diwali-eda

September 19, 2023

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

[2]:
```
df = pd.read_csv('Diwali_Sales.csv',encoding='latin1')
df.head()
```

[2]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN	NaN
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0	NaN	NaN
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0	NaN	NaN
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0	NaN	NaN

[3]:
```
df.shape
```

[3]:
```
(11251, 15)
```

[4]:
```
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
```

[5]:
```
df.isnull().sum()
```

[5]:
```
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
```

[6]:
```
df.drop(['Status','unnamed1'],inplace=True,axis=1)
```

[7]:
```
df.head()
```

[7]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	\
0	1002903	Sanskriti	P00125942	F	26-35	28	0	
1	1000732	Kartik	P00110942	F	26-35	35	1	

```
2  1001990      Bindu  P00118542      F     26-35   35                1
3  1001425     Sudevi  P00237842      M      0-17   16                0
4  1000588       Joni  P00057942      M     26-35   28                1

            State      Zone       Occupation Product_Category  Orders   Amount
0     Maharashtra   Western       Healthcare             Auto       1  23952.0
1  Andhra Pradesh  Southern             Govt             Auto       3  23934.0
2   Uttar Pradesh   Central       Automobile             Auto       3  23924.0
3       Karnataka  Southern     Construction             Auto       2  23912.0
4         Gujarat   Western  Food Processing             Auto       2  23877.0
```

[8]:
```
df.shape
```

[8]:
```
(11251, 13)
```

[9]:
```
#droping null values from amount feature
df.dropna(inplace=True)
```

[10]:
```
df.shape
```

[10]:
```
(11239, 13)
```

[11]:
```
df['Amount'] = df['Amount'].astype('int')
```

[12]:
```
#changing the data type from float to int
df['Amount'].dtype
```

[12]:
```
dtype('int32')
```

[13]:
```
df.rename(columns={'Gender':'Sex'}) # how to rename columns
```

[13]:
```
       User_ID    Cust_name Product_ID Sex Age Group  Age  Marital_Status  \
0      1002903    Sanskriti  P00125942   F     26-35   28               0
1      1000732       Kartik  P00110942   F     26-35   35               1
2      1001990        Bindu  P00118542   F     26-35   35               1
3      1001425       Sudevi  P00237842   M      0-17   16               0
4      1000588         Joni  P00057942   M     26-35   28               1
…        …           …         …   ..       …  …             …
11246  1000695      Manning  P00296942   M     18-25   19               1
11247  1004089  Reichenbach  P00171342   M     26-35   33               0
11248  1001209        Oshin  P00201342   F     36-45   40               0
11249  1004023       Noonan  P00059442   M     36-45   37               0
11250  1002744      Brumley  P00281742   F     18-25   19               0

                State      Zone       Occupation Product_Category  Orders  \
0         Maharashtra   Western       Healthcare             Auto       1
1      Andhra Pradesh  Southern             Govt             Auto       3
```

```
2       Uttar Pradesh   Central       Automobile             Auto       3
3           Karnataka  Southern     Construction             Auto       2
4             Gujarat   Western  Food Processing             Auto       2
…                  …       …               …      …      …
11246     Maharashtra   Western         Chemical           Office       4
11247         Haryana  Northern       Healthcare       Veterinary       3
11248  Madhya Pradesh   Central          Textile           Office       4
11249       Karnataka  Southern      Agriculture           Office       3
11250     Maharashtra   Western       Healthcare           Office       3

       Amount
0       23952
1       23934
2       23924
3       23912
4       23877
…         …
11246     370
11247     367
11248     213
11249     206
11250     188

[11239 rows x 13 columns]
```

[14]:
```
df.describe()
```

[14]:
```
            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
```

[15]:
```
df['State'].describe()
```

[15]:
```
count             11239
unique               16
top       Uttar Pradesh
freq                1944
Name: State, dtype: object
```

[16]:
```
df['State'].value_counts()
```

[16]:
```
Uttar Pradesh       1944
Maharashtra         1525
Karnataka           1304
Delhi               1104
Madhya Pradesh       921
Andhra Pradesh       811
Himachal Pradesh     608
Kerala               453
Haryana              452
Bihar                434
Gujarat              427
Jharkhand            380
Uttarakhand          320
Rajasthan            231
Punjab               200
Telangana            125
Name: State, dtype: int64
```

[17]:
```
df['State'].count()
```

[17]:
```
11239
```

[18]:
```
df['State'].nunique()
```

[18]:
```
16
```

[ ]:
```

```

1 Exploratory Data Analysis

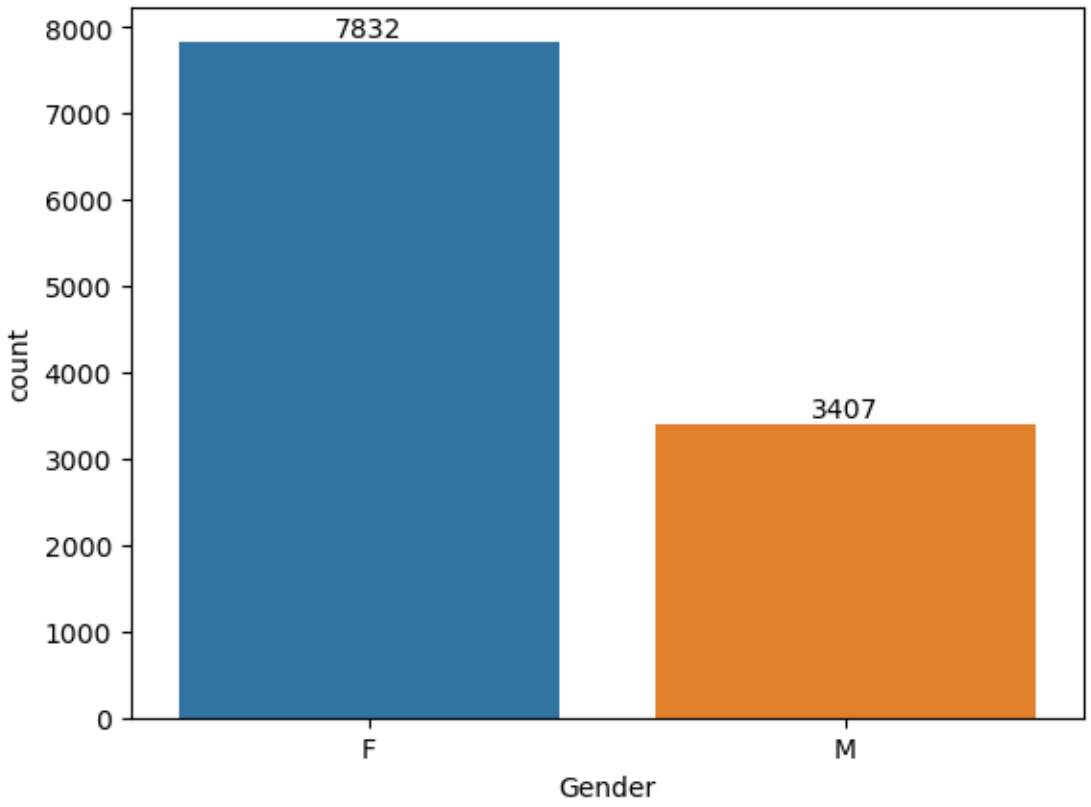
[19]:
```
df.head()
```

[19]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877

[ ]:

[20]:
```
#Gender
ax = sns.countplot(x='Gender',data=df)

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



#OBSERVATION

1.Female made more transaction than male

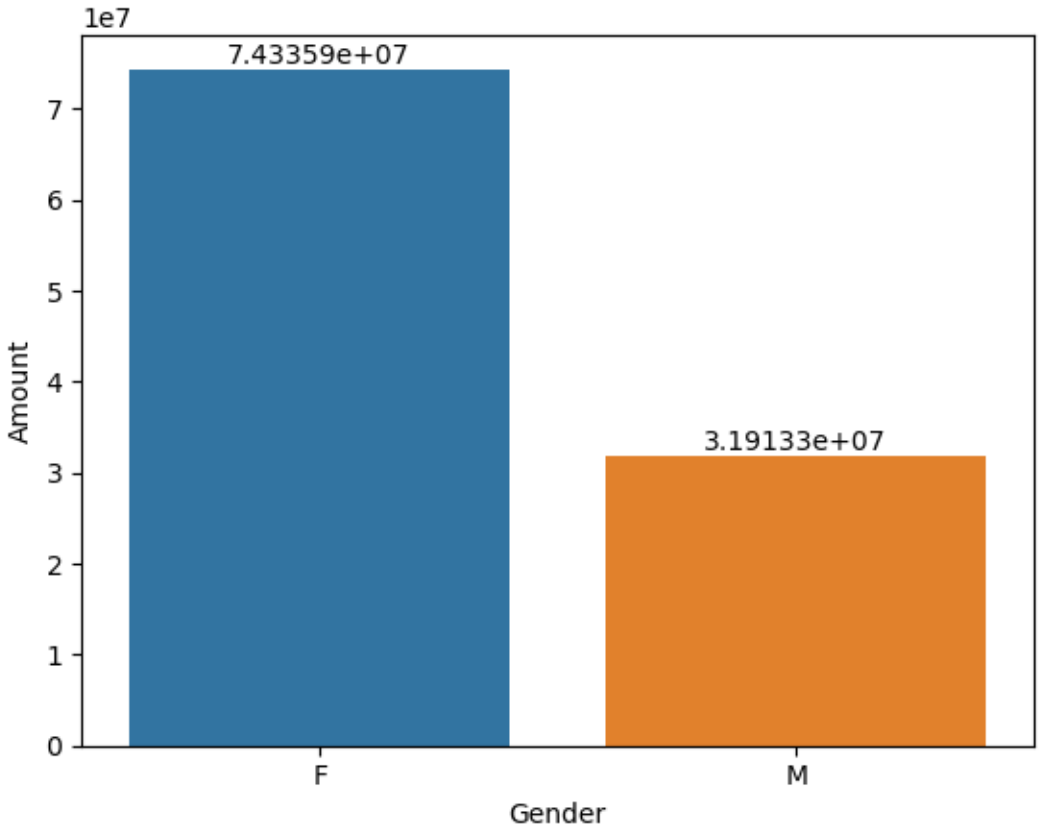
[21]:
```
#Amount  by gender
amt_gen = df.groupby(['Gender'])['Amount'].sum().reset_index()
amt_gen
```

[21]:

	Gender	Amount
0	F	74335853
1	M	31913276

[22]:
```
ax=sns.barplot(x='Gender',y='Amount',data=amt_gen)
```

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



#OBSERVATION

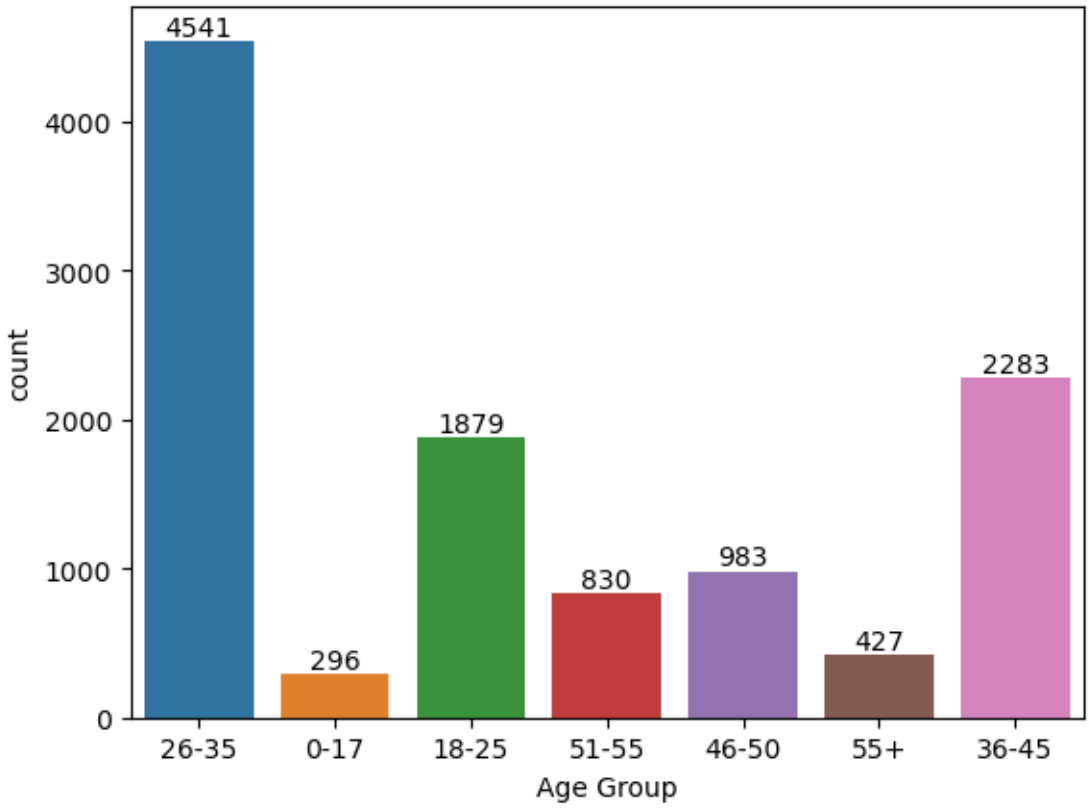
1.Female Purchase more than Men

[23]:
```
df.columns
```

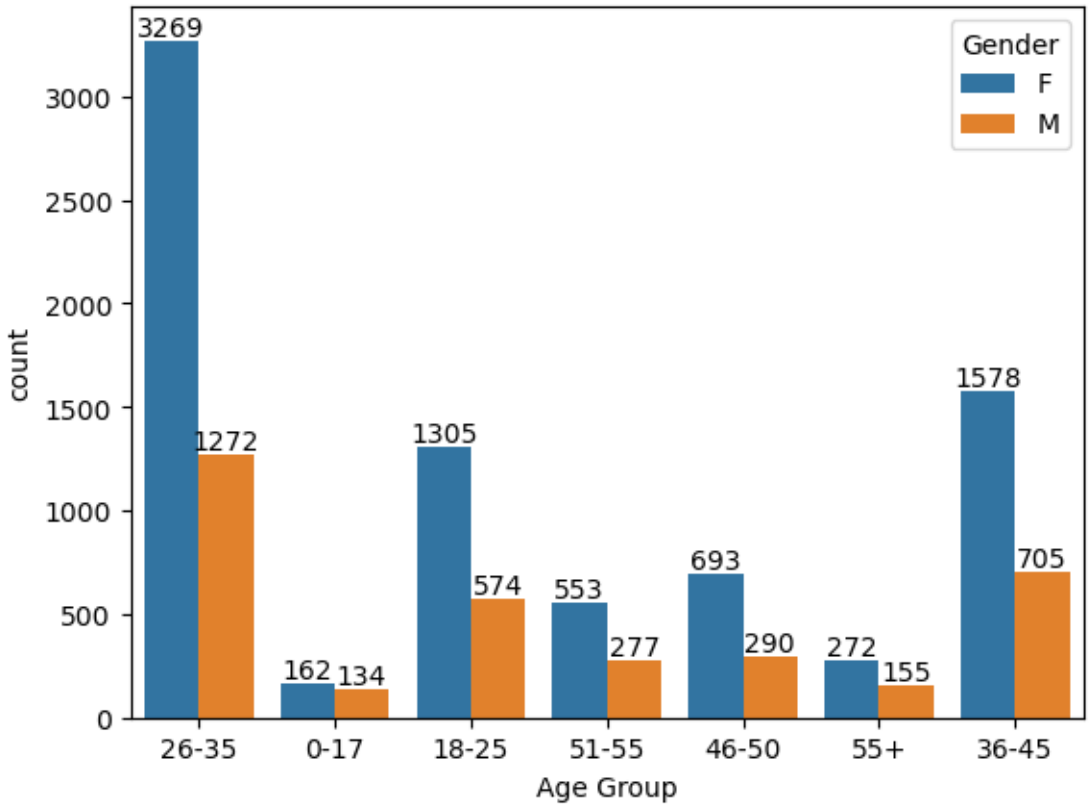
[23]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

[ ]:
```

```

[24]:
```
#age group count
ax = sns.countplot(x='Age Group',data=df)
for x in ax.containers:
    ax.bar_label(x)
```



[25]:
```
ax = sns.countplot(x='Age Group',data=df,hue='Gender')

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



#OBSERVATION

1.Most of the customers are from 26-35 Age Group

2.Female are dominant in 26-35 Age Group

[ ]:

[ ]:

[35]:
```
# Amount vs Age Group
AgeGroup_Amount = df.groupby(['Age Group'])['Amount'].sum().reset_index().
 ↪sort_values(by ='Amount', ascending=False)
AgeGroup_Amount
```

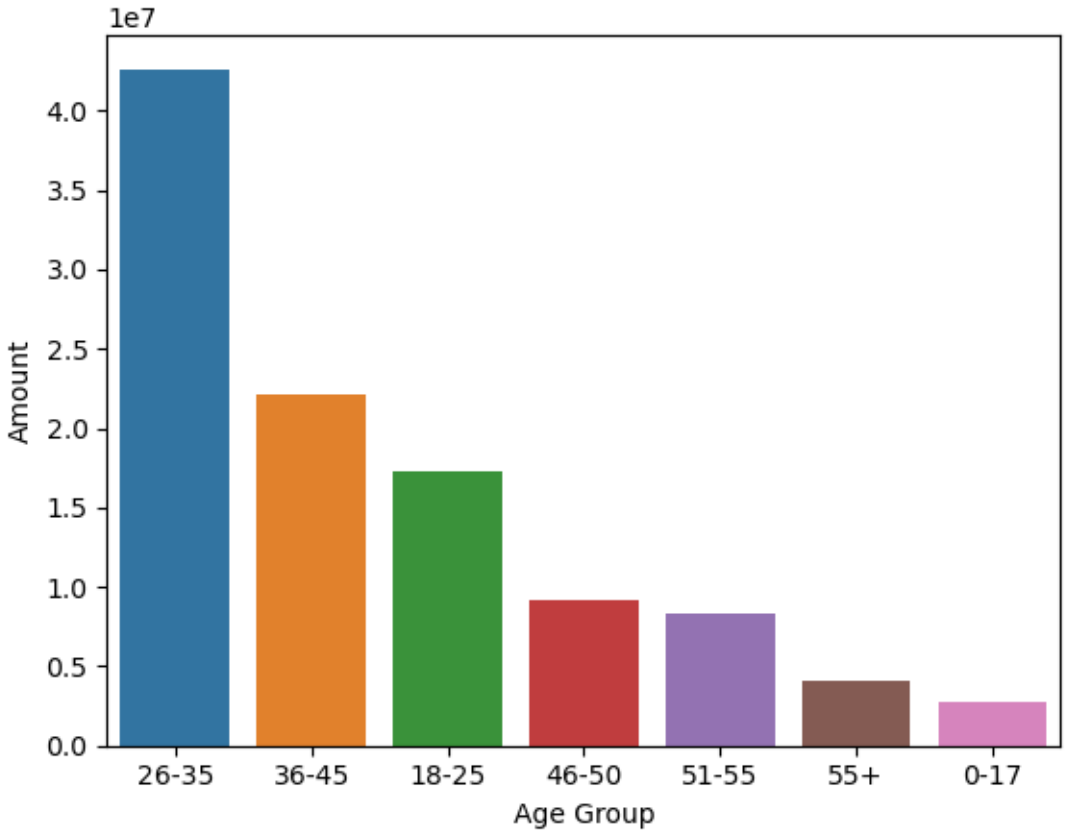
[35]:

	Age Group	Amount
2	26-35	42613442
3	36-45	22144994
1	18-25	17240732
4	46-50	9207844
5	51-55	8261477
6	55+	4080987

```
0      0-17   2699653
```

[38]:
```
sns.barplot(x='Age Group',y='Amount', data=AgeGroup_Amount)
```

[38]:
```
<AxesSubplot:xlabel='Age Group', ylabel='Amount'>
```



#OBSERVATION

1.Most of the Amount spend under 26-25 Age group

[42]:
```
AgeGender_amount = df.groupby(['Age Group','Gender'])['Amount'].sum().
↪reset_index().sort_values(by='Amount',ascending=False)
AgeGender_amount
```

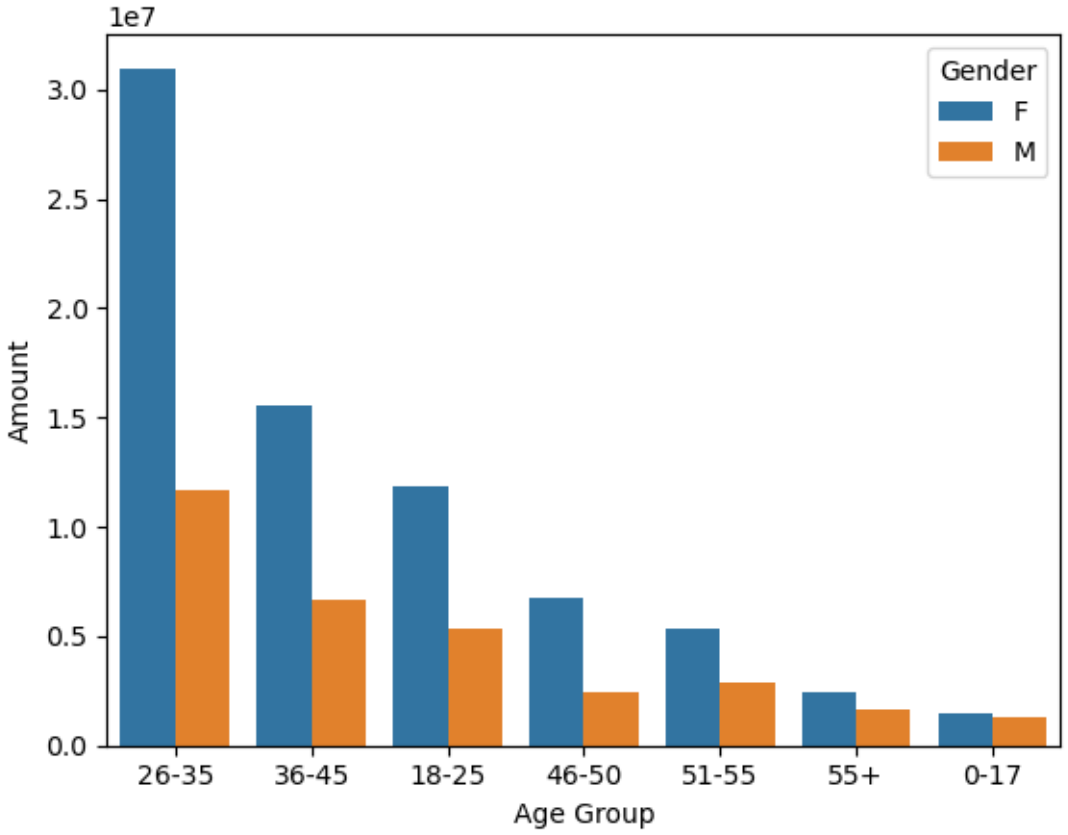
[42]:

	Age Group	Gender	Amount
4	26-35	F	30963953
6	36-45	F	15509956
2	18-25	F	11887003
5	26-35	M	11649489
8	46-50	F	6743393
7	36-45	M	6635038

```
10      51-55      F   5385208
3       18-25      M   5353729
11      51-55      M   2876269
9       46-50      M   2464451
12        55+      F   2404931
13        55+      M   1676056
0       0-17       F   1441409
1       0-17       M   1258244
```

[48]:
```
sns.barplot(x='Age Group',y='Amount' ,data = AgeGender_amount ,hue='Gender')
```

[48]:
```
<AxesSubplot:xlabel='Age Group', ylabel='Amount'>
```



#Observation

1.From all the Age Groups,Female dominates in purchasing

[ ]:

[ ]:

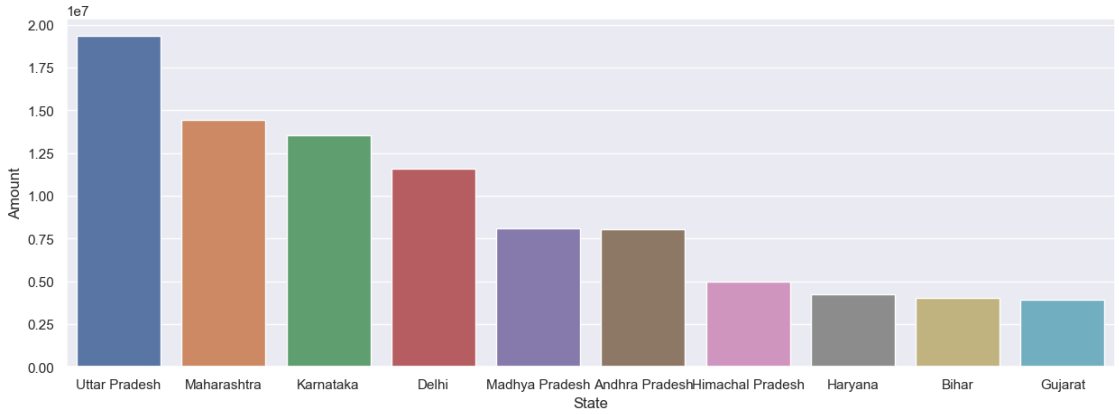
[59]:
```
#State by Amount

state_amount = df.groupby(['State'])['Amount'].sum().reset_index().
 ↪sort_values(by='Amount',ascending=False).head(10)
state_amount
```

[59]:

	State	Amount
14	Uttar Pradesh	19374968
10	Maharashtra	14427543
7	Karnataka	13523540
2	Delhi	11603818
9	Madhya Pradesh	8101142
0	Andhra Pradesh	8037146
5	Himachal Pradesh	4963368
4	Haryana	4220175
1	Bihar	4022757
3	Gujarat	3946082

[63]:
```
sns.set(rc={'figure.figsize':(15,5)})

sns.barplot(x='State',y='Amount',data=state_amount)
```

[63]: <AxesSubplot:xlabel='State', ylabel='Amount'>



[74]:
```
#State by Amount

State_Amt_Ord = df.groupby(['State'])['Orders'].sum().reset_index().
 ↪sort_values(by='Orders',ascending=False).head(10)
State_Amt_Ord
```

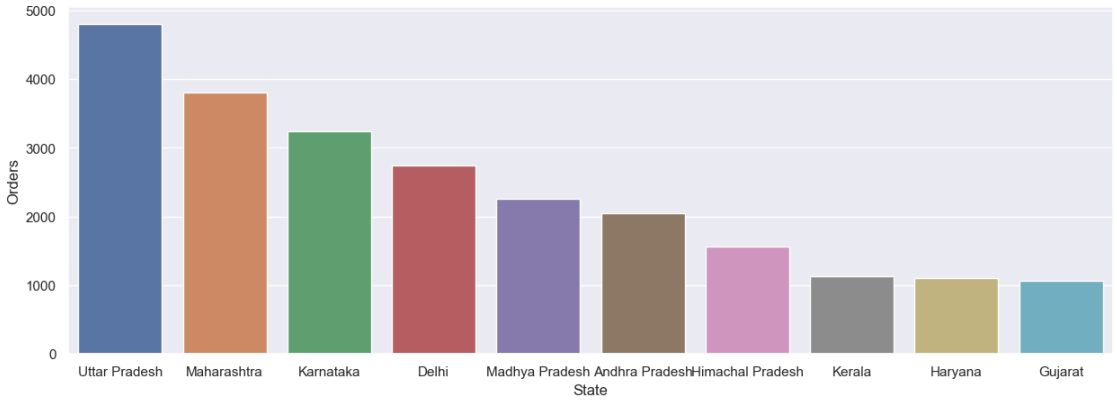
[74]:

	State	Orders
14	Uttar Pradesh	4807

```
10       Maharashtra    3810
7          Karnataka    3240
2              Delhi    2740
9      Madhya Pradesh    2252
0      Andhra Pradesh    2051
5   Himachal Pradesh    1568
8             Kerala    1137
4            Haryana    1109
3            Gujarat    1066
```

[75]:
```
sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x='State',y='Orders',data=State_Amt_Ord)
```

[75]:
```
<AxesSubplot:xlabel='State', ylabel='Orders'>
```



#Oberservation

1.Top 5 state by Amount and Order are UP,Maharashtra,Karnataka,Delhi,MP

[ ]:
```
```

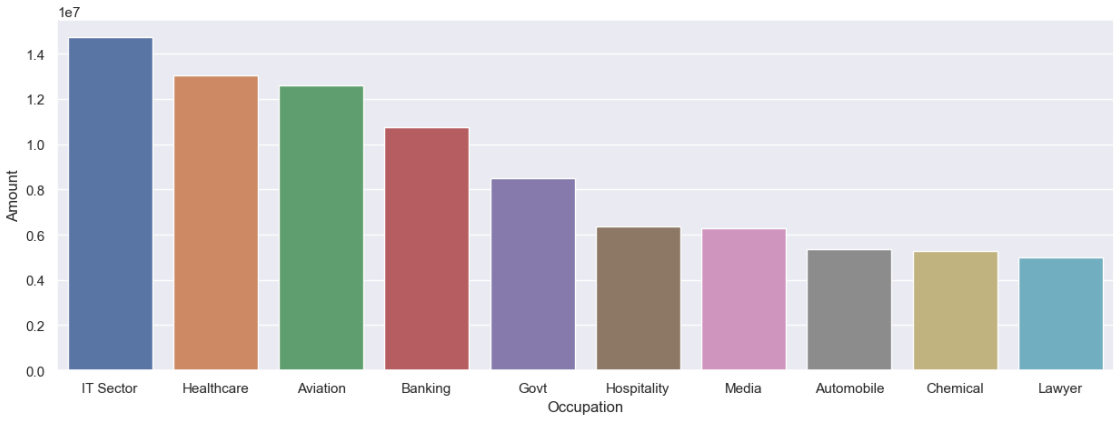
[79]:
```
#Occupation by Amount

Occup_Amt = df.groupby(['Occupation'])['Amount'].sum().reset_index().
 ↪sort_values(by='Amount',ascending=False).head(10)
Occup_Amt
```

[79]:
```
     Occupation    Amount
10    IT Sector  14755079
8    Healthcare  13034586
2      Aviation  12602298
3       Banking  10770610
7          Govt   8517212
```

```
9   Hospitality   6376405
12        Media   6295832
1    Automobile   5368596
4      Chemical   5297436
11       Lawyer   4981665
```

[80]:
```
sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x='Occupation',y='Amount',data=Occup_Amt)
```

[80]: <AxesSubplot:xlabel='Occupation', ylabel='Amount'>

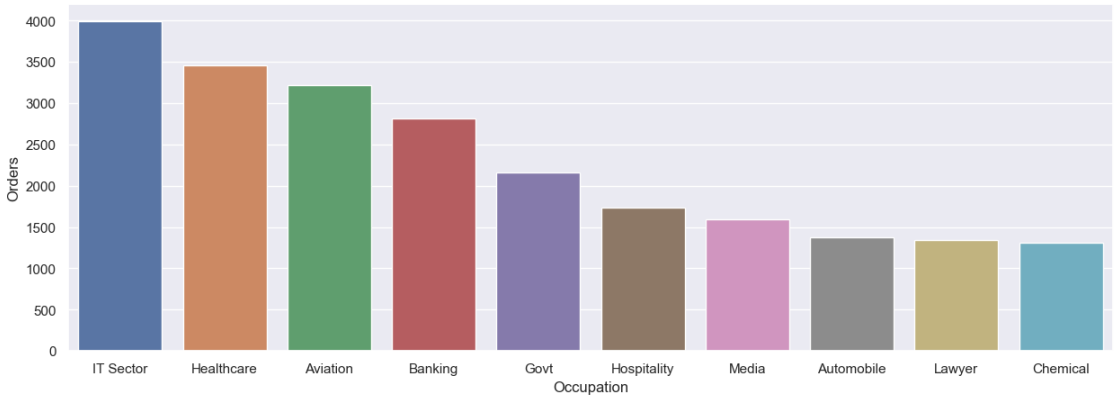


[82]:
```
#Occupation by Orders
Occup_ords = df.groupby(['Occupation'])['Orders'].sum().reset_index().
 ↪sort_values(by='Orders',ascending=False).head(10)
Occup_ords
```

[82]:
```
      Occupation  Orders
10     IT Sector    3997
8     Healthcare    3455
2       Aviation    3215
3        Banking    2817
7           Govt    2155
9   Hospitality    1739
12        Media    1596
1    Automobile    1371
11       Lawyer    1344
4      Chemical    1309
```

[83]:
```
sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x='Occupation',y='Orders',data=Occup_ords)
```

[83]: <AxesSubplot:xlabel='Occupation', ylabel='Orders'>



#OBSERVATION

1.Customers who are purchasing more are from IT Sector,Healthcare,Aviation,Banking and Govt Occupation

[ ]:

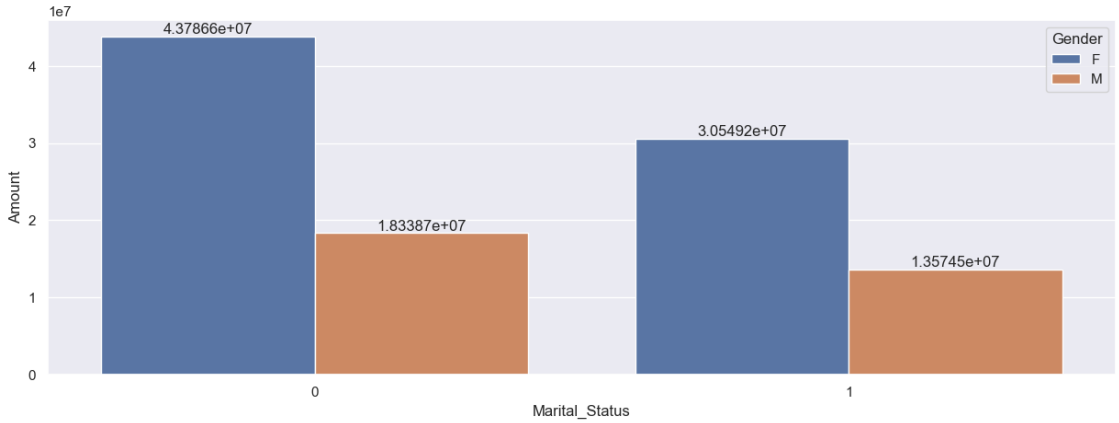
[ ]:
```
# Marital_Status by Amount
```

[88]:
```
MS_gender_Amount = df.groupby(['Marital_Status','Gender'])['Amount'].sum().
↪reset_index()
MS_gender_Amount
```

[88]:

	Marital_Status	Gender	Amount
0	0	F	43786646
1	0	M	18338738
2	1	F	30549207
3	1	M	13574538

[91]:
```
ax = sns.
↪barplot(x='Marital_Status',y='Amount',data=MS_gender_Amount,hue='Gender')

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



#OBSERVATION

Married Women Purchased more than Unmaried Women

[ ]:

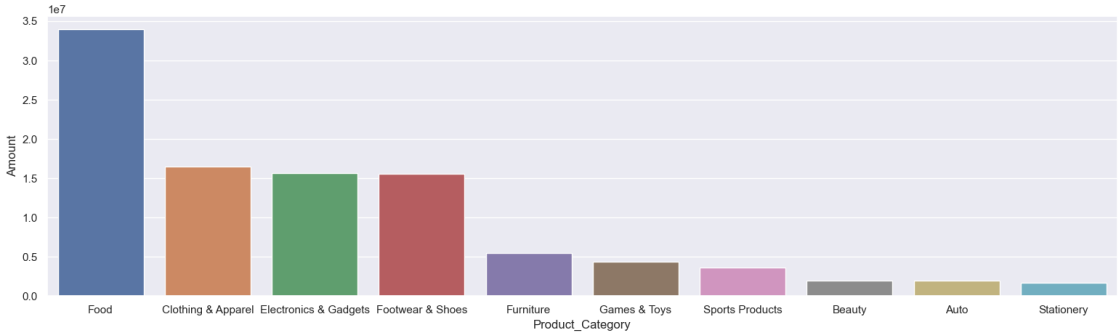
[98]:
```
#Product category by Amount

PC_Amt = df.groupby(['Product_Category'])['Amount'].sum().reset_index().
 ↪sort_values(by='Amount',ascending=False).head(10)
PC_Amt
```

[98]:

	Product_Category	Amount
6	Food	33933883
3	Clothing & Apparel	16495019
5	Electronics & Gadgets	15643846
7	Footwear & Shoes	15575209
8	Furniture	5440051
9	Games & Toys	4331694
14	Sports Products	3635933
1	Beauty	1959484
0	Auto	1958609
15	Stationery	1676051

[109]:
```
sns.set(rc={'figure.figsize':(19,5)})

sns.barplot(x='Product_Category',y='Amount',data=PC_Amt)
```

[109]:
```
<AxesSubplot:xlabel='Product_Category', ylabel='Amount'>
```



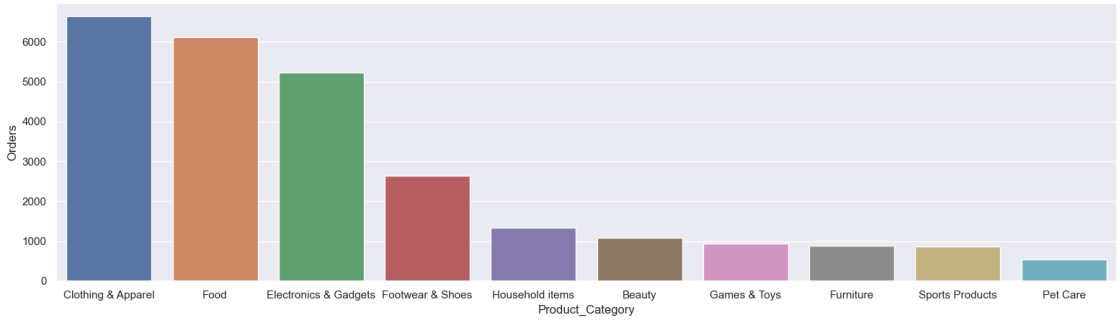
[104]:
```
#Product category by Orders
PC_Ords = df.groupby(['Product_Category'])['Orders'].sum().reset_index().
 ↪sort_values(by='Orders',ascending=False).head(10)
PC_Ords
```

[104]:

	Product_Category	Orders
3	Clothing & Apparel	6634
6	Food	6110
5	Electronics & Gadgets	5226
7	Footwear & Shoes	2646
11	Household items	1331
1	Beauty	1086
9	Games & Toys	940
8	Furniture	889
14	Sports Products	870
13	Pet Care	536

[108]:
```
sns.set(rc={'figure.figsize':(19,5)})

sns.barplot(x='Product_Category',y='Orders',data=PC_Ords)
```

[108]: <AxesSubplot:xlabel='Product_Category', ylabel='Orders'>



#OBSERVATION from above product category by Amount and orders

1.Clothing & Apparel have more orders than Food caetegory but Food Category is top 1 in terms of revenue

2.Clothing & Apparel,Food And Electronics & Gadgets Are Top 3 in terms Of revenue and orders

[110]:
```
df.head()
```

[110]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: