

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

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

	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						
...
...						
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]

df.shape

(11251, 15)

df.head(10)

	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
5	1000588	Joni	P00057942	M	26-35	28		1
6	1001132	Balk	P00018042	F	18-25	25		1
7	1002092	Shivangi	P00273442	F	55+	61		0
8	1003224	Kushal	P00205642	M	26-35	35		0
9	1003650	Ginny	P00031142	F	26-35	26		1

Orders \	State	Zone	Occupation	Product_Category
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				
5	Himachal Pradesh	Northern	Food Processing	Auto
1				
6	Uttar Pradesh	Central	Lawyer	Auto
4				
7	Maharashtra	Western	IT Sector	Auto
1				
8	Uttar Pradesh	Central	Govt	Auto
2				
9	Andhra Pradesh	Southern	Media	Auto
4				

	Amount	Status	unnamed1
0	23952.00	NaN	NaN
1	23934.00	NaN	NaN
2	23924.00	NaN	NaN
3	23912.00	NaN	NaN
4	23877.00	NaN	NaN
5	23877.00	NaN	NaN
6	23841.00	NaN	NaN
7	NaN	NaN	NaN
8	23809.00	NaN	NaN
9	23799.99	NaN	NaN

```
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

```
dtypes: float64(3), int64(4), object(8)
memory usage: 1.3+ MB
```

```
df.info()
```

#	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

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

	User_ID	Cust_name	Product_ID	Gender	Age	Group	Age \
0	False	False	False	False		False	False
1	False	False	False	False		False	False
2	False	False	False	False		False	False
3	False	False	False	False		False	False
4	False	False	False	False		False	False
...
11246	False	False	False	False		False	False
11247	False	False	False	False		False	False
11248	False	False	False	False		False	False
11249	False	False	False	False		False	False

11250	False	False	False	False	False	False
	Marital_Status	State	Zone	Occupation	Product_Category	
Orders \						
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...
11246	False	False	False	False	False	False
11247	False	False	False	False	False	False
11248	False	False	False	False	False	False
11249	False	False	False	False	False	False
11250	False	False	False	False	False	False

	Amount
0	False
1	False
2	False
3	False
4	False
...	...
11246	False
11247	False
11248	False
11249	False
11250	False

[11251 rows x 13 columns]

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

```

```
df.dropna(inplace=True)
```

```
df.shape
```

```
(11239, 13)
```

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

```
df['Amount'].dtypes
```

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

```
df['Marital_Status'] = df['Marital_Status'].map({0 : 'Married', 1 : 'Unmarried'})
```

```
df.head(5)
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status
0	1002903	Sanskriti	P00125942	F	26-35	28	Married
1	1000732	Kartik	P00110942	F	26-35	35	Unmarried
2	1001990	Bindu	P00118542	F	26-35	35	Unmarried
3	1001425	Sudevi	P00237842	M	0-17	16	Married

4	1000588	Joni	P00057942	M	26-35	28	Unmarried
---	---------	------	-----------	---	-------	----	-----------

	State	Zone	Occupation	Product_Category	Orders	Amount
0	Maharashtra	Western	Healthcare	Auto	1	23952
1	Andhra Pradesh	Southern	Govt	Auto	3	23934
2	Uttar Pradesh	Central	Automobile	Auto	3	23924
3	Karnataka	Southern	Construction	Auto	2	23912
4	Gujarat	Western	Food Processing	Auto	2	23877

```
df.columns
```

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

```
df.rename(columns={'Age Group':'Age_Group'},inplace=True)
```

```
df.columns
```

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

```
df.describe()
```

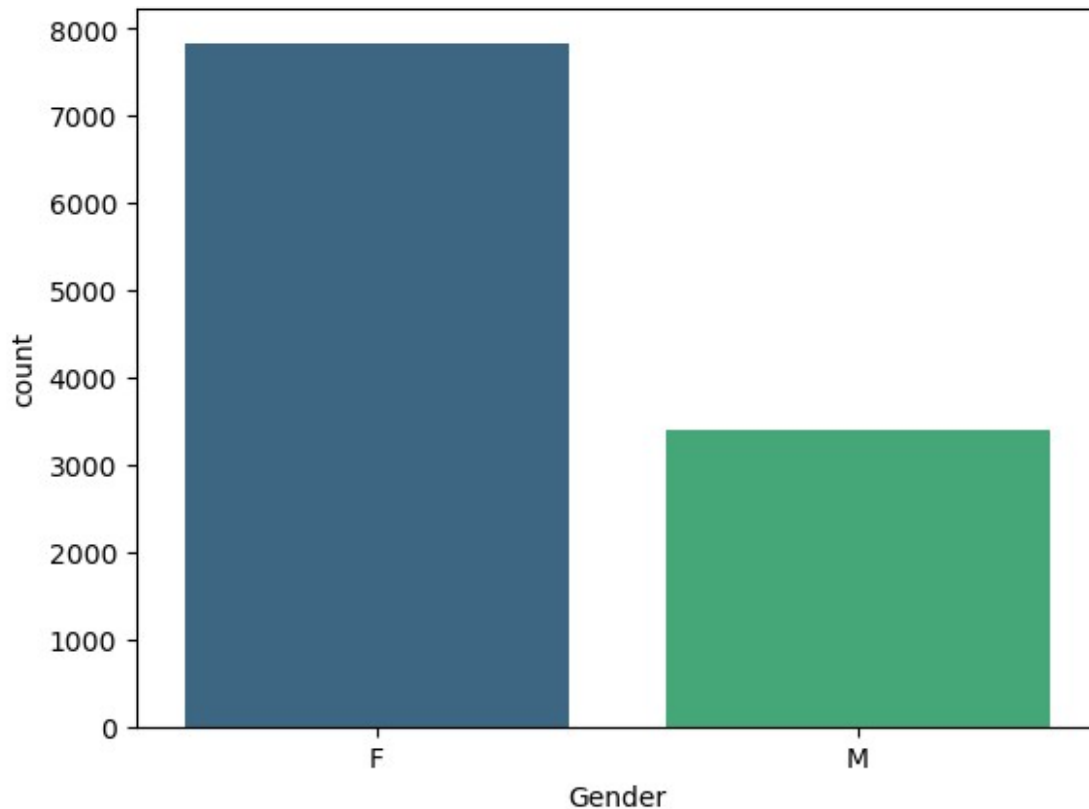
	User_ID	Age	Orders	Amount
count	1.123900e+04	11239.000000	11239.000000	11239.000000
mean	1.003004e+06	35.410357	2.489634	9453.610553
std	1.716039e+03	12.753866	1.114967	5222.355168
min	1.000001e+06	12.000000	1.000000	188.000000
25%	1.001492e+06	27.000000	2.000000	5443.000000
50%	1.003064e+06	33.000000	2.000000	8109.000000
75%	1.004426e+06	43.000000	3.000000	12675.000000
max	1.006040e+06	92.000000	4.000000	23952.000000

```
#Gender
```

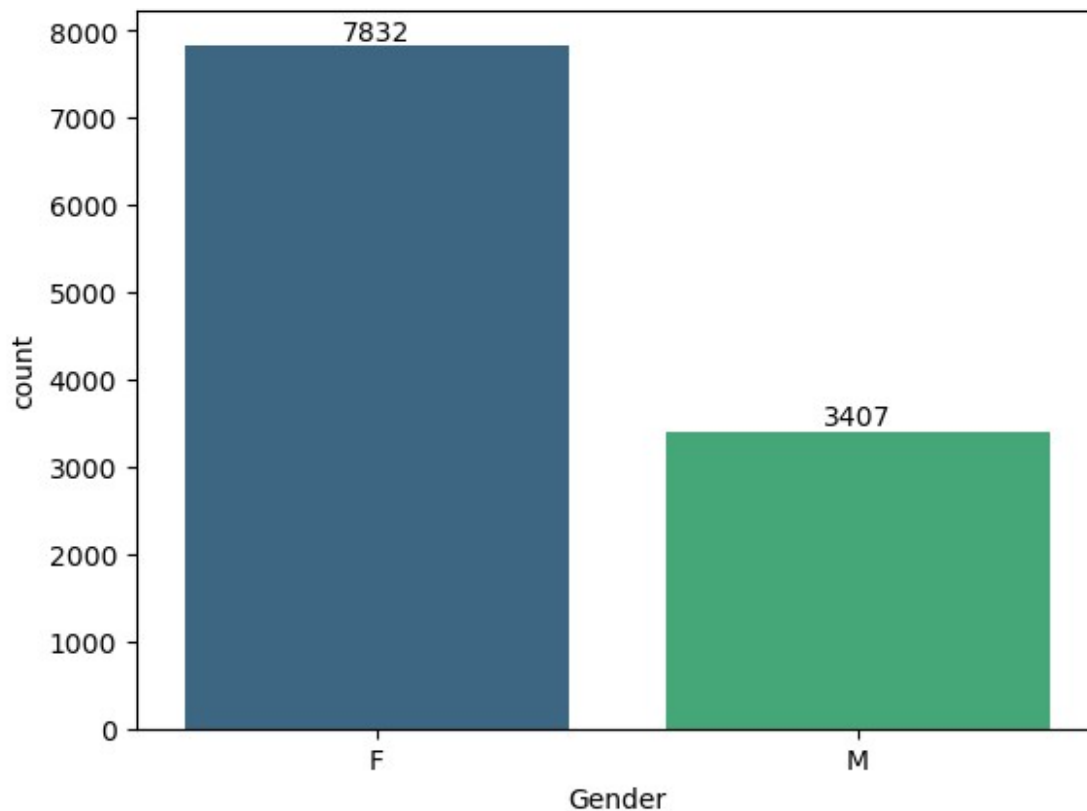
```
df.columns
```

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

```
ax = sns.countplot(x = 'Gender',data = df,hue='Gender', palette =  
'viridis')
```



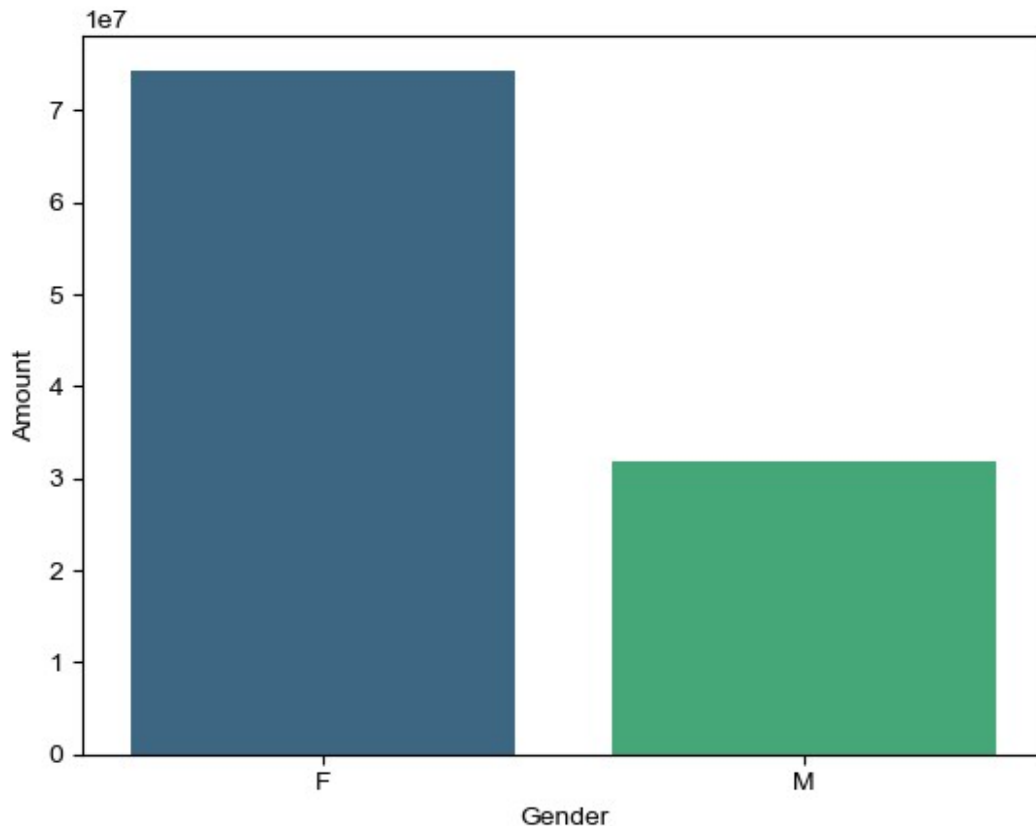
```
ax = sns.countplot(x = 'Gender',data = df, hue='Gender', palette =  
'viridis')  
for bars in ax.containers:  
    ax.bar_label(bars)
```

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

  Gender  Amount
0      F  74335853
1      M  31913276

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',
palette = 'viridis')
sns.set(rc={'figure.figsize':(7,5)})
```



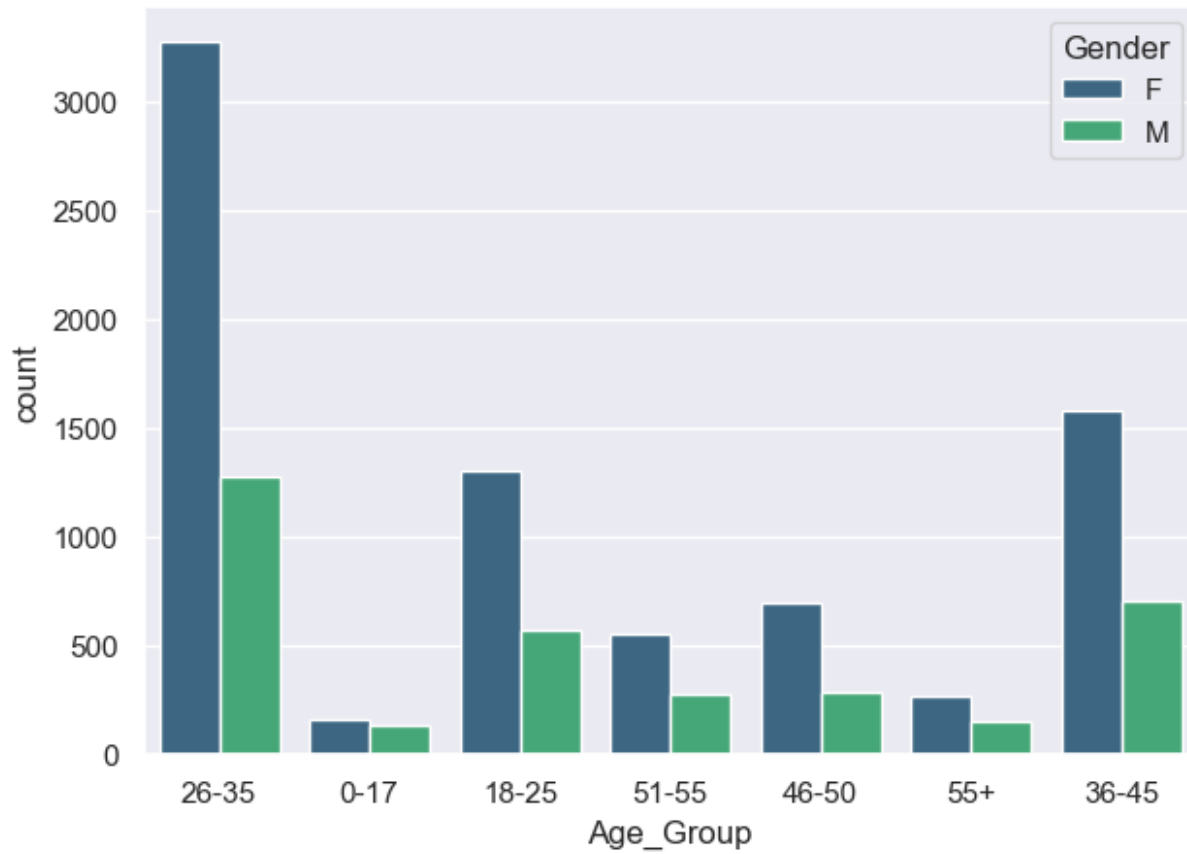
The graph shows that females contribute approximately 70 million, while males contribute around 40 million, indicating higher purchasing behavior among females.

```
# Age
df.columns

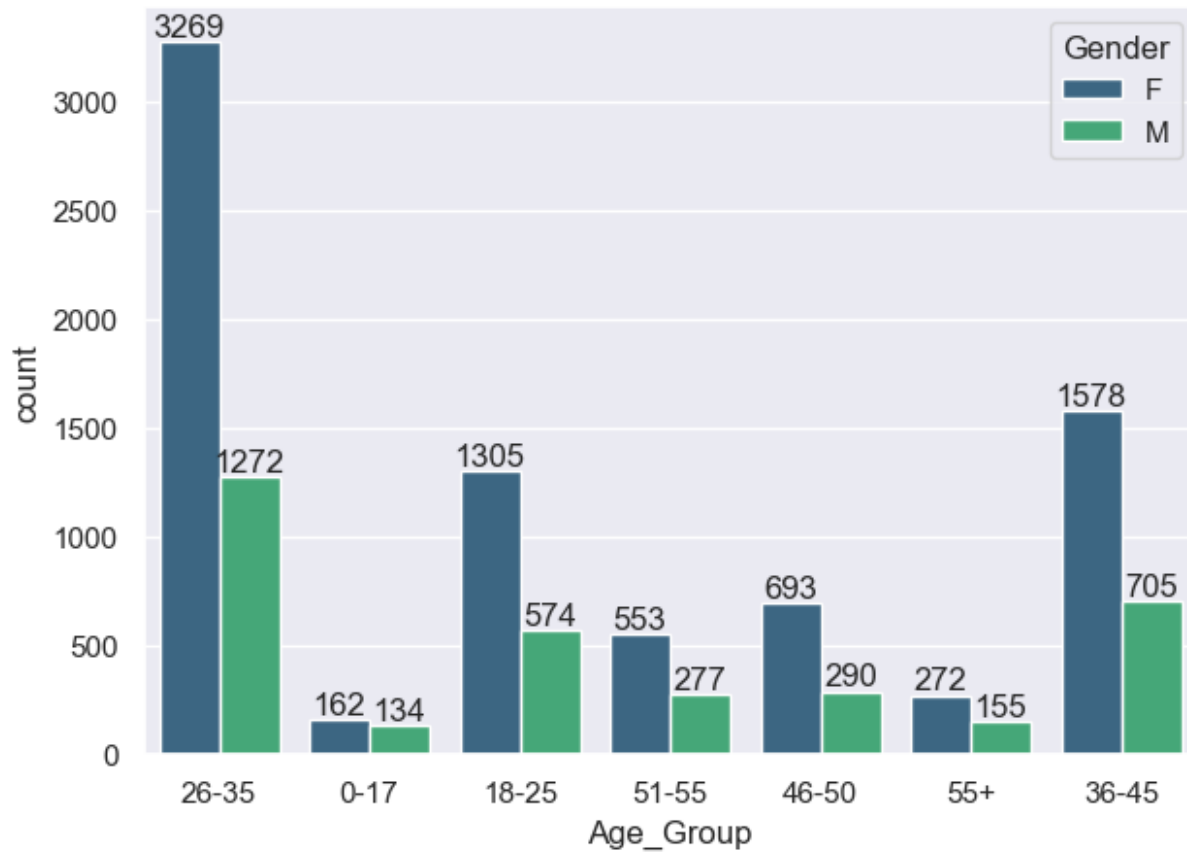
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age_Group',
      'Age',
      'Marital_Status', 'State', 'Zone', 'Occupation',
      'Product_Category',
      'Orders', 'Amount'],
      dtype='object')

sns.countplot(x='Age_Group', data = df , hue = 'Gender', palette =
'viridis')

<Axes: xlabel='Age_Group', ylabel='count'>
```



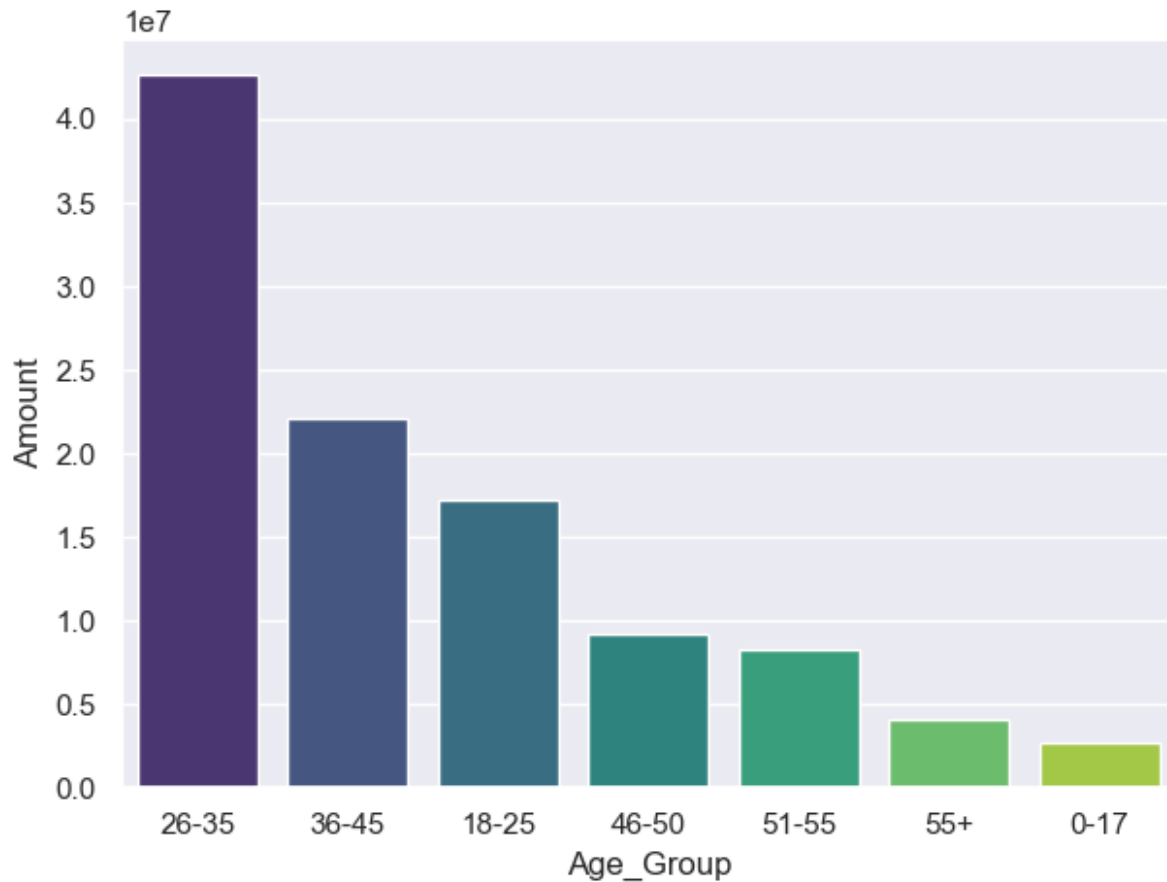
```
ax = sns.countplot(x='Age_Group', data = df , hue = 'Gender', palette =  
'viridis')  
  
for bars in ax.containers:  
    ax.bar_label(bars)
```



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

sns.barplot(x= 'Age_Group',y='Amount',data=sales_age,hue =
'Age_Group',palette = 'viridis' )

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



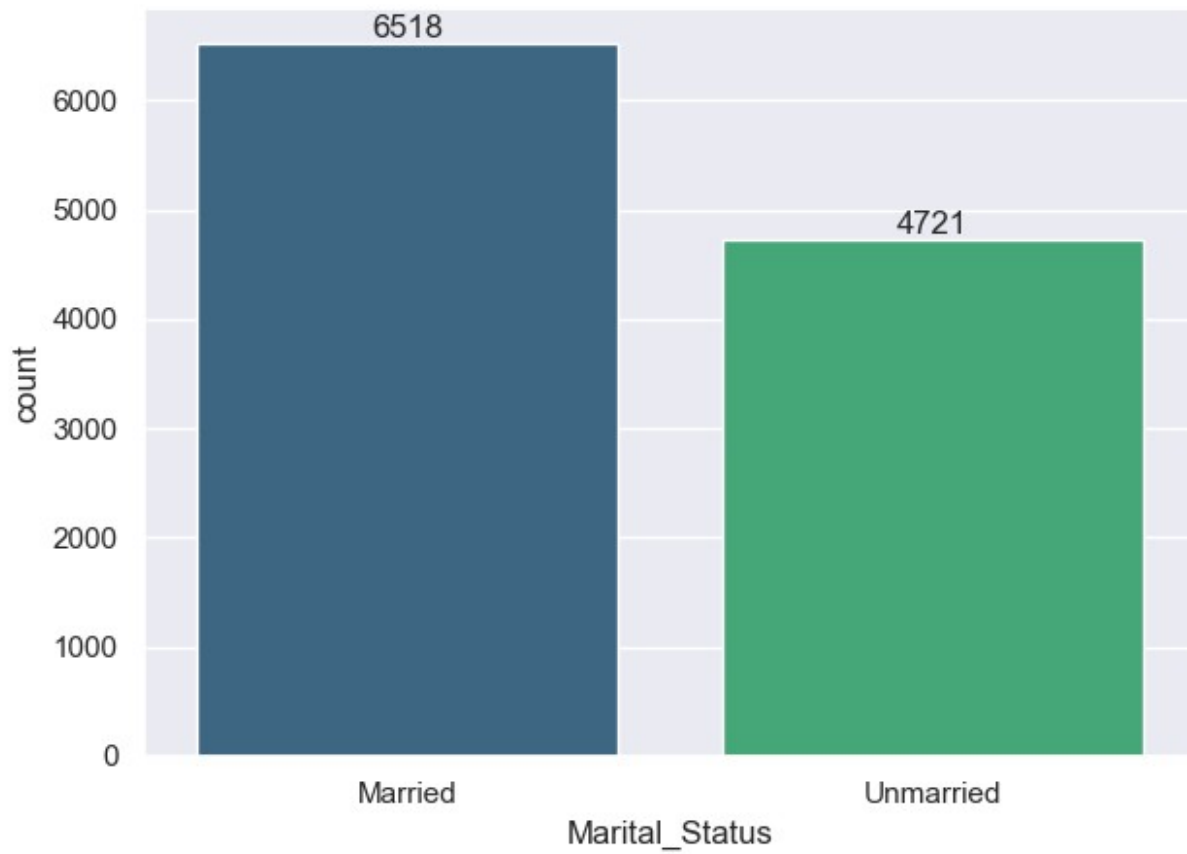
From above graph The highest purchase amount (over 40 million) is from females in the 26-35 age group.

```
# Marital_Status
df.columns

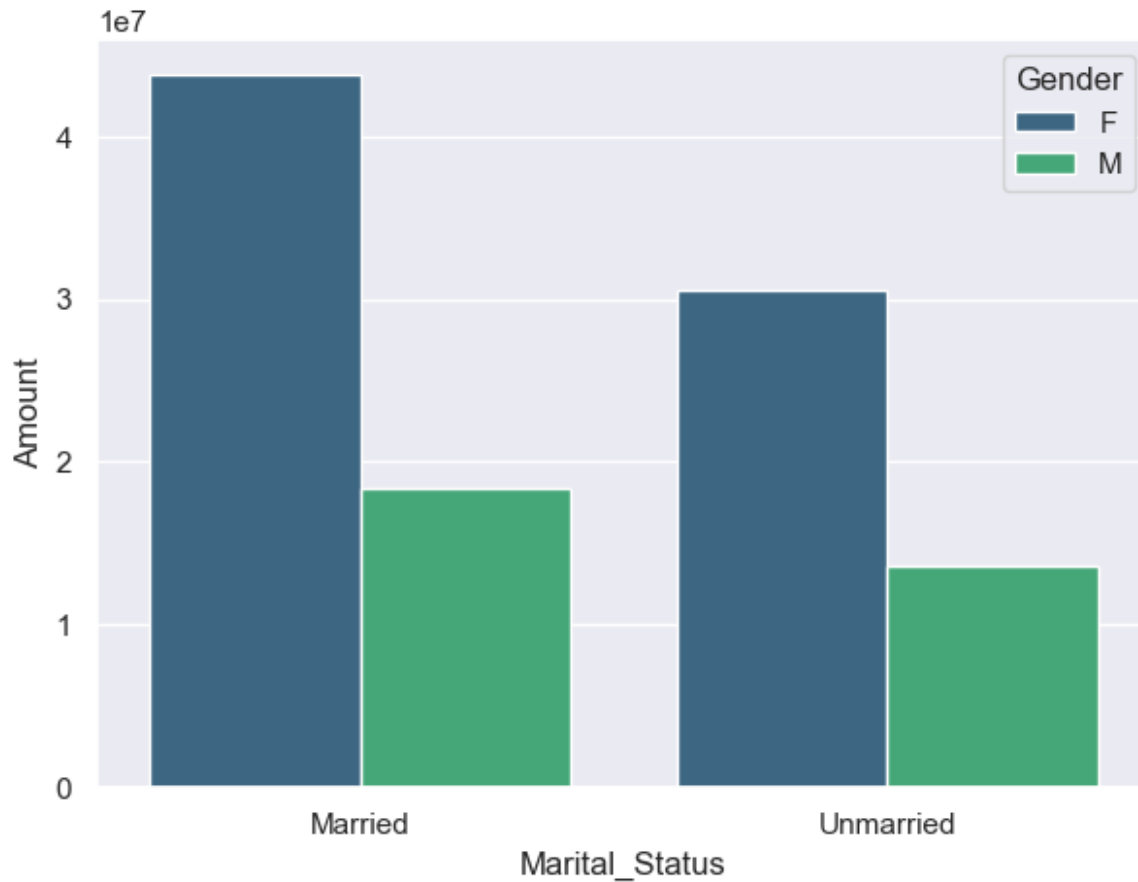
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age_Group',
      'Age',
      'Marital_Status', 'State', 'Zone', 'Occupation',
      'Product_Category',
      'Orders', 'Amount'],
      dtype='object')

ax = sns.countplot(x='Marital_Status', data = df , hue =
"Marital_Status", palette= 'viridis' )

sns.set(rc={'figure.figsize':(17,5)})
for bars in ax.containers:
    ax.bar_label(bars)
```



```
sales_marital_status = df.groupby(['Marital_Status',  
'Gender'],as_index=False)  
['Amount'].sum().sort_values(by='Amount',ascending=False)  
  
sns.set(rc={'figure.figsize':(7,5)})  
sns.barplot(x= 'Marital_Status',y='Amount',data=sales_marital_status,  
hue = 'Gender',palette = 'viridis' )  
<Axes: xlabel='Marital_Status', ylabel='Amount'>
```



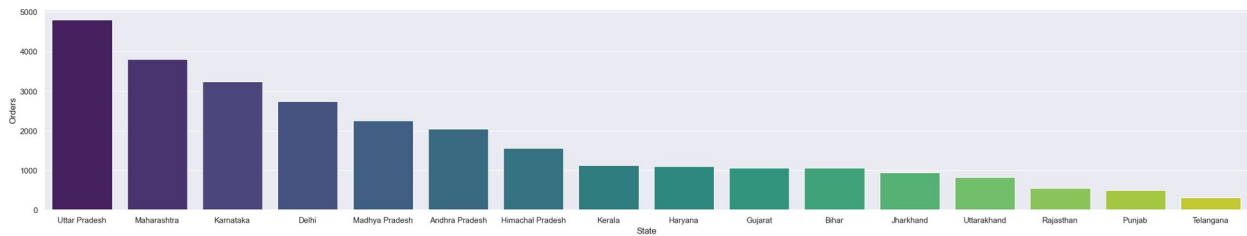
From above graphs we can see that the Married females have the highest purchase amount (over 40 million), while unmarried males have the lowest purchase amount (under 15 million).

```
#State
df.columns

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

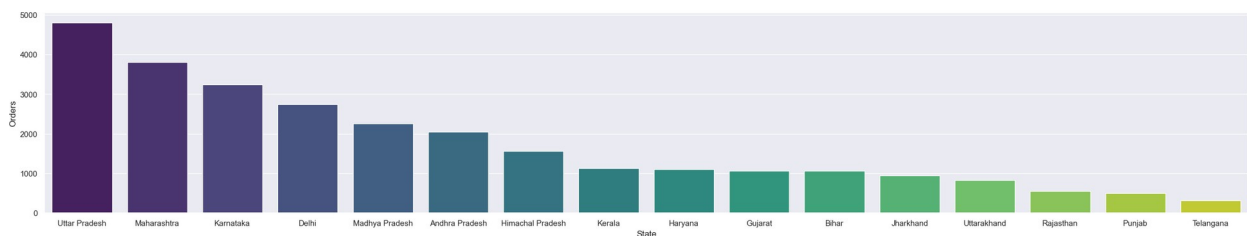
sales_state = df.groupby(['State'],as_index=False)
['Orders'].sum().sort_values(by = 'Orders' , ascending=False)

sns.barplot(x = 'State',y = 'Orders', data = sales_state ,hue =
'State', palette = 'viridis')
sns.set(rc={'figure.figsize':(30,5)})
```



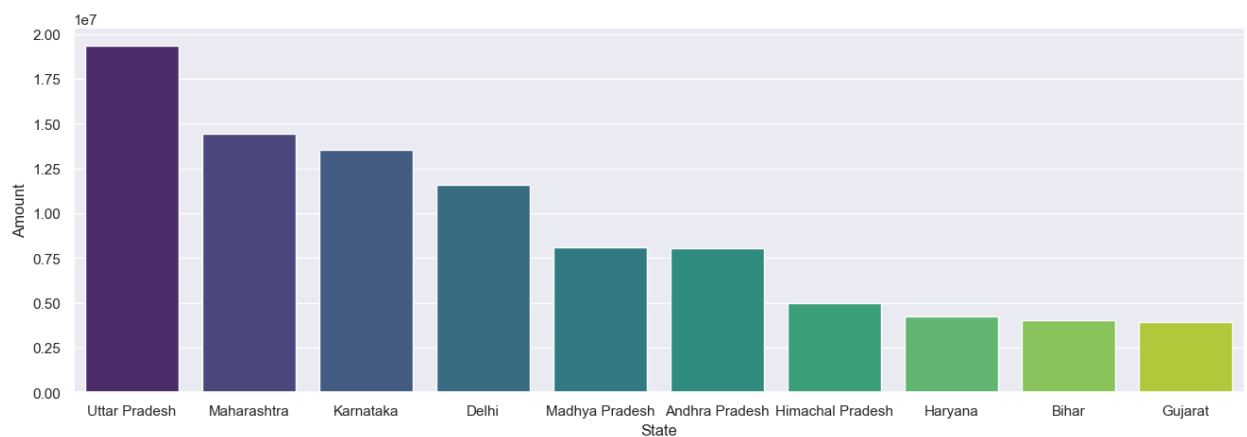
```
sales_state = df.groupby(['State'],as_index=False)
['Orders'].sum().sort_values(by = 'Orders' , ascending=False)

sns.barplot(x = 'State',y = 'Orders', data = sales_state, hue =
'State' , palette = 'viridis')
sns.set(rc={'figure.figsize':(16,5)})
```



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

sns.barplot(x = 'State',y = 'Amount', data = sales_state, hue =
'State' , palette = 'viridis')
sns.set(rc={'figure.figsize':(16,5)})
```



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

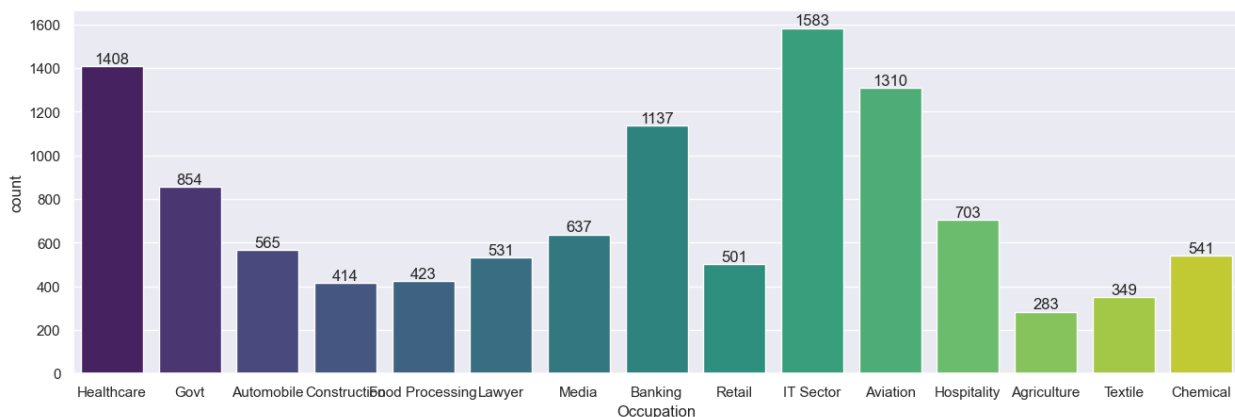
```
# Occupation
df.columns
```



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

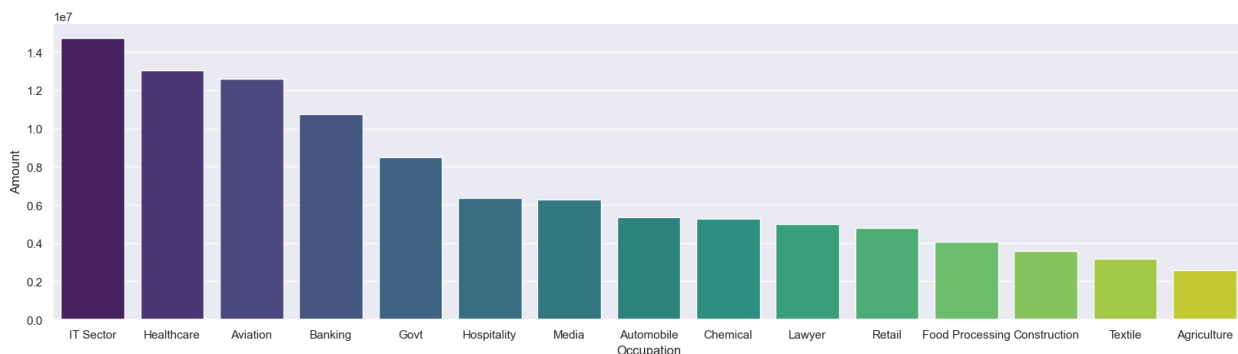
```
ax = sns.countplot(x='Occupation', data = df, hue = 'Occupation' ,
                  palette = 'viridis')
sns.set(rc={'figure.figsize':(20,5)})
```

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



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

sns.barplot(x = 'Occupation', y = 'Amount', data = sales_occupation,
            hue = 'Occupation' , palette = 'viridis')
sns.set(rc={'figure.figsize':(16,5)})
```



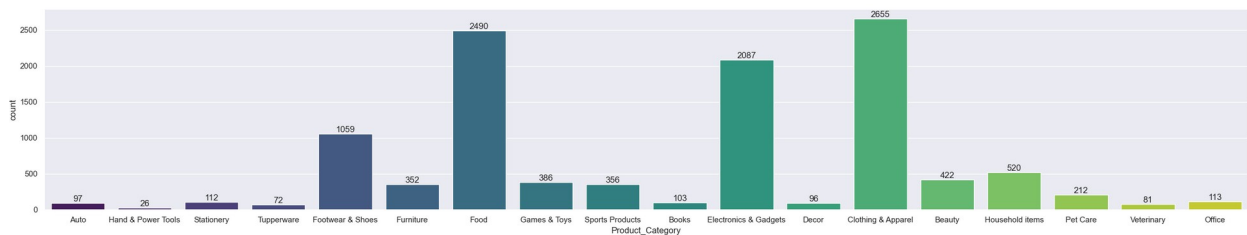
From the graph, we can see that most of the buyers are from the IT, Healthcare, and Aviation sectors, with the IT sector having the highest purchase amount (over 14 million), while lawyers have the lowest purchase amount (under 6 million).

```
# Product Category
df.columns

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

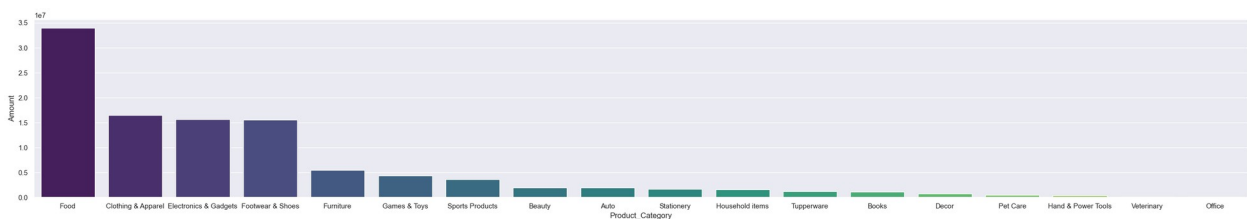
ax = sns.countplot(x='Product_Category', data = df, hue =
'Product_Category' , palette = 'viridis')
sns.set(rc={'figure.figsize':(34,5)})

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



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

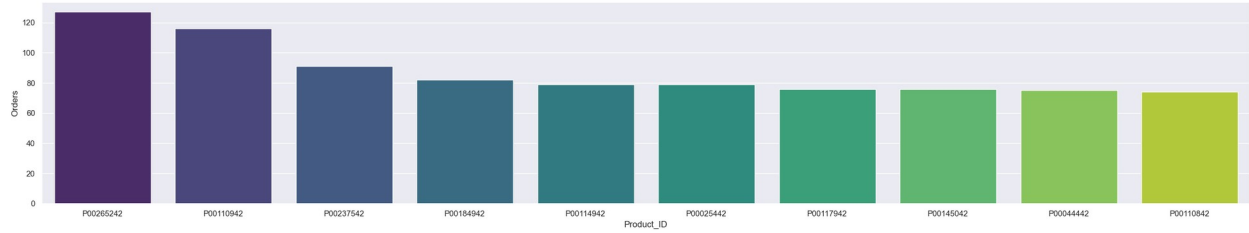
sns.barplot(x = 'Product_Category',y = 'Amount', data = sales_Pro_Cat,
hue = 'Product_Category' , palette = 'viridis')
sns.set(rc={'figure.figsize':(30,5)})
```



From the above graph, we can see that the highest sales amount is in the Food category (over 3 million), followed by Clothing & Apparel, Electronics & Gadgets, and Footwear & Shoes, while categories like Veterinary, Hand & Power Tools, and Pet Care have the lowest sales amounts.

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

sns.barplot(x = 'Product_ID',y = 'Orders', data = sales_Pro_ID, hue =
'Product_ID' , palette = 'viridis')
sns.set(rc={'figure.figsize':(15,5)})
```



```
# Zone
df.columns

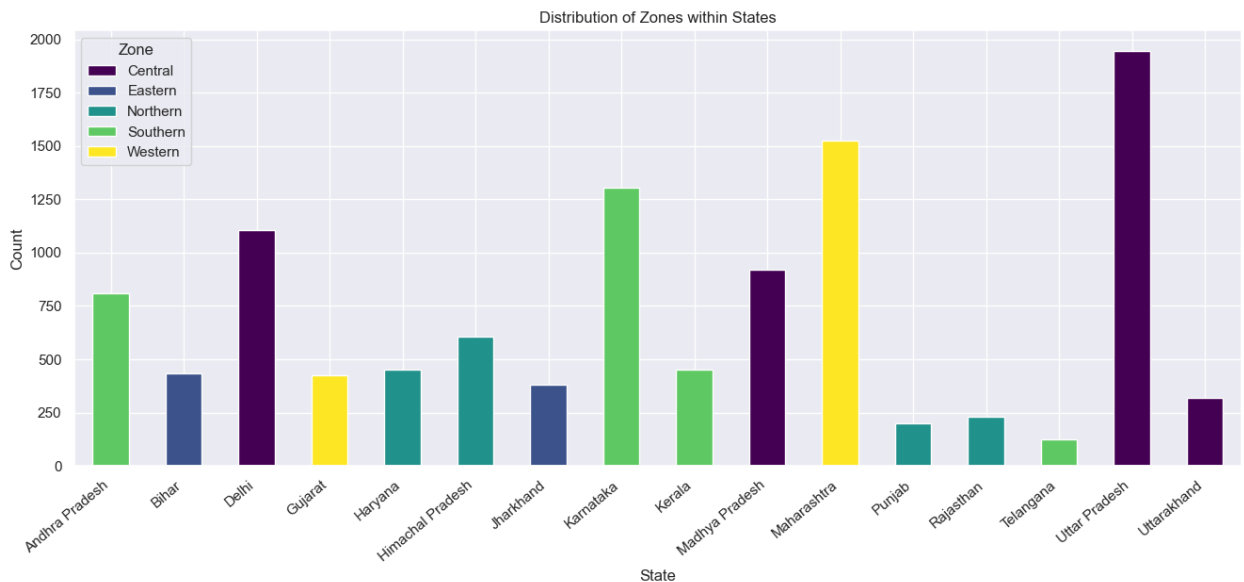
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age_Group',
      'Age',
      'Marital_Status', 'State', 'Zone', 'Occupation',
      'Product_Category',
      'Orders', 'Amount'],
      dtype='object')

zone_state_counts = df.groupby(["State", "Zone"]).size().unstack()

zone_state_counts.plot(kind="bar", stacked=True, figsize=(16, 6),
                      colormap="viridis")

plt.xlabel("State")
plt.ylabel("Count")
plt.title("Distribution of Zones within States")
plt.xticks(rotation=40, ha="right")

plt.legend(title="Zone")
plt.show()
```



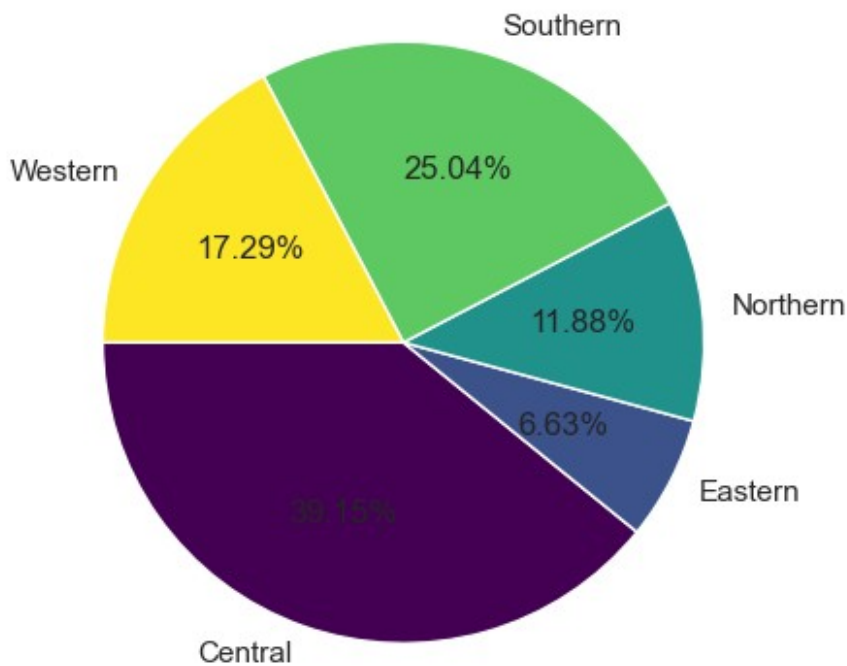
```

zone_sales = df.groupby("Zone")["Amount"].sum()

plt.figure(figsize=(5, 8))
zone_sales.plot.pie(autopct='%1.2f%%', cmap = "viridis",
startangle=180)
plt.ylabel("")

plt.show()

```



The Central zone dominates sales with 39.15%, followed by Southern (25.04%) and Western (17.29%), while Northern (11.88%) and Eastern (6.63%) show lower contributions, indicating regional sales disparity.

Conclusion

Married women age group 26-35 from Uttar Pradesh (Central Zone), Karnataka(Southern Zone) and Maharashtra(Western Zone), working in IT, Healthcare, and Aviation, are the most likely buyers.They prefer products from Food, Clothing, and Electronics categories, with the Central Zone contributing the highest sales, highlighting its strong consumer demand.