Importing essential libraries and data

In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [8]:
```
df = pd.read_csv('Diwali Sales Data.csv',encoding='unicode_escape')
```

Understanding and cleaning data

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

Out[10]:
```
(11251, 15)
```

In [12]:
```
df.head()
```

Out[12]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Mah
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar
3	1001425	Sudevi	P00237842	M	0-17	16	0	Ka
4	1000588	Joni	P00057942	M	26-35	28	1	

In [14]:
```
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 [20]:
```
df.drop(['Status','unnamed1'],axis=1,inplace=True)
```

In [22]:
```
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 [24]:
```
pd.isnull(df).sum()
```

Out[24]:
```
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 [26]:
```
df.dropna(inplace=True)
```

In [36]:
```
pd.isnull(df).sum()
```

Out[36]:
```
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 [28]:
```
df['Amount'] = df['Amount'].astype('int')
```

In [30]:
```
df['Amount'].dtypes
```

Out[30]:
```
dtype('int64')
```

In [32]:
```
df.columns
```

Out[32]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [38]:
```
df.describe()
```

Out[38]:

	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[['Age','Orders','Amount']].describe()
```

Out[46]:

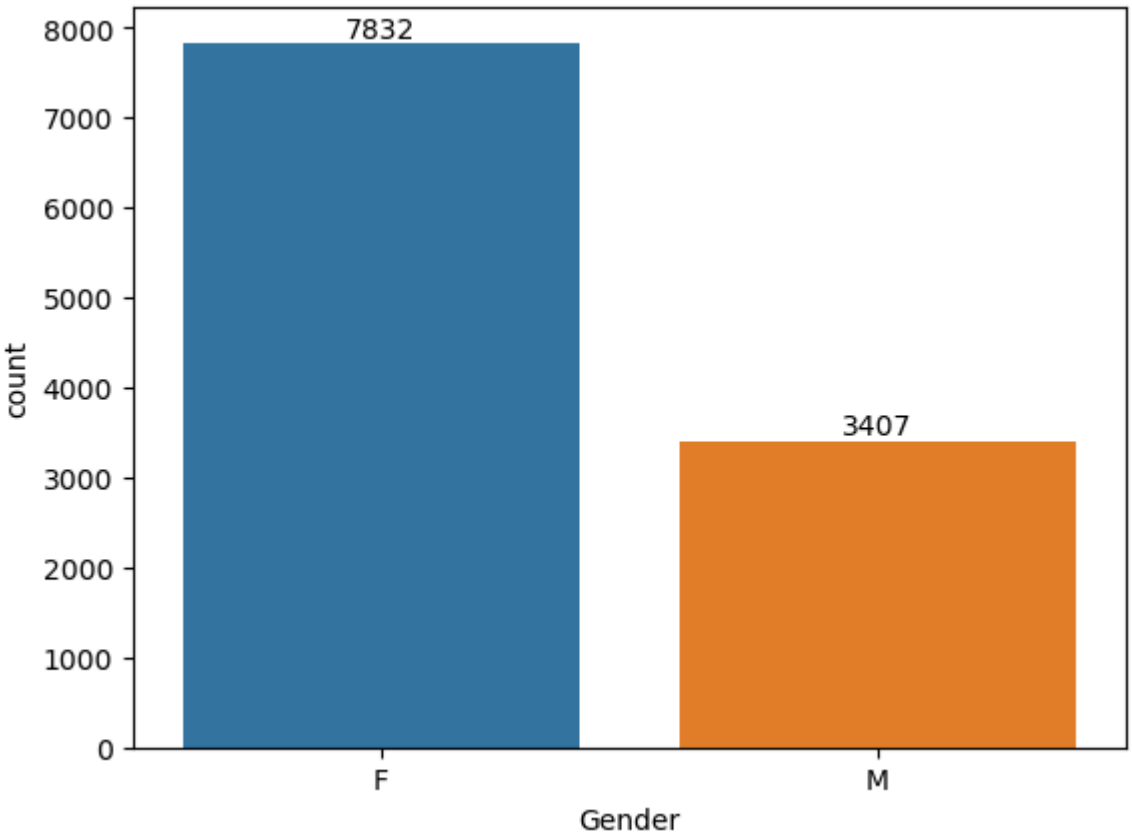
	Age	Orders	Amount
count	11239.000000	11239.000000	11239.000000
mean	35.410357	2.489634	9453.610553
std	12.753866	1.114967	5222.355168
min	12.000000	1.000000	188.000000
25%	27.000000	2.000000	5443.000000
50%	33.000000	2.000000	8109.000000
75%	43.000000	3.000000	12675.000000
max	92.000000	4.000000	23952.000000

EDA

Gender

In [56]:

```
ax = sns.countplot(x='Gender',data=df)
for bars in ax.containers:
    ax.bar_label(bars)
```



In [134…

```
salesGen = df.groupby(['Gender'],as_index=False)['Amount'].sum()
salesGen
```

Out[134…

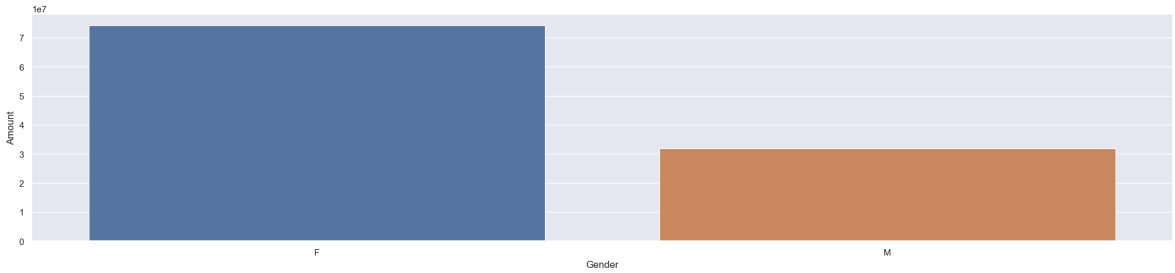
	Gender	Amount
0	F	74335853
1	M	31913276

In [136…

```
salesGen = df.groupby(['Gender'],as_index=False)['Amount'].sum()
sns.barplot(x='Gender',y='Amount',data=salesGen)
```

Out[136…

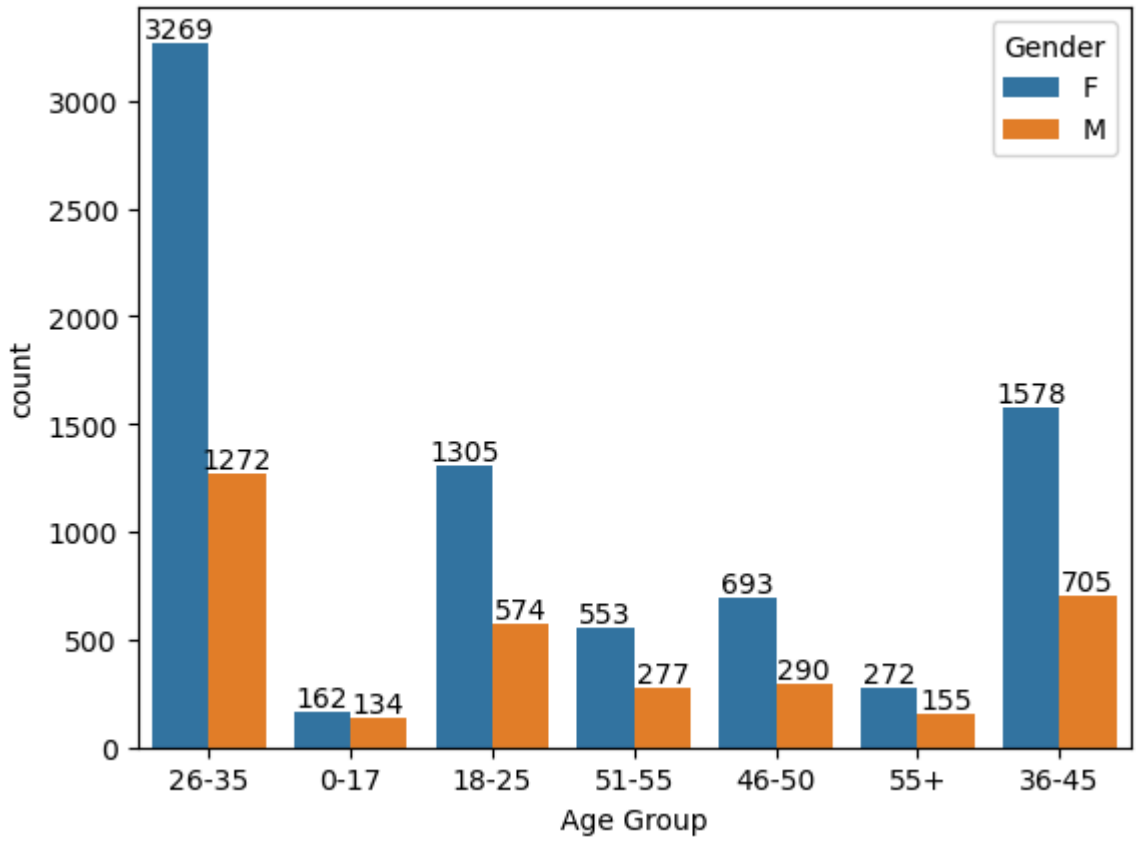
```
<Axes: xlabel='Gender', ylabel='Amount'>
```



Age

In [71]:

```
ax = sns.countplot(data=df,x='Age Group',hue='Gender')
for bars in ax.containers:
    ax.bar_label(bars)
```



In [138…

```
salesAge = df.groupby(['Age Group'], as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
salesAge
```

Out[138…

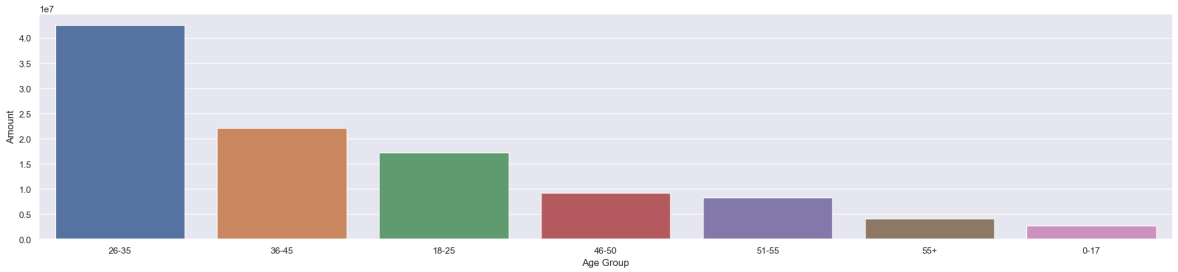
	Age Group	Amount
2	26-35	42613442
3	36-45	22144994
1	18-25	17240732
4	46-50	9207844
5	51-55	8261477
6	55+	4080987
0	0-17	2699653

In [140…

```
salesAge = df.groupby(['Age Group'], as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
sns.barplot(x='Age Group',y='Amount',data = salesAge)
```

Out[140…

```
<Axes: xlabel='Age Group', ylabel='Amount'>
```



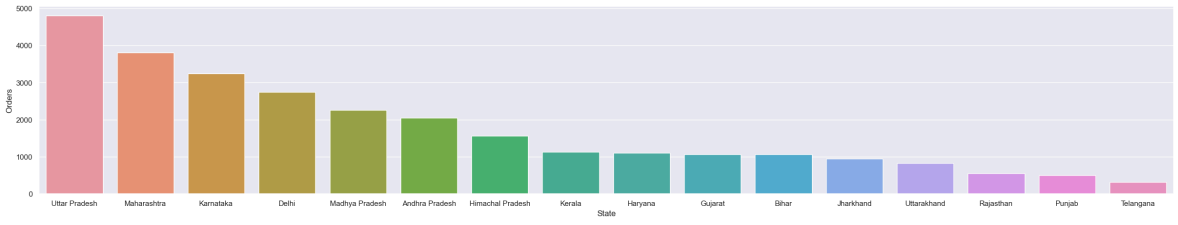
State

In [92]:

```
salesState = df.groupby(['State'],as_index=False)
['Orders'].sum().sort_values(by='Orders',ascending = False)
sns.set(rc={'figure.figsize':(30,5)})
sns.barplot(data=salesState,x='State',y='Orders')
```

Out[92]:

```
<Axes: xlabel='State', ylabel='Orders'>
```



In [142…

```
salesState = df.groupby(['State'],as_index=False)
['Amount'].sum().sort_values(by ='Amount',ascending=False)
salesState
```

Out[142...

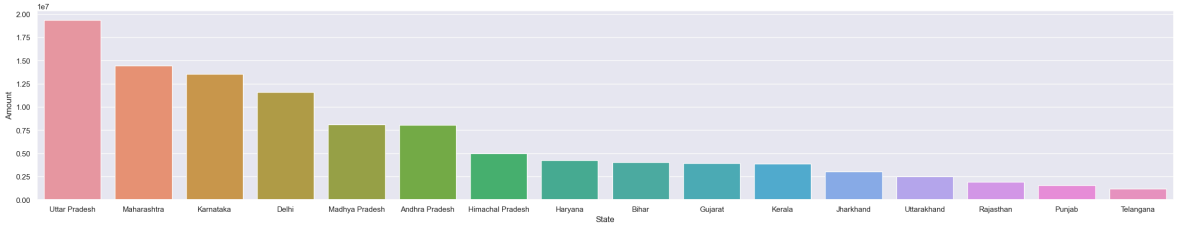
	State	Amount
14	Uttar Pradesh	19374968
10	Maharashtra	14427543
7	Karnataka	13523540
2	Delhi	11603818
9	Madhya Pradesh	8101142
0	Andhra Pradesh	8037146
5	Himachal Pradesh	4963368
4	Haryana	4220175
1	Bihar	4022757
3	Gujarat	3946082
8	Kerala	3894491
6	Jharkhand	3026456
15	Uttarakhand	2520944
12	Rajasthan	1909409
11	Punjab	1525800
13	Telangana	1151490

In [144...

```
salesState = df.groupby(['State'],as_index=False)
['Amount'].sum().sort_values(by ='Amount',ascending=False)
sns.set(rc={'figure.figsize':(30,5)})
sns.barplot(data=salesState,x='State',y='Amount')
```

Out[144...

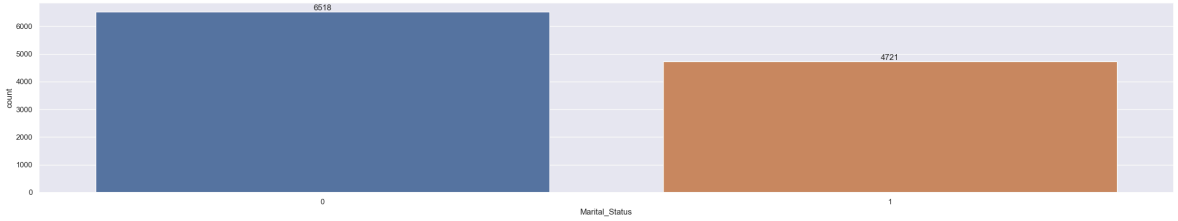
```
<Axes: xlabel='State', ylabel='Amount'>
```



Marital Status

In [109...

```
ax = sns.countplot(data=df,x='Marital_Status')
sns.set(rc={'figure.figsize':(6,5)})
for bars in ax.containers:
    ax.bar_label(bars)
```



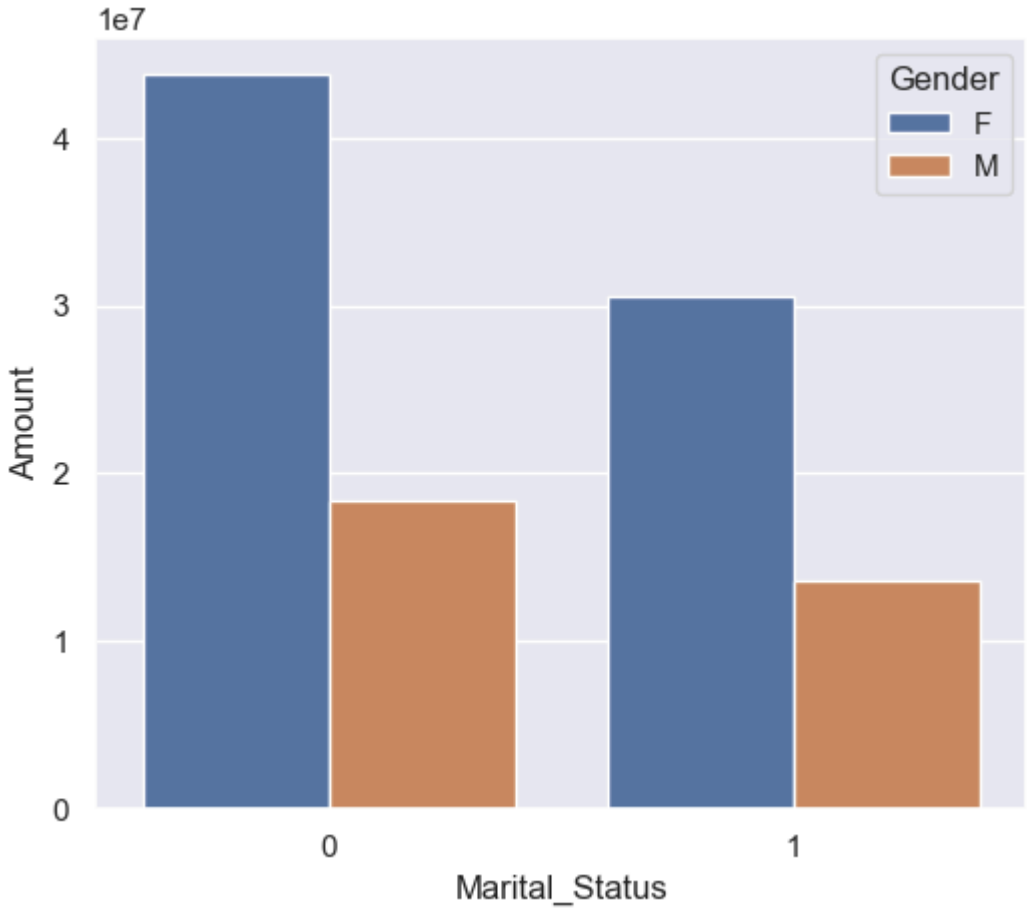
In [146…
```
salesState = df.groupby(['Marital_Status','Gender'],as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
salesState
```

Out[146…

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

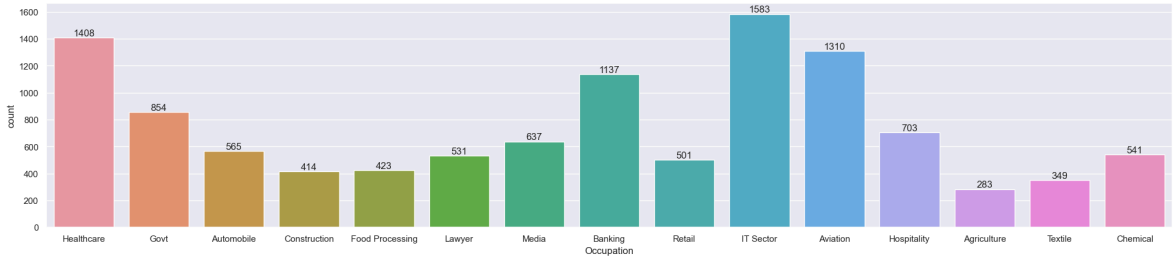
In [121…
```
salesState = df.groupby(['Marital_Status','Gender'],as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
sns.set(rc={'figure.figsize':(6,5)})
sns.barplot(data=salesState, x='Marital_Status',y='Amount',hue='Gender')
```

Out[121…
```
<Axes: xlabel='Marital_Status', ylabel='Amount'>
```



Occupation

In [124…
```
sns.set(rc={'figure.figsize':(25,5)})
ax = sns.countplot(data=df,x='Occupation')
for bars in ax.containers:
    ax.bar_label(bars)
```



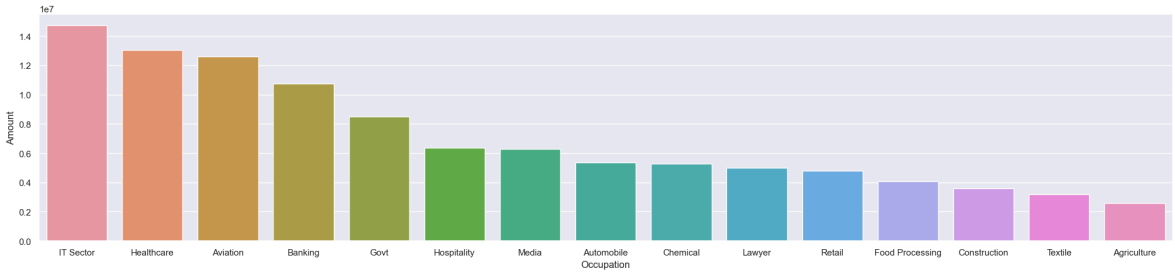
In [148...
```
salesState = df.groupby(['Occupation'],as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
salesState
```

Out[148...

	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
13	Retail	4783170
6	Food Processing	4070670
5	Construction	3597511
14	Textile	3204972
0	Agriculture	2593087

In [152...
```
sns.set(rc={'figure.figsize':(25,5)})
sns.barplot(data=salesState,x='Occupation',y='Amount')
```

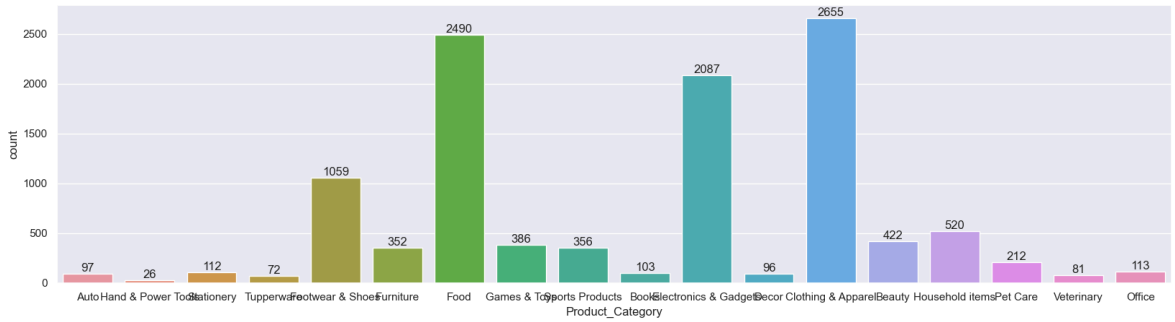
Out[152...
```
<Axes: xlabel='Occupation', ylabel='Amount'>
```



Product Category

In [155...
```
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 [167…

```
salesState = df.groupby(['Product_Category'],as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
salesState
```

Out[167…

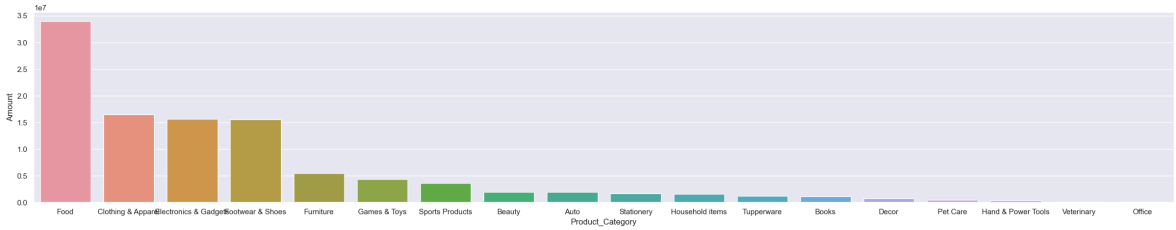
	Product_Category	Amount
6	Food	33933883
3	Clothing & Apparel	16495019
5	Electronics & Gadgets	15643846
7	Footwear & Shoes	15575209
8	Furniture	5440051
9	Games & Toys	4331694
14	Sports Products	3635933
1	Beauty	1959484
0	Auto	1958609
15	Stationery	1676051
11	Household items	1569337
16	Tupperware	1155642
2	Books	1061478
4	Decor	730360
13	Pet Care	482277
10	Hand & Power Tools	405618
17	Veterinary	112702
12	Office	81936

In [163…

```
salesState = df.groupby(['Product_Category'],as_index=False)
['Amount'].sum().sort_values(by='Amount',ascending=False)
sns.set(rc={'figure.figsize':(30,5)})
sns.barplot(data=salesState,x='Product_Category',y='Amount')
```

Out[163…

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


1e7
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Amount
Food
Clothing & Apparel
Electronics & Gadgets
Footwear & Shoes
Furniture
Games & Toys
Sports Products
Beauty
Auto
Stationery
Household items
Tupperware
Books
Decor
Pet Care
Hand & Power Tools
Veterinary
Office
Product_Category