In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt # visualizing data
%matplotlib inline
import seaborn as sns
```

In [4]:

```python
df = pd.read_csv('Diwali Sales Data.csv',encoding= 'unicode_escape')
```

In [5]:

```python
df.shape
```

Out[5]:

```
(11251, 15)
```

In [6]:

```python
df.head
```

Out[6]:

```
<bound method NDFrame.head of        User_ID    Cust_name Product_ID Gender Age Group  A
ge  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]:

```python
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 [8]:
```
df.drop(['Status', 'unnamed1'], axis=1, inplace=True)
```

In [9]:
```
pd.isnull(df).sum()
```

Out[9]:
```
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 [11]:
```
df.dropna(inplace=True)
```

In [12]:
```
df.head(10)
```

Loading [MathJax]/extensions/Safe.js

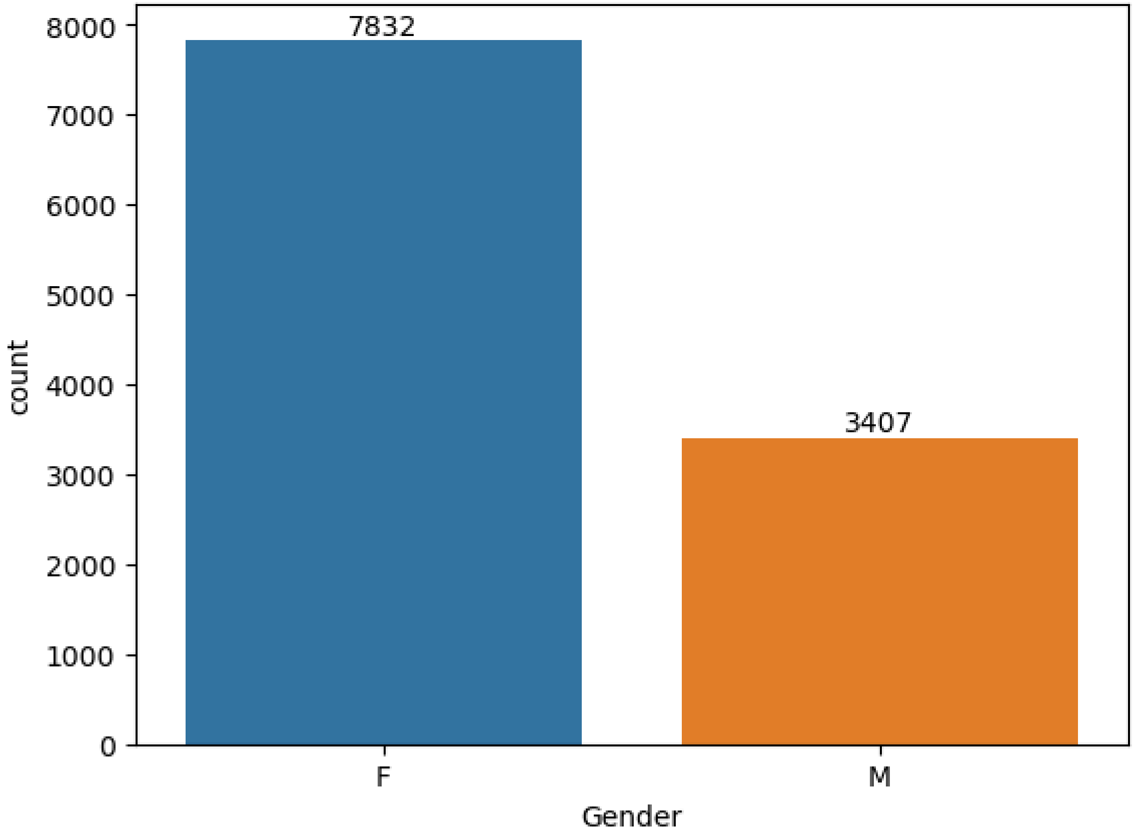
Out[12]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupatio
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcar
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Gov
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobil
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Constructio
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Foo Processin
5	1000588	Joni	P00057942	M	26-35	28	1	Himachal Pradesh	Northern	Foo Processin
6	1001132	Balk	P00018042	F	18-25	25	1	Uttar Pradesh	Central	Lawye
8	1003224	Kushal	P00205642	M	26-35	35	0	Uttar Pradesh	Central	Gov
9	1003650	Ginny	P00031142	F	26-35	26	1	Andhra Pradesh	Southern	Medi
10	1003829	Harshita	P00200842	M	26-35	34	0	Delhi	Central	Bankin

EDA

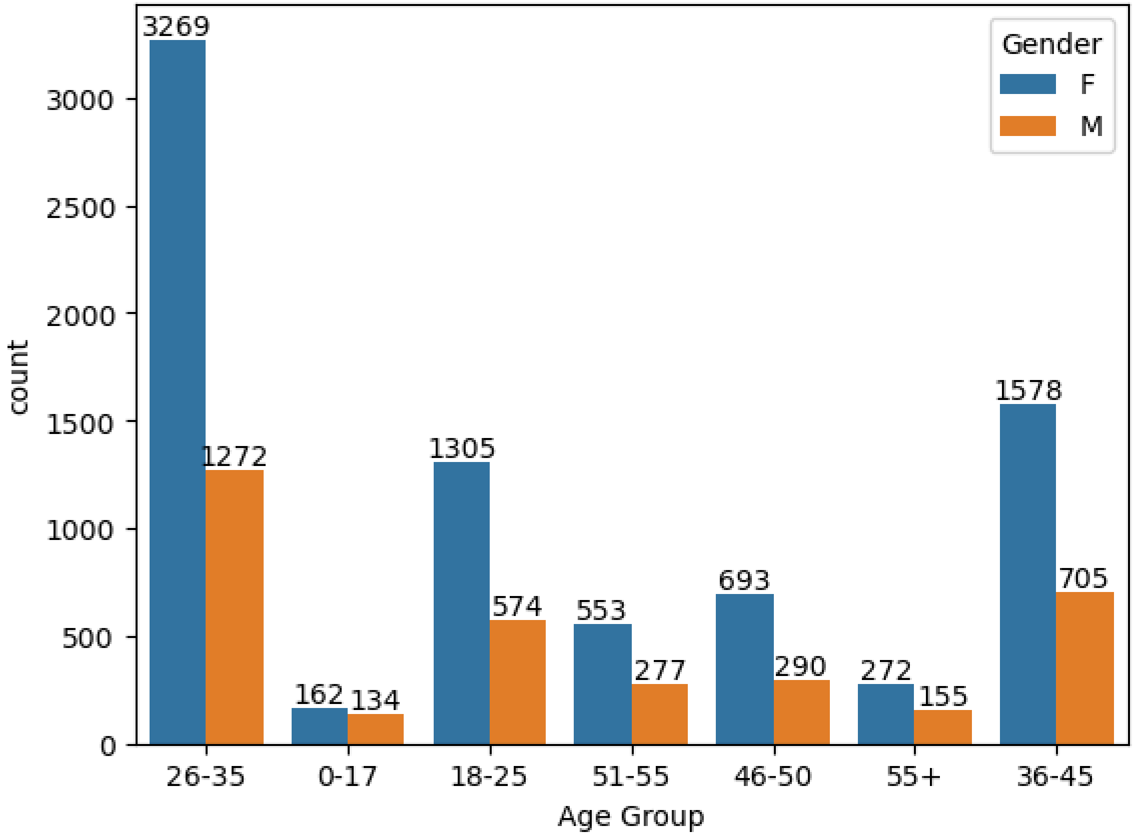
In [13]:

```
# plotting a bar chart for Gender and it's count

ax = sns.countplot(x = 'Gender',data = df)

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



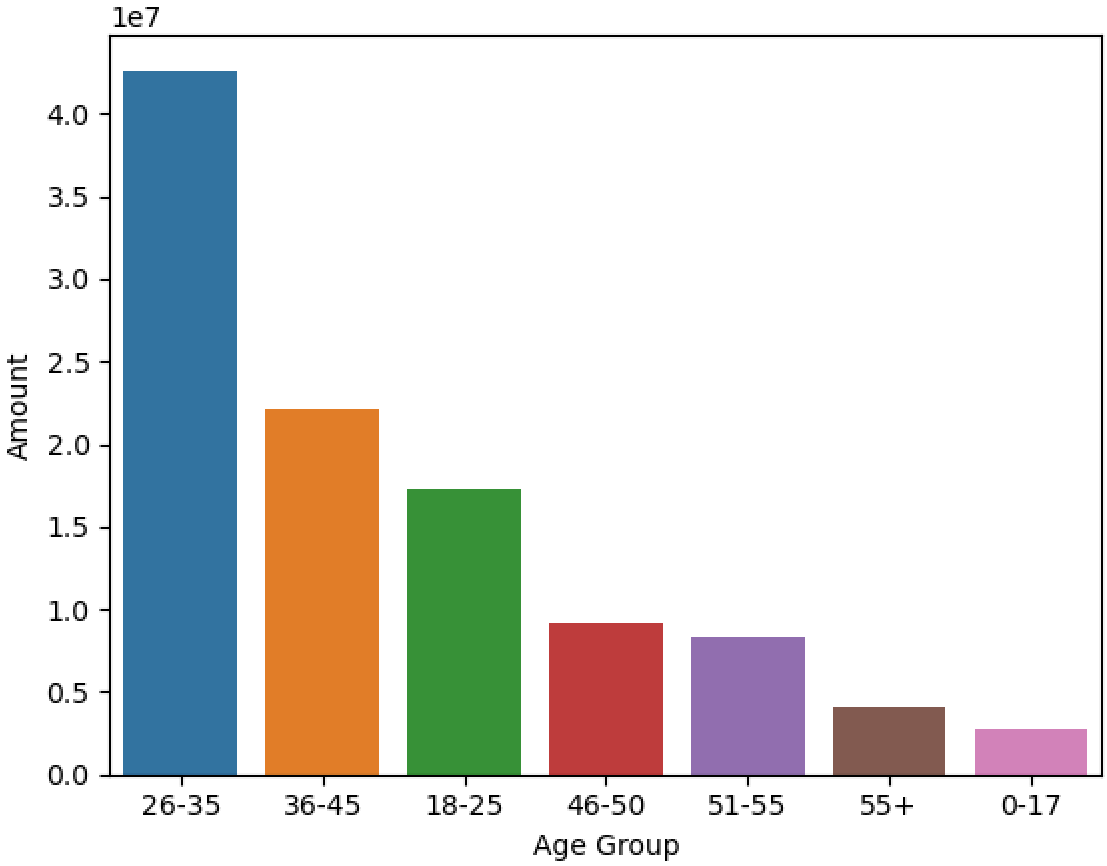
In [16]:

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

for bars in ax.containers:
        ıbel(bars)
```



In [17]:
```python
# Total Amount vs Age Group
sales_age = df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values(by='Am

sns.barplot(x = 'Age Group',y= 'Amount' ,data = sales_age)
```

Out[17]:
```
<AxesSubplot:xlabel='Age Group', ylabel='Amount'>
```



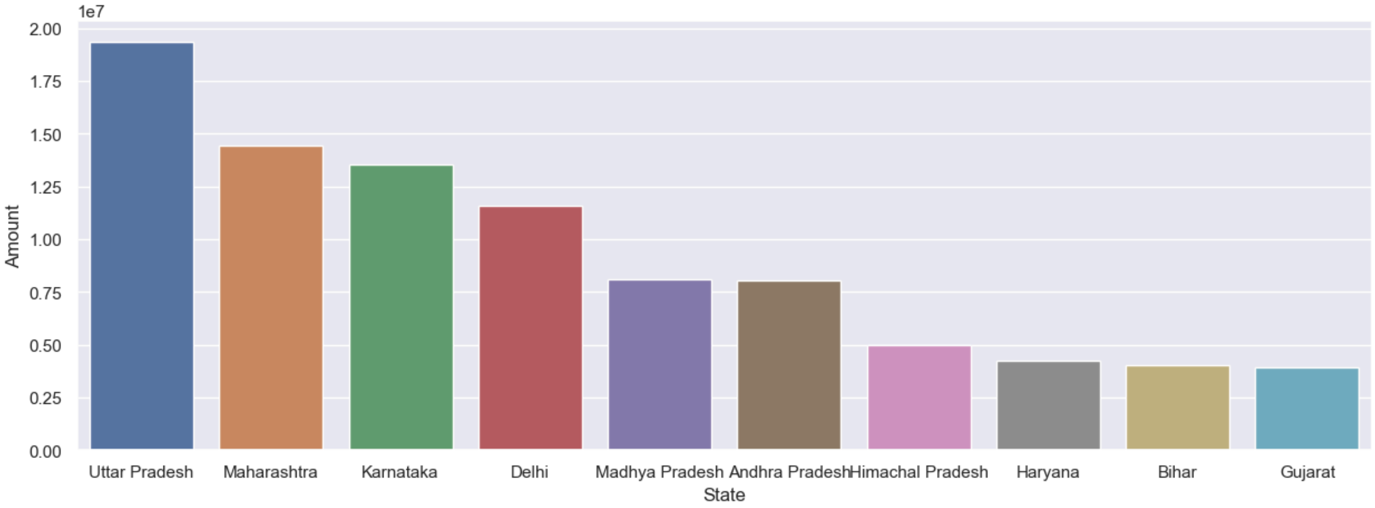
State

In [18]:

```
# total amount/sales from top 10 states

sales_state = df.groupby(['State'], as_index=False)['Amount'].sum().sort_values(by='Amou

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

Out[18]:

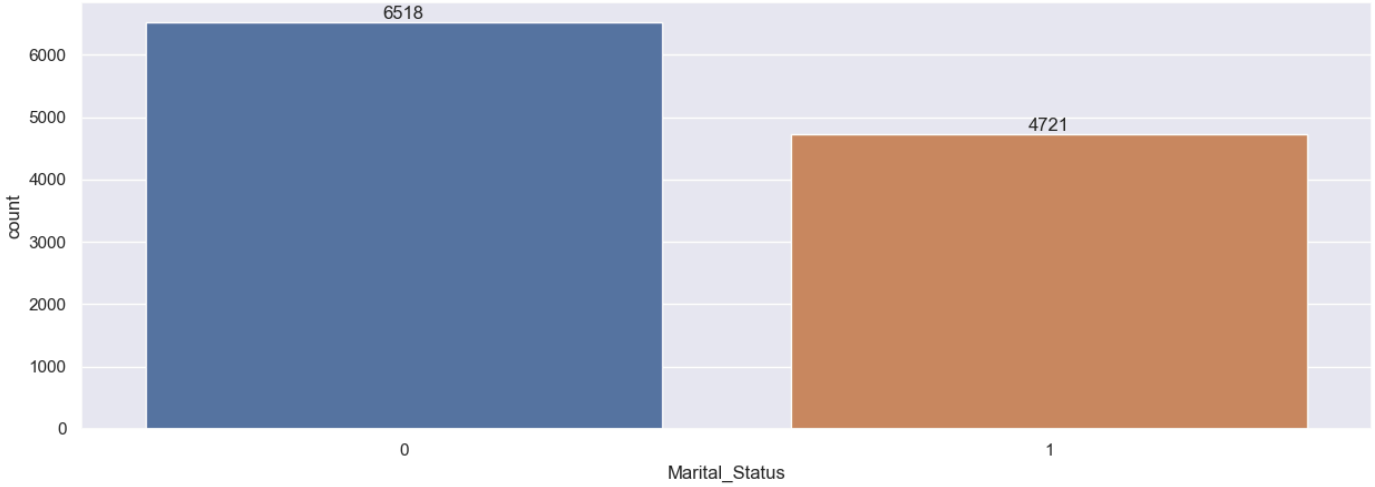
```
<AxesSubplot:xlabel='State', ylabel='Amount'>
```



Marital Status

In [21]:

```
ax = sns.countplot(data = df, x = 'Marital_Status')

sns.set(rc={'figure.figsize':(7,5)})
for bars in ax.containers:
    ax.bar_label(bars)
```



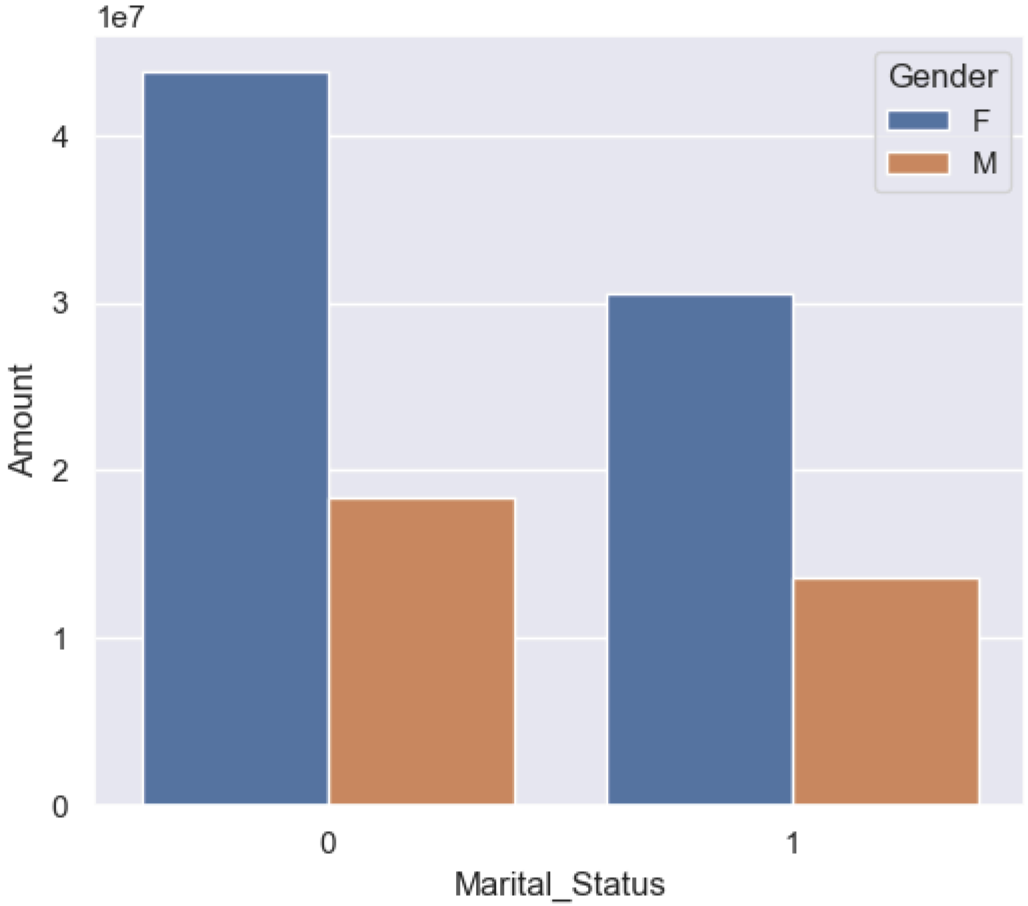
In [22]:

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

sns.set(rc={'figure.figsize':(6,5)})
sns.barplot(data = sales_state, x = 'Marital_Status',y= 'Amount', hue='Gender')
```

Out[22]:

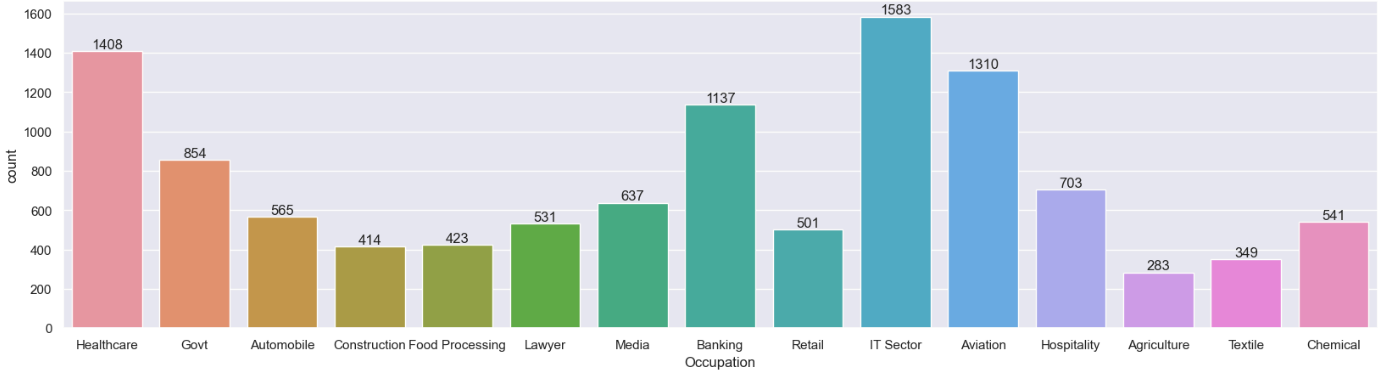
```
<AxesSubplot:xlabel='Marital_Status', ylabel='Amount'>
```

Loading [MathJax]/extensions/Safe.js



Occupation

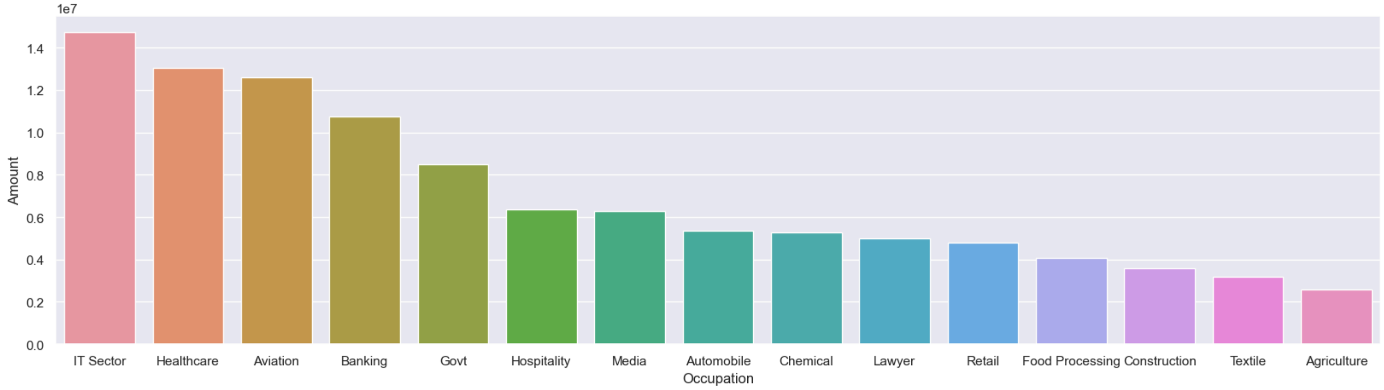
In [23]:
```
sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Occupation')

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



In [24]:
```
sales_state = df.groupby(['Occupation'], as_index=False)['Amount'].sum().sort_values(by=

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

Out[24]:
```
<AxesSubplot:xlabel='Occupation', ylabel='Amount'>
```

Loading [MathJax]/extensions/Safe.js



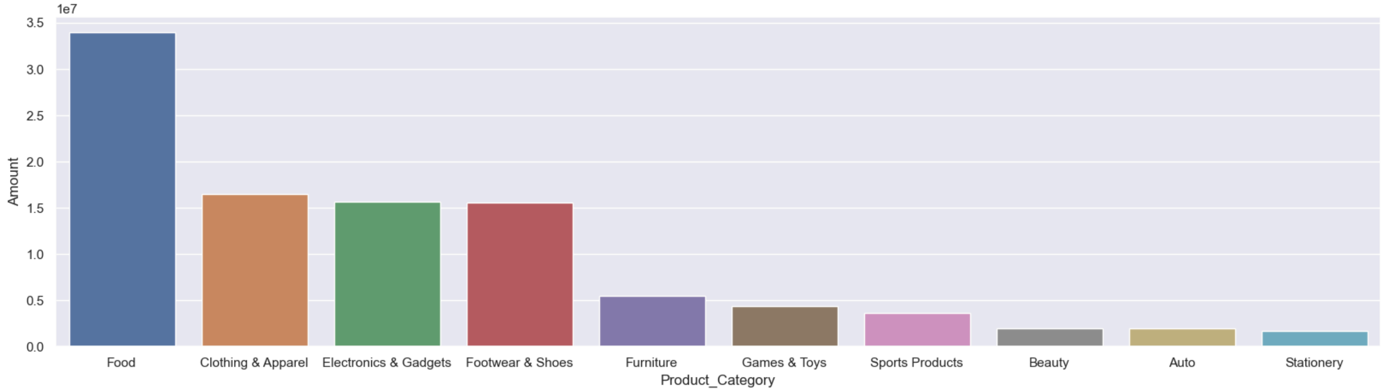
Product Category

In [25]:

```
sales_state = df.groupby(['Product_Category'], as_index=False)['Amount'].sum().sort_valu

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

Out[25]:

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



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

```
In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
```

Loading [MathJax]/extensions/Safe.js

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: