

## Problem Statement:

### Enhancing Customer Experience and Sales through Data Analysis

A company operates in a competitive market where customer experience is a crucial differentiator. Despite having a strong product lineup and a dedicated sales team, they observed a stagnation in their sales growth and customer satisfaction scores over the past few quarters. To maintain their competitive edge and drive growth, it is imperative to leverage data-driven insights to understand their customers better and optimize their sales strategies.

### Objective:

The primary objective of this initiative is to improve customer experience and boost sales by systematically analyzing available data. By understanding customer behaviors, preferences, and pain points, and aim to develop actionable strategies that will enhance customer satisfaction and increase sales conversions.

### Expected Outcomes:

By executing this data-driven approach, we anticipate the following outcomes: 1.Improved customer satisfaction and loyalty through enhanced personalized experiences, 2.Increased sales conversions and revenue growth, 3.More efficient and effective sales processes, 4.Better understanding of customer needs and market trends, 5.Continuous improvement through ongoing measurement and analysis.

## Installing Libraries

```
In [3]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [4]: df = pd.read_csv('C:\Users\as945\OneDrive\Documents\Sales Analysis\sales.csv', encoding='unicode_escape')
```

## Exploring and Cleaning the data

```
In [6]: df.shape
Out[6]: (11251, 15)
```

```
In [4]: df.head()
```

```
Out[4]:
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status		State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
0	1002903	Sanjiv	P00125942	F	26-35	28	0		Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1	1000732	Karika	P00110942	F	26-35	35	1		Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN	NaN
2	1001990	Bindu	P00118542	F	26-35	35	1		Uttar Pradesh	Central	Automobile	Auto	3	23924.0	NaN	NaN
3	1001425	Sudevi	P00237842	M	0-17	16	0		Karnataka	Southern	Construction	Auto	2	23912.0	NaN	NaN
4	1000588	Jon	P00057942	M	26-35	28	1		Gujarat	Western	Food Processing	Auto	2	23877.0	NaN	NaN

```
In [5]: 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 
memory usage: 1.3+ MB
```

```
In [6]: df.drop(['Status', 'unnamed1'], axis=1, inplace=True)
```

```
pd.isnull(df).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

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

```
In [9]: df['Amount'] = df['Amount'].astype('int')
```

```
In [10]: df['Amount'].dtypes
```

```
Out[10]: dtype('int64')
```

```
In [11]: df.columns
```

```
In [13]: Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
              'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
              'Orders', 'Amount'],
            dtype='object')
```

```
In [10]: df.describe()
```

```
Out[13]:
```

	User_ID	Age	Marital_Status	Orders	Amount
count	11239.000000	11239.000000	11239.000000	11239.000000	11239.000000
mean	1.003004e+06	35.410357	0.420555	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%	1.003964e+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 [14]: df[['Age', 'Orders', 'Amount']].describe()
```

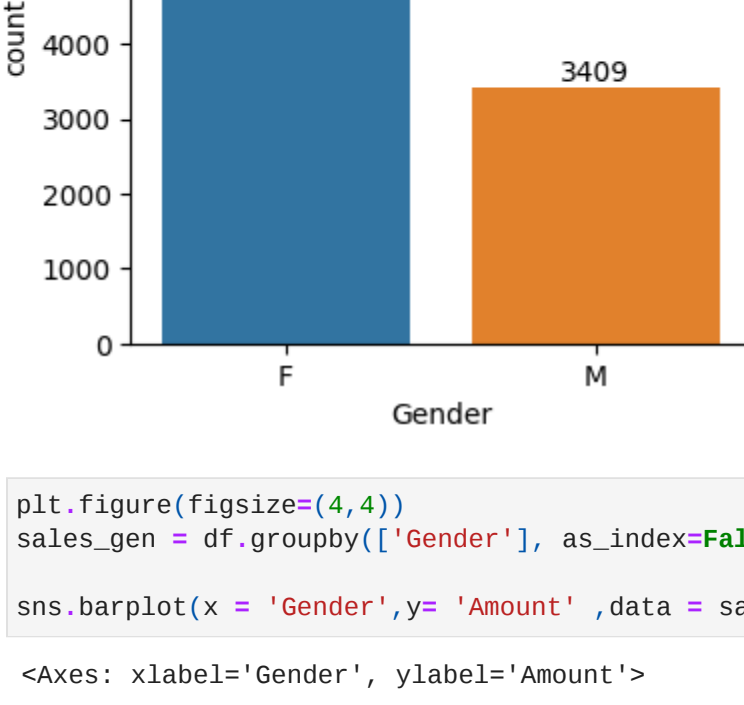
```
Out[14]:
```

	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 Analysis

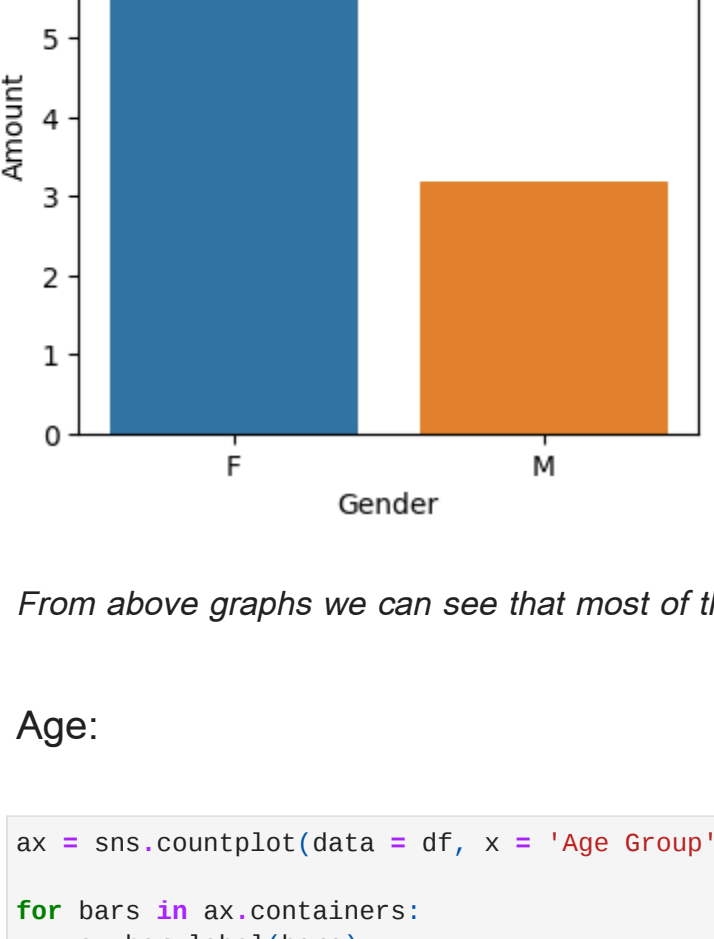
### Gender:

```
In [7]: plt.figure(figsize=(4,4))
sales_gen = sns.countplot(x = 'Gender', data = df, hue='Gender')
for bars in ax.containers:
    ax.bar_label(bars)
```



```
In [21]: plt.figure(figsize=(4,4))
sales_gen = df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)
sns.barplot(x = 'Gender', y = 'Amount', data = sales_gen, hue='Gender')
```

```
Out[21]: <Axes: xlabel='Gender', ylabel='Amount'>
```



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

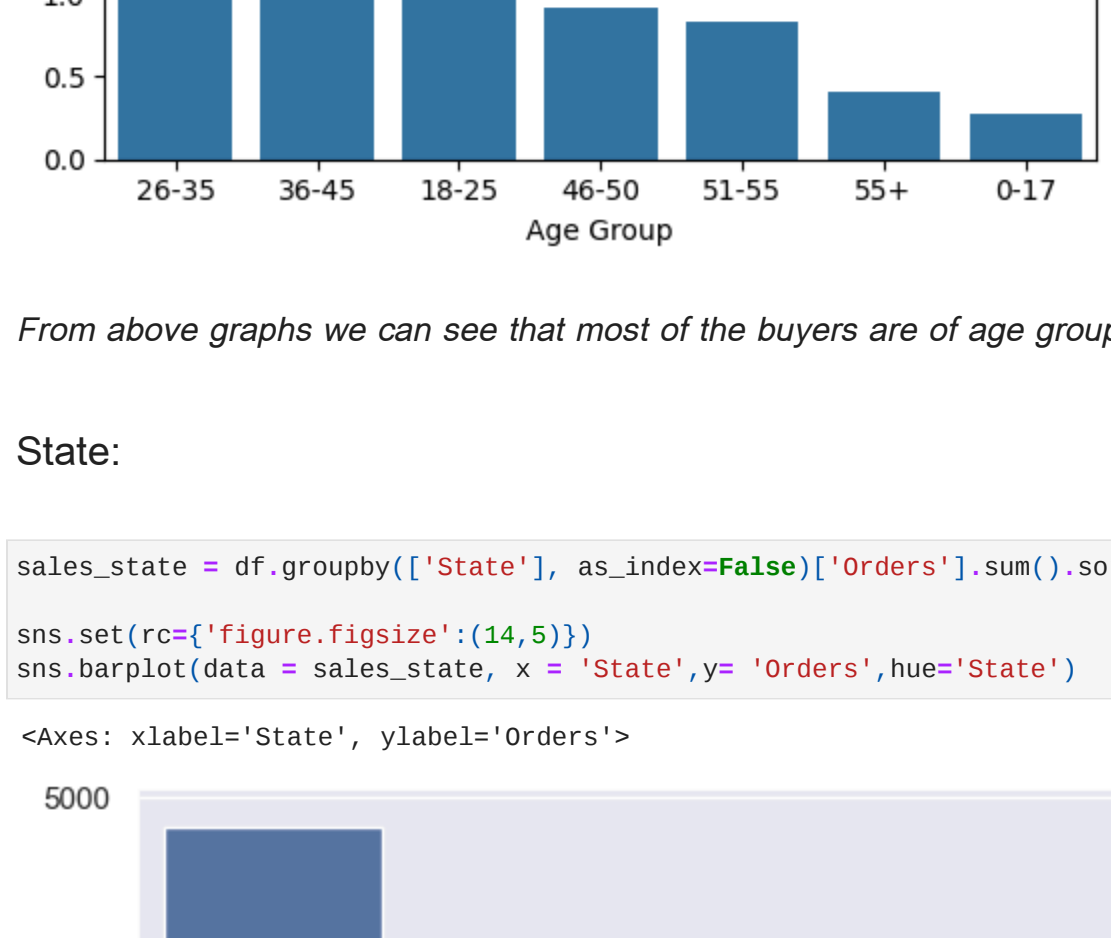
### Age:

```
In [22]: ax = sns.countplot(data = df, x = 'Age Group', hue = 'Gender')
for bars in ax.containers:
    ax.bar_label(bars)
```



```
In [23]: sales_age = df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False)
sns.barplot(x = 'Age Group', y = 'Amount', data = sales_age)
```

```
Out[23]: <Axes: xlabel='Age Group', ylabel='Amount'>
```



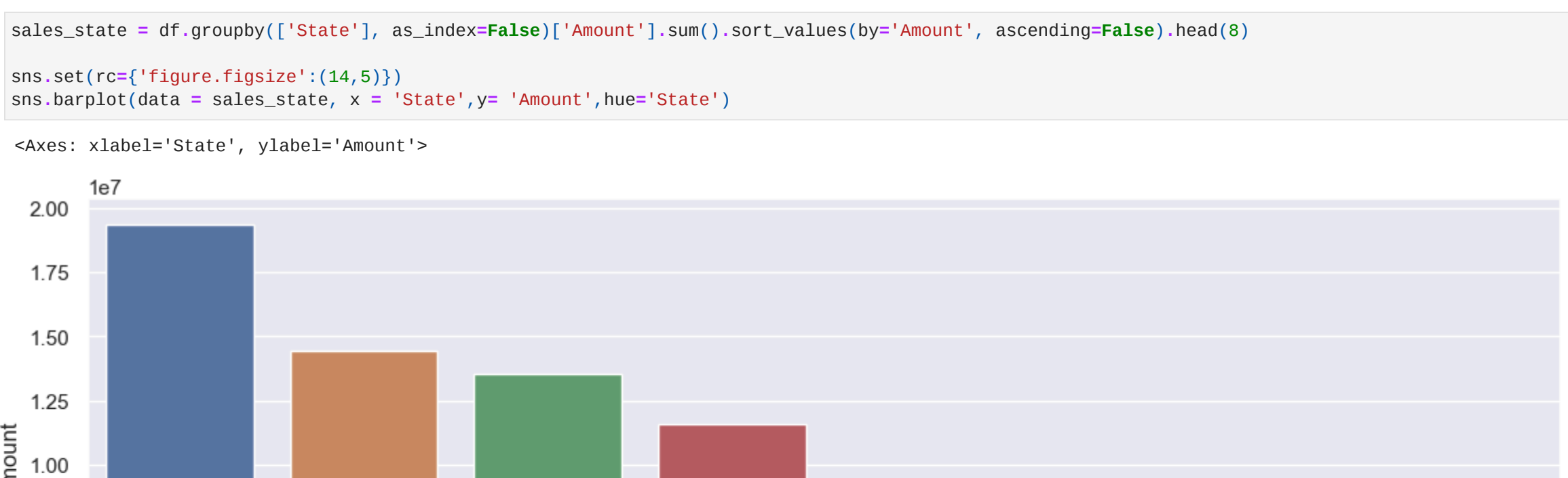
From above graphs we can see that most of the buyers are of age group between 26-35 yrs female

### State:

```
In [47]: sales_state = df.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by='Orders', ascending=False).head(8)
```

```
sns.set(rc={'figure.figsize':(14,5)})
sns.barplot(data = sales_state, x = 'State', y = 'Orders', hue='State')
```

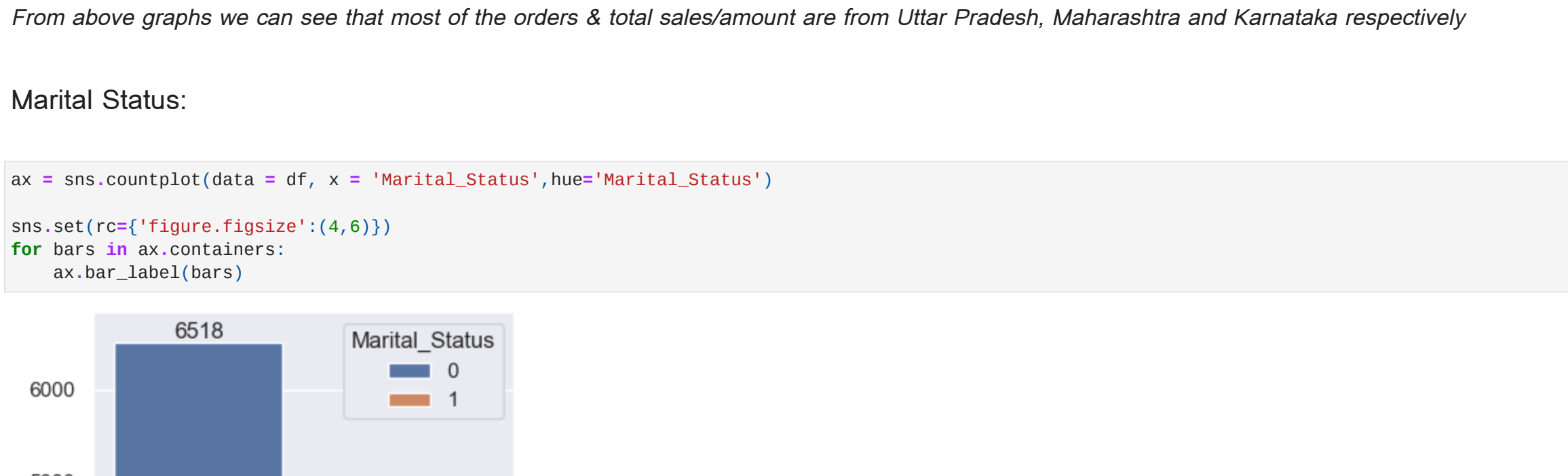
```
Out[47]: <Axes: xlabel='State', ylabel='Orders'>
```



```
In [40]: sales_state = df.groupby(['State'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False).head(8)
```

```
sns.set(rc={'figure.figsize':(14,5)})
sns.barplot(data = sales_state, x = 'State', y = 'Amount', hue='State')
```

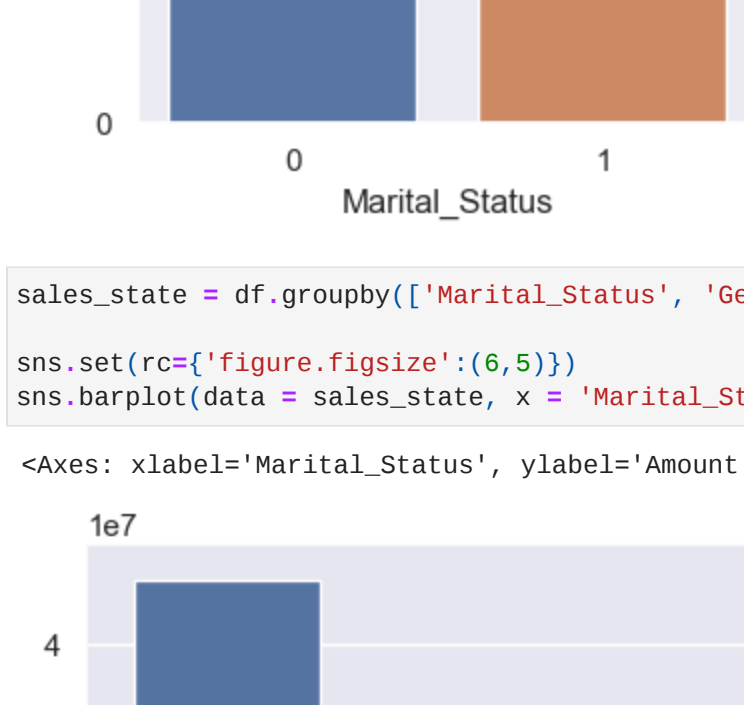
```
Out[40]: <Axes: xlabel='State', ylabel='Amount'>
```



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

### Marital Status:

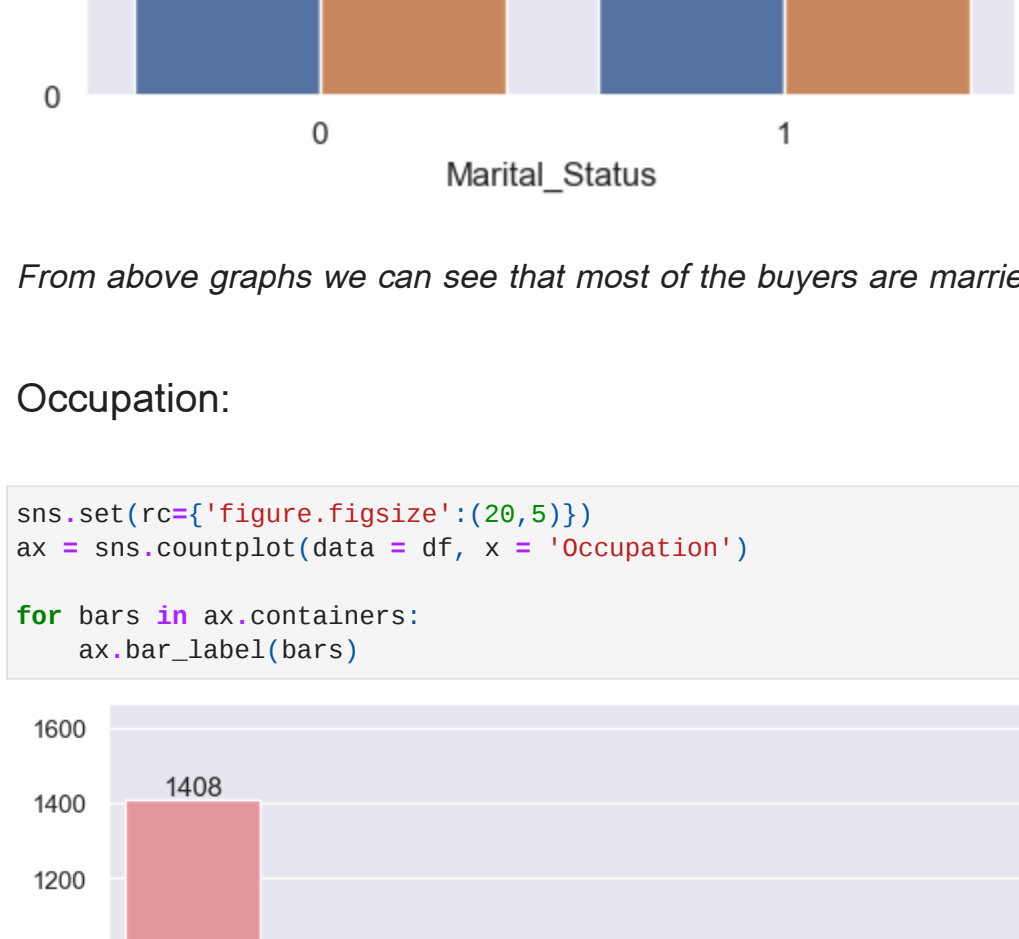
```
In [44]: ax = sns.countplot(data = df, x = 'Marital_Status', hue='Marital_Status')
sns.set(rc={'figure.figsize':(4,6)})
for bars in ax.containers:
    ax.bar_label(bars)
```



```
In [22]: sales_state = 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 = sales_state, x = 'Marital_Status', y = 'Amount', hue='Gender')
```

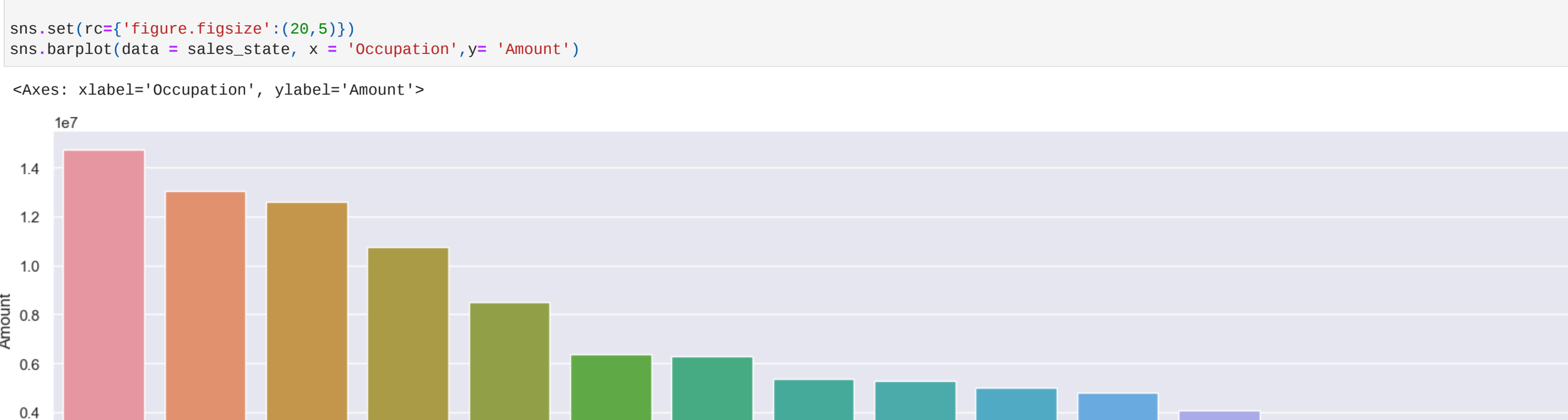
```
Out[22]: <Axes: xlabel='Marital_Status', ylabel='Amount'>
```



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

### Occupation:

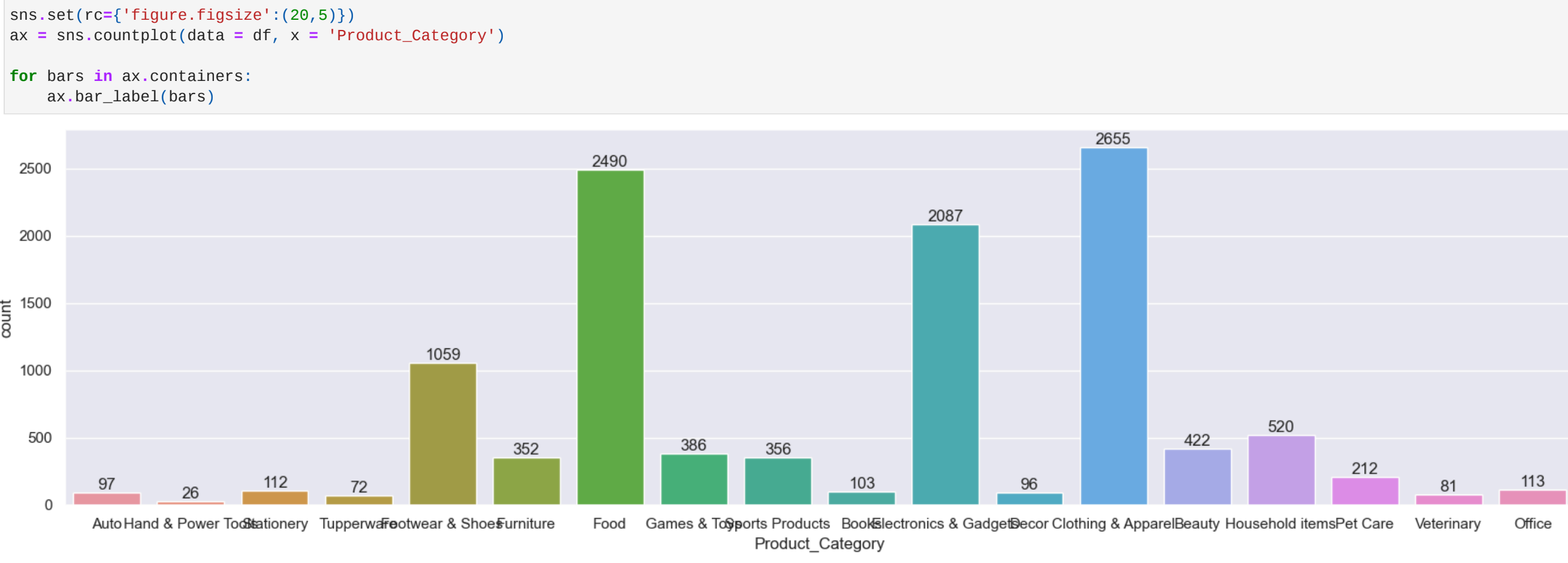
```
In [23]: sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Occupation')
for bars in ax.containers:
    ax.bar_label(bars)
```



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

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

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



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

### Product Category:

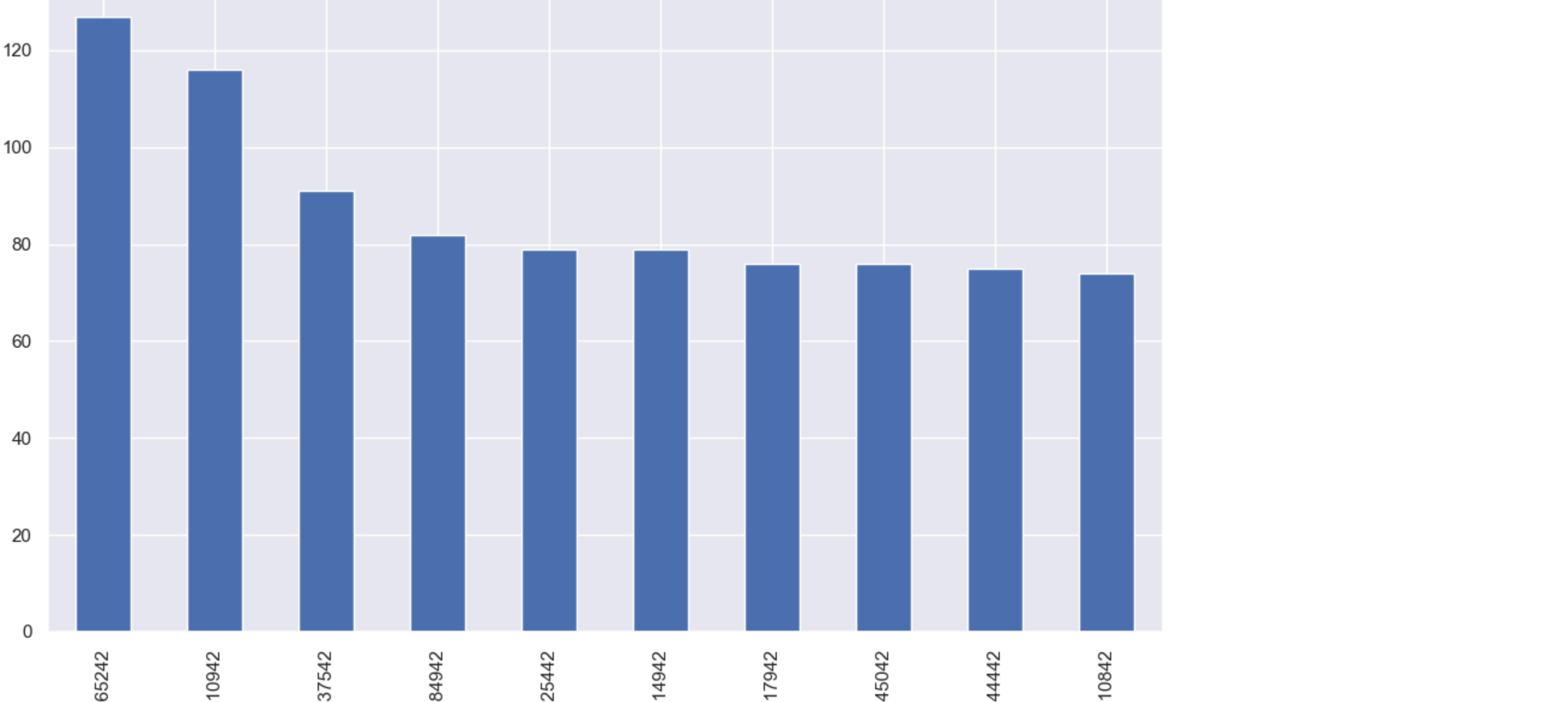
```
In [25]: sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Product_Category')
for bars in ax.containers:
    ax.bar_label(bars)
```



```
In [26]: sales_state = df.groupby(['Product_Category'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending=False).head(10)
```

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

```
Out[26]: <Axes: xlabel='Product_Category', ylabel='Amount'>
```



```
In [27]: sales_state = df.groupby(['Product_ID'], as_index=False)['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)
```

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

```
Out[27]: <Axes: xlabel='Product_ID', ylabel='Orders'>
```



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

## 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

Thank you!