In [26]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
```

In [27]:
```
df=pd.read_csv("Diwali Sales Data.csv",encoding="unicode_escape")
df
```

Out[27]:

	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

11251 rows × 15 columns

In [28]:
```
df.shape
```

Out[28]: (11251, 15)

In [29]:
```
df.head(10)
```

Out[29]:

	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 [30]:
```
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 [31]:
```
df.describe()
```

Out[31]:

	User_ID	Age	Marital_Status	Orders	Amount	Status	unr
count	1.125100e+04	11251.000000	11251.000000	11251.000000	11239.000000	0.0	
mean	1.003004e+06	35.421207	0.420318	2.489290	9453.610858	NaN	
std	1.716125e+03	12.754122	0.493632	1.115047	5222.355869	NaN	
min	1.000001e+06	12.000000	0.000000	1.000000	188.000000	NaN	
25%	1.001492e+06	27.000000	0.000000	1.500000	5443.000000	NaN	
50%	1.003065e+06	33.000000	0.000000	2.000000	8109.000000	NaN	
75%	1.004430e+06	43.000000	1.000000	3.000000	12675.000000	NaN	
max	1.006040e+06	92.000000	1.000000	4.000000	23952.000000	NaN	

In [32]:
```
df.drop(["Status","unnamed1"],axis=1,inplace=True)
```

In [33]:
```
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 [34]:
```
pd.isnull(df).sum()
```

Out[34]:
```
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 [35]:
```
avg=np.mean(df["Amount"])
```

In [36]:
```
avg=np.round(avg,decimals=2)
avg
```

Out[36]:
```
9453.61
```

In [37]:
```
df.fillna(avg,inplace=True)
```

In [38]:
```
pd.isnull(df).sum()
```

Out[38]:
```
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              0
dtype: int64
```

In [39]:
```
df["Amount"]=df["Amount"].astype("int")
```

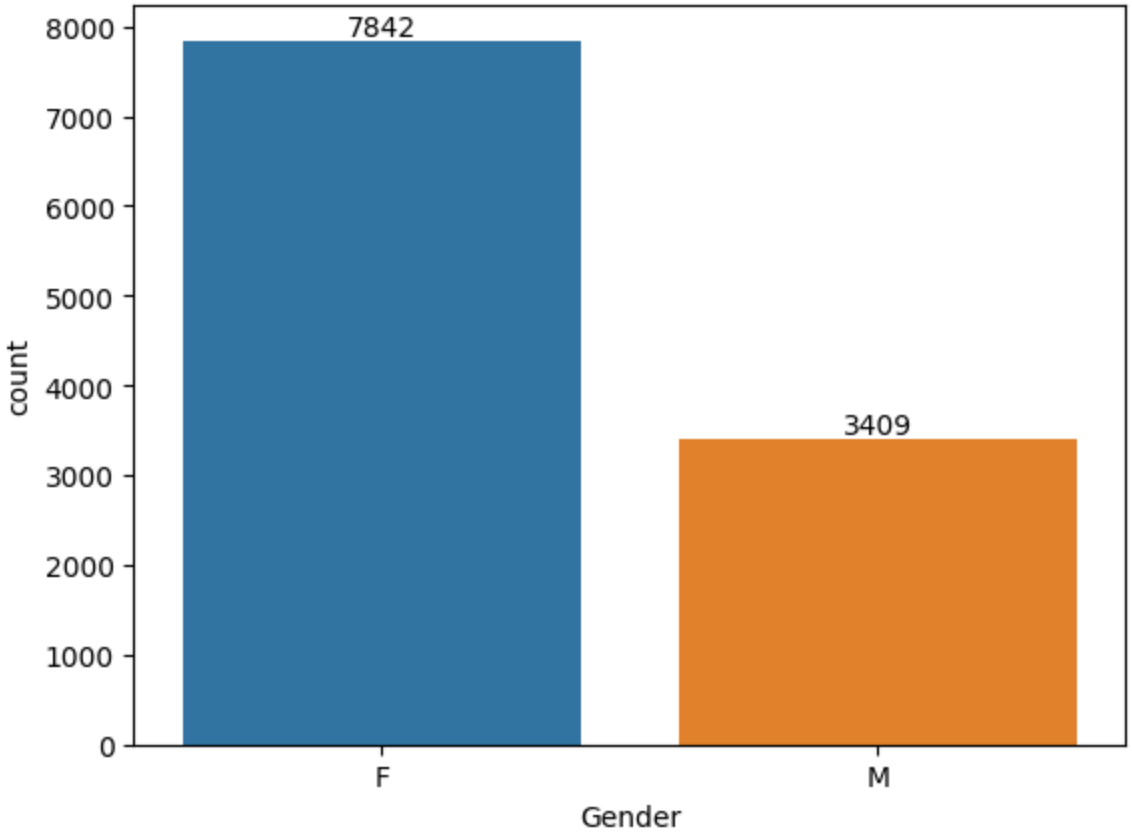
In [40]:
```
df.info()
```

```
<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            11251 non-null  int32
dtypes: int32(1), int64(4), object(8)
memory usage: 1.1+ MB
```

Exploratory Data Analysis

Gender

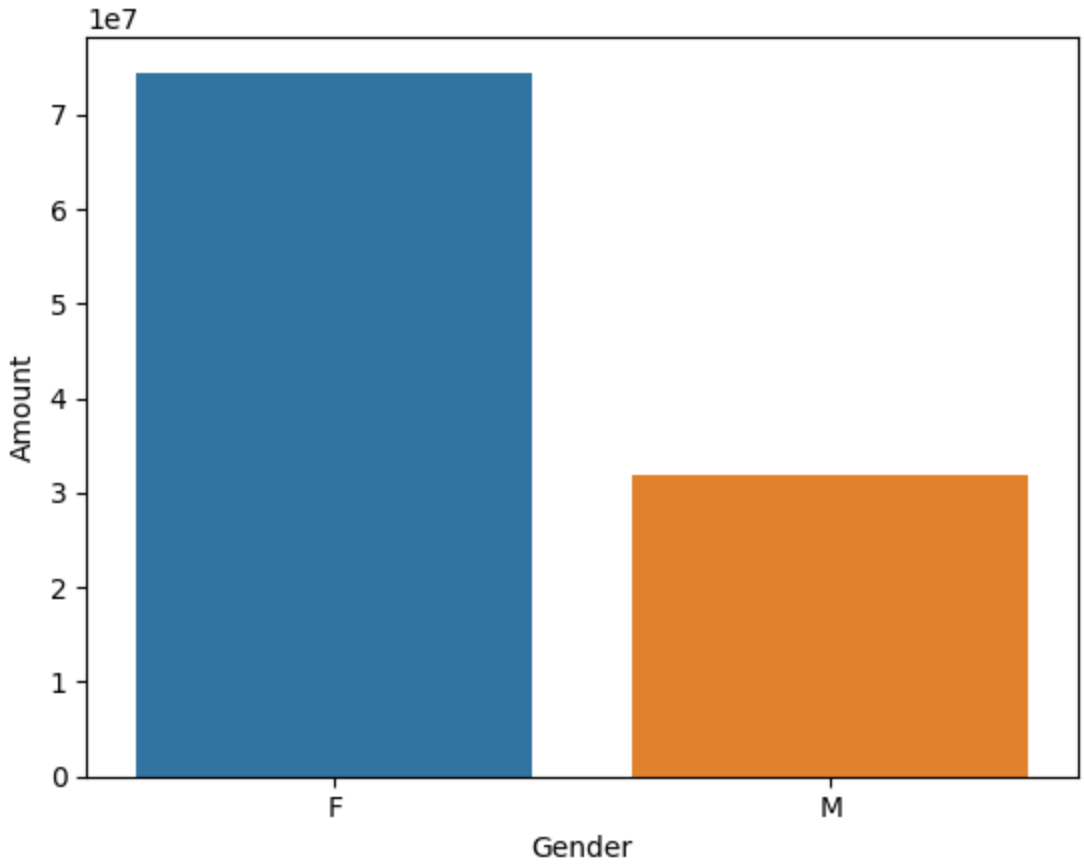
In [66]:
```
ax=sb.countplot(x="Gender",data=df)
for bars in ax.containers:
    ax.bar_label(bars)
```



In [64]:
```
money_expen=df.groupby(["Gender"],as_index=False)['Amount'].sum().sort_values(by='A
```

In [65]:
```
sb.barplot(x='Gender',y='Amount',data=money_expen)
```

Out[65]:
```
<Axes: xlabel='Gender', ylabel='Amount'>
```

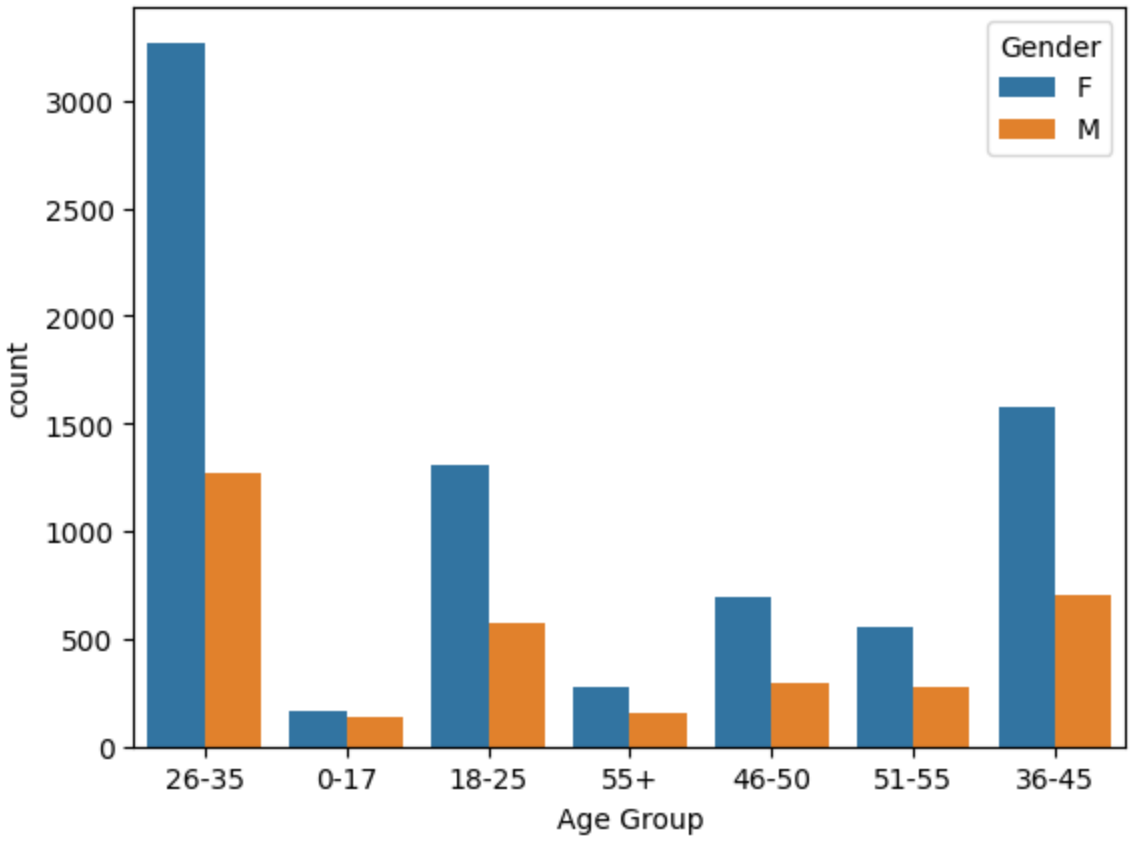


from above graphs we can see that most of the buyes are female and purchasing power of women are greater the men

Age

In [44]:
```
ag=sb.countplot(x="Age Group",hue="Gender",data=df)
ag
```

Out[44]:
```
<Axes: xlabel='Age Group', ylabel='count'>
```



In [46]:
```
df.columns
```

Out[46]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

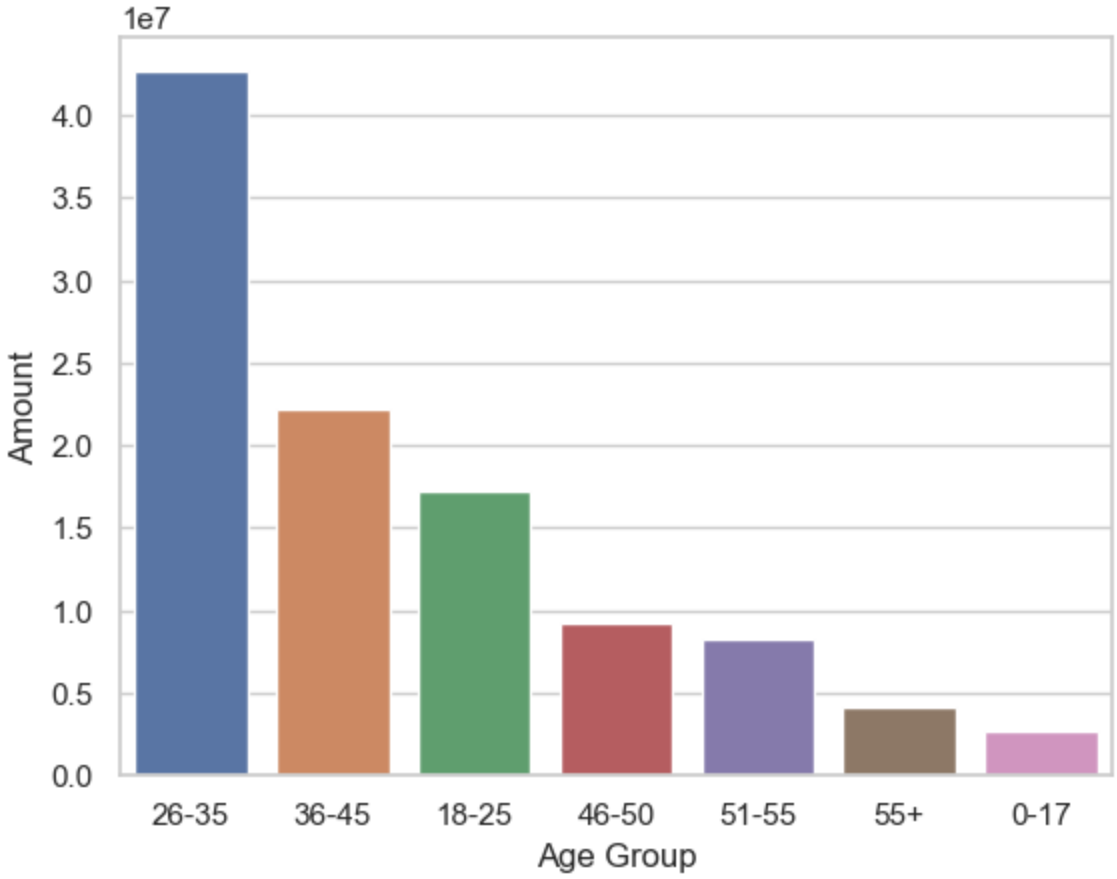
In [84]:
```
sales_age=df.groupby(["Age Group"],as_index=False)['Amount'].sum().sort_values(by='
sales_age
```

Out[84]:

	Age Group	Amount
2	26-35	42632348
3	36-45	22173353
1	18-25	17240732
4	46-50	9245656
5	51-55	8280383
6	55+	4090440
0	0-17	2699653

In [85]:
```
sb.barplot(x='Age Group',y='Amount',data=sales_age)
```

Out[85]:
```
<Axes: xlabel='Age Group', ylabel='Amount'>
```



In [ ]:

state

In [95]:
```
state_orders=df.groupby(["State"],as_index=False)['Orders'].sum().sort_values(by='O
state_orders
```

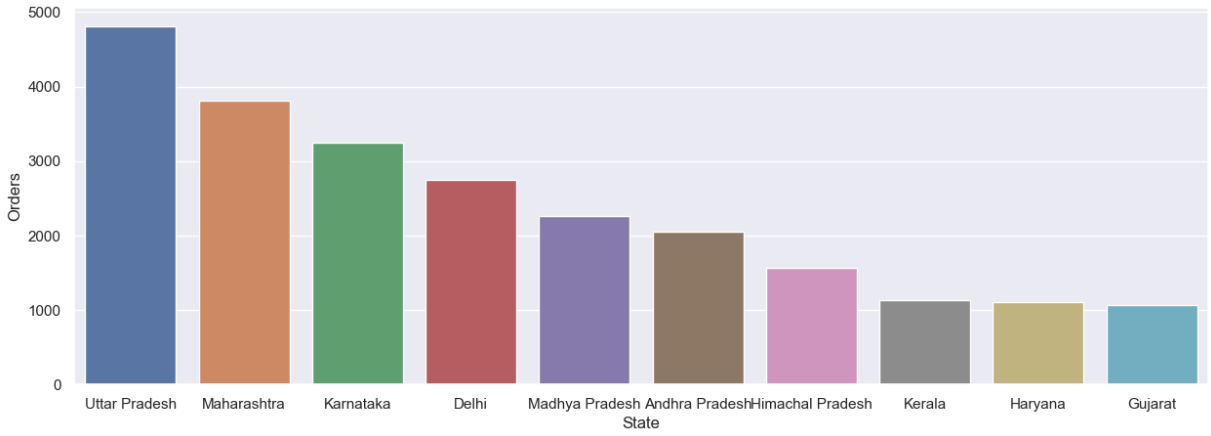
Out[95]:

	State	Orders
14	Uttar Pradesh	4813
10	Maharashtra	3811
7	Karnataka	3241
2	Delhi	2744
9	Madhya Pradesh	2259
0	Andhra Pradesh	2054
5	Himachal Pradesh	1568
8	Kerala	1137
4	Haryana	1109
3	Gujarat	1070

In [89]:
```
sb.set(rc={'figure.figsize':(15,5)})
```

In [97]:
```
sb.barplot(data=state_orders,x='State',y='Orders')
```

Out[97]: <Axes: xlabel='State', ylabel='Orders'>



In [98]:
```
state_amount=df.groupby(["State"],as_index=False)['Amount'].sum().sort_values(by='A
state_amount
```

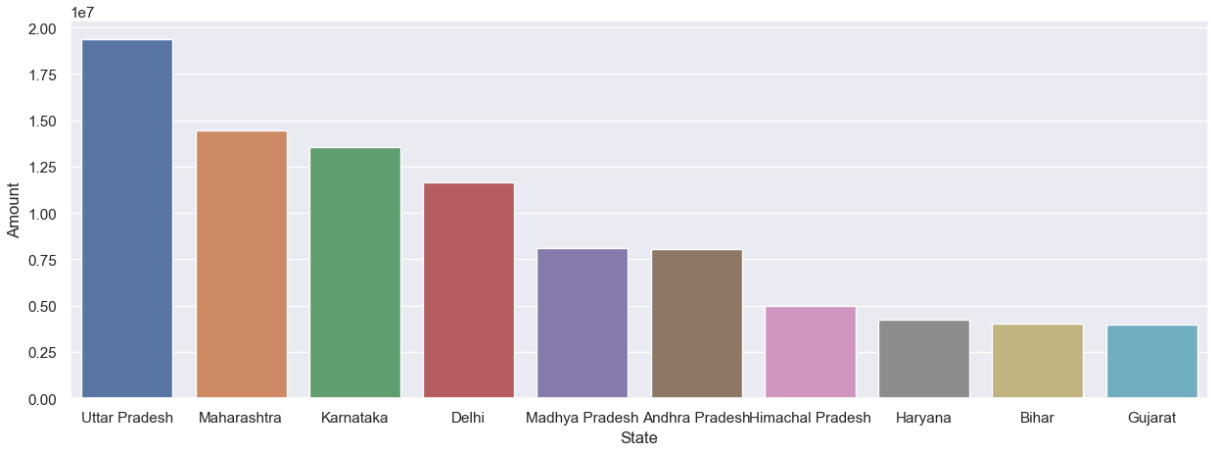
Out[98]:

	State	Amount
14	Uttar Pradesh	19393874
10	Maharashtra	14436996
7	Karnataka	13532993
2	Delhi	11632177
9	Madhya Pradesh	8120048
0	Andhra Pradesh	8046599
5	Himachal Pradesh	4963368
4	Haryana	4220175
1	Bihar	4022757
3	Gujarat	3964988

In [100…

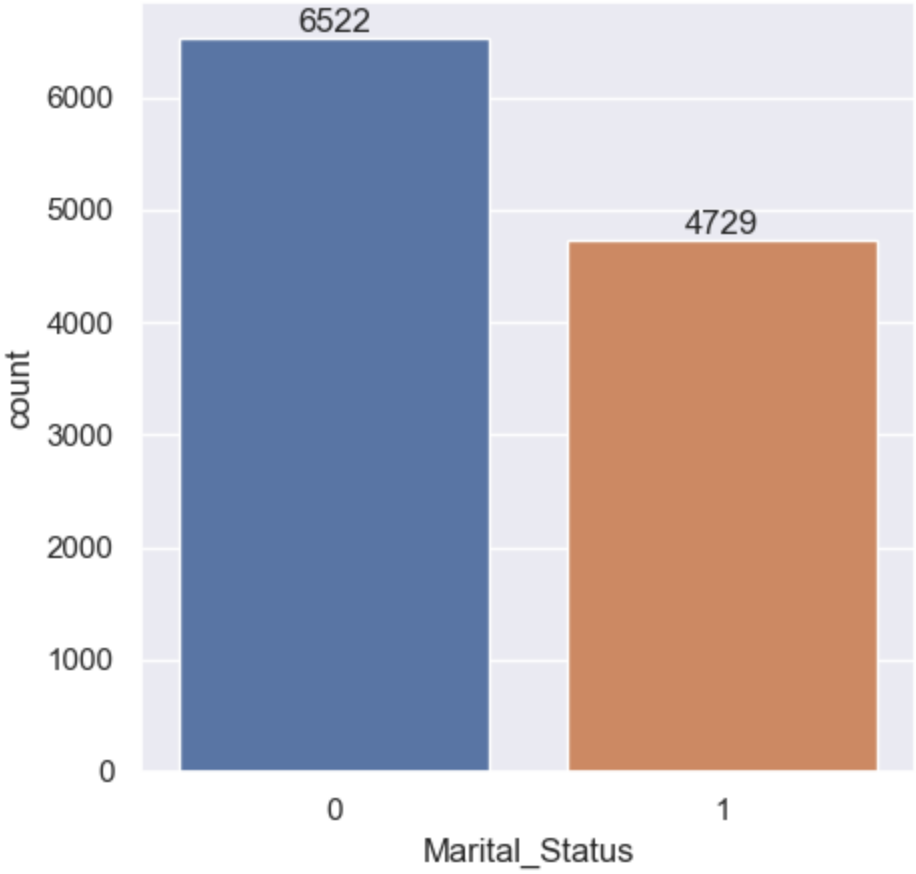
```
sb.set(rc={'figure.figsize':(15,5)})
sb.barplot(data=state_amount,x='State',y='Amount')
```

Out[100]: <Axes: xlabel='State', ylabel='Amount'>



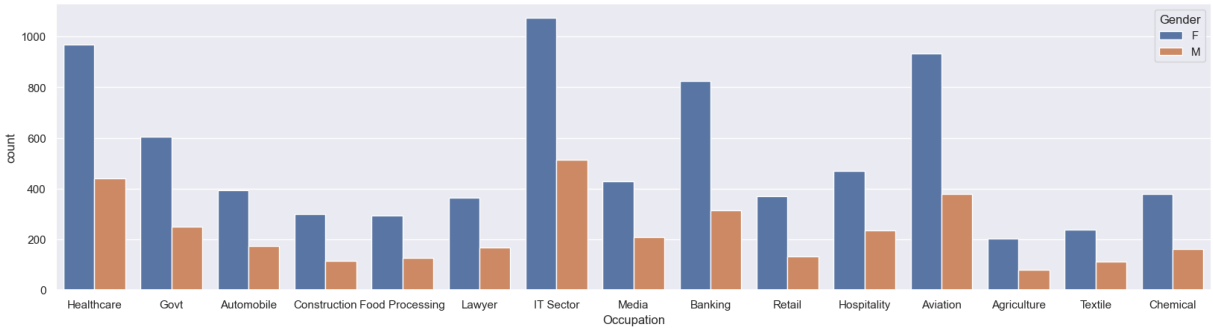
In [104…

```
ms=sb.countplot(x="Marital_Status",data=df)
sb.set(rc={'figure.figsize':(5,5)})
for bar in ms.containers:
    ms.bar_label(bar)
```



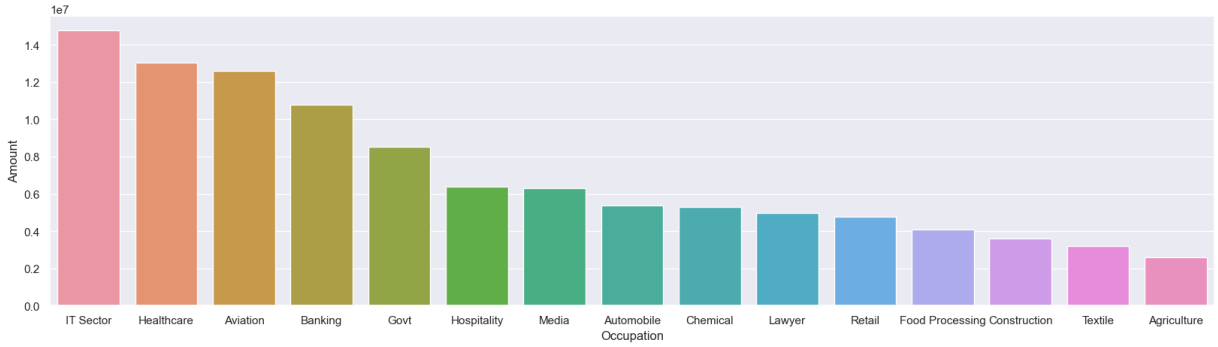
Occupation

In [110…
```
o=sb.countplot(x="Occupation",data=df,hue='Gender')
sb.set(rc={'figure.figsize':(20,5)})
```



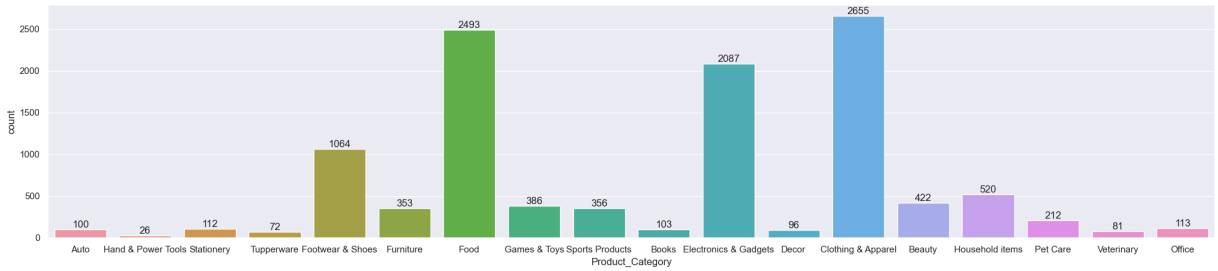
In [118…
```
occu_amount=df.groupby(["Occupation"],as_index=False)['Amount'].sum().sort_values(b
sb.set(rc={'figure.figsize':(20,5)})
sb.barplot(data=occu_amount,x='Occupation',y='Amount')
```

Out[118]: <Axes: xlabel='Occupation', ylabel='Amount'>



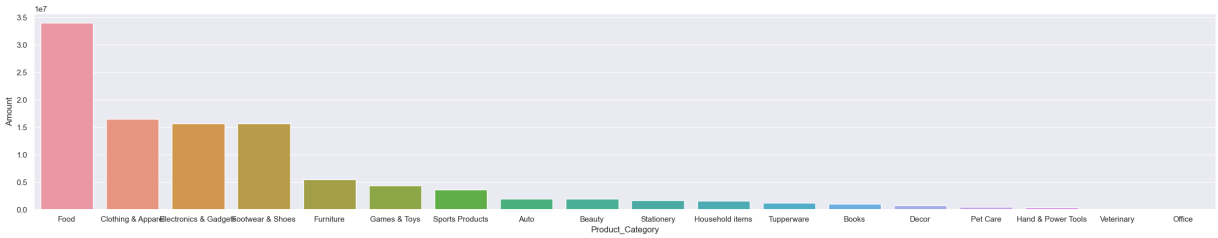
Product Category

In [122…
```
pc=sb.countplot(x="Product_Category",data=df)
sb.set(rc={'figure.figsize':(25,5)})
for bar in pc.containers:
    pc.bar_label(bar)
```



In [125…
```
pc_amount=df.groupby(["Product_Category"],as_index=False)['Amount'].sum().sort_valu
sb.set(rc={'figure.figsize':(30,5)})
sb.barplot(data=pc_amount,x='Product_Category',y='Amount')
```

Out[125]: `<Axes: xlabel='Product_Category', ylabel='Amount'>`



Conclusion

married women withage group 26-35 from UP,Maharastra and Karnataka working in IT,Healtcare and Aviation are most likely to buy product from Food, Clothing and Electronic Category.