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

In [3]:
```
df = pd.read_csv('Sales Data.csv',encoding='unicode_escape')
```

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

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

In [5]:
```
df.head(10)
```

Out[5]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat
5	1000588	Joni	P00057942	M	26-35	28	1	Himachal Pradesh
6	1001132	Balk	P00018042	F	18-25	25	1	Uttar Pradesh
7	1002092	Shivangi	P00273442	F	55+	61	0	Maharashtra
8	1003224	Kushal	P00205642	M	26-35	35	0	Uttar Pradesh
9	1003650	Ginny	P00031142	F	26-35	26	1	Andhra Pradesh

In [6]:
```
df.info
```

Out[6]:
```
<bound method DataFrame.info of        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
...        ...          ...       ...    ...       ...  ...             ...
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  Status  unnamed1
0      23952.0     NaN       NaN
1      23934.0     NaN       NaN
2      23924.0     NaN       NaN
3      23912.0     NaN       NaN
4      23877.0     NaN       NaN
...        ...     ...       ...
11246    370.0     NaN       NaN
11247    367.0     NaN       NaN
11248    213.0     NaN       NaN
11249    206.0     NaN       NaN
11250    188.0     NaN       NaN

[11251 rows x 15 columns]>
```

In [7]:
```
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 [14]:
```
pd.isnull(df)
```

Out[14]:

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

11251 rows × 13 columns

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

Out[17]:
```
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 [18]:
```
df.shape
```

Out[18]:
```
(11251, 13)
```

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

In [20]:
```
df.shape
```

Out[20]:
```
(11239, 13)
```

In [21]:
```
df
```

Out[21]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	S
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharas
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Prac
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Prac
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karna
4	1000588	Joni	P00057942	M	26-35	28	1	Guj
...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharas
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Hary
11248	1001209	Oshin	P00201342	F	36-45	40	0	Mac Prac
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karna
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharas

11239 rows × 13 columns

In [27]:
```
data = [['Umang',22],['Mayank',27],['Vanish',],['Mihir',22]]
test_data = pd.DataFrame(data, columns=['Name','Age'])
```

In [28]:
```
test_data
```

Out[28]:

	Name	Age
0	Umang	22.0
1	Mayank	27.0
2	Vanish	NaN
3	Mihir	22.0

In [29]:
```
test_data.dropna()
```

Out[29]:

	Name	Age
0	Umang	22.0
1	Mayank	27.0
3	Mihir	22.0

In [30]:
```
test_data
```

Out[30]:

	Name	Age
0	Umang	22.0
1	Mayank	27.0
2	Vanish	NaN
3	Mihir	22.0

In [31]:
```
test_data.dropna(inplace=True)
```

In [32]:
```
test_data
```

Out[32]:

	Name	Age
0	Umang	22.0
1	Mayank	27.0
3	Mihir	22.0

In [33]:
```
df
```

Out[33]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	S
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharas
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Prac
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Prac
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karna
4	1000588	Joni	P00057942	M	26-35	28	1	Guj
...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharas
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Hary
11248	1001209	Oshin	P00201342	F	36-45	40	0	Mac Prac
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karna
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharas

11239 rows × 13 columns

In [34]:
```
df.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
```

In [38]:
```
# Change the Data
df['Amount'] = df['Amount'].astype('int')
```

In [39]:
```
df
```

Out[39]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	S
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharas
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Prac
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Prac
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karna
4	1000588	Joni	P00057942	M	26-35	28	1	Guj
...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharas
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Hary
11248	1001209	Oshin	P00201342	F	36-45	40	0	Mac Prac
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karna
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharas

11239 rows × 13 columns

In [40]:
```
df.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  int32 
dtypes: int32(1), int64(4), object(8)
memory usage: 1.2+ MB
```

In [41]:
```
# Check the data types by Users
```

In [41]:
```
df['Amount'].dtype
```

Out[41]:
```
dtype('int32')
```

In [43]:
```
df.columns
```

Out[43]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [50]:
```
# Rename Columns
df.rename(columns= {'Marital_Status':'Married'})
```

Out[50]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Married	State
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat
...	...	...	...	...	...	...	...	...
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra

11239 rows × 13 columns

In [45]:
```
df.describe()
```

Out[45]:

	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

In [46]:
```
df.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  int32 
dtypes: int32(1), int64(4), object(8)
memory usage: 1.2+ MB
```

In [51]:
```
df.columns
```

Out[51]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [52]:
```
df.describe()
```

Out[52]:

	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

Exploratory Data Analysis

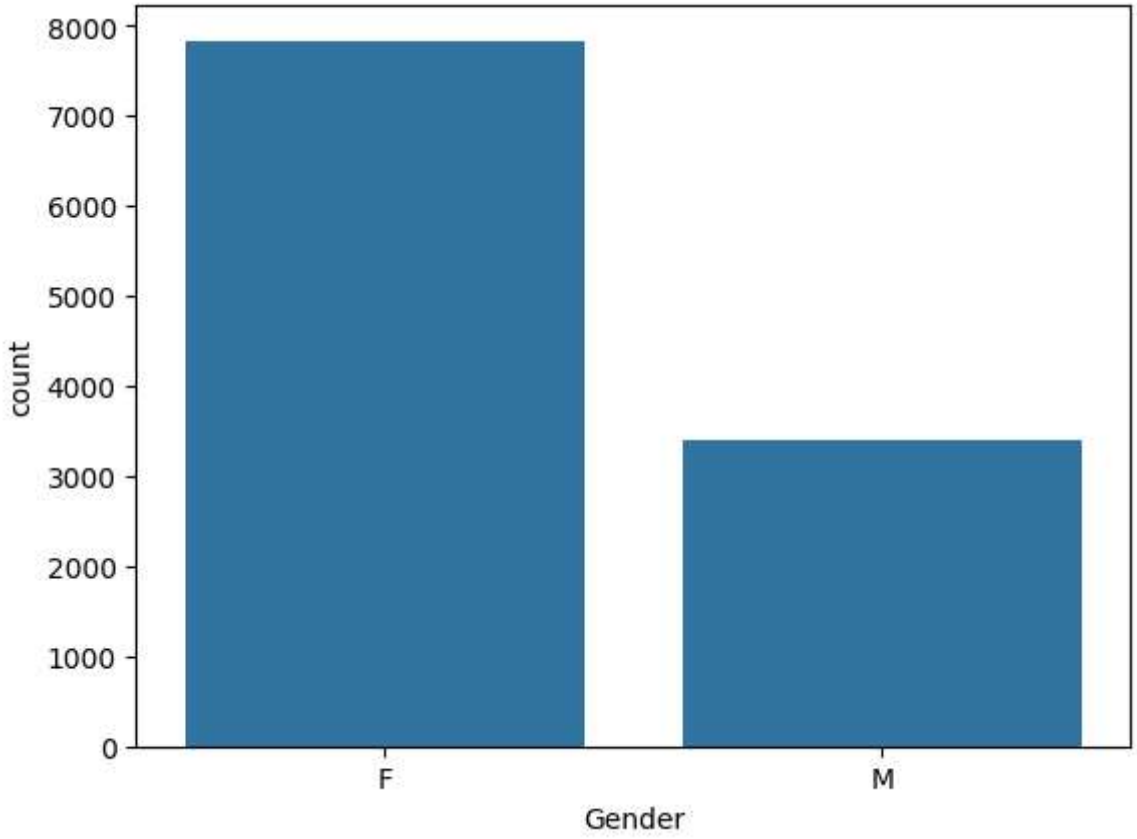
Gender Find Data

In [56]:
```
df.columns
```

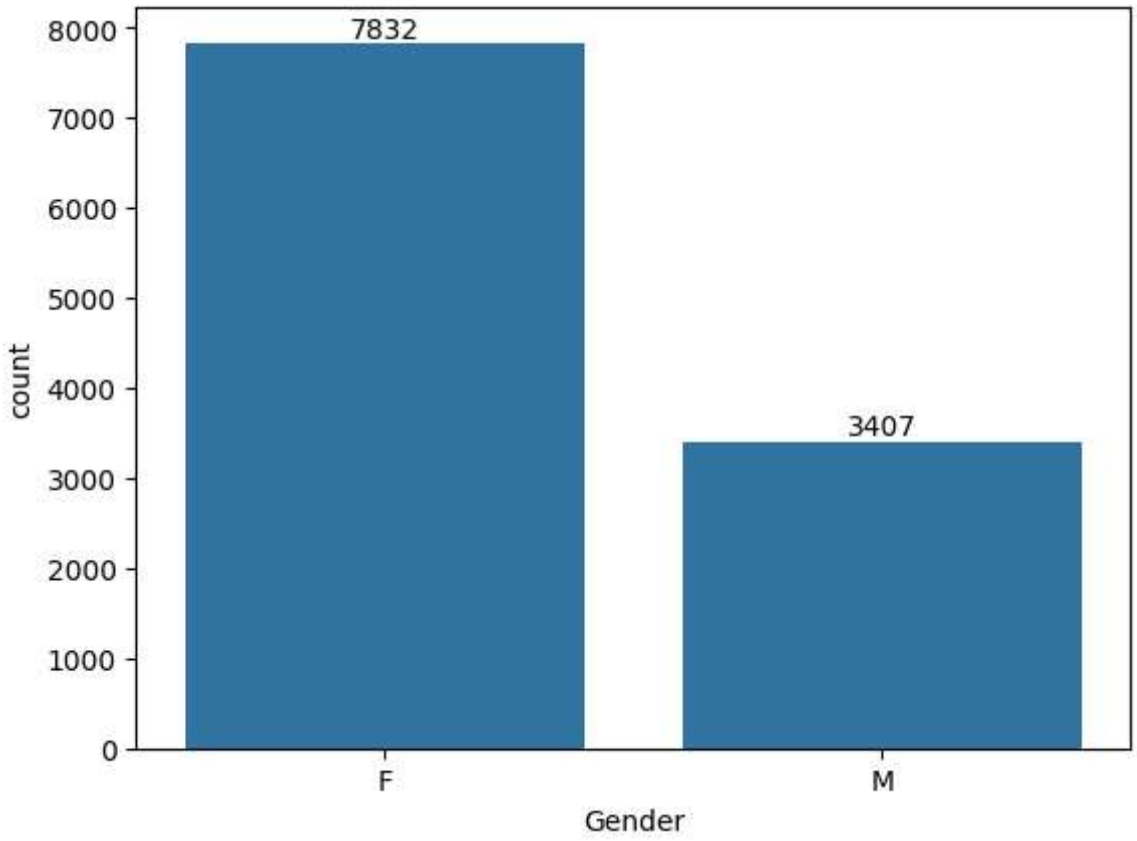
Out[56]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [60]:
```
sn.countplot(x = 'Gender',data = df)
```

Out[60]:
```
<Axes: xlabel='Gender', ylabel='count'>
```



In [62]:
```
dataset = sn.countplot(x = 'Gender',data = df)
for i in dataset.containers:
       dataset.bar_label(i)
```



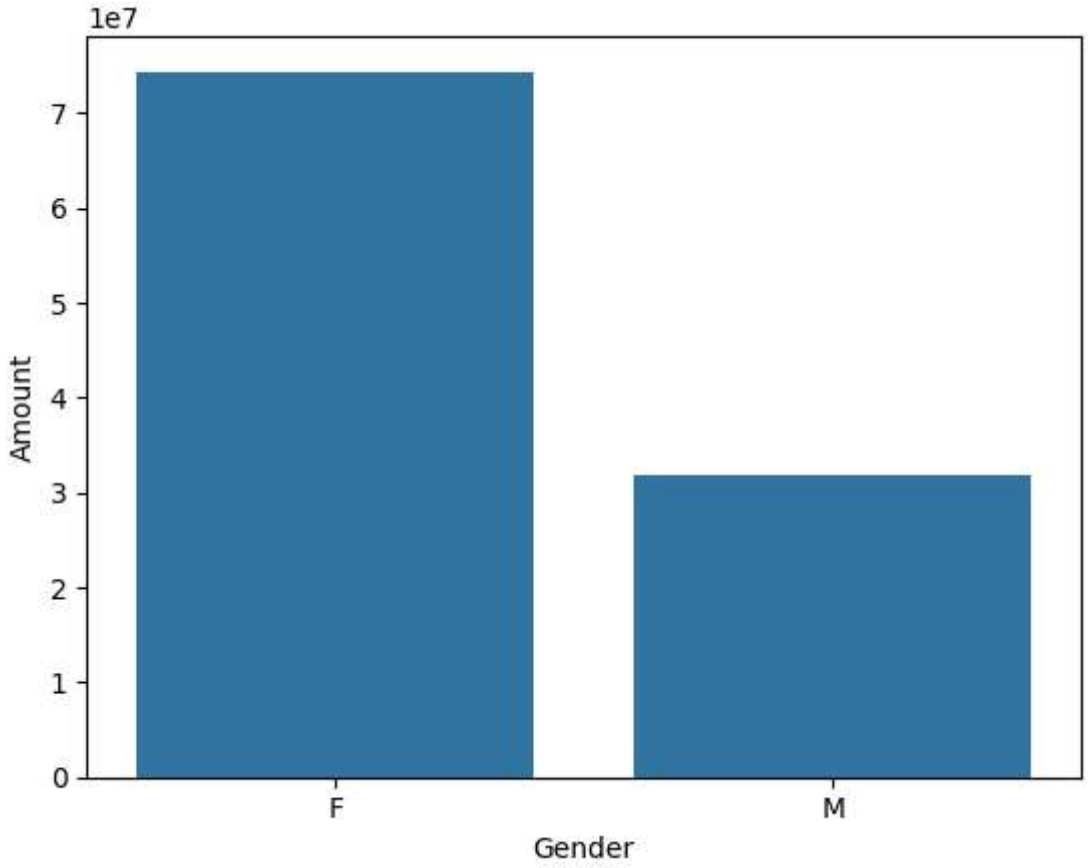
In [75]:
```
df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount', asc
```

Out[75]:

	Gender	Amount
0	F	74335853
1	M	31913276

In [79]:
```
set = df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount

sn.barplot(x = 'Gender', y = 'Amount', data=set)
```

Out[79]:
```
<Axes: xlabel='Gender', ylabel='Amount'>
```



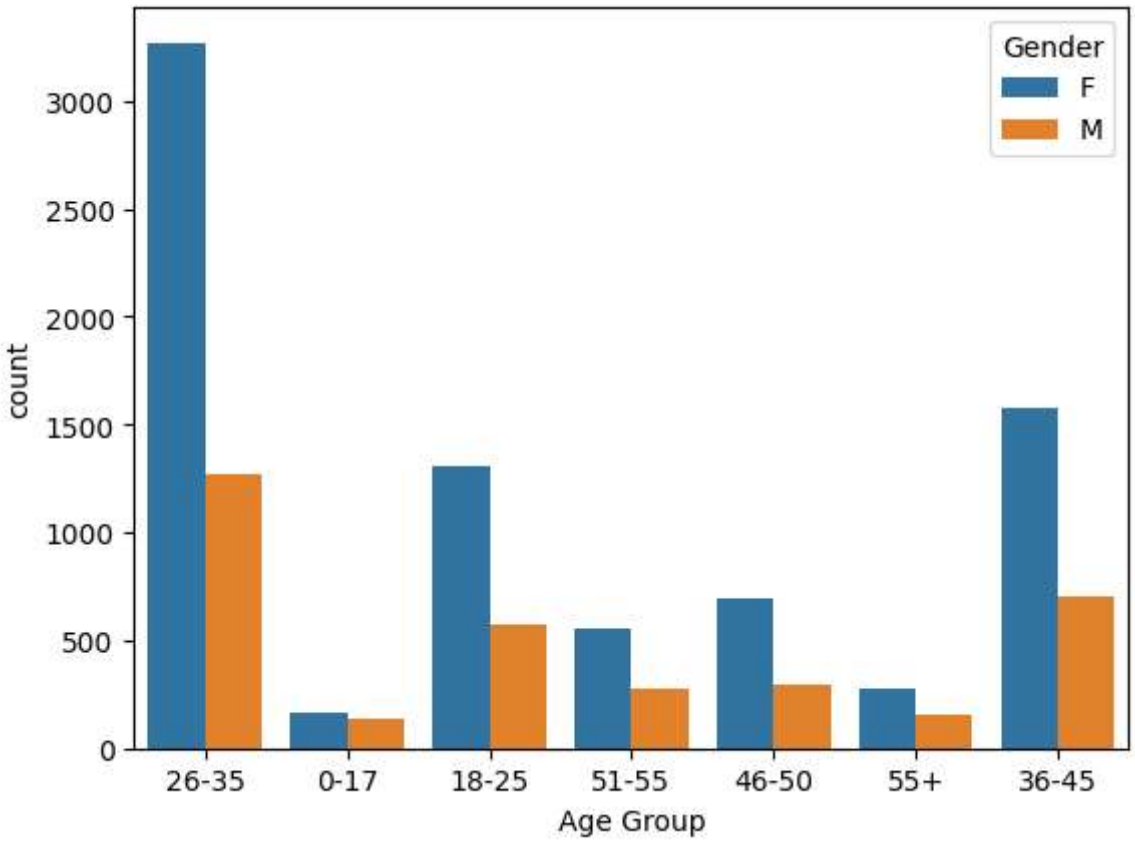
Age

In [80]:
```
df.columns
```

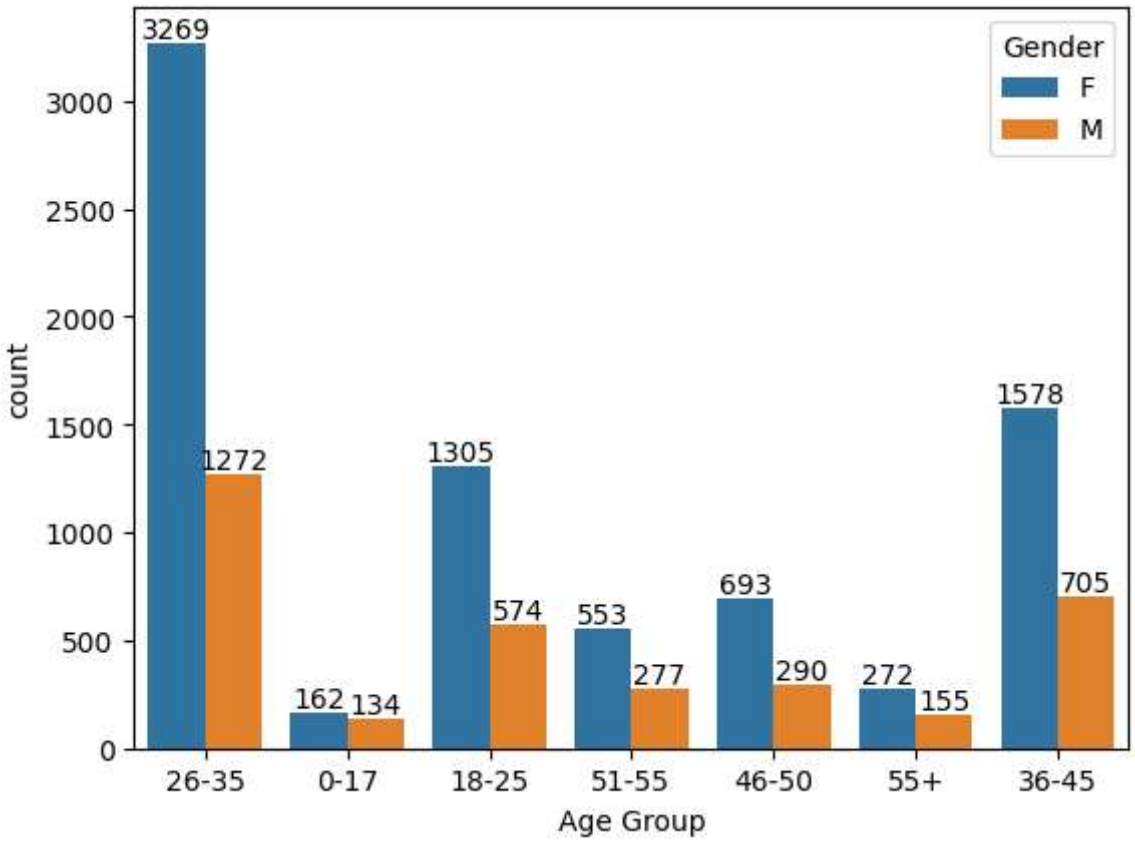
Out[80]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [89]:
```
sn.countplot(data = df, x = 'Age Group', hue='Gender')
```

Out[89]:
```
<Axes: xlabel='Age Group', ylabel='count'>
```

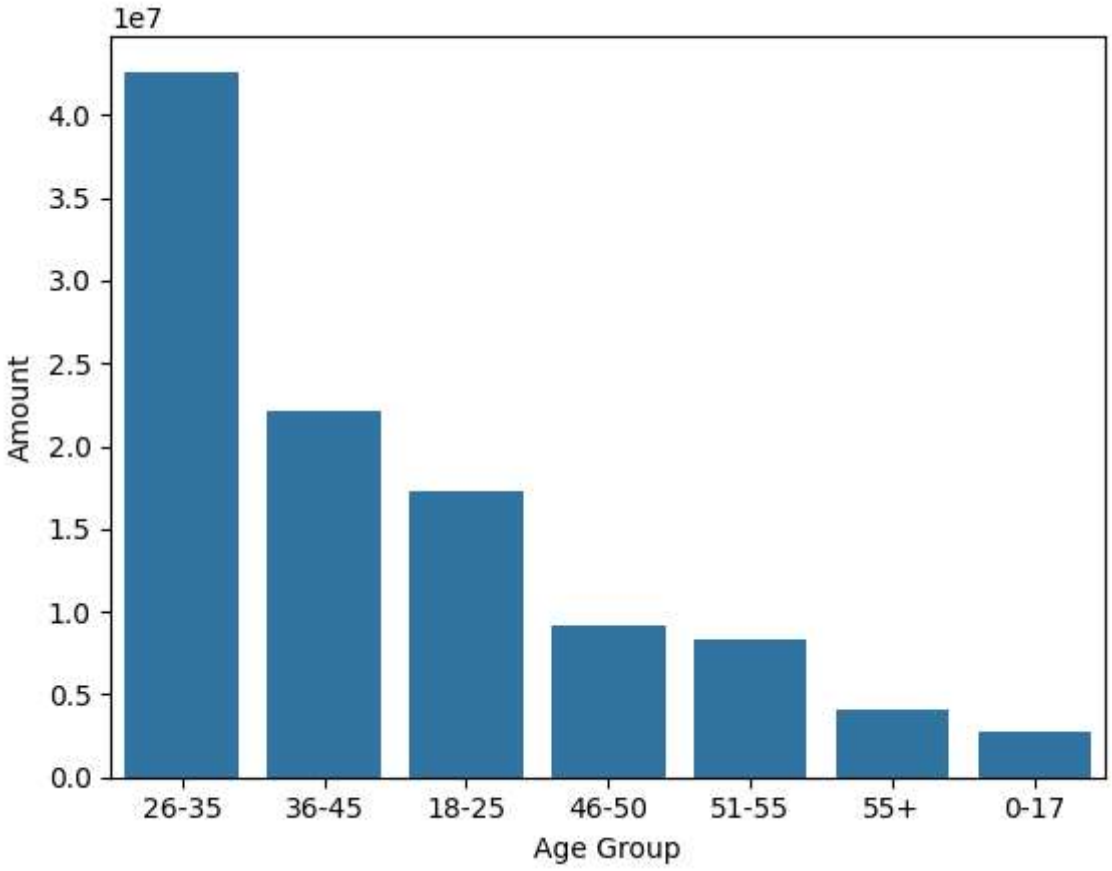


In [93]:
```
set = dataset = sn.countplot(data = df, x = 'Age Group',hue='Gender')
for i in set.containers:
       set.bar_label(i)
```



In [95]:
```
set = df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values(by='Amo

sn.barplot(x = 'Age Group', y = 'Amount', data=set)
```

Out[95]:
```
<Axes: xlabel='Age Group', ylabel='Amount'>
```



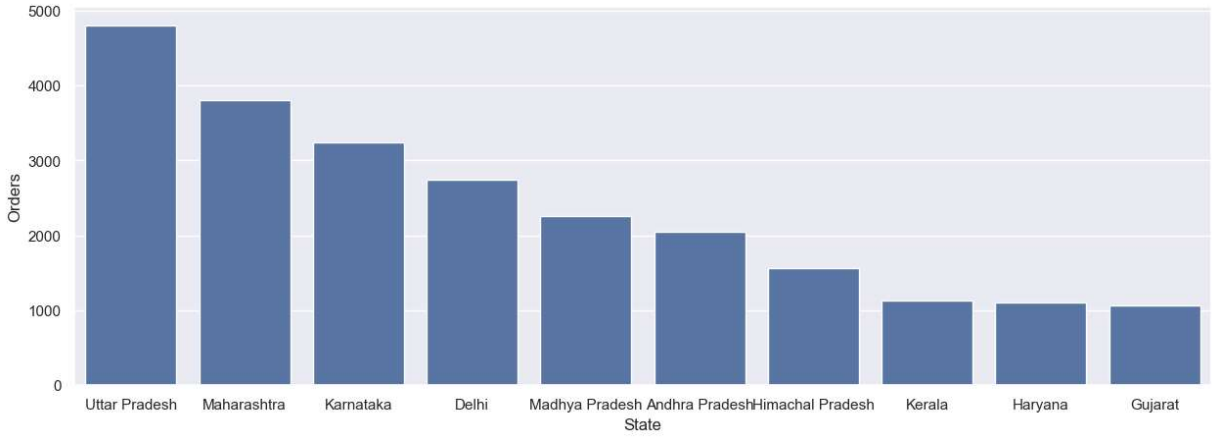
State

In [96]:
```
df.columns
```

Out[96]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

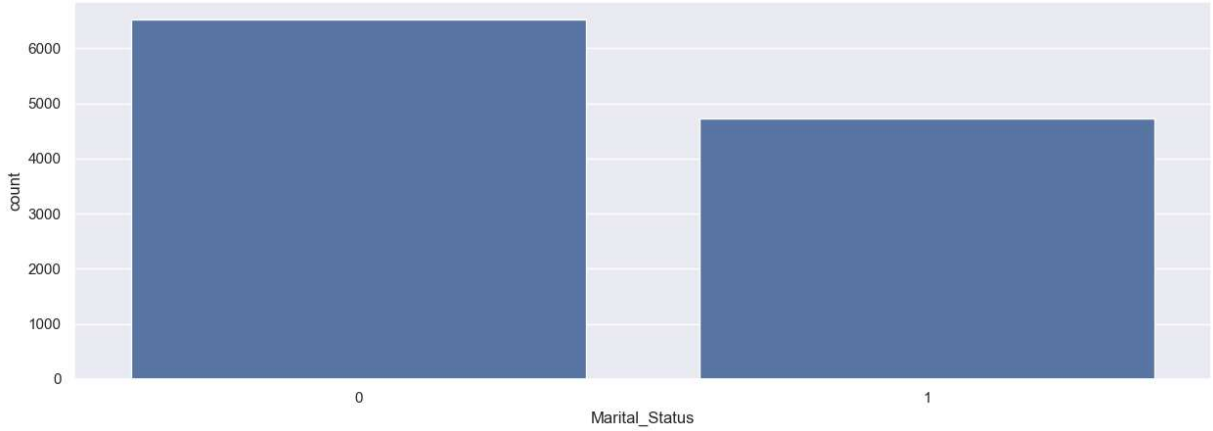
In [100…
```
set = df.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by='Orders'

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

Out[100…
```
<Axes: xlabel='State', ylabel='Orders'>
```

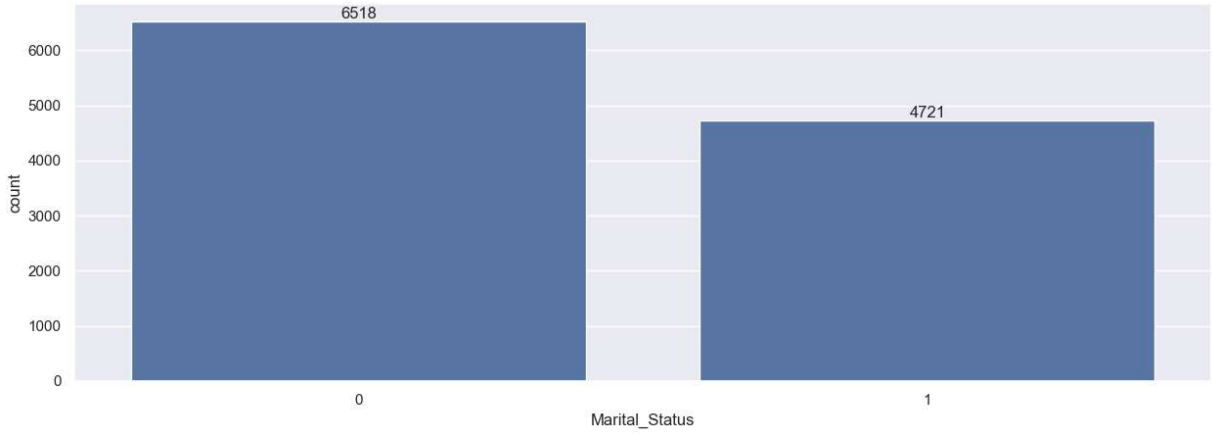


Marital Status

In [103…
```
sn.countplot(data = df, x = 'Marital_Status')
sn.set(rc={'figure.figsize':(15,5)})
```



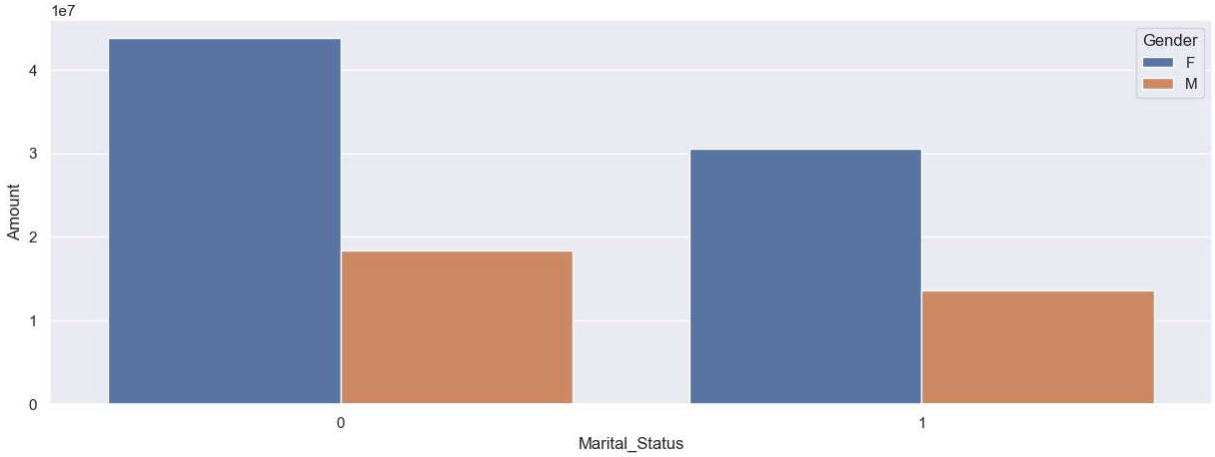
In [105…
```
ms = sn.countplot(data = df, x = 'Marital_Status')
sn.set(rc={'figure.figsize':(15,5)})

for i in ms.containers:
    ms.bar_label(i)
```



In [110…
```
set = df.groupby(['Marital_Status', 'Gender'], as_index=False)['Amount'].sum().sort
```

```
sn.barplot(x = 'Marital_Status', y = 'Amount', hue='Gender', data=set)
```

Out[110… <Axes: xlabel='Marital_Status', ylabel='Amount'>

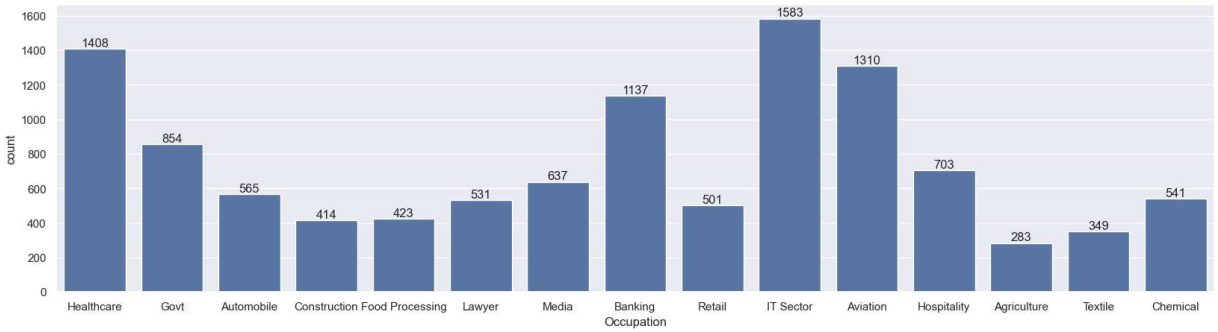


Occupation

In [112…
```
df.columns
```

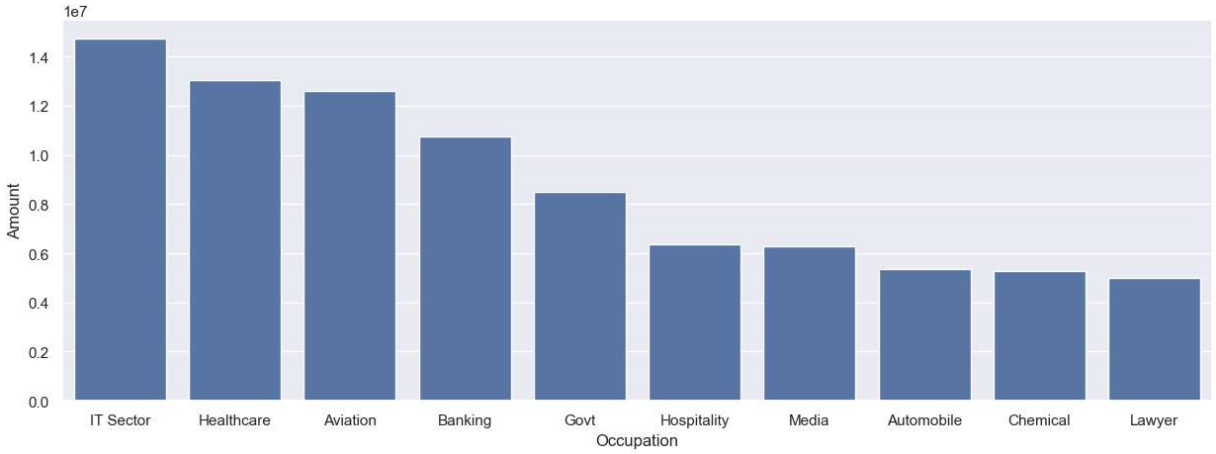
Out[112…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [115…
```
ms = sn.countplot(data = df, x = 'Occupation')
sn.set(rc={'figure.figsize':(22,5)})

for i in ms.containers:
    ms.bar_label(i)
```



In [117…
```
set = df.groupby(['Occupation'], as_index=False)['Amount'].sum().sort_values(by='Am

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

Out[117… <Axes: xlabel='Occupation', ylabel='Amount'>



In [118…
```
df.columns
```

Out[118…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

END PROJECT

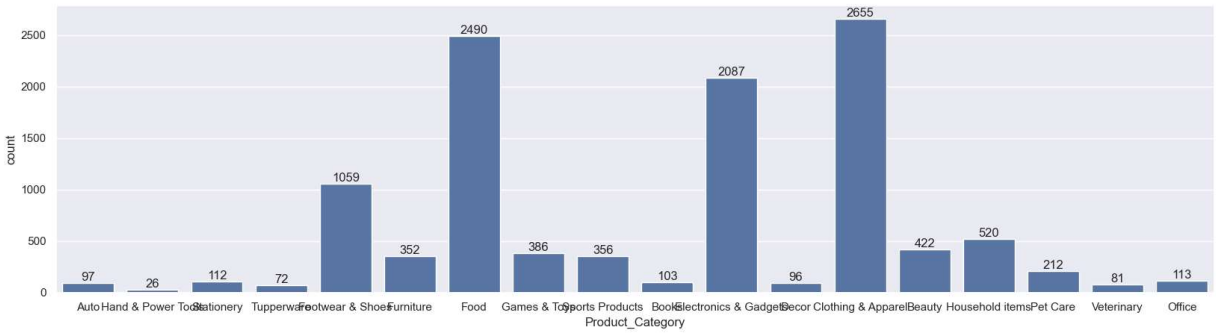
In [160…
```
df.columns
```

Out[160…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [161…
```
df.columns
```

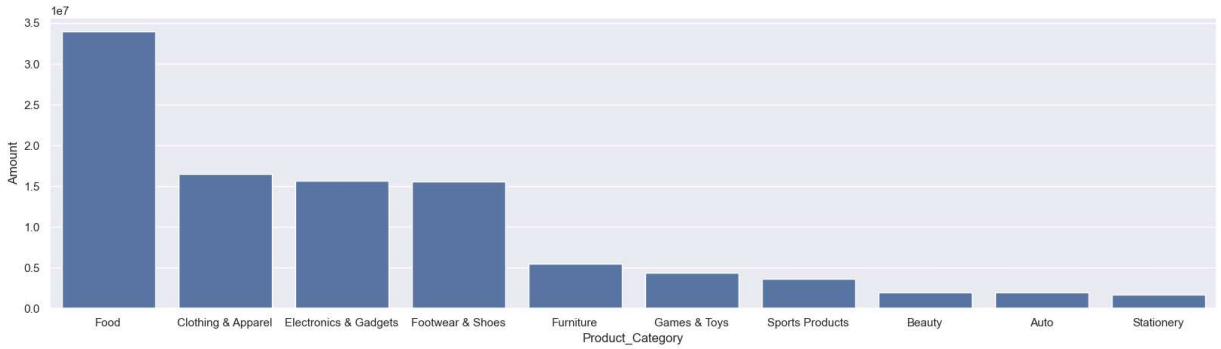
Out[161…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [162…
```
ms = sn.countplot(data = df, x = 'Product_Category')
sn.set(rc={'figure.figsize':(22,5)})

for i in ms.containers:
    ms.bar_label(i)
```



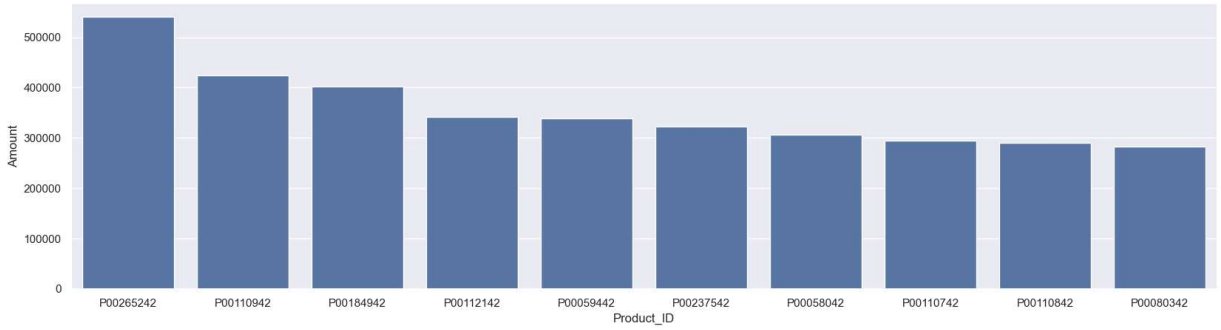
In [164…
```
set = df.groupby(['Product_Category'], as_index=False)['Amount'].sum().sort_values(

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

Out[164… <Axes: xlabel='Product_Category', ylabel='Amount'>



In [165…
```
set = df.groupby(['Product_ID'], as_index=False)['Amount'].sum().sort_values(by='Am

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

Out[165… <Axes: xlabel='Product_ID', ylabel='Amount'>



END PROJECT

PROJECT BY : UMANG MODI

In [ ]: