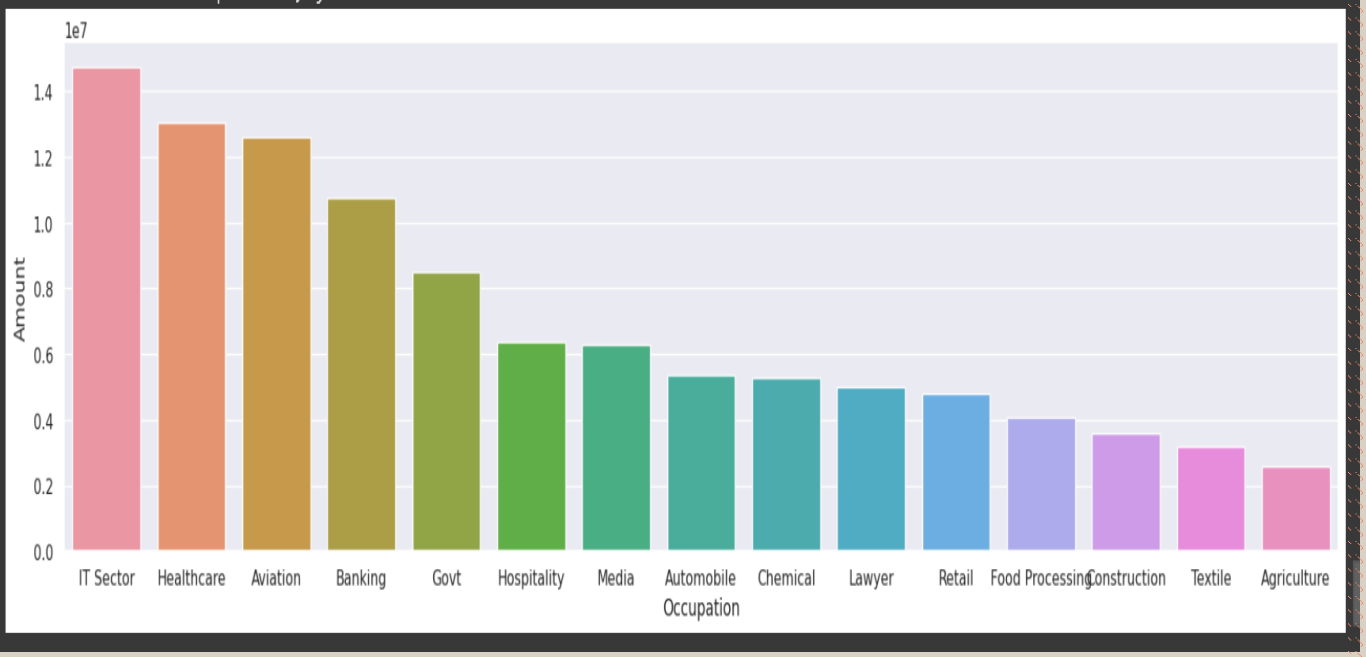



```
sales_state = df.groupby(['Occupation'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)

sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Occupation',y= 'Amount')
```

```
sns.barplot(data = sales_state, x = 'Occupation',y= 'Amount')
```

```
<Axes: xlabel='Occupation', ylabel='Amount'>
```



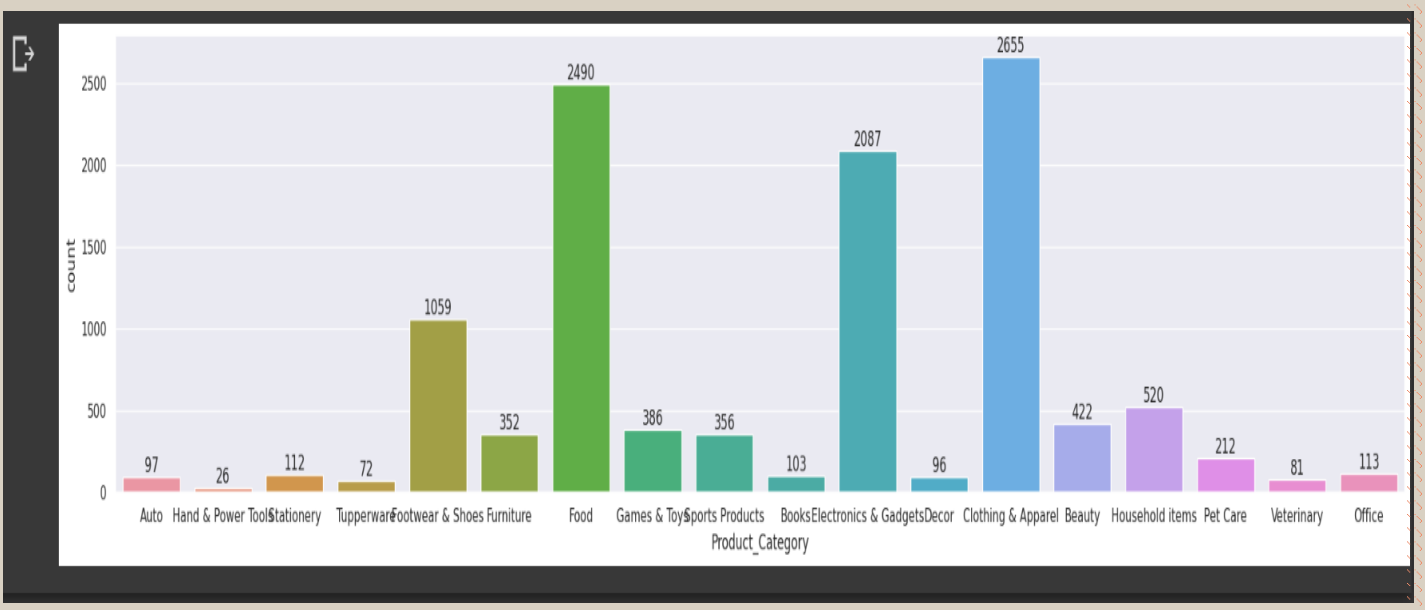
So it shows that people working in IT sector are spending more amount on purchasing as compared to remaining sectors like: Healthcare, Aviation, Banking & Agriculture etc.

Product Category

Let's see which category of item is sailing more and giving maximum profit.

```
sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Product_Category')

for bars in ax.containers:
    ax.bar_label(bars)
```



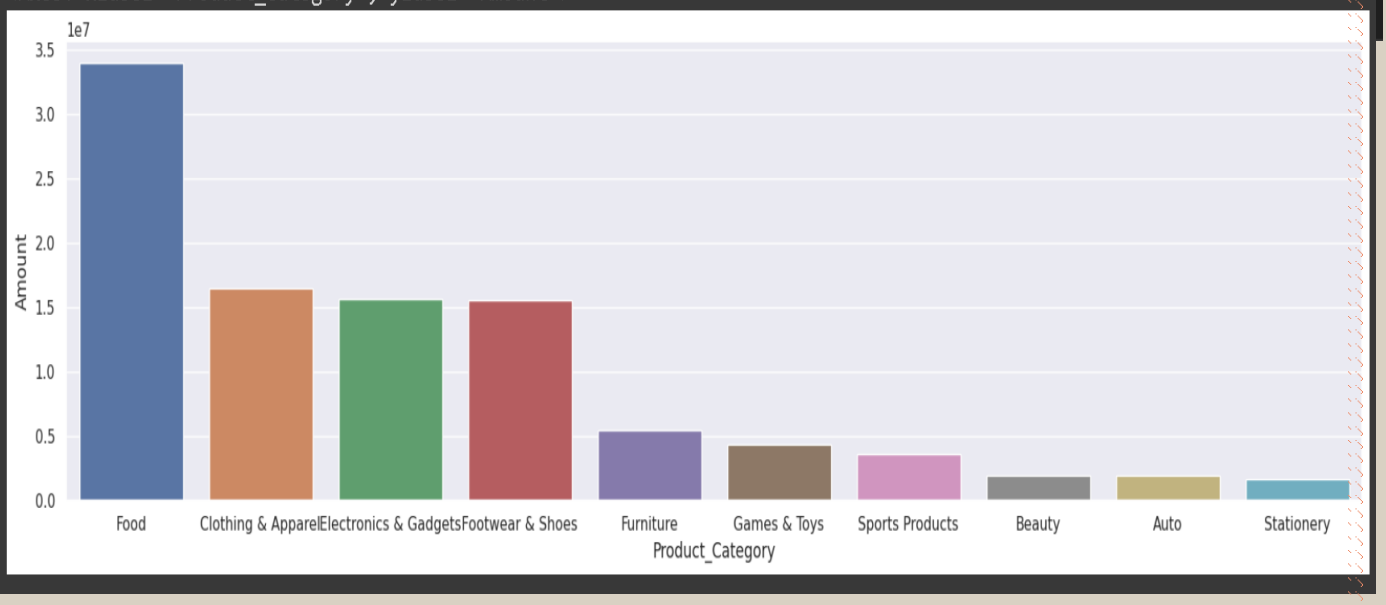
So, we can see maximum count on clothes and appeal then food.

But let's see which generate more amount

```
sales_state = df.groupby(['Product_Category'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)

sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Product_Category', y= 'Amount')
```

```
<Axes: xlabel='Product_Category', ylabel='Amount'>
```



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

Conversely maximum profit generated by the food.

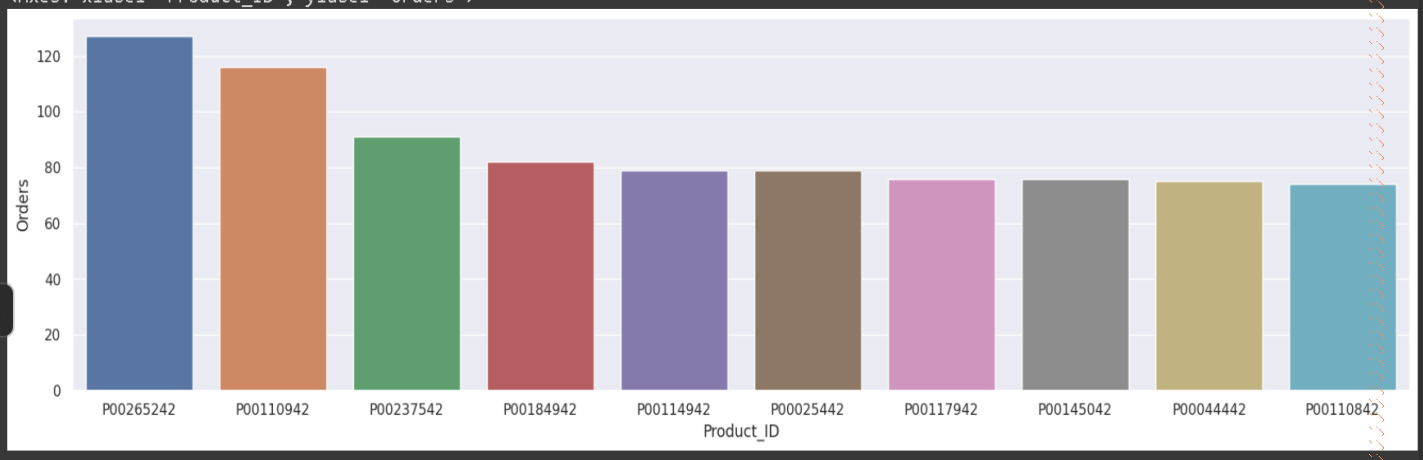
Product

Now let's know top selling product

```
sales_state = df.groupby(['Product_ID'], as_index=False)['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)

sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Product_ID',y= 'Orders')
```

```
<Axes: xlabel='Product_ID', ylabel='Orders'>
```



0s completed at 5:20 PM

Conclusion:

Married women age group 26-35 years from Uttar Pradesh, Maharashtra and Karnataka working in IT, Healthcare and Aviation are more likely to buy products from Food, Clothing and Electronics category.

Project learning

- Data cleaning and manipulation.
- How to perform EDA on which field & how to get insight.
- Changed my mentality we have to apply EDA to give conclusion which can generate company revenue we have to tell them useful areas on which they can work on.
- Used seaborne library and how it contributes a lot in this analysis without it will be difficult.
- In last this analysis will definitely improve the company sales if company will work on expects which I given in above summary it also helped to plan inventory and meets demands.

Bibliography

Data set link

https://drive.google.com/file/d/19GnZWj5lXc5Sa_wHTQsxMMhyiORAkL0-/view?usp=sharing

Jupiter notebook code

https://drive.google.com/file/d/1BErUEnR_ItYbORWFM4MUlZ3pJ-2tpgs0/view?usp=sharing


Thank
you!