

```
In [1]: !pip install numpy
```

```
Requirement already satisfied: numpy in c:\users\subhe\anaconda3\lib\site-packages (1.24.3)
```

```
In [2]: !pip install seaborn
```

```
Requirement already satisfied: seaborn in c:\users\subhe\anaconda3\lib\site-packages (0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\subhe\anaconda3\lib\site-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in c:\users\subhe\anaconda3\lib\site-packages (from seaborn) (1.5.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\subhe\anaconda3\lib\site-packages (from seaborn) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (23.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\subhe\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\subhe\anaconda3\lib\site-packages (from pandas>=0.25->seaborn) (2022.7)
Requirement already satisfied: six>=1.5 in c:\users\subhe\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
```

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

```
In [4]: df = pd.read_csv('Diwali Sales Data.csv', encoding = 'unicode_escape')
```

```
In [5]: df.shape
```

```
Out[5]: (11251, 13)
```

```
In [6]: df.head()
```

```
Out[6]:
```

| | User_ID | Cust_name | Product_ID | Gender | Age Group | Age | Marital_Status | State | Zone | Occupation |
|---|---------|-----------|------------|--------|-----------|-----|----------------|----------------|----------|-----------------|
| 0 | 1002903 | Sanskriti | P00125942 | F | 26-35 | 28 | 0 | Maharashtra | Western | Healthcare |
| 1 | 1000732 | Kartik | P00110942 | F | 26-35 | 35 | 1 | Andhra Pradesh | Southern | Govt |
| 2 | 1001990 | Bindu | P00118542 | F | 26-35 | 35 | 1 | Uttar Pradesh | Central | Automobile |
| 3 | 1001425 | Sudevi | P00237842 | M | 0-17 | 16 | 0 | Karnataka | Southern | Construction |
| 4 | 1000588 | Joni | P00057942 | M | 26-35 | 28 | 1 | Gujarat | Western | Food Processing |

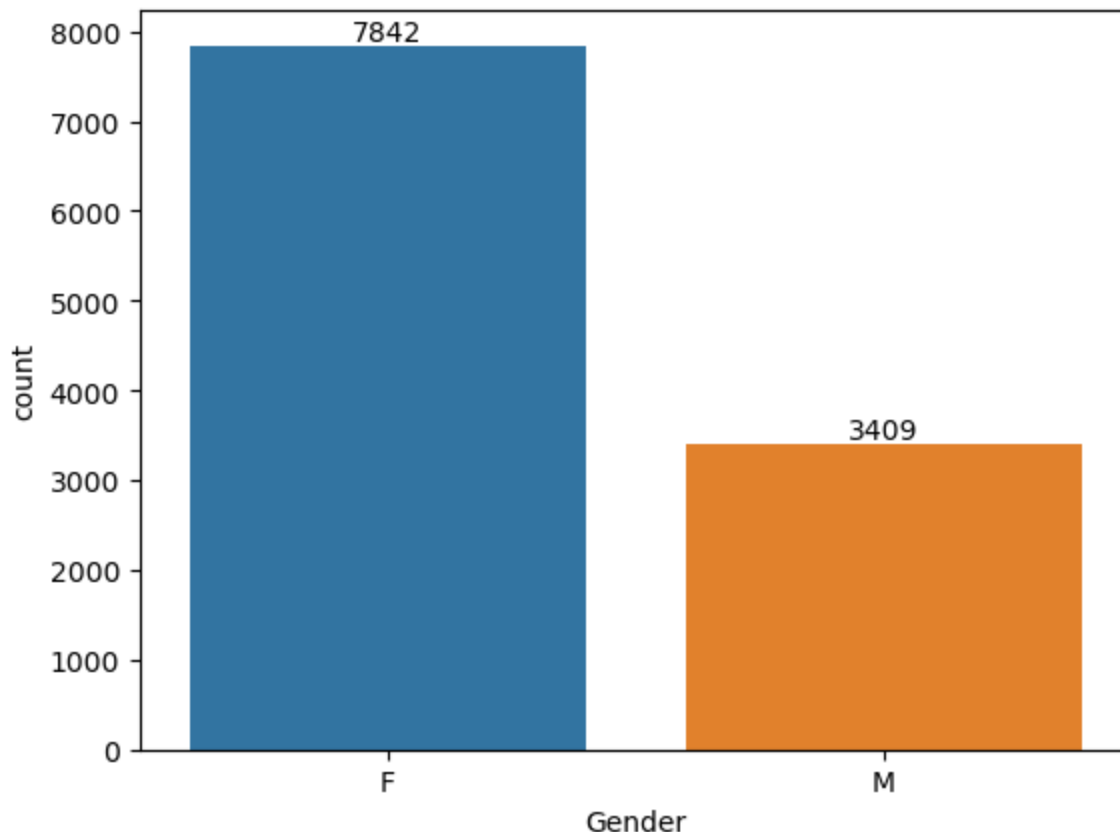
```
In [7]: df.info()
```

```
<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
```

EXPLORATORY DATA ANALYSIS EDA

```
In [9]: ax = sns.countplot(x='Gender', data=df)
```

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



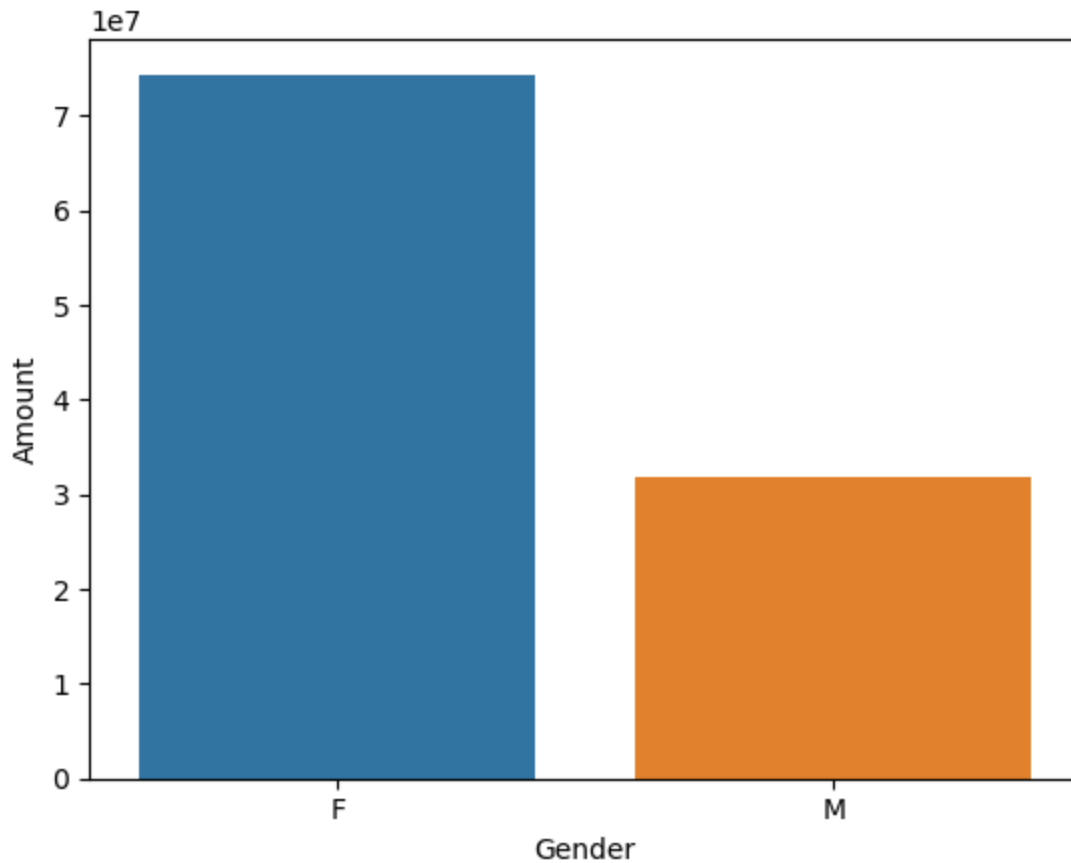
```
In [10]: df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount', ascending
```

```
Out[10]:
```

| | Gender | Amount |
|---|--------|-------------|
| 0 | F | 74335856.43 |
| 1 | M | 31913276.00 |

```
In [17]: sales_gen = df.groupby(['Gender'], as_index=False)['Amount'].sum().sort_values(by='Amount')
sns.barplot(x='Gender', y='Amount', data = sales_gen)
```

```
Out[17]: <Axes: xlabel='Gender', ylabel='Amount'>
```

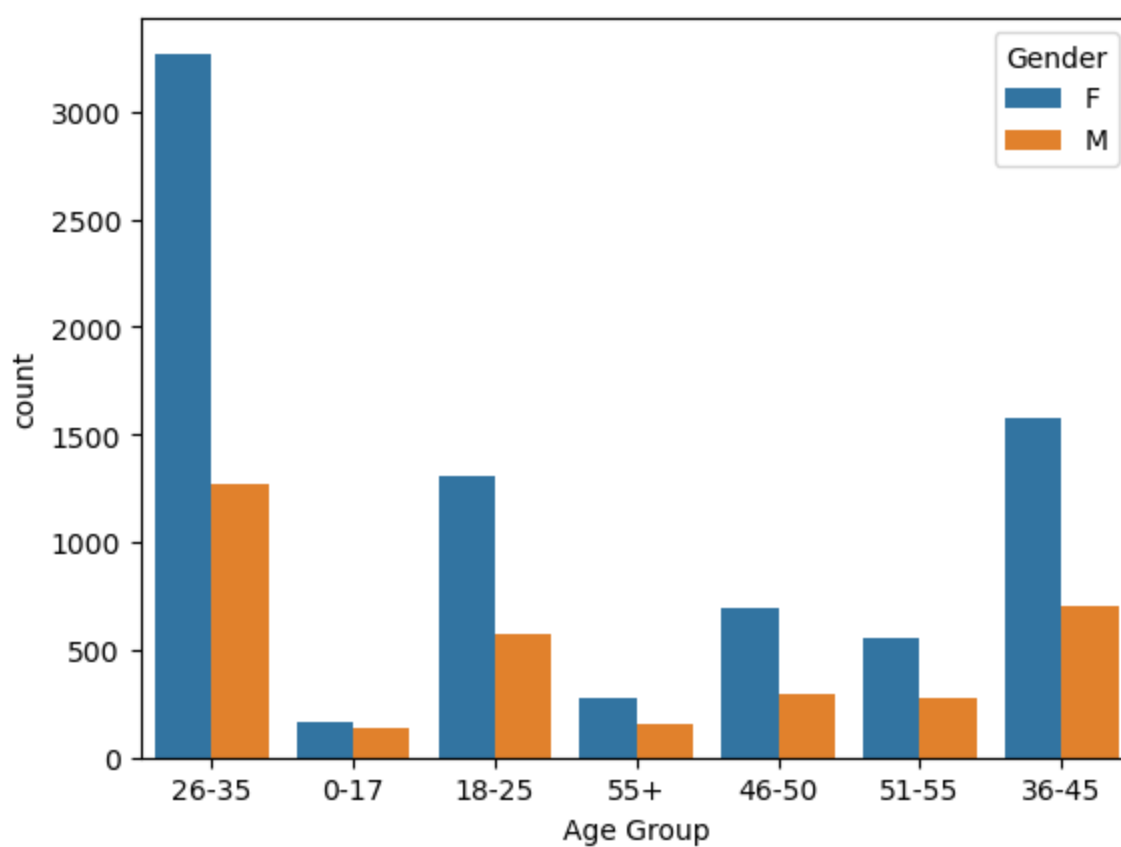


From the above graph we can see that most of the buyers are female and the purchasing power of female is more than male.

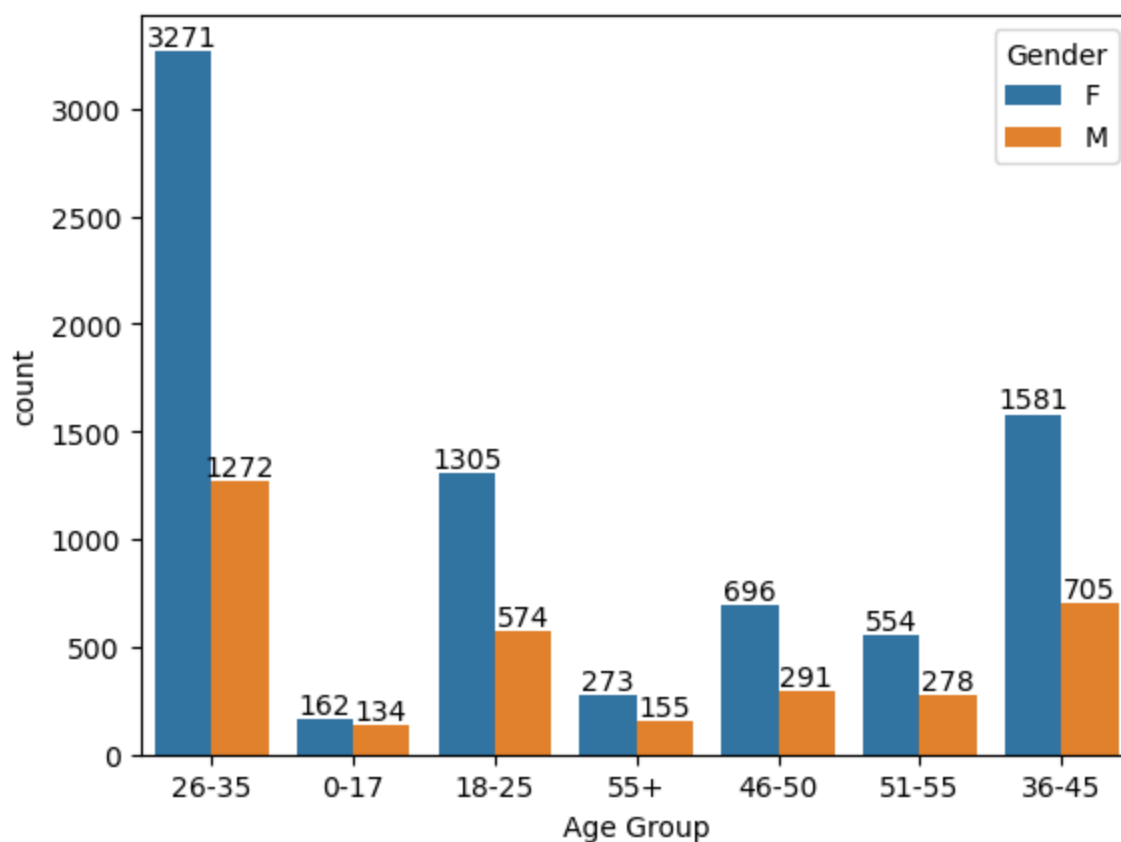
AGE

```
In [19]: sns.countplot(data=df, x='Age Group', hue='Gender')
```

```
Out[19]: <Axes: xlabel='Age Group', ylabel='count'>
```

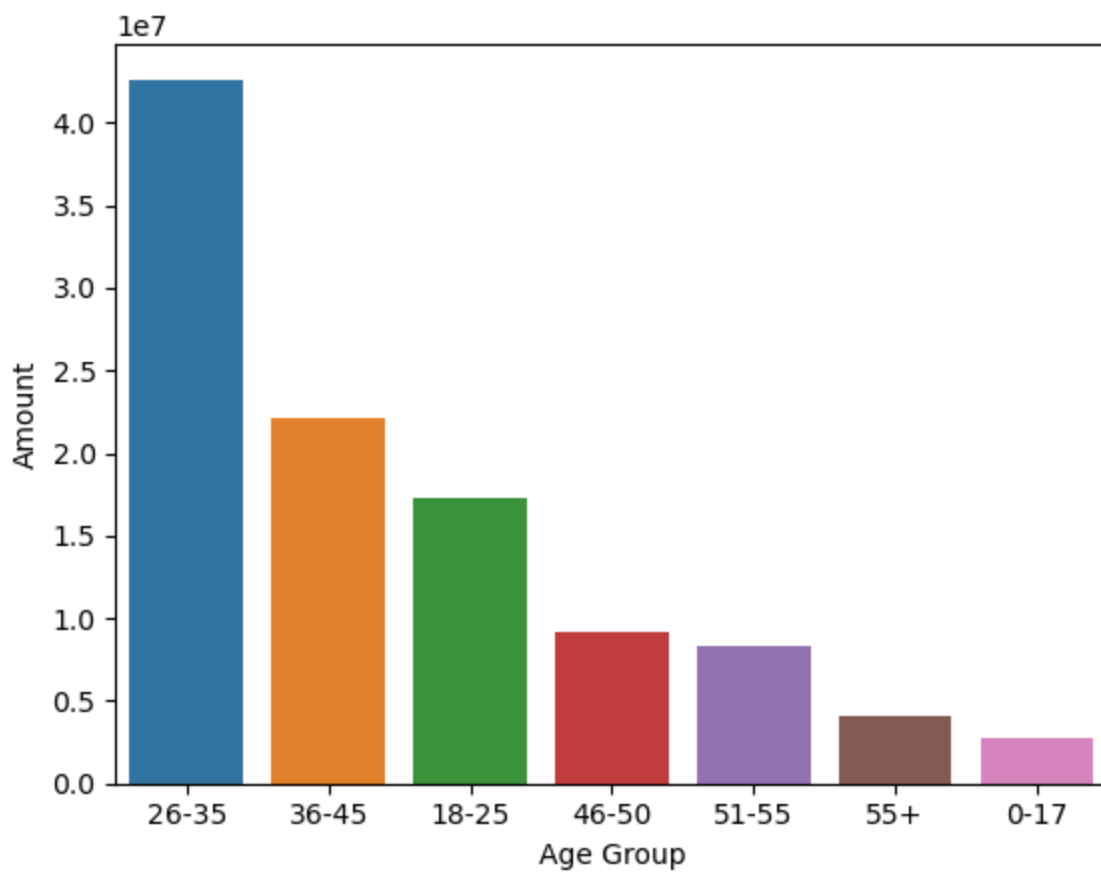


```
In [20]: ax=sns.countplot(data=df,x='Age Group', hue='Gender')
for bars in ax.containers:
    ax.bar_label(bars)
```



```
In [25]: sales_age=df.groupby(['Age Group'], as_index=False)['Amount'].sum().sort_values(by='Amount')
sns.barplot(x='Age Group',y='Amount' , data = sales_age)
```

```
Out[25]: <Axes: xlabel='Age Group', ylabel='Amount'>
```

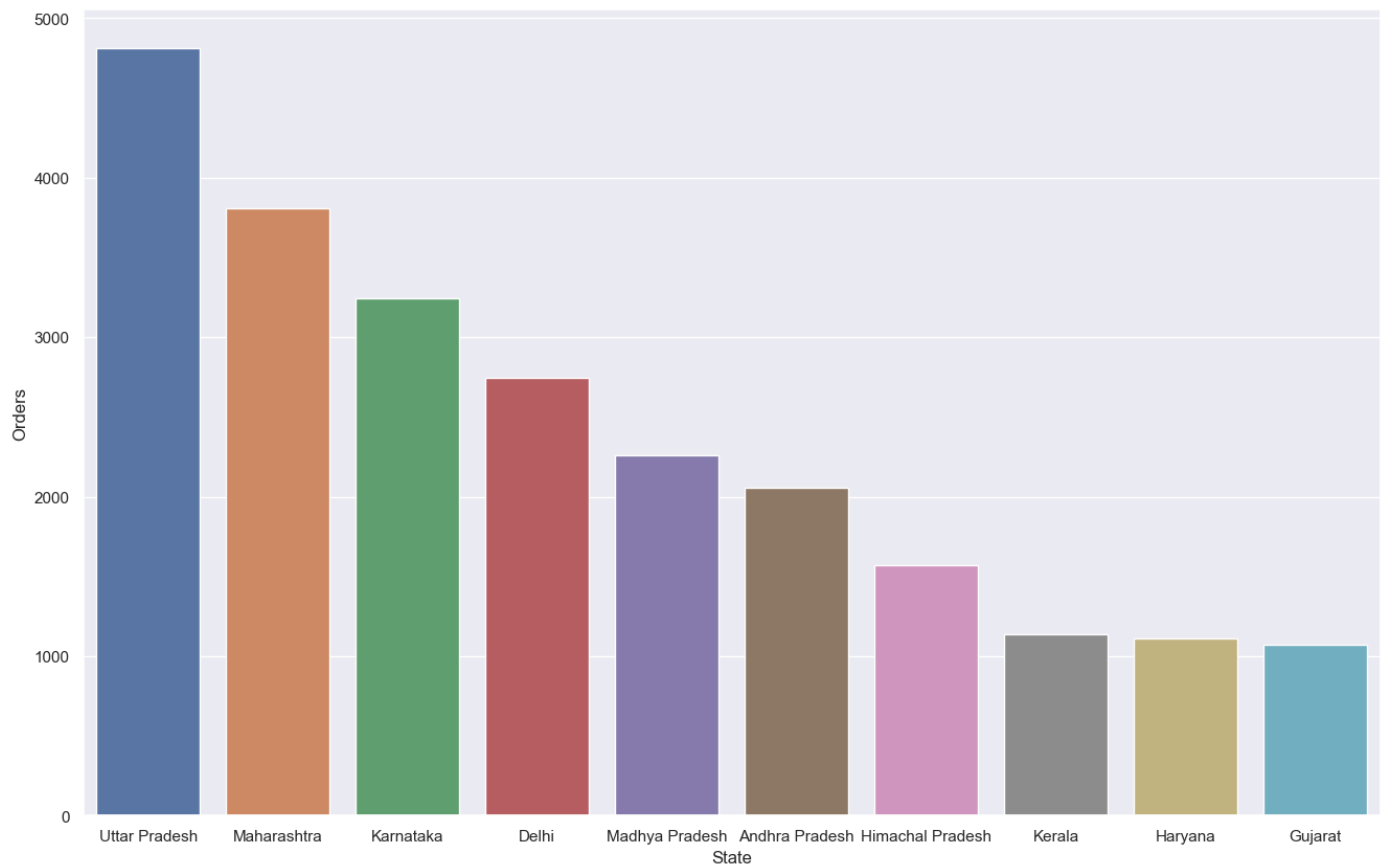


From the above graph we can see that most of the buyers are between the age group 26-35.

STATE

```
In [35]: sale_state=df.groupby(['State'], as_index=False)['Orders'].sum().sort_values(by='Orders')
sns.set(rc={'figure.figsize':(16,10)})
sns.barplot(x='State',y='Orders',data=sale_state)
```

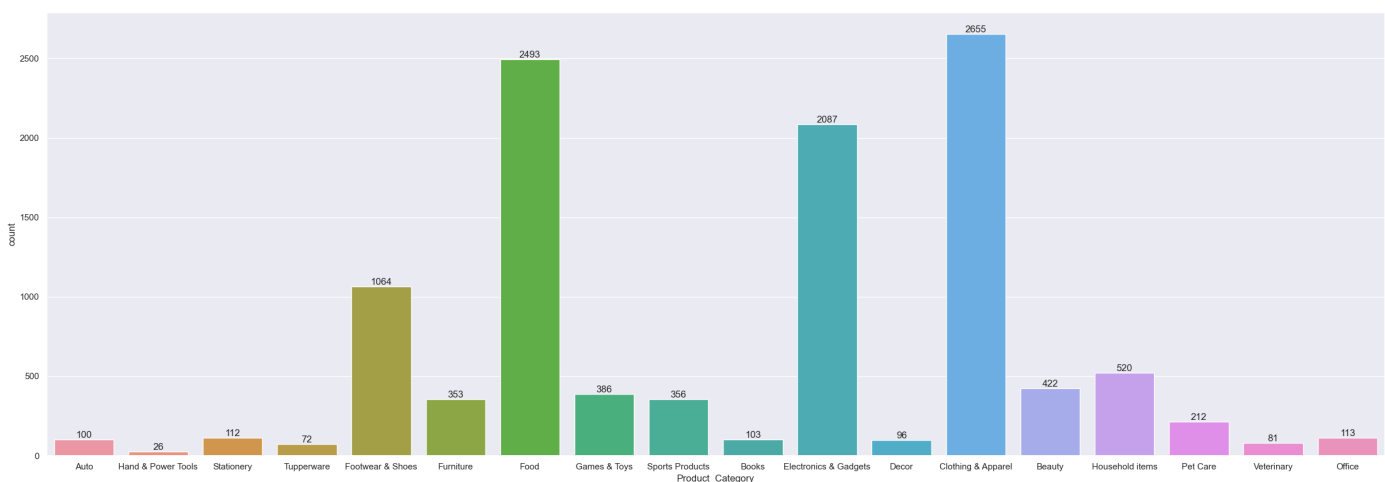
```
Out[35]: <Axes: xlabel='State', ylabel='Orders'>
```



From the above graph we can see that most of the orders are placed from Uttarpradesh.

PRODUCT CATEGORY

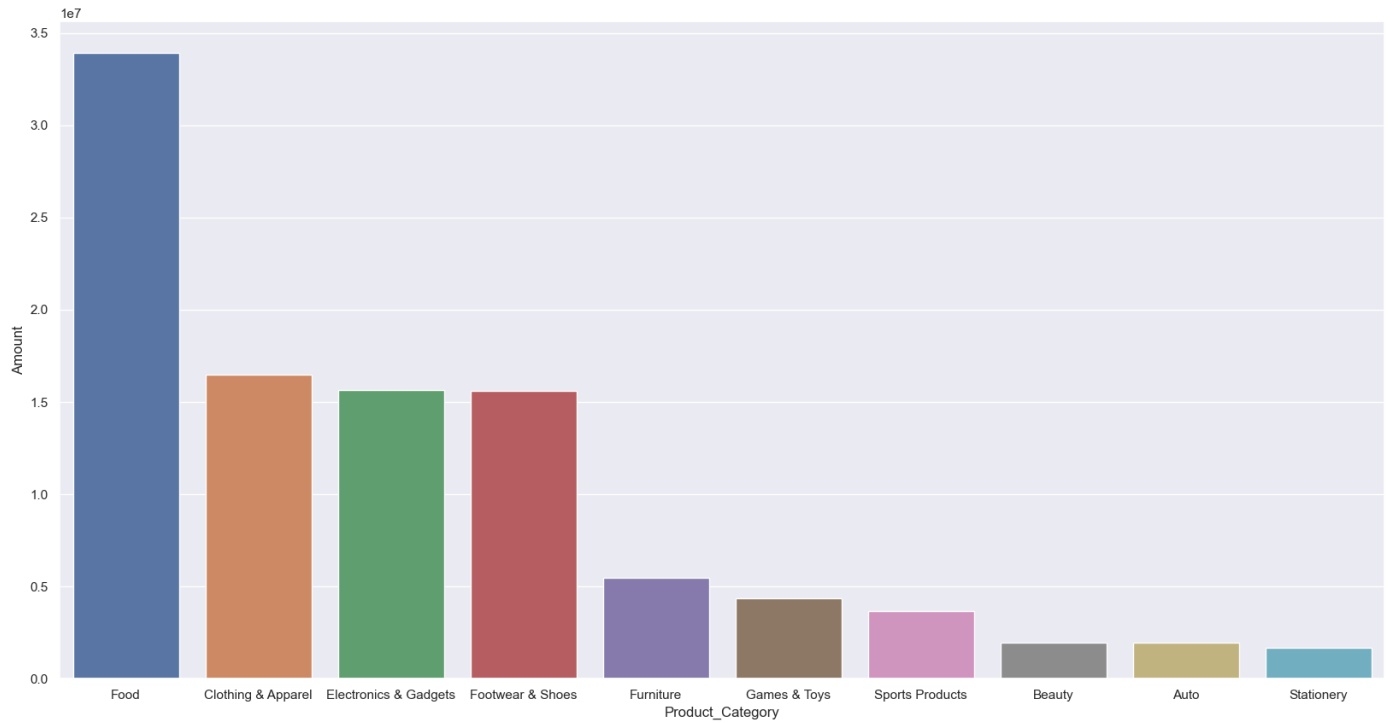
```
In [44]: sns.set(rc={'figure.figsize':(30,10)})
ax=sns.countplot(data=df,x='Product_Category')
for bars in ax.containers:
    ax.bar_label(bars)
```



The above graph shows the category of product with highest number of sales.

```
In [56]: sale_product=df.groupby(['Product_Category'],as_index=False)['Amount'].sum().sort_values
sns.set(rc={'figure.figsize':(20,10)})
sns.barplot(data=sale_product,x='Product_Category',y='Amount')
```

```
Out[56]: <Axes: xlabel='Product_Category', ylabel='Amount'>
```

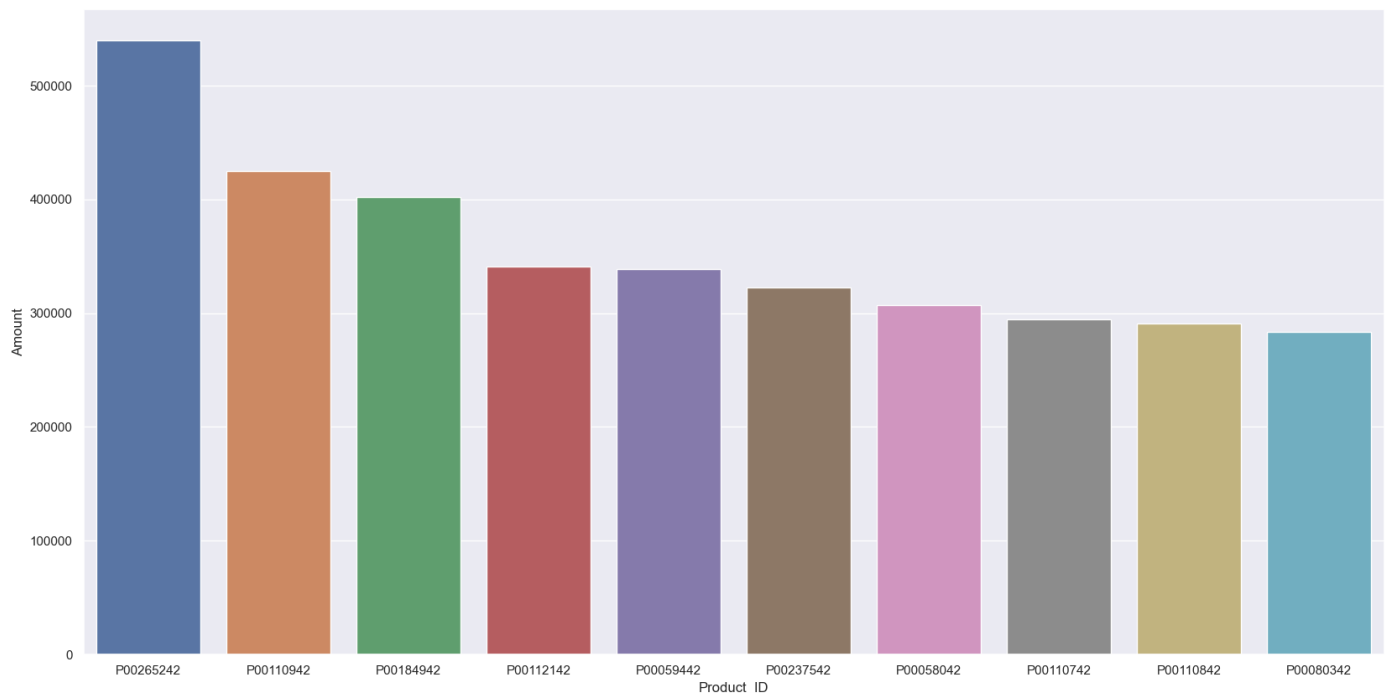


In the above graph we could see that maximum amount is spent on Fooding.

TOP SELLING PRODUCT

```
In [57]: sale_product=df.groupby(['Product_ID'],as_index=False)['Amount'].sum().sort_values(by='Amount',ascending=False)
sns.set(rc={'figure.figsize':(20,10)})
sns.barplot(data=sale_product,x='Product_ID',y='Amount')
```

```
Out[57]: <Axes: xlabel='Product_ID', ylabel='Amount'>
```



Product Id P00265242 is the top selling product.

```
In [ ]:
```