Importing Necessary Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

Importing the Dataset

```
df = pd.read_csv(r'C:\Users\Lenovo\Downloads\
Python_Diwali_Sales_Analysis-main\Python_Diwali_Sales_Analysis-main\
Diwali Sales Data.csv', encoding= 'unicode_escape')
```

Basic Description of the Dataset

```
df.shape
```

```
(11251, 15)
```

```
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
```

```
df.head()

   User_ID  Cust_name Product_ID Gender Age Group  Age 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


            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


    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
```

Data Cleaning Process

Deleting the Null Values

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

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
```

Identifying the Null Values

```
pd.isnull(df).sum()

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
```

Dropping the Null Values

```
df.dropna(inplace=True)

pd.isnull(df).sum()

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
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 11239 entries, 0 to 11250
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   User_ID           11239 non-null  int64  
 1   Cust_name         11239 non-null  object 
 2   Product_ID        11239 non-null  object 
 3   Gender            11239 non-null  object 
 4   Age Group         11239 non-null  object 
 5   Age               11239 non-null  int64  
 6   Marital_Status    11239 non-null  int64  
 7   State             11239 non-null  object 
 8   Zone              11239 non-null  object 
 9   Occupation        11239 non-null  object 
 10  Product_Category  11239 non-null  object 
 11  Orders            11239 non-null  int64  
 12  Amount            11239 non-null  float64
dtypes: float64(1), int64(4), object(8)
memory usage: 1.2+ MB
```

Changing the Data Type for the Numerical Analysis

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

```
df['Amount'].dtype
```

```
dtype('int32')
```

```
df.rename (columns={'Marital_Status':'Nikah'},inplace=True)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 11239 entries, 0 to 11250
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   User_ID           11239 non-null  int64  
 1   Cust_name         11239 non-null  object 
 2   Product_ID        11239 non-null  object 
 3   Gender            11239 non-null  object 
 4   Age Group         11239 non-null  object 
 5   Age               11239 non-null  int64  
 6   Nikah             11239 non-null  int64
 7   State             11239 non-null  object
 8   Zone              11239 non-null  object
 9   Occupation        11239 non-null  object
 10  Product_Category  11239 non-null  object
 11  Orders            11239 non-null  int64
 12  Amount            11239 non-null  int32
dtypes: int32(1), int64(4), object(8)
memory usage: 1.2+ MB
```

Basic Statistical Data Analysis

```
df.describe()
```

```
            User_ID           Age         Nikah        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
```

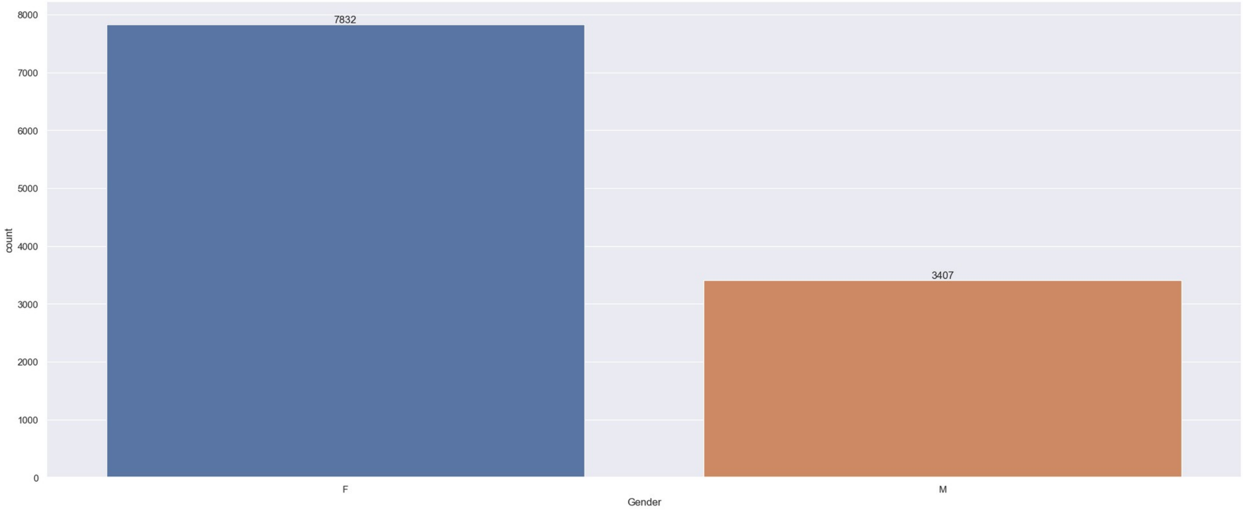
```
df[['Age','Orders','Amount']].describe()
```

```
                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: Exploratory Data Analysis

Ranking Customers based on their Gender

```
ax = sns.countplot(x = 'Gender',data = df, hue='Gender')
sns.set (rc={'figure.figsize':(7,3)})
for bars in ax.containers:
    ax.bar_label(bars)
```



Most of the customers are Female

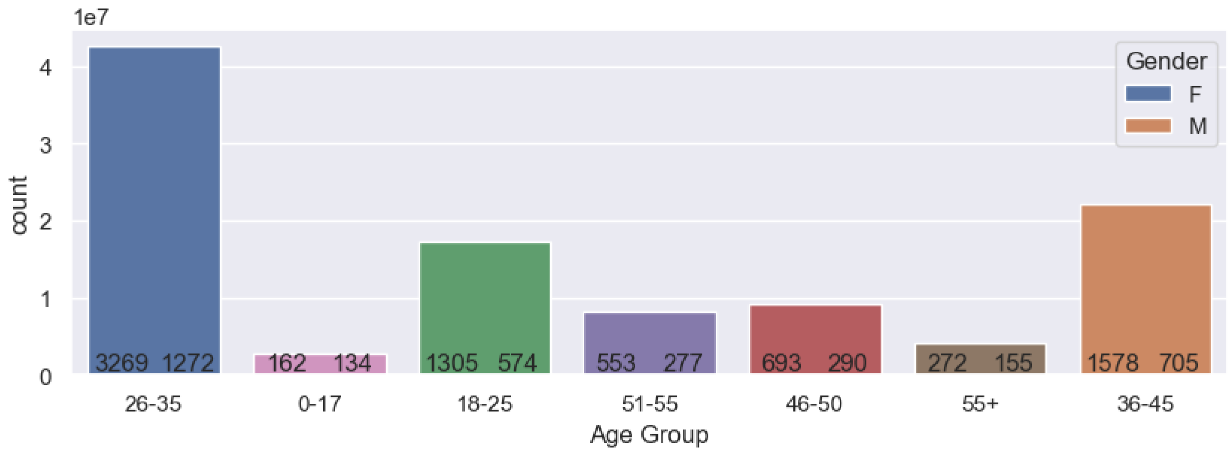
```
df.columns
```

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

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

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

sales_age = df.groupby(['Age Group'], as_index=False)
['Amount'].sum().sort_values(by='Amount', ascending=False)
sns.barplot(x = 'Age Group',y= 'Amount', hue ='Age Group' ,data =
sales_age)
sns.set(rc={'figure.figsize':(12,3)})
```

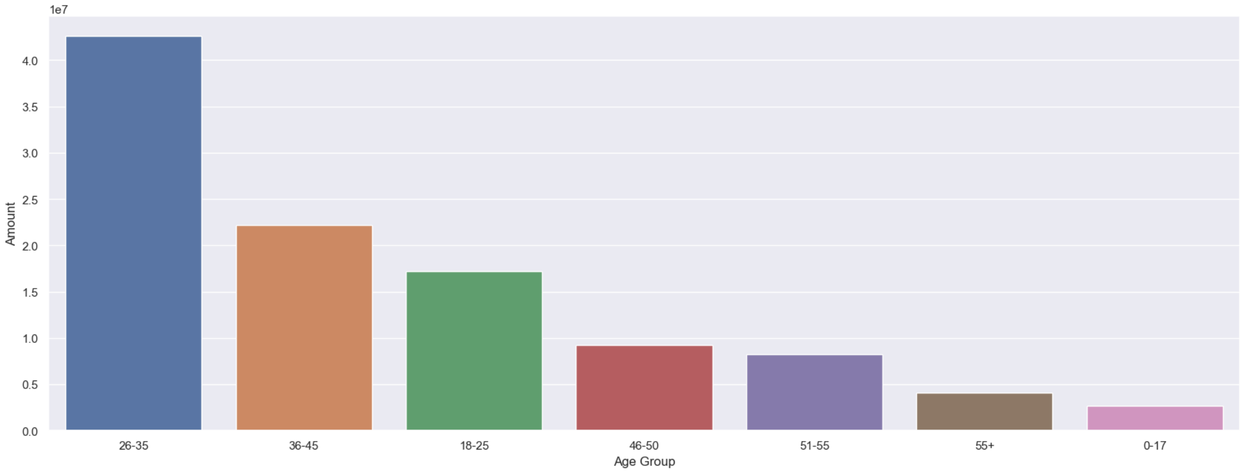


Ranking Customers based on their age group

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

sns.barplot(x = 'Age Group',y= 'Amount', hue= 'Age Group',data =
sales_By_age)

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

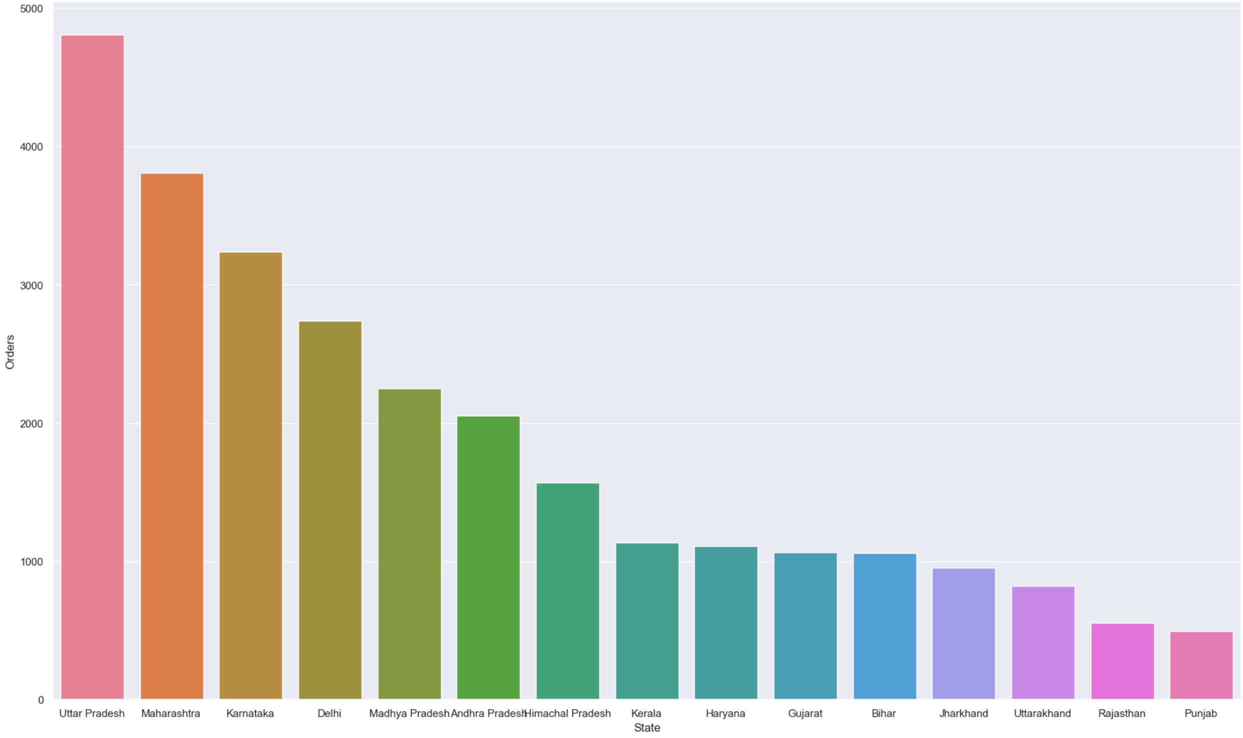


Most of the customers are from age group 26-35. On the other hand age group 0-17 hase the least number of Customers

```
Sales_order_By_State = df.groupby(['State'], as_index=False)
['Orders'].sum().sort_values(
    by='Orders', ascending=False).head(15)

sns.set(rc={'figure.figsize':(22,13)})
sns.barplot(data = Sales_order_By_State, x = 'State',y= 'Orders',
hue='State')
```

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

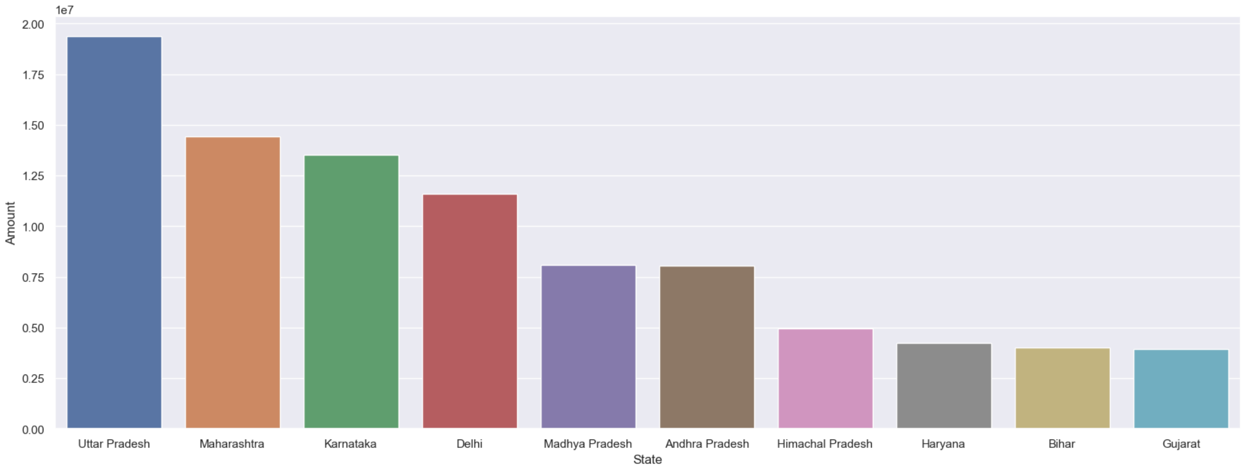


Customers ranking by States

```
Sales_Amount_By_State = df.groupby(['State'], as_index=False)
['Amount'].sum().sort_values(
    by='Amount', ascending=False).head(10)

sns.set(rc={'figure.figsize':(20,7)})
sns.barplot(data = Sales_Amount_By_State, x = 'State',y=
'Amount',hue='Nikah')
```

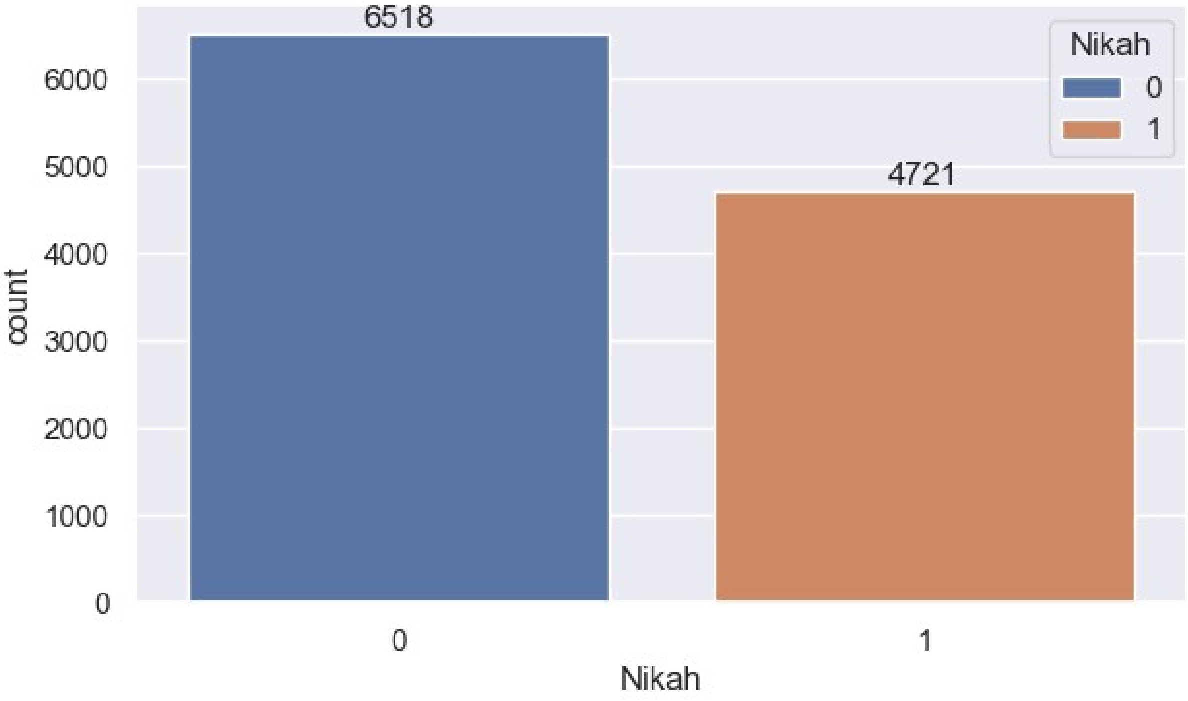
```
<Axes: xlabel='State', ylabel='Amount'>
```



Most of the customers are from Uttar Pradesh and Gujrat with the least number of cutomers

Customers Marital Status

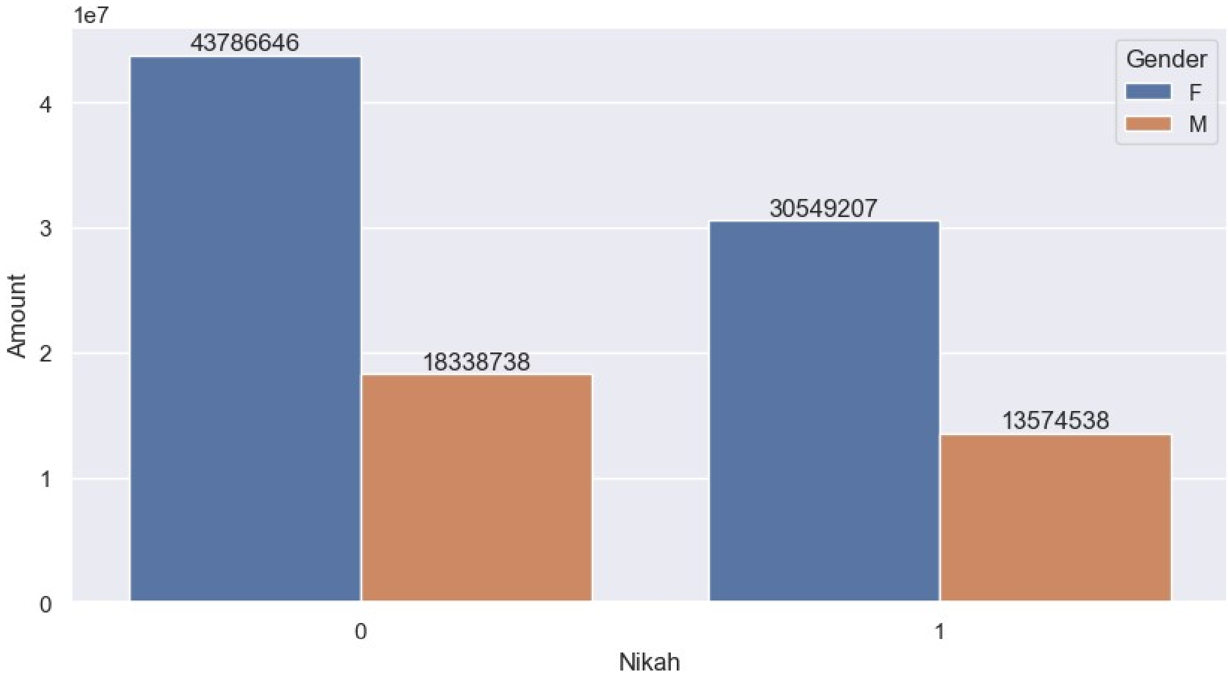
```
ax= sns.countplot (data = df, x= 'Nikah', hue='Nikah')
sns.set (rc={'figure.figsize':(7,3)})
for bars in ax.containers:
    ax.bar_label(bars)
```



Most Customers are Unmarrid or single

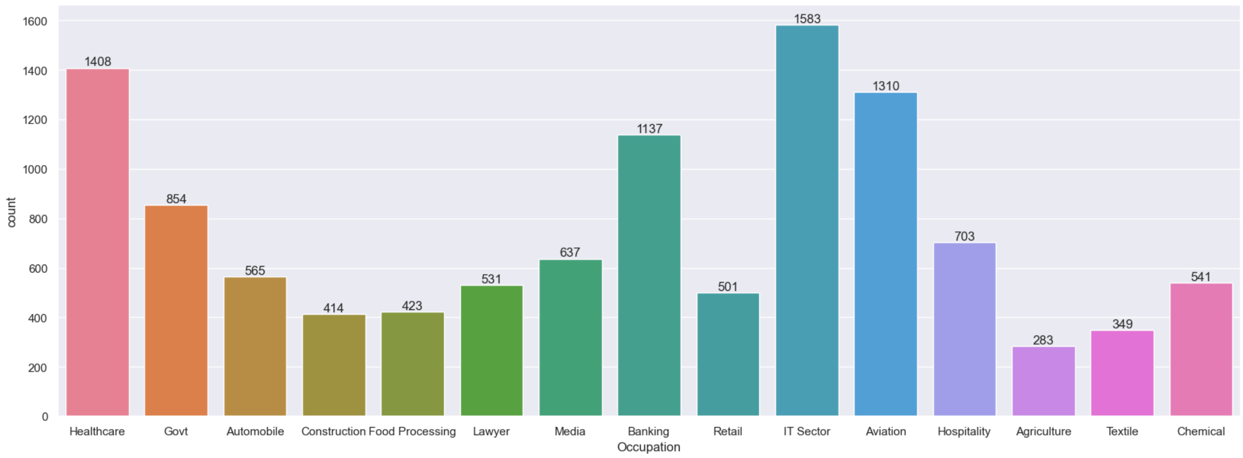
Ranking amount of sold product based on Gender and Marital status

```
Sales_amount_by_Marital_Status_and_Gender =
(df.groupby(['Nikah','Gender'], as_index=False)
['Amount'] .sum() .sort_values(by='Amount', ascending=False))
sns.set(rc={'figure.figsize': (10,5)})
ax = sns.barplot(data=Sales_amount_by_Marital_Status_and_Gender,
x='Nikah', y='Amount', hue='Gender')
for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f')
```



Single females are the ones with most purchases based on amount of sale

```
sns.set(rc=({'figure.figsize': (20,7)}))
ax=sns.countplot(data=df, x='Occupation', hue= 'Occupation')
for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f')
```

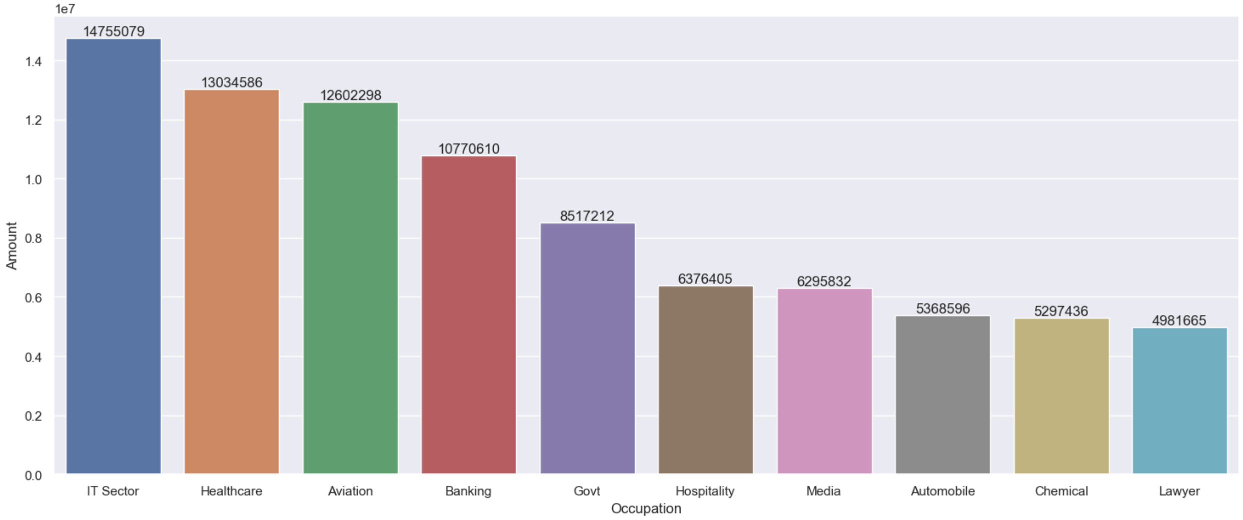


Ranking Sales based on customer Occupation

```
Sale_Amount_By_Occupation = (df.groupby(['Occupation'],
as_index=False)['Amount'] .sum() .sort_values(by='Amount',
ascending=False) .head(10))

sns.set(rc={'figure.figsize': (18,7)})
ax = sns.barplot( data=Sale_Amount_By_Occupation, x='Occupation',
y='Amount', hue='Occupation')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f')
```

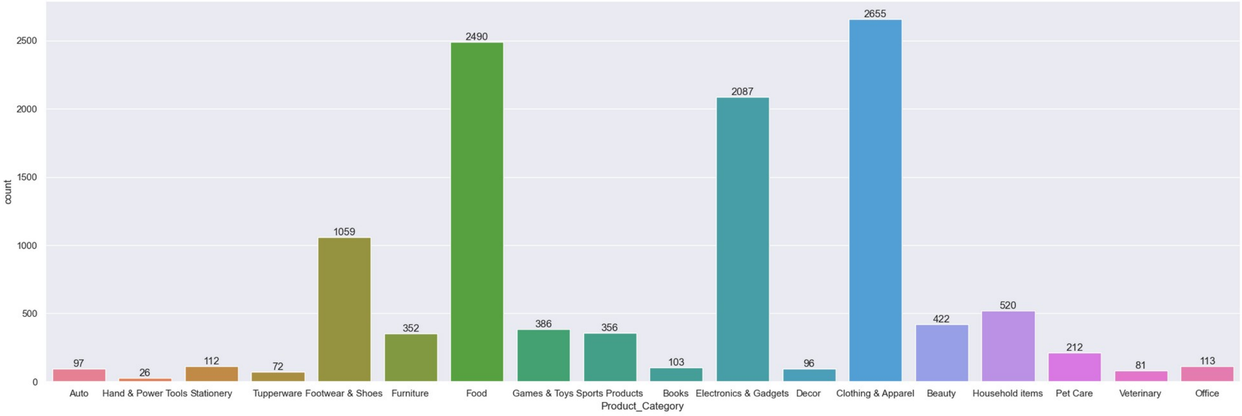


People from IT sector profession has the most purchase whereas, Lawyers are the profession with the lowest purchase

Product category that has highest products

```
sns.set(rc={'figure.figsize':(25,8)})
ax = sns.countplot(data = df, x = 'Product_Category', hue=
'Product_Category')

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



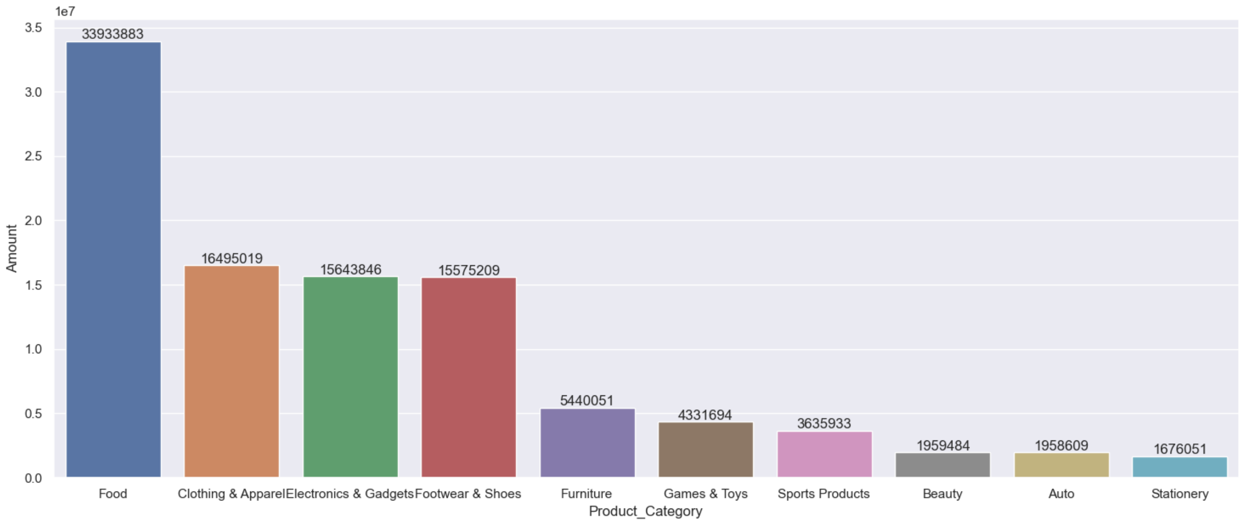
Clothing and Appearel is the most sold product Category

Sale amount based on Product Category

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

sns.set(rc={'figure.figsize': (18,7)})
ax = sns.barplot( data=Sale_Amount_By_Product_Catagory,
x='Product_Category',  y='Amount', hue='Product_Category')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f')
```



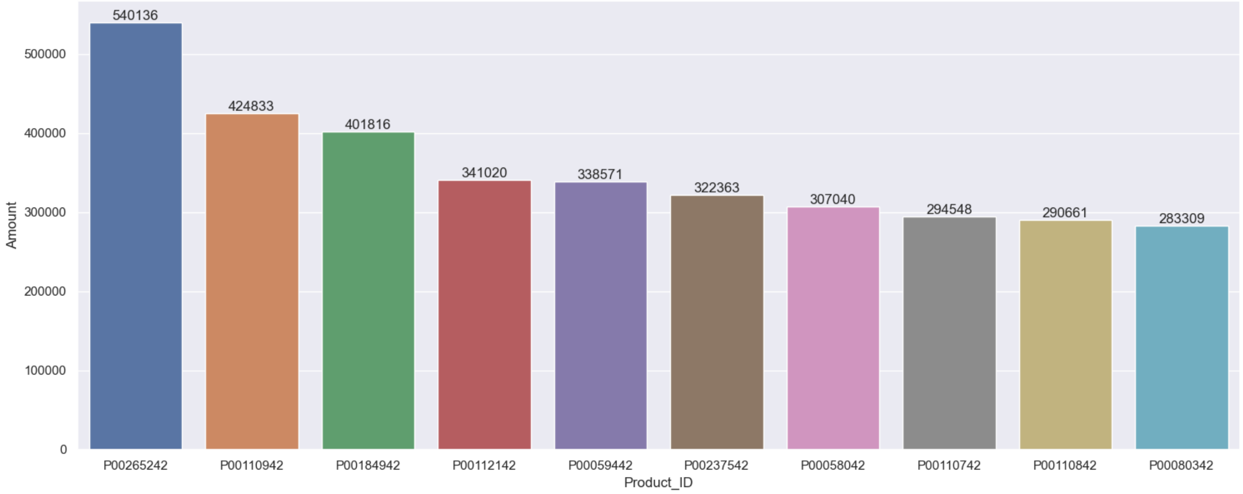
Food items are the most sold Product Category and Stationary items are the least sold Product Catagory

Sale amount by ProductID

```
Sale_Amount_By_ProductID = (df.groupby(['Product_ID'], as_index=False)
['Amount'] .sum() .sort_values(by='Amount',
ascending=False) .head(10))

sns.set(rc={'figure.figsize': (18,7)})
ax = sns.barplot( data=Sale_Amount_By_ProductID, x='Product_ID',
y='Amount', hue='Product_ID')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f')
```



P00265242 is the highest sold ProductID product

Ranking to 10 products based on amoun of Sales:

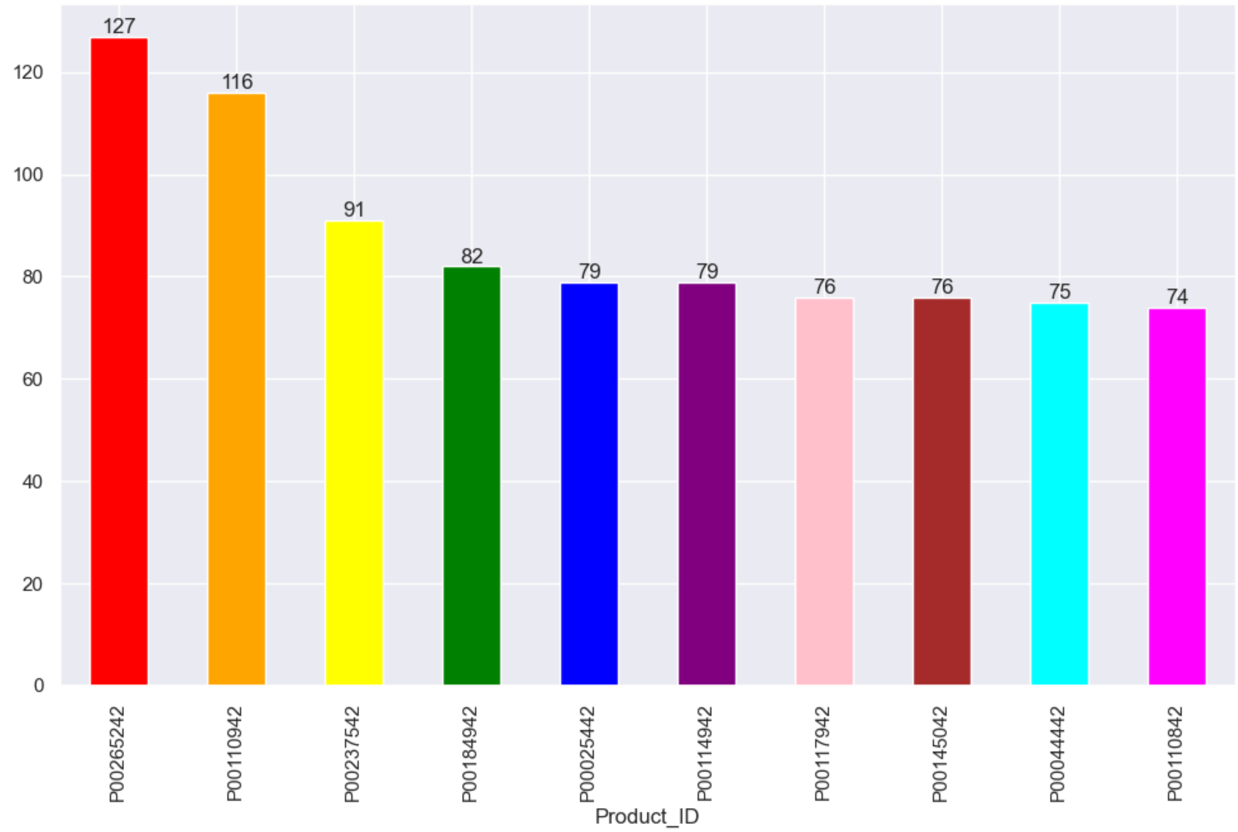
```
fig1, ax1 = plt.subplots(figsize=(12,7))

data = df.groupby('Product_ID')
['Orders'].sum().nlargest(10).sort_values(ascending=False)

data.plot(kind='bar', ax=ax1,

color=['red','orange','yellow','green','blue','purple','pink','brown',
'cyan','magenta'])

for bar in ax1.patches:
    ax1.annotate(
        f'{bar.get_height():.0f}',
        (bar.get_x() + bar.get_width()/2, bar.get_height()),
        ha='center', va='bottom'
    )
```



P00265242 is the highest sold product based on amount sold and P00110842 is the lowst sold product based on amount sold

Conlusion:

Women in the 26–35 age group who are married and reside in Uttar Pradesh, Maharashtra, and Karnataka, particularly those employed in the IT, healthcare, and aviation sectors, demonstrate a higher likelihood of purchasing products from the food, clothing, and electronics categories.