DIWALI SALES DATA EXPLORATORY DATA ANALYSIS PROJECT

Objective of the project:

1) To get statistical insights from the dataset 2) To know customers of which gender and age group are the maximum buyers 3) Top 10 states from which most of the orders were booked 4) Marital status of the consumers 5) Occupation of the consumers 6) Top 10 product categories and product IDs

Tools used in this project :

1. NumPy
2. Pandas
3. Matplotlib
4. Seaborn

Importing the Dependencies

In [3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

The Dataset

In [5]:
```
df= pd.read_csv(r'C:\Users\HP\Downloads\Python_Diwali_Sales_Analysis-main\Python_Diwali_Sales_Analysis-main\Diw
```

In [6]:
```
df.head()
```

Out[6]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	A
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	

In [7]:
```
df.shape
```

Out[7]:
```
(11251, 15)
```

This dataset has 11251 rows and 15 columns.

Checking for the null values

In [10]:
```
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
```

In [11]:
```
df.isnull().sum()
```

Out[11]:
```
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
Status             11251
unnamed1           11251
dtype: int64
```

'Status' and 'unnamed1' columns have 11251 number of null values.This huge number will affect the analysis. Hence we need to remove these two columns from the dataset to get a better and clearer insight of the dataset.

Removing columns 'Status' and 'unnamed1'

In [14]:
```
df.drop(columns=['Status','unnamed1'],axis=1,inplace=True)
```

In [15]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 13 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
dtypes: float64(1), int64(4), object(8)
memory usage: 1.1+ MB
```

In [16]:
```
df.shape
```

Out[16]:
```
(11251, 13)
```

The updated dataset has 13 columns.

In [17]:
```
pd.isnull(df)
```

Out[17]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...
11246	False	False	False	False	False	False	False	False	False	False	False	False	False
11247	False	False	False	False	False	False	False	False	False	False	False	False	False
11248	False	False	False	False	False	False	False	False	False	False	False	False	False
11249	False	False	False	False	False	False	False	False	False	False	False	False	False
11250	False	False	False	False	False	False	False	False	False	False	False	False	False

11251 rows × 13 columns

In [18]:
```
pd.isnull(df).sum()
```

Out[18]:
```
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
```

In [19]:
```
df.dropna(inplace=True)
```

In [21]:
```
df.shape
```

Out[21]:
```
(11239, 13)
```

Finally after data cleaning the dataset has 11239 rows and 13 columns.

Changing the data types

In [22]:
```
df['Amount']=df['Amount'].astype('int')
```

In [23]:
```
df['Amount'].dtypes
```

Out[23]:
```
dtype('int32')
```

In [26]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 11239 entries, 0 to 11250
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   User_ID           11239 non-null  int64 
 1   Cust_name         11239 non-null  object
 2   Product_ID        11239 non-null  object
 3   Gender            11239 non-null  object
 4   Age Group         11239 non-null  object
 5   Age               11239 non-null  int64 
 6   Marital_Status    11239 non-null  int64 
 7   State             11239 non-null  object
 8   Zone              11239 non-null  object
 9   Occupation        11239 non-null  object
 10  Product_Category  11239 non-null  object
 11  Orders            11239 non-null  int64 
 12  Amount            11239 non-null  int32 
dtypes: int32(1), int64(4), object(8)
memory usage: 1.2+ MB
```

The datatype of 'Amount' has been changed to integer from float for easier calculation.

Statistical insights of the dataset

Statistical insight of the dataset

In [27]:
```
df.describe()
```

Out[27]:

	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

The minimum order amount is of 188 rupees while the maximum order amount is of 23952 rupees.

Statistical insight of the columns :'Age','Orders','Amount'

In [28]:
```
df[['Age','Orders','Amount']].describe()
```

Out[28]:

	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

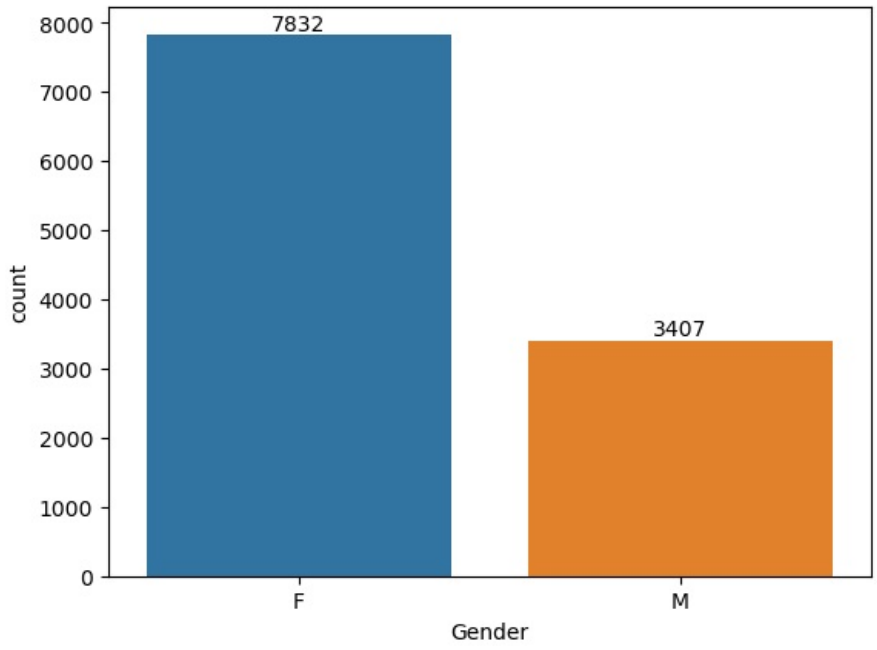
Exploratory Data Analysis

In [30]:
```
df.columns
```

Out[30]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

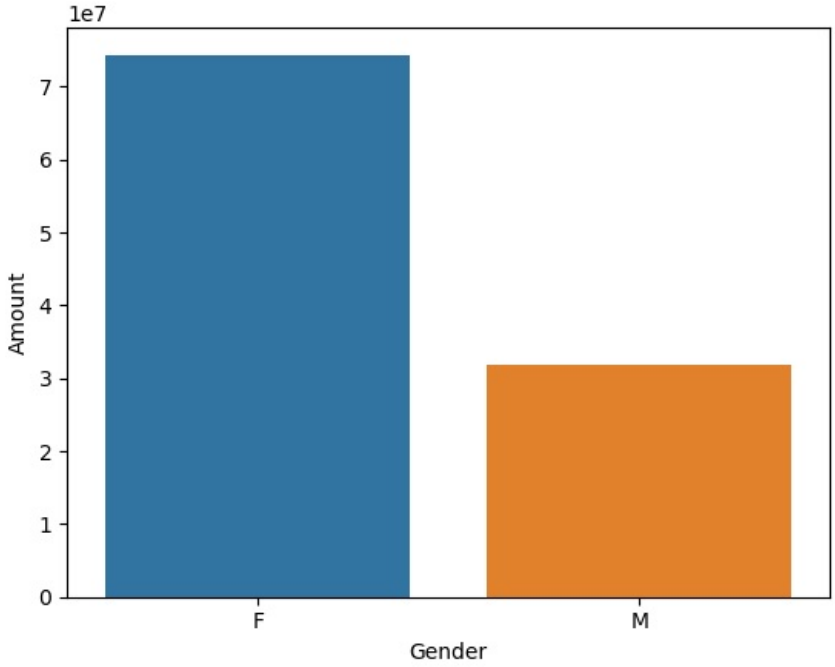
Gender

In [31]:
```
ax = sns.countplot(x='Gender',data=df)

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



In [33]:
```
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)
```

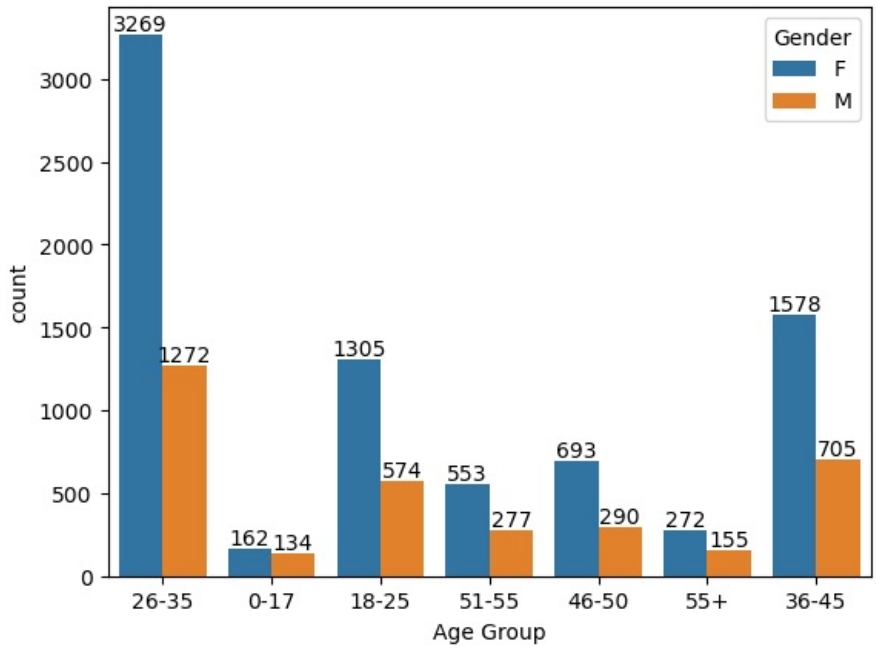
Out[33]:
```
<Axes: xlabel='Gender', ylabel='Amount'>
```



Most of the consumers are female and female consumers have booked the maximum number of orders.

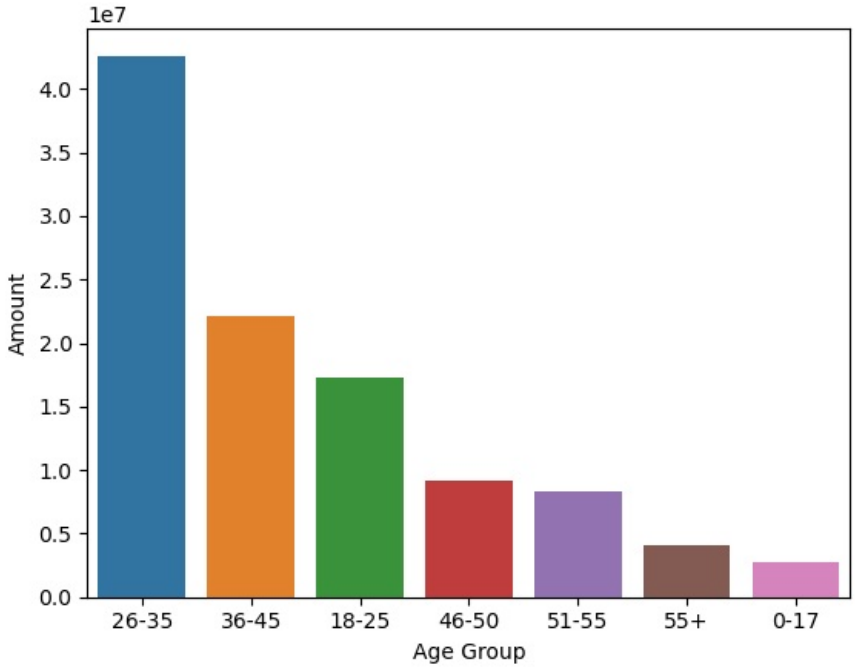
Age

In [34]:
```
ax=sns.countplot(data=df,x='Age Group',hue='Gender')

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



In [35]:
```
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)
```

Out[35]:
```
<Axes: xlabel='Age Group', ylabel='Amount'>
```

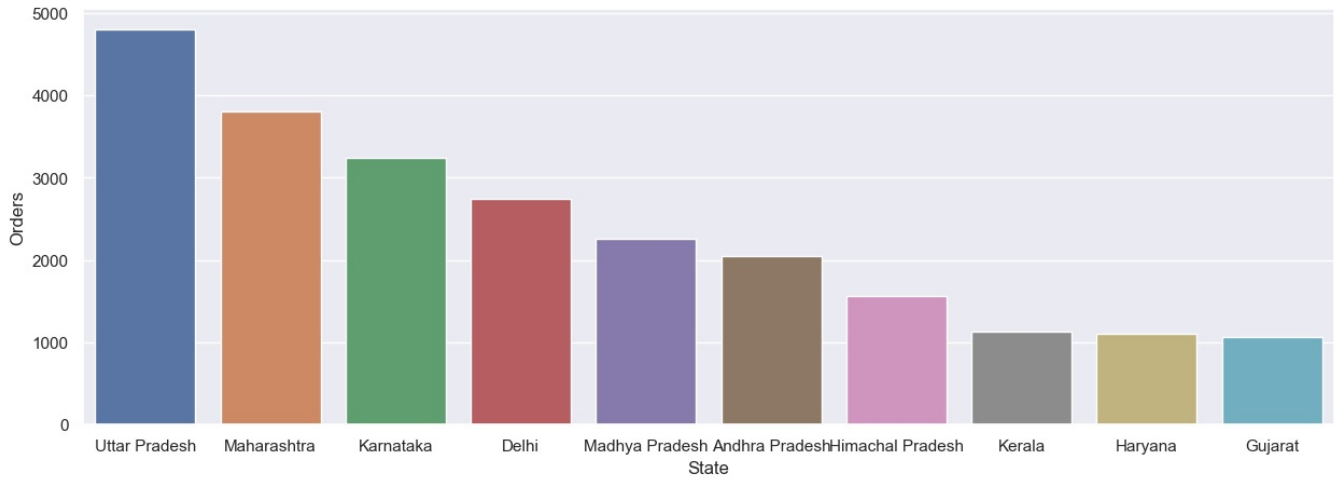


Most of the buyers are women of age group (26-35) years.

State

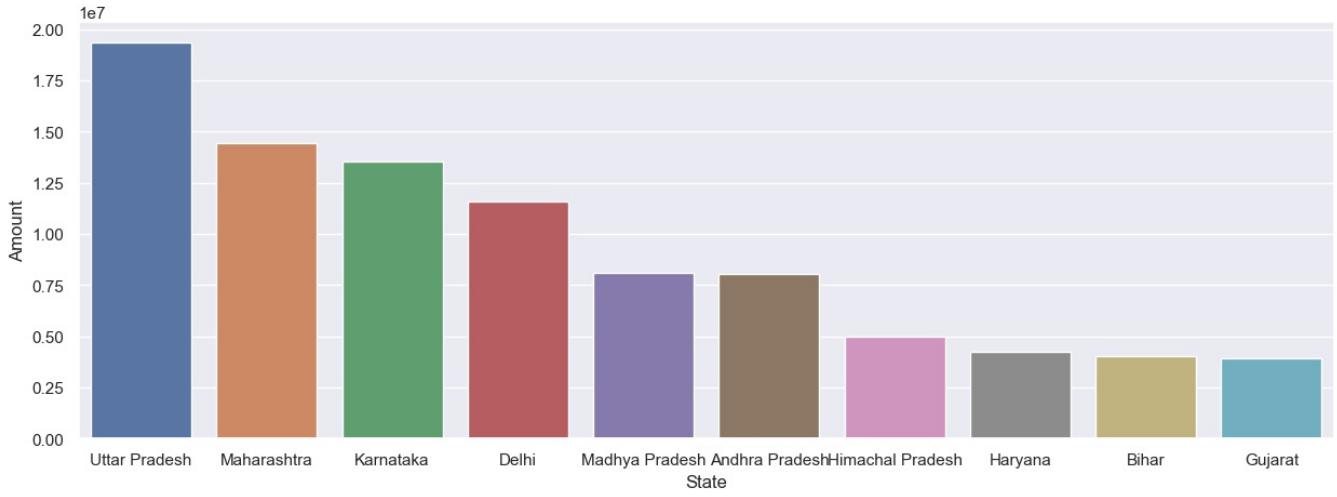
In [37]:
```
sales_state = df.groupby(['State'],as_index=False)['Orders'].sum().sort_values(by='Orders',ascending=False).hea

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

Out[37]:
```
<Axes: xlabel='State', ylabel='Orders'>
```



In [38]:
```
sales_state = df.groupby(['State'],as_index=False)['Amount'].sum().sort_values(by='Amount',ascending=False).hea

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

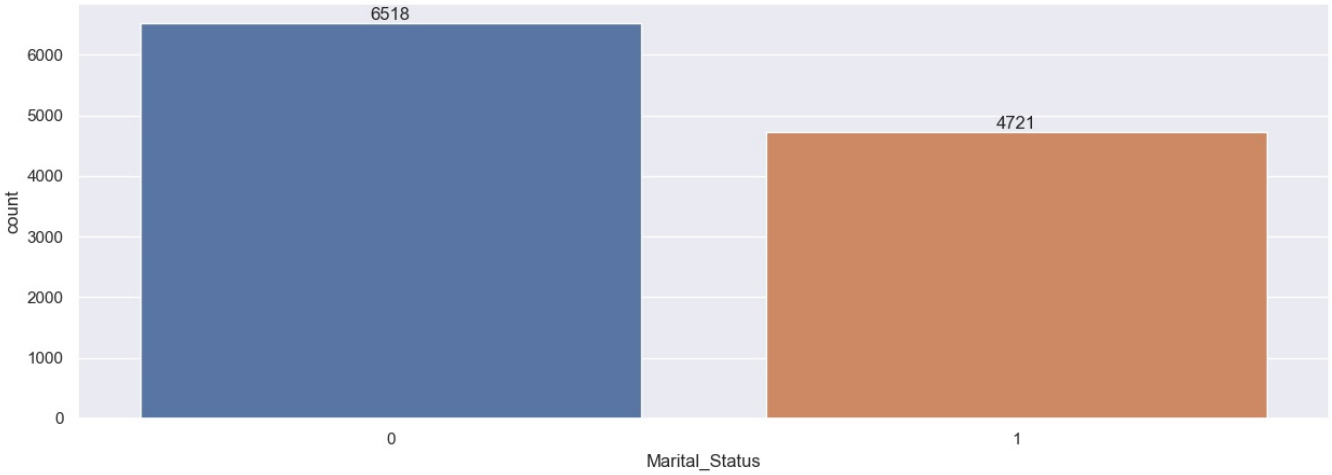
Out[38]:
```
<Axes: xlabel='State', ylabel='Amount'>
```



Most of the orders were booked from Uttar Pradesh, Maharastra, Karnataka, Delhi, Madhya Pradesh and the largest amount of order was from Delhi.

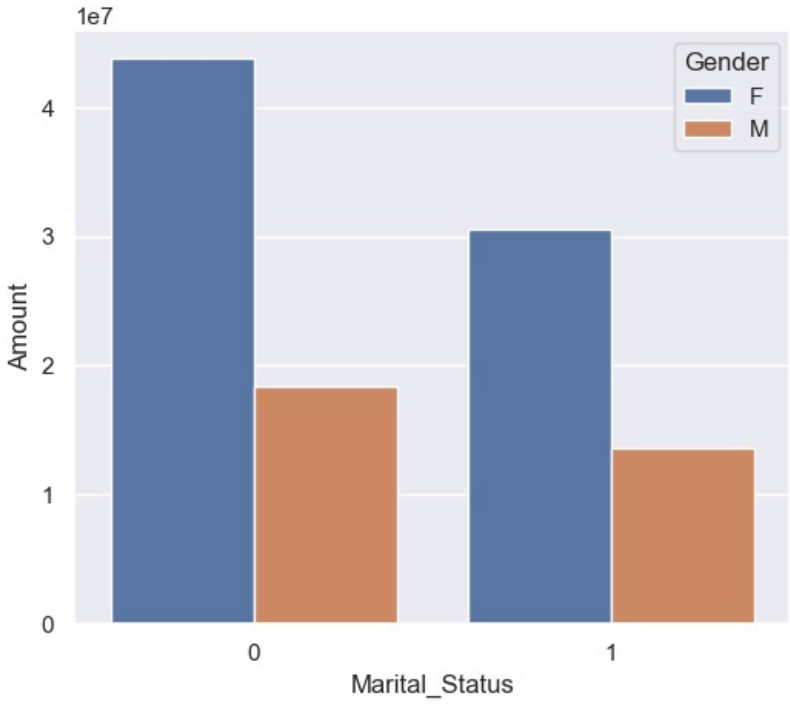
Marital status

In [39]:
```
ax=sns.countplot(data=df,x='Marital_Status')

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



In [40]:
```
sales_state = df.groupby(['Marital_Status','Gender'],as_index=False)['Amount'].sum().sort_values(by='Amount',as

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

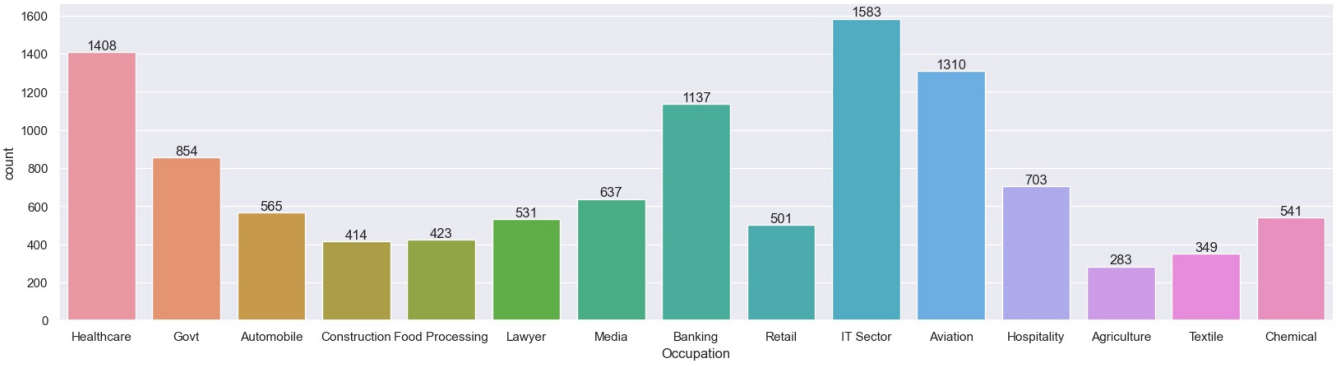
Out[40]:
```
<Axes: xlabel='Marital_Status', ylabel='Amount'>
```



The customers who are married women have booked maximum number of orders.

Occupation

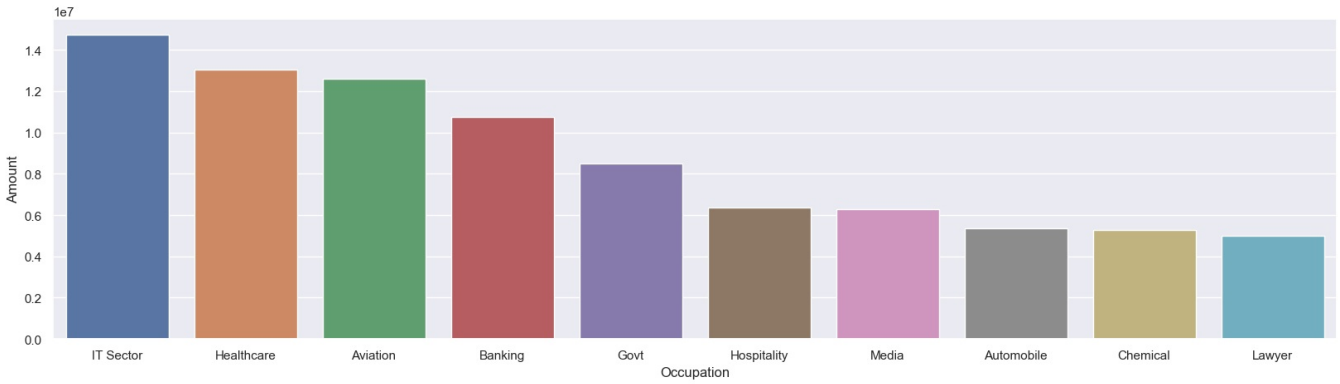
In [41]:
```
sns.set(rc={'figure.figsize':(20,5)})
ax=sns.countplot(data=df,x='Occupation')

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



In [42]:
```
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')
```

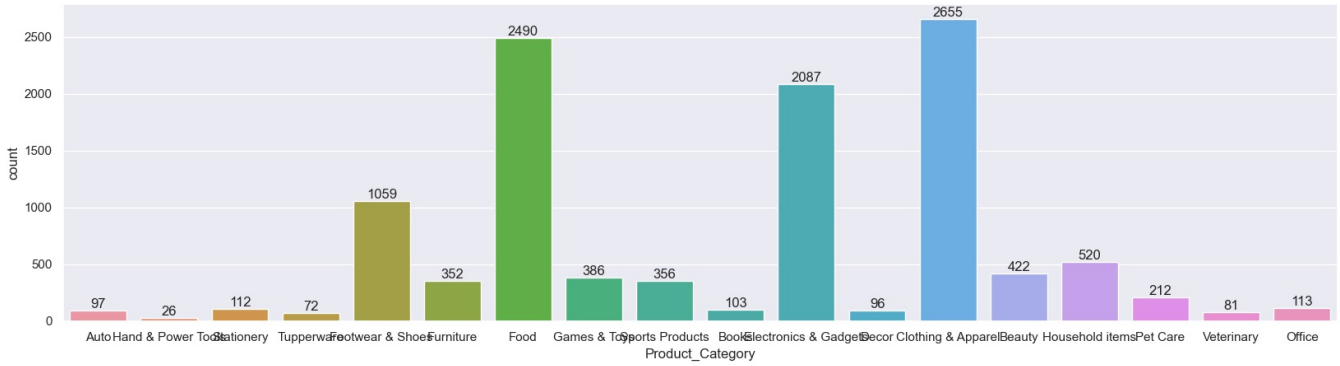
Out[42]:
```
<Axes: xlabel='Occupation', ylabel='Amount'>
```



Most of the consumers are married women working in IT sector,healthcare sector , aviation sector and their purchasing power is also very high than others.

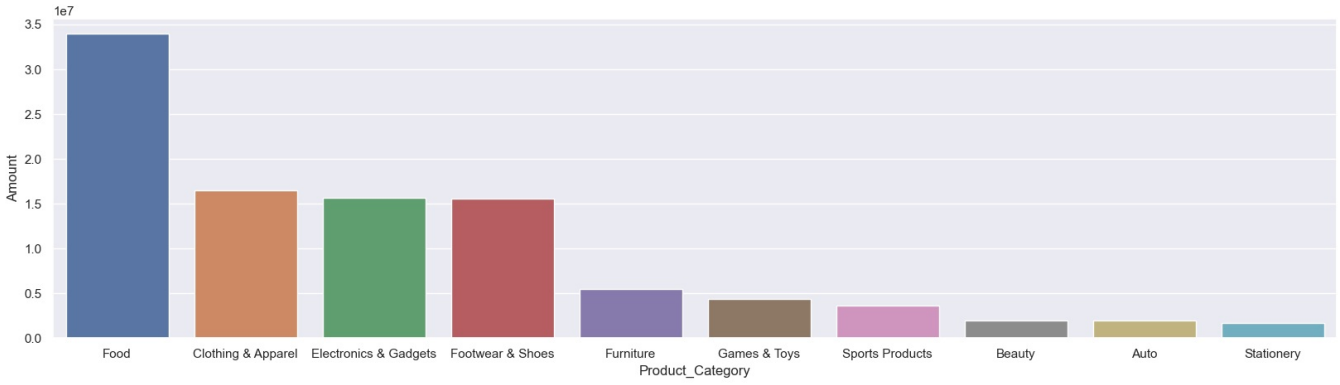
Product Category

In [43]:
```
sns.set(rc={'figure.figsize':(20,5)})
ax=sns.countplot(data=df,x='Product_Category')

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



In [44]:
```
sales_state = df.groupby(['Product_Category'],as_index=False)['Amount'].sum().sort_values(by='Amount',ascending

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

Out[44]:
```
<Axes: xlabel='Product_Category', ylabel='Amount'>
```



Order booking wise top 5 product categories : Clothing and apparel , food , electronics and gadgets , footware and shoe , household items.

Amount wise top 5 product categories : food,clothing and apparel , electronics and gadgets, footware and shoes , furniture.

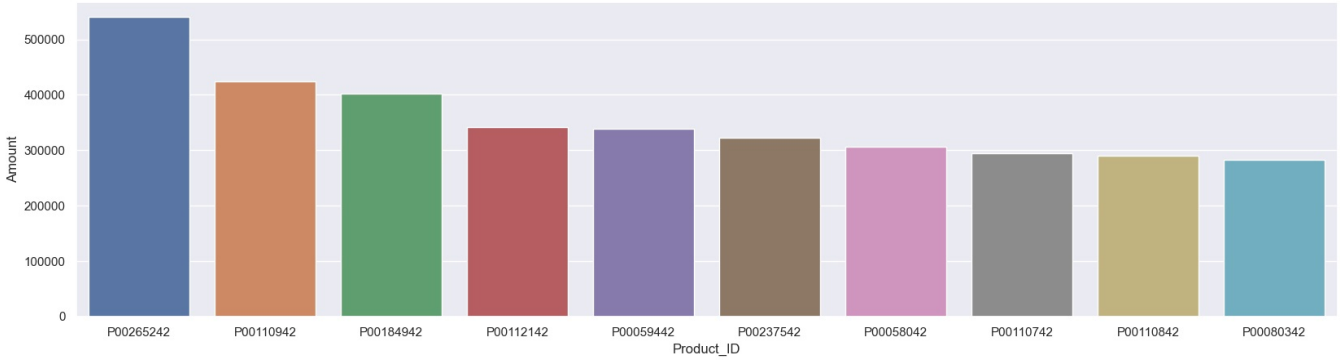
Product ID

In [45]:
```
sales_state = df.groupby(['Product_ID'],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_ID',y='Amount')
```

Out[45]:

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



Product ID P00265242 has seen the largest number of amount purchased.

Conclusion:

1) Most of the consumers are female of age group (26-35) years. 2) Top 10 states from which most of the orders were booked : Uttar Pradesh,MahaRashtra,Karnataka,Delhi,MadhyaPradesh,Andhra Pradesh,Himachal Pradesh,Haryana,Bihar,Gujarat and the largest amount of booking was from Delhi. 3) Most of the consumers are married women. 4) Most of the consumers are from IT sector, healthcare industry, aviation sector,banking sector and government service. 5) Top 5 product categories by order : Clothing and apparel , food , electronics and gadgets , footware and shoes , household items. 5) Top 5 product categories by amount : food,clothing and apparel,electronic and gadgets, footware and shoes,furniture. 6) Top 5 Product IDs : P00265242, P00110942, P00184942, P00112142, P00237542

In [ ]: