Diwali Sales Analysis

Importing Libraries-

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

import plotly.express as px
import plotly.graph_objects as go
```

Loading Dataset-

In [3]:
```
data = pd.read_csv("Diwali Sales Data.csv", encoding = "latin-1")
data.head()
```

Out[3]:

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

In [4]:
```
data.shape
```

Out[4]:
```
(11251, 15)
```

In [5]:
```
data.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
```

Dropping Unrelated/Blank Columns-

In [6]:
```
data.drop(['Status', 'unnamed1'], axis=1, inplace=True, errors='ignore')
data.head()
```

Out[6]:

	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	23952.0
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0

Checking Null Values-

In [7]:
```
data.isnull().sum()
```

Out[7]:
```
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
```

Droping Null Values-

In [8]:
```
data.dropna(inplace=True)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 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  float64
dtypes: float64(1), int64(4), object(8)
memory usage: 1.2+ MB
```

Changing Datatype-

In [9]:
```
data["Amount"]=data["Amount"].astype("int")
data["Amount"].dtypes
```

Out[9]:
```
dtype('int32')
```

In [10]:
```
data.columns
```

Out[10]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

Rename Column-

In [11]:
```
data.rename(columns={"Marital_Status" : "Shadi"}, inplace=True)
data.columns
```

Out[11]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Shadi', 'State', 'Zone', 'Occupation', 'Product_Category', 'Orders',
       'Amount'],
      dtype='object')
```

Discriptive Analysis-

In [12]:
```
data.describe()
```

Out[12]:

	User_ID	Age	Shadi	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

In [13]:
```
#use decribe for sepecific columns-
data[["Age", "Orders", "Amount"]].describe()
```

Out[13]:

	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 Analsysis-

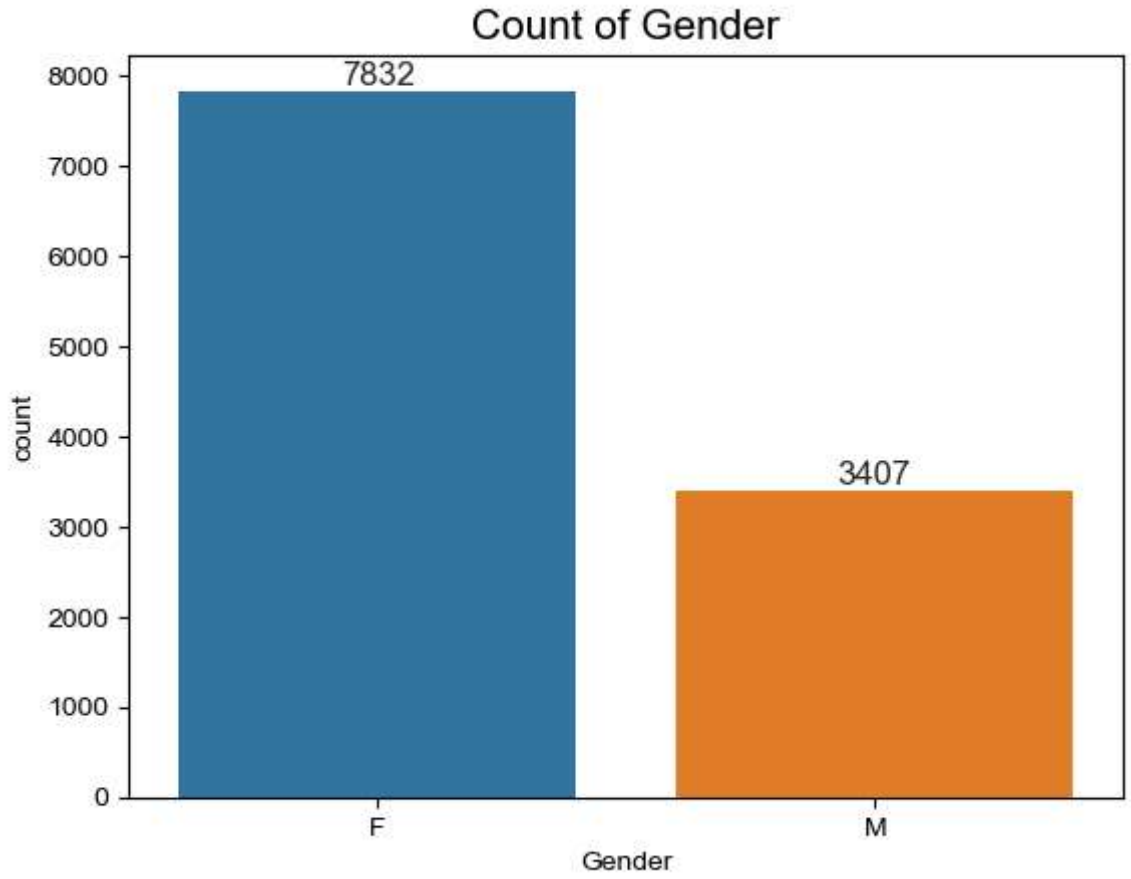
(i)Gender-

In [14]:
```
data["Gender"].value_counts()
```

Out[14]:
```
Gender
F    7832
M    3407
Name: count, dtype: int64
```

In [15]:
```
fig = sns.countplot(data, x="Gender")
sns.set(rc={'figure.figsize':(15,5)})

for bars in fig.containers:
    fig.bar_label(bars)
plt.title("Count of Gender", fontsize=15)
plt.show()
```



Gender v/s Total Amount-

In [16]:
```
data.head(2)
```

Out[16]:

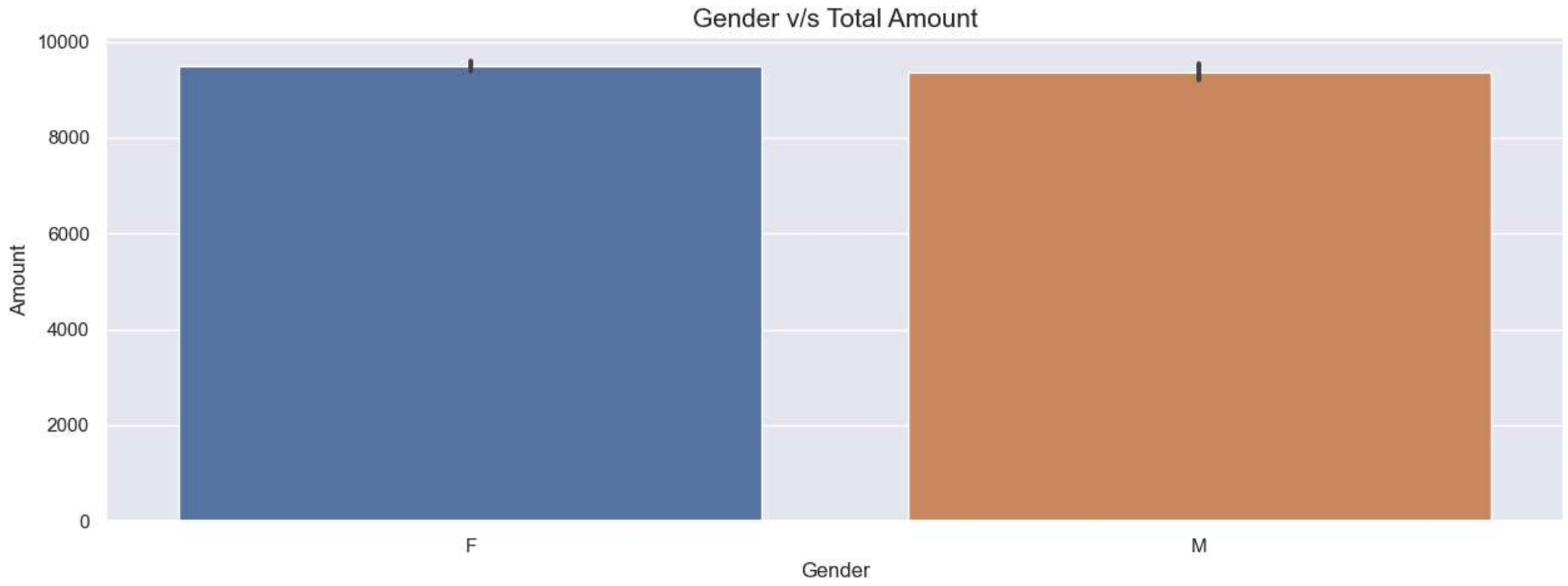
	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934

In [17]:
```
Amount_by_Gender = data.groupby("Gender")["Amount"].sum().reset_index()
Amount_by_Gender
```

Out[17]:

	Gender	Amount
0	F	74335853
1	M	31913276

In [18]:
```
fig = sns.barplot(data, x="Gender", y="Amount")
sns.set(rc={'figure.figsize':(15,5)})
plt.title("Gender v/s Total Amount", fontsize=15)
plt.show()
```



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

(ii) Age-

In [19]:
```
data.head()
```

Out[19]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	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

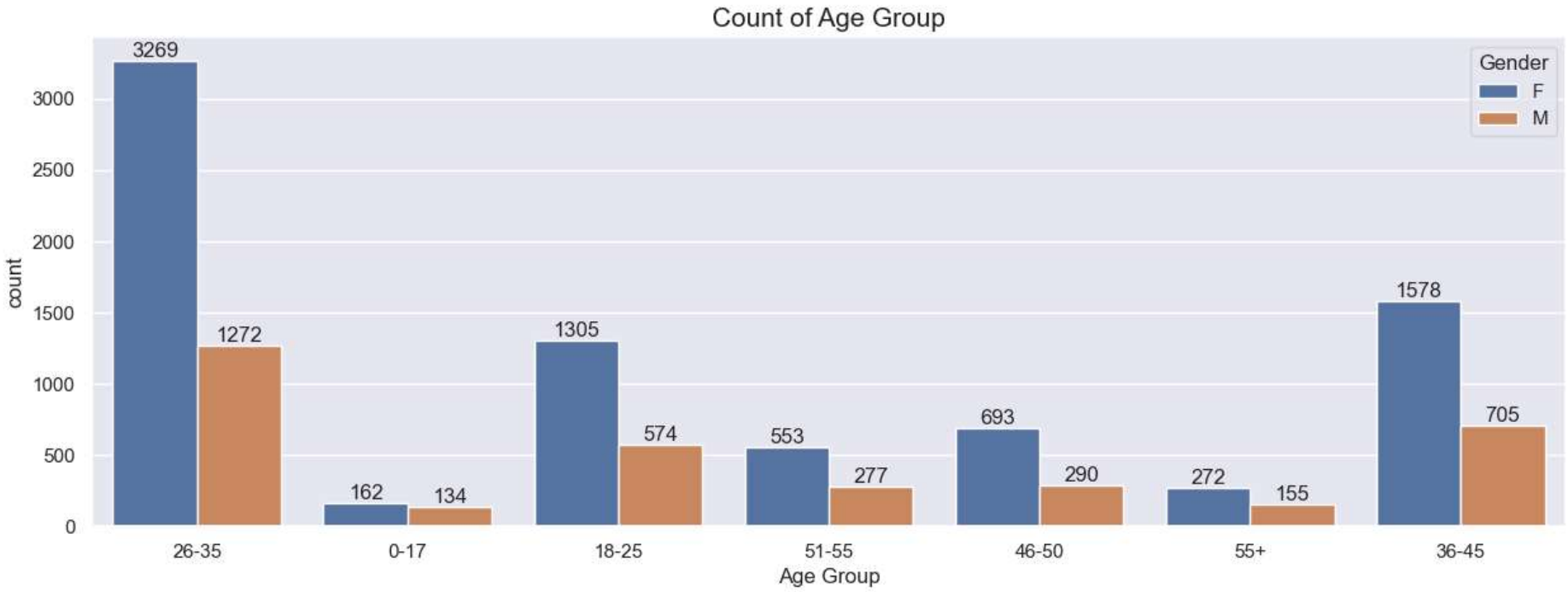
In [20]:
```
data["Age Group"].value_counts()
```

Out[20]:
```
Age Group
26-35    4541
36-45    2283
18-25    1879
46-50     983
51-55     830
55+       427
0-17      296
Name: count, dtype: int64
```

In [21]:
```
fig=sns.countplot(data, x="Age Group", hue="Gender")

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

for bars in fig.containers:
    fig.bar_label(bars)
plt.title("Count of Age Group", fontsize=15)
plt.show()
```



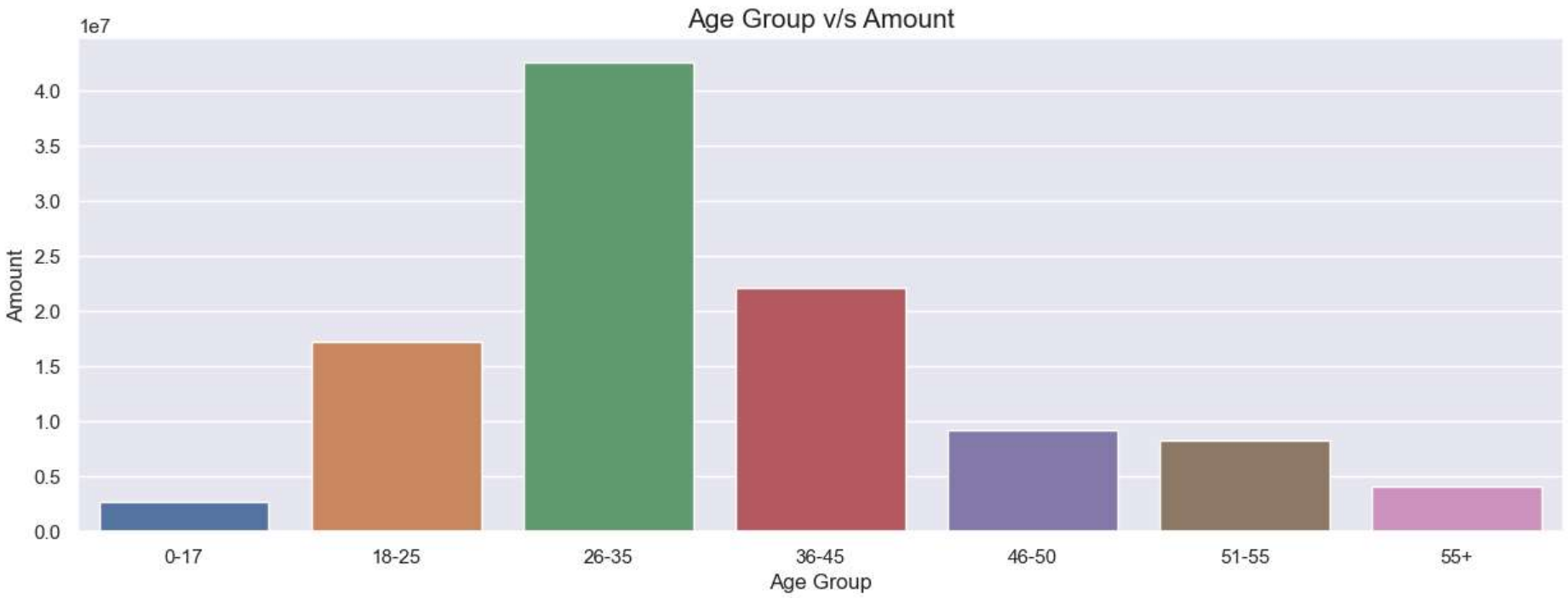
Age Group v/s Total Amount-

In [22]:
```
Amount_by_Age = data.groupby("Age Group")["Amount"].sum().reset_index()
Amount_by_Age
```

Out[22]:

	Age Group	Amount
0	0-17	2699653
1	18-25	17240732
2	26-35	42613442
3	36-45	22144994
4	46-50	9207844
5	51-55	8261477
6	55+	4080987

In [23]:
```
fig = sns.barplot(Amount_by_Age, x="Age Group", y="Amount")
sns.set(rc={'figure.figsize':(15,5)})
plt.title("Age Group v/s Amount", fontsize=15)
plt.show()
```



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

(iii) State-

In [24]:
```
Top_States = data.sort_values(by="Orders", ascending=False).head(10)
Top_States["State"]
```

Out[24]:
```
2234    Uttar Pradesh
7299          Gujarat
7291      Maharashtra
7288          Gujarat
2596        Karnataka
7285      Uttarakhand
2602        Karnataka
2603    Uttar Pradesh
7278      Maharashtra
7277        Karnataka
Name: State, dtype: object
```

In [25]:
```
orders_by_states = data.groupby("State")["Orders"].sum().reset_index()
orders_by_states
```

Out[25]:

	State	Orders
0	Andhra Pradesh	2051
1	Bihar	1062
2	Delhi	2740
3	Gujarat	1066
4	Haryana	1109
5	Himachal Pradesh	1568
6	Jharkhand	953
7	Karnataka	3240
8	Kerala	1137
9	Madhya Pradesh	2252
10	Maharashtra	3810
11	Punjab	495
12	Rajasthan	555
13	Telangana	312
14	Uttar Pradesh	4807
15	Uttarakhand	824

Total number of Orders from Top 10 states-

In [26]:
```
Orders_by_State = data.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)
Orders_by_State
```

Out[26]:

	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

In [27]:

```
fig = sns.barplot(Orders_by_State, x="State", y="Orders")
sns.set(rc={"figure.figsize":(15,5)})
plt.title("Top 10 States by Orders", fontsize=15)
plt.show()
```

Total Amount/Sales from Top 10 States-

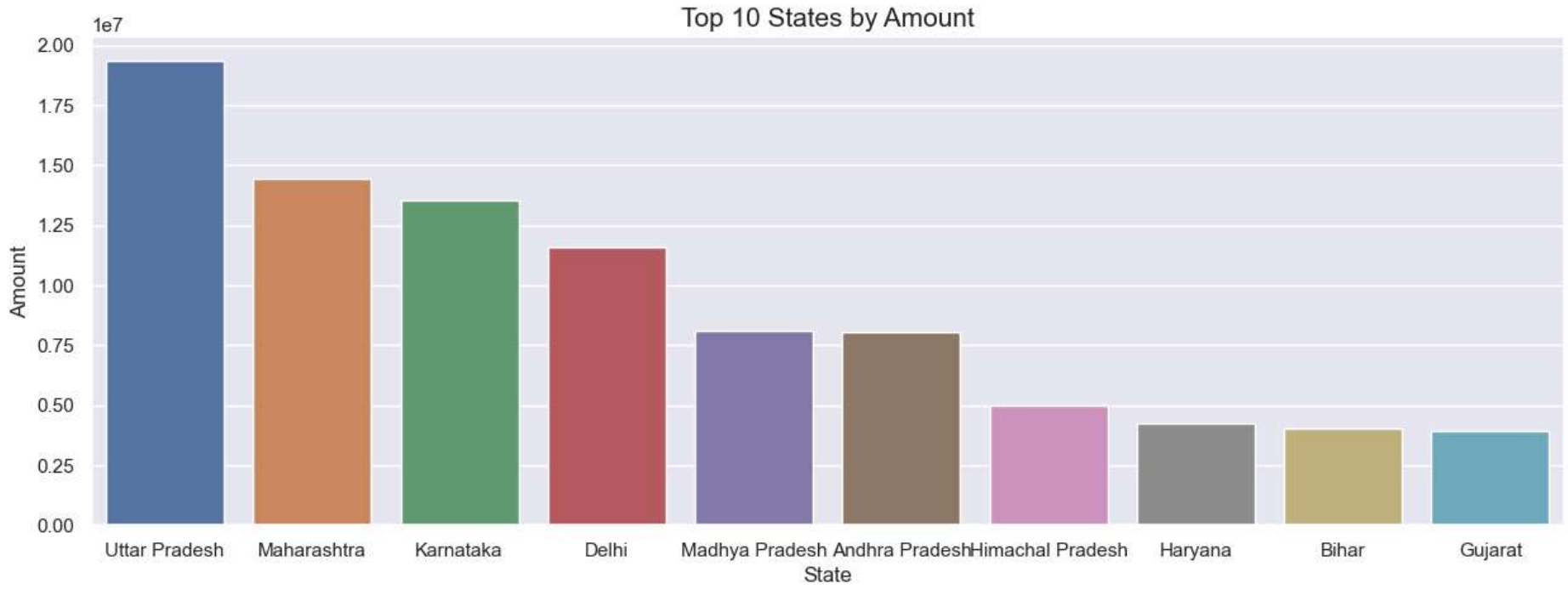
In [28]:
```
Amount_by_States = data.groupby(["State"], as_index=False)["Amount"].sum().sort_values(by="Amount", ascending=False).head(10)
Amount_by_States
```

Out[28]:

	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

In [29]:

```
fig = sns.barplot(Amount_by_States, x="State", y="Amount")
plt.title("Top 10 States by Amount", fontsize=15)
plt.show()
```



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

(iv) Marital Status-

In [30]:
```
data.head()
```

Out[30]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	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

In [31]:
```
data["Shadi"].value_counts()
```

Out[31]:
```
Shadi
0    6518
1    4721
Name: count, dtype: int64
```

In [32]:
```
fig = sns.countplot(data, x="Shadi", hue="Gender")

for bars in fig.containers:
    fig.bar_label(bars)
plt.title("Marital Status", fontsize=15)
plt.show()
```

In [33]:
```
Marital_by_Amount = data.groupby(["Shadi","Gender"], as_index=False)["Amount"].sum()
Marital_by_Amount
```

Out[33]:

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

In [34]:
```
fig = sns.barplot(Marital_by_Amount, x="Shadi", y="Amount", hue = "Gender")
plt.title("Orders based on Marital Status", fontsize=20)
plt.show()
```

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

(v) Occupation-

In [35]:
```
data.head()
```

Out[35]:

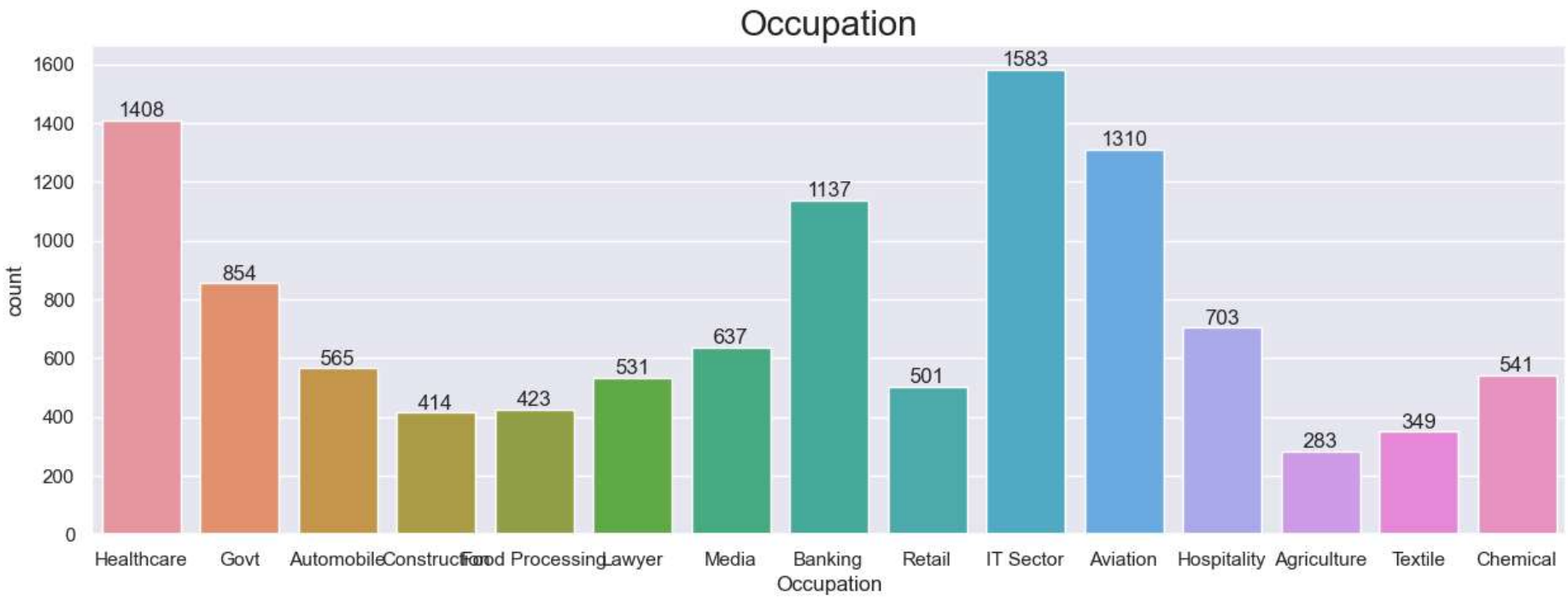
	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934
2	1001990	Bindu	P00118542	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

In [36]:
```
data["Occupation"].value_counts()
```

Out[36]:
```
Occupation
IT Sector          1583
Healthcare         1408
Aviation           1310
Banking            1137
Govt                854
Hospitality         703
Media               637
Automobile          565
Chemical            541
Lawyer              531
Retail              501
Food Processing     423
Construction        414
Textile             349
Agriculture         283
Name: count, dtype: int64
```

In [37]:
```
fig = sns.countplot(data, x="Occupation")

for bars in fig.containers:
    fig.bar_label(bars)
plt.title("Occupation", fontsize=20)
plt.show()
```



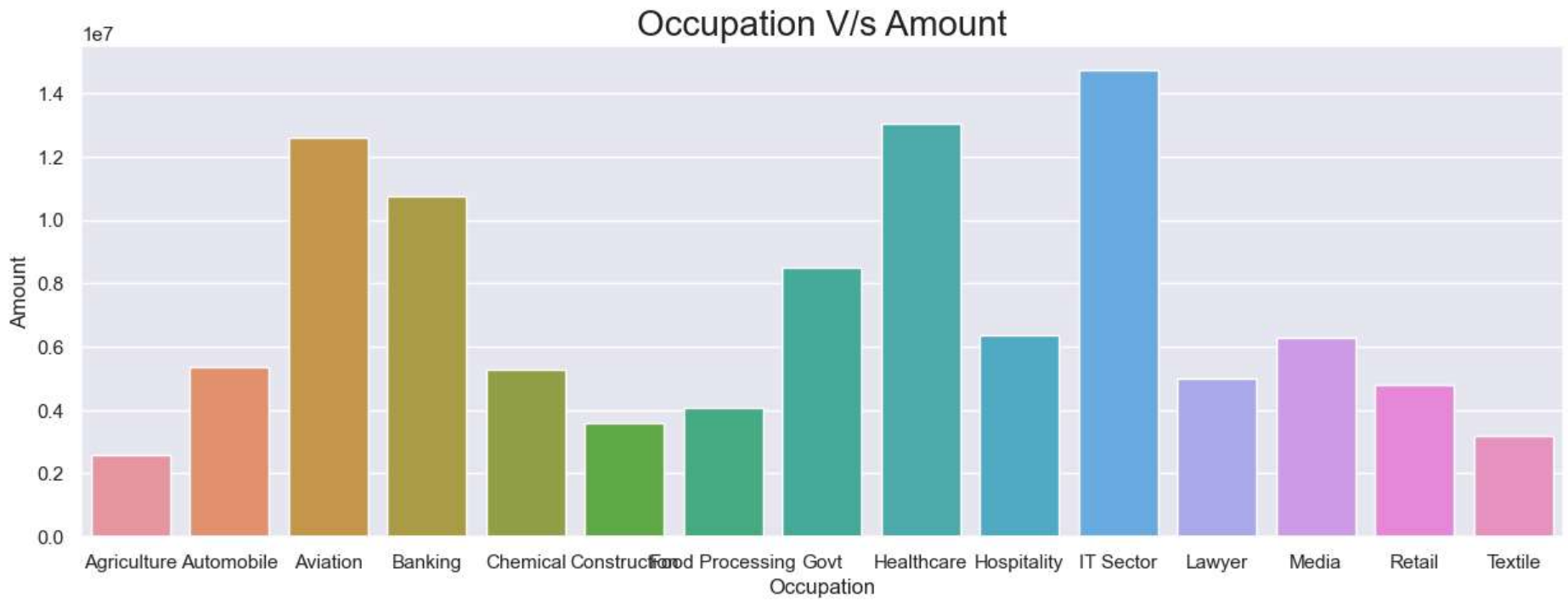
In [38]:
```
Occupation_by_Amount = data.groupby("Occupation")["Amount"].sum().reset_index()
Occupation_by_Amount
```

Out[38]:

	Occupation	Amount
0	Agriculture	2593087
1	Automobile	5368596
2	Aviation	12602298
3	Banking	10770610
4	Chemical	5297436
5	Construction	3597511
6	Food Processing	4070670
7	Govt	8517212
8	Healthcare	13034586
9	Hospitality	6376405
10	IT Sector	14755079
11	Lawyer	4981665
12	Media	6295832
13	Retail	4783170
14	Textile	3204972

In [39]:

```
fig = sns.barplot(Occupation_by_Amount, x="Occupation", y="Amount")
plt.title("Occupation V/s Amount", fontsize=20)
plt.show()
```



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

(vi) Product Category-

In [40]:
```
data.head()
```

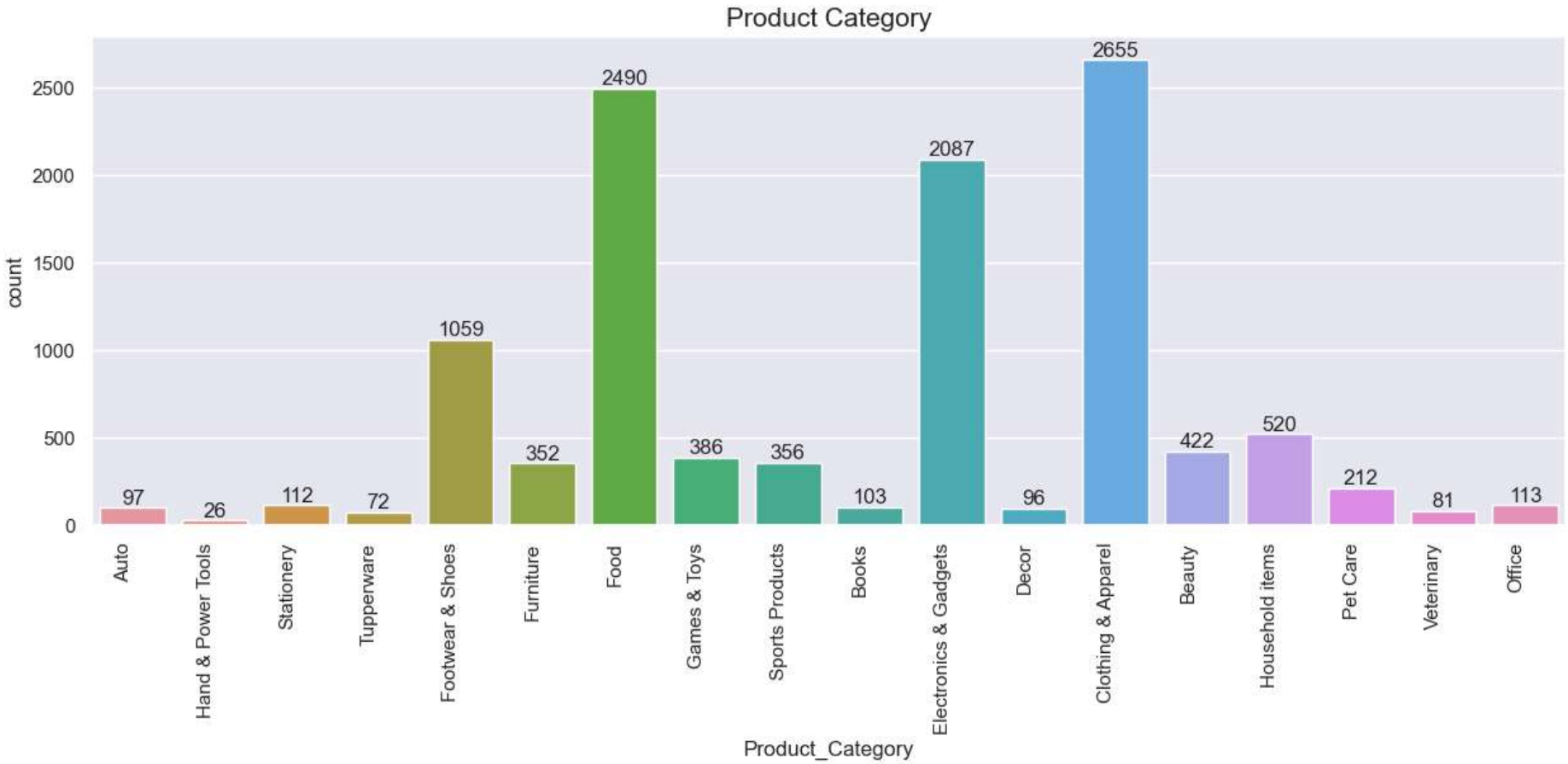
Out[40]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	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

In [41]:
```
data["Product_Category"].value_counts()
```

Out[41]:
```
Product_Category
Clothing & Apparel       2655
Food                     2490
Electronics & Gadgets    2087
Footwear & Shoes         1059
Household items           520
Beauty                    422
Games & Toys              386
Sports Products           356
Furniture                 352
Pet Care                  212
Office                    113
Stationery                112
Books                     103
Auto                       97
Decor                      96
Veterinary                 81
Tupperware                 72
Hand & Power Tools         26
Name: count, dtype: int64
```

In [60]:
```
fig = sns.countplot(data, x="Product_Category")
for bars in fig.containers:
    fig.bar_label(bars)
plt.title("Product Category", fontsize=15)
plt.xticks(rotation=90, ha='right')
plt.show()
```



In [44]:
```
Category_by_Amount =  data.groupby(["Product_Category"], as_index=False)["Amount"].sum().sort_values(by = "Amount", ascending = F
Category_by_Amount
```

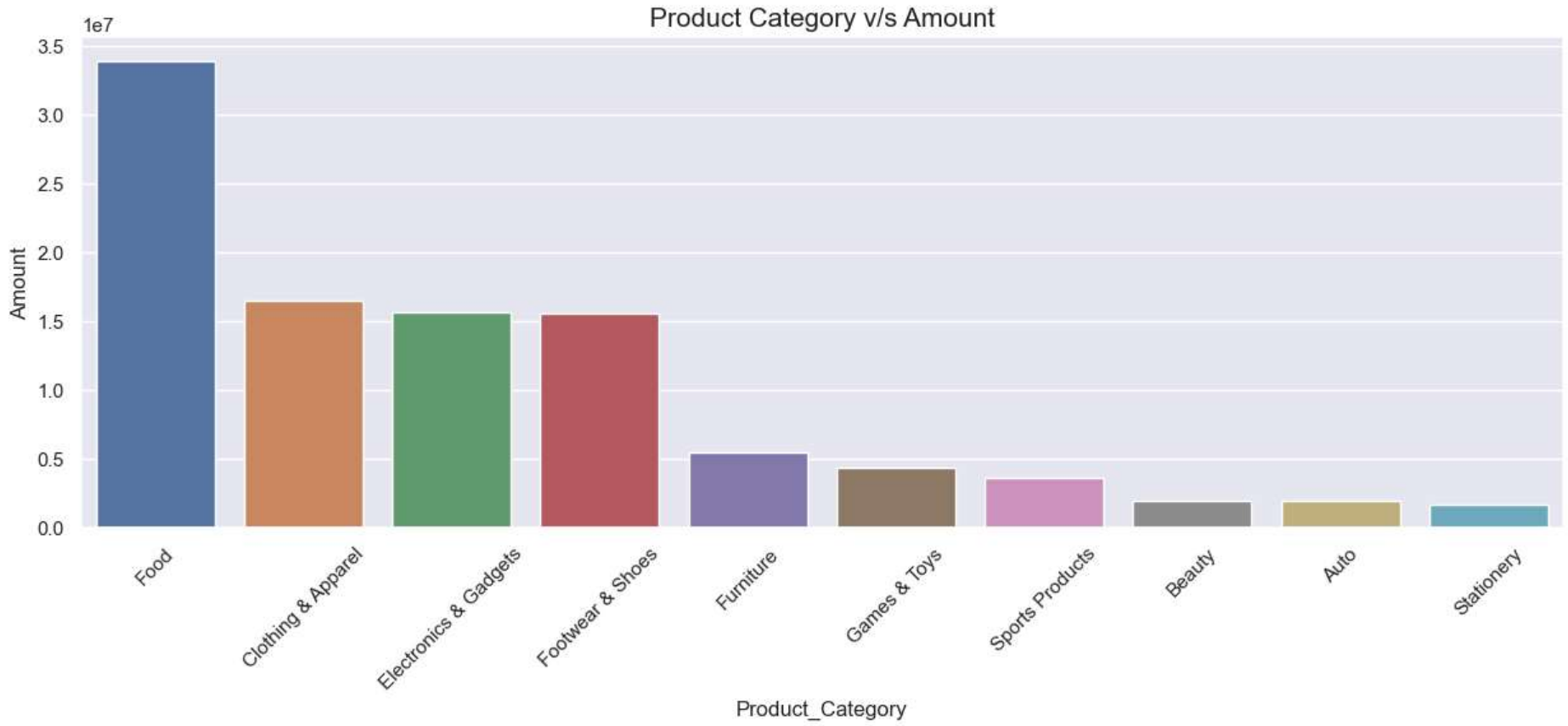
Out[44]:

	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

Product Category v/s Amount-

In [64]:

```
fig = sns.barplot(Category_by_Amount, x="Product_Category", y="Amount")
plt.xticks(rotation=45)
plt.title("Product Category v/s Amount", fontsize=15)
plt.show()
```



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

Product Category v/s Orders-

In [59]:
```
Category_by_Orders = data.groupby(["Product_Category"], as_index=False)["Orders"].sum().sort_values(by="Orders",
                                                                                                    ascending=False).head(10)
Category_by_Orders
```

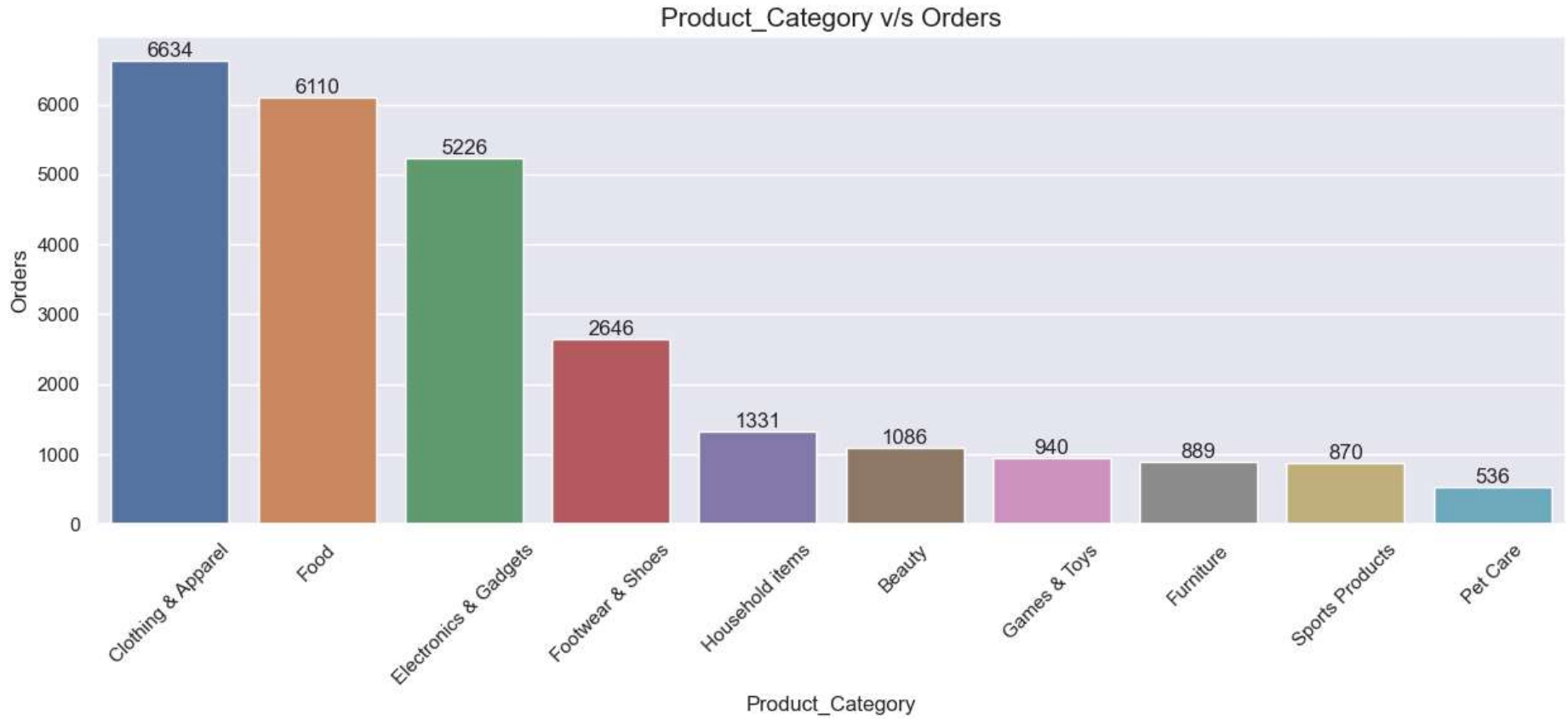
Out[59]:

	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

In [68]:

```
fig = sns.barplot(Category_by_Orders, x="Product_Category", y="Orders")
for bars in fig.containers:
    fig.bar_label(bars)

plt.xticks(rotation=45)
plt.title("Product_Category v/s Orders",fontsize=15)
plt.show()
```



Top 10 Most sold Product-

In [53]:
```
data.head()
```

Out[53]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shadi	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

In [71]:
```
Top_Category = data["Product_ID"].value_counts().head(10)
Top_Category
```

Out[71]:
```
Product_ID
P00265242    53
P00110942    44
P00184942    37
P00237542    35
P00112142    34
P00114942    33
P00110742    32
P00112542    30
P00110842    30
P00145042    30
Name: count, dtype: int64
```

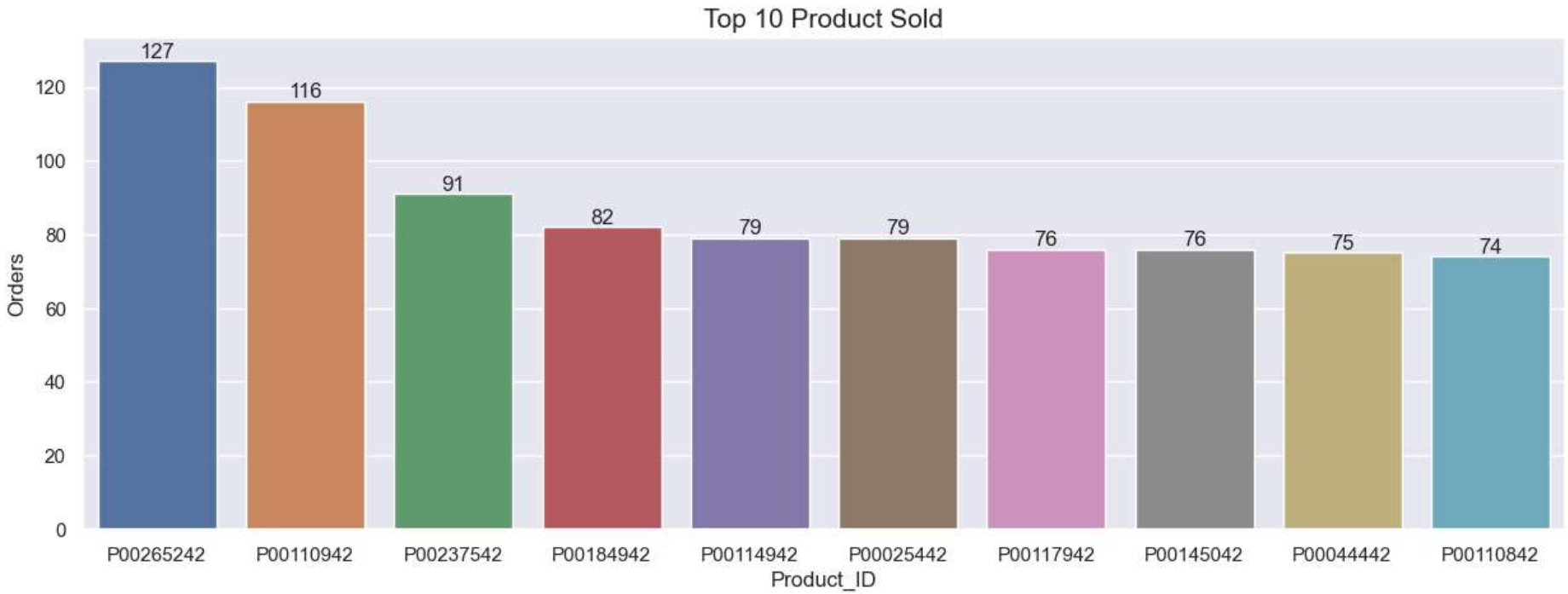
In [73]:
```
Top_Product = data.groupby(["Product_ID"], as_index=False)["Orders"].sum().sort_values(by="Orders", ascending=False).head(10)
Top_Product
```

Out[73]:

	Product_ID	Orders
1679	P00265242	127
644	P00110942	116
1504	P00237542	91
1146	P00184942	82
679	P00114942	79
171	P00025442	79
708	P00117942	76
888	P00145042	76
298	P00044442	75
643	P00110842	74

In [78]:
```
fig = sns.barplot(Top_Product, x="Product_ID", y="Orders")
for bars in fig.containers:
    fig.bar_label(bars)
```

```
plt.title("Top 10 Product Sold", fontsize=15)
plt.show()
```



Conclusion: Married women age group 26-35 yrs from UP, Maharastra and Karnataka working in IT, Healthcare and Aviation are more likely to buy products from Food, Clothing and Electronics category.