

Festive Sales visualization

Python
project

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

```
[ ] # To avoid encoding error,use "latin-1"  
df=pd.read_csv("/content/Diwali Sales Data.csv",encoding='latin-1')  
df
```



	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1	1000732	Kartik	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
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	Office	3	206.0	NaN	NaN
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	Office	3	188.0	NaN	NaN

11251 rows × 15 columns

```
[ ] df.shape
```

```
[ ] (11251, 15)
```

```
[ ] df.head()
```



	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN	NaN

[] df.tail()

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status		State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
11246	1000695	Manning	P00296942	M	18-25	19		1	Maharashtra	Western	Chemical	Office	4	370.0	NaN	NaN
11247	1004089	Reichenbach	P00171342	M	26-35	33		0	Haryana	Northern	Healthcare	Veterinary	3	367.0	NaN	NaN
11248	1001209	Oshin	P00201342	F	36-45	40		0	Madhya Pradesh	Central	Textile	Office	4	213.0	NaN	NaN
11249	1004023	Noonan	P00059442	M	36-45	37		0	Karnataka	Southern	Agriculture	Office	3	206.0	NaN	NaN
11250	1002744	Brumley	P00281742	F	18-25	19		0	Maharashtra	Western	Healthcare	Office	3	188.0	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
 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

[] # Drop unrelated/Blank Columns.
df.drop(["Status","unnamed1"],axis=1,inplace=True)

▶ # Drop unneeded/Blank Columns.

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



	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0
...
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	Chemical	Office	4	370.0
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	Healthcare	Veterinary	3	367.0
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	Textile	Office	4	213.0
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	Office	3	206.0
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	Office	3	188.0

11251 rows × 13 columns

```
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
```

```
# To check Null value  
df.isnull()
```

```
# Count of Null values.  
df.isnull().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.shape  
→ (11251, 13)  
  
▶ # To delete null values in the Rows.  
df.dropna(inplace=True)  
df
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status		State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28		0	Maharashtra	Western	Healthcare	Auto	1	23952.0
1	1000732	Kartik	P00110942	F	26-35	35		1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0
2	1001990	Bindu	P00118542	F	26-35	35	*	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0
3	1001425	Sudevi	P00237842	M	0-17	16		0	Karnataka	Southern	Construction	Auto	2	23912.0
4	1000588	Joni	P00057942	M	26-35	28		1	Gujarat	Western	Food Processing	Auto	2	23877.0
...
11246	1000695	Manning	P00296942	M	18-25	19		1	Maharashtra	Western	Chemical	Office	4	370.0
11247	1004089	Reichenbach	P00171342	M	26-35	33		0	Haryana	Northern	Healthcare	Veterinary	3	367.0
11248	1001209	Oshin	P00201342	F	36-45	40		0	Madhya Pradesh	Central	Textile	Office	4	213.0
11249	1004023	Noonan	P00059442	M	36-45	37		0	Karnataka	Southern	Agriculture	Office	3	206.0
11250	1002744	Brumley	P00281742	F	18-25	19		0	Maharashtra	Western	Healthcare	Office	3	188.0

```
[ ] df.shape  
→ (11239, 13)
```

Before the delete Null values. There were 11251 rows and 13 columns. After delete the null values there are 11239 Rows and 13 Columns in the data.

Now the Question is arise that df.drop(inplace=True) and df.drop() are same or not?

Ans:- When you use inplace=True, the DataFrame is modified in place, and the operation does not return a new DataFrame.

When you don't use inplace=True, the DataFrame is not modified in place. Instead, the operation returns a new DataFrame with the specified modifications.

```
[ ] # Change the data type of Amount  
df["Amount"] = df["Amount"].astype("Int")
```

```
▶ df["Amount"].dtype
```

```
→ dtype('int64')
```

```
[ ] df["Amount"] = df["Amount"].astype("Float")
```

```
df["Amount"].dtype
```

```
→ dtype('O')
```

```
▶ 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	Marital_Status	•	Orders
count	1.123900e+04	11239.000000	11239.000000	11239.000000	
mean	1.003004e+06	35.410357	0.420055	2.489634	
std	1.716039e+03	12.753866	0.493589	1.114967	
min	1.000001e+06	12.000000	0.000000	1.000000	
25%	1.001492e+06	27.000000	0.000000	2.000000	
50%	1.003064e+06	33.000000	0.000000	2.000000	
75%	1.004426e+06	43.000000	1.000000	3.000000	
max	1.006040e+06	92.000000	1.000000	4.000000	

```
▶ min 1.000001e+06 12.000000 0.000000 1.000000  
→ 25% 1.001492e+06 27.000000 0.000000 2.000000  
    50% 1.003064e+06 33.000000 0.000000 2.000000  
    75% 1.004426e+06 43.000000 1.000000 3.000000  
    max 1.006040e+06 92.000000 1.000000 4.000000
```

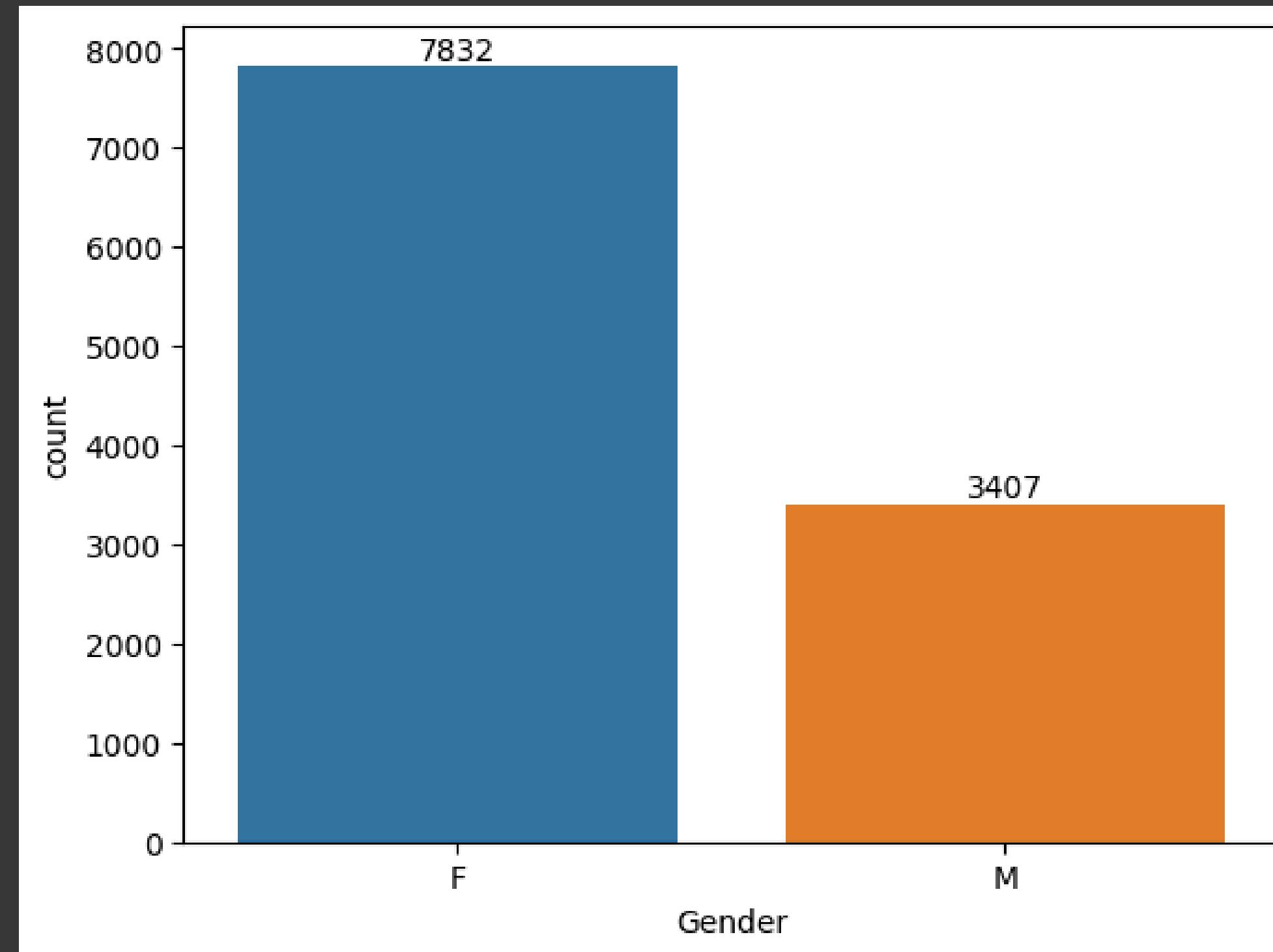
```
[ ] df[["Age","Orders","Amount"]].describe()
```



	Age	Orders
count	11239.000000	11239.000000
mean	35.410357	2.489634
std	12.753866	1.114967
min	12.000000	1.000000
25%	27.000000	2.000000
50%	33.000000	2.000000
75%	43.000000	3.000000
max	92.000000	4.000000

▼ Gender

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

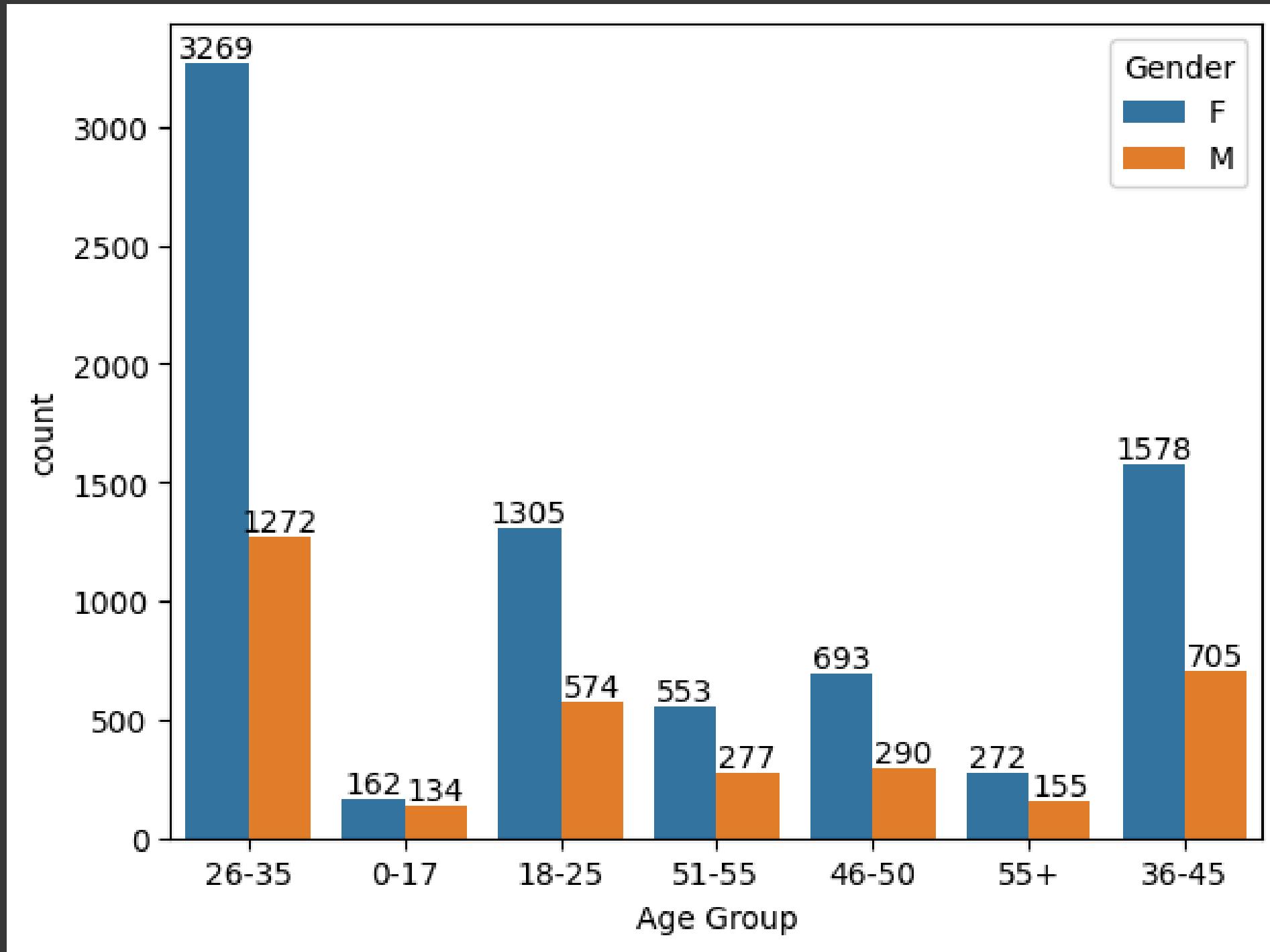


From above graphs we can see that most of the buyers are females and even the purchasing power of females are greater than men.

▼ Age

```
▶ ax=sns.countplot(x="Age Group",data=df,hue="Gender")
for bars in ax.containers:
    ax.bar_label(bars)
```

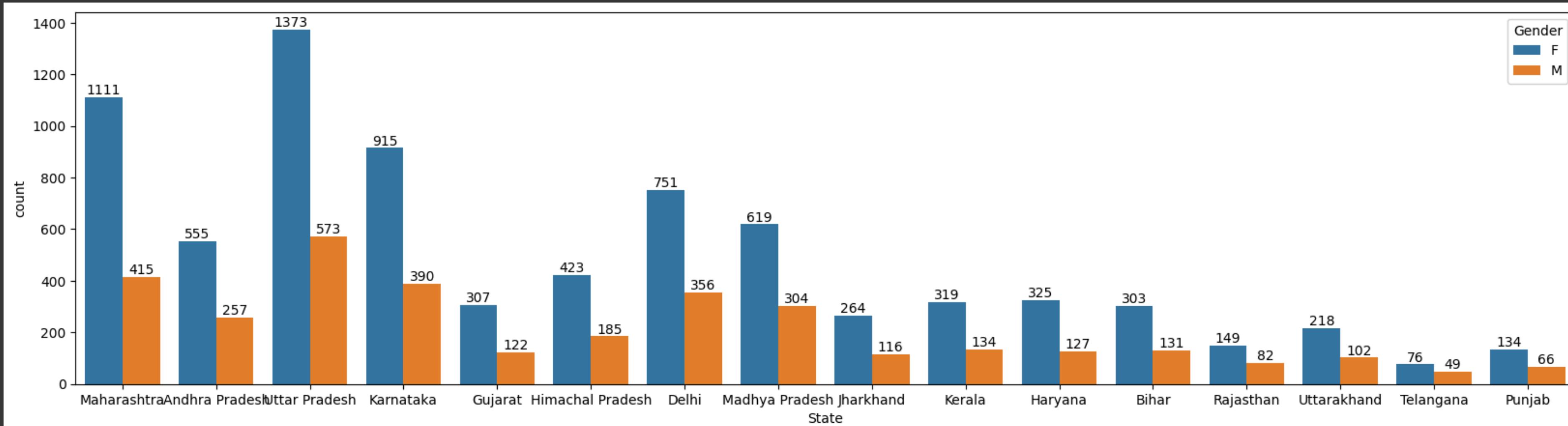
→



As we can see above graphs The age between 26-35 Females purchasing power is greater than other age group.

State

```
plt.figure(figsize=(20,5))
ax=sns.countplot(x="State",data=df,hue="Gender")
for bars in ax.containers:
    ax.bar_label(bars)
```



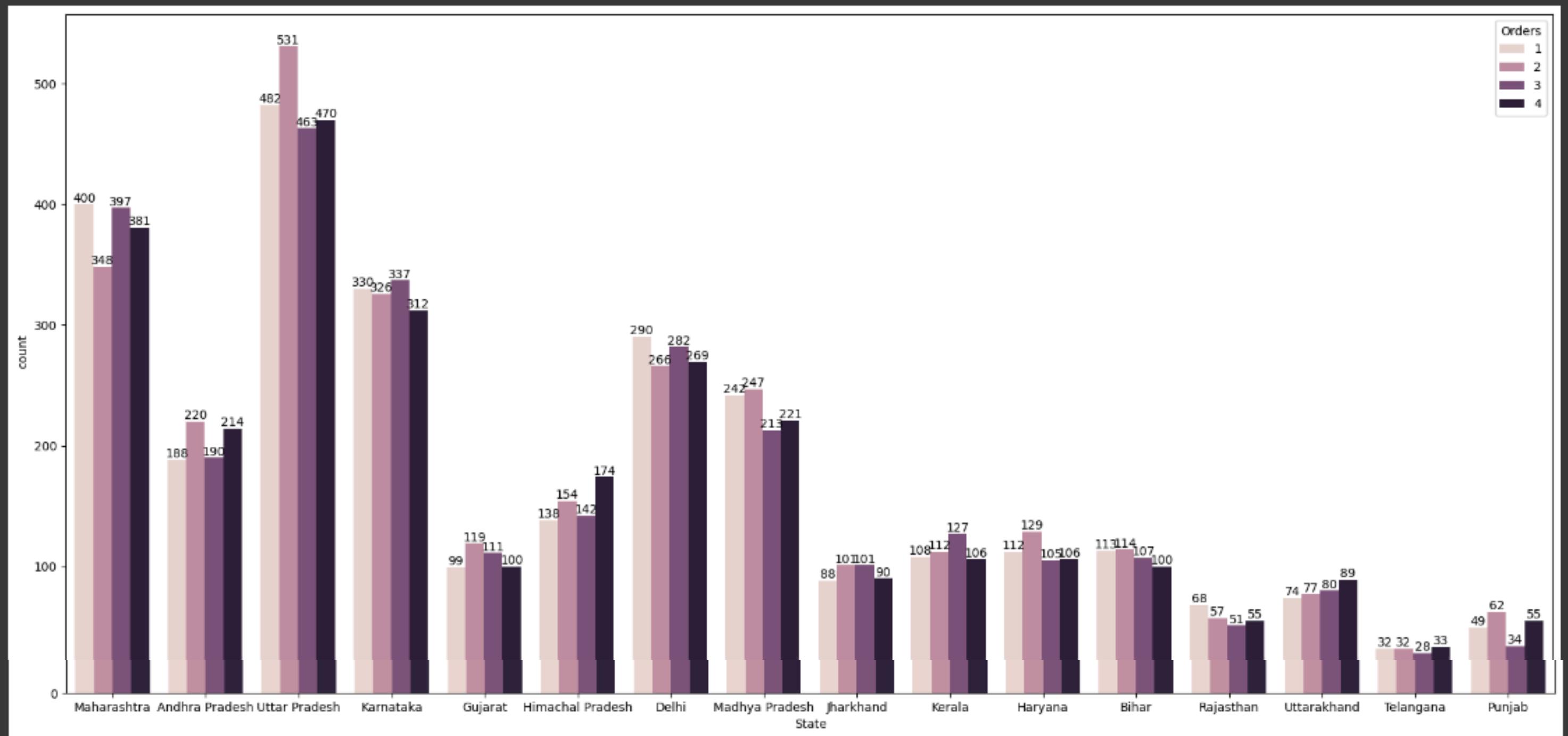
As we can see above graph. It Shows that uttar pradesh females purchasing more in comparision to other states.

f.columns

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

Orders

```
plt.figure(figsize=(22,10))
ax=sns.countplot(x="State",data=df,hue="Orders")
for bars in ax.containers:
    ax.bar_label(bars)
```



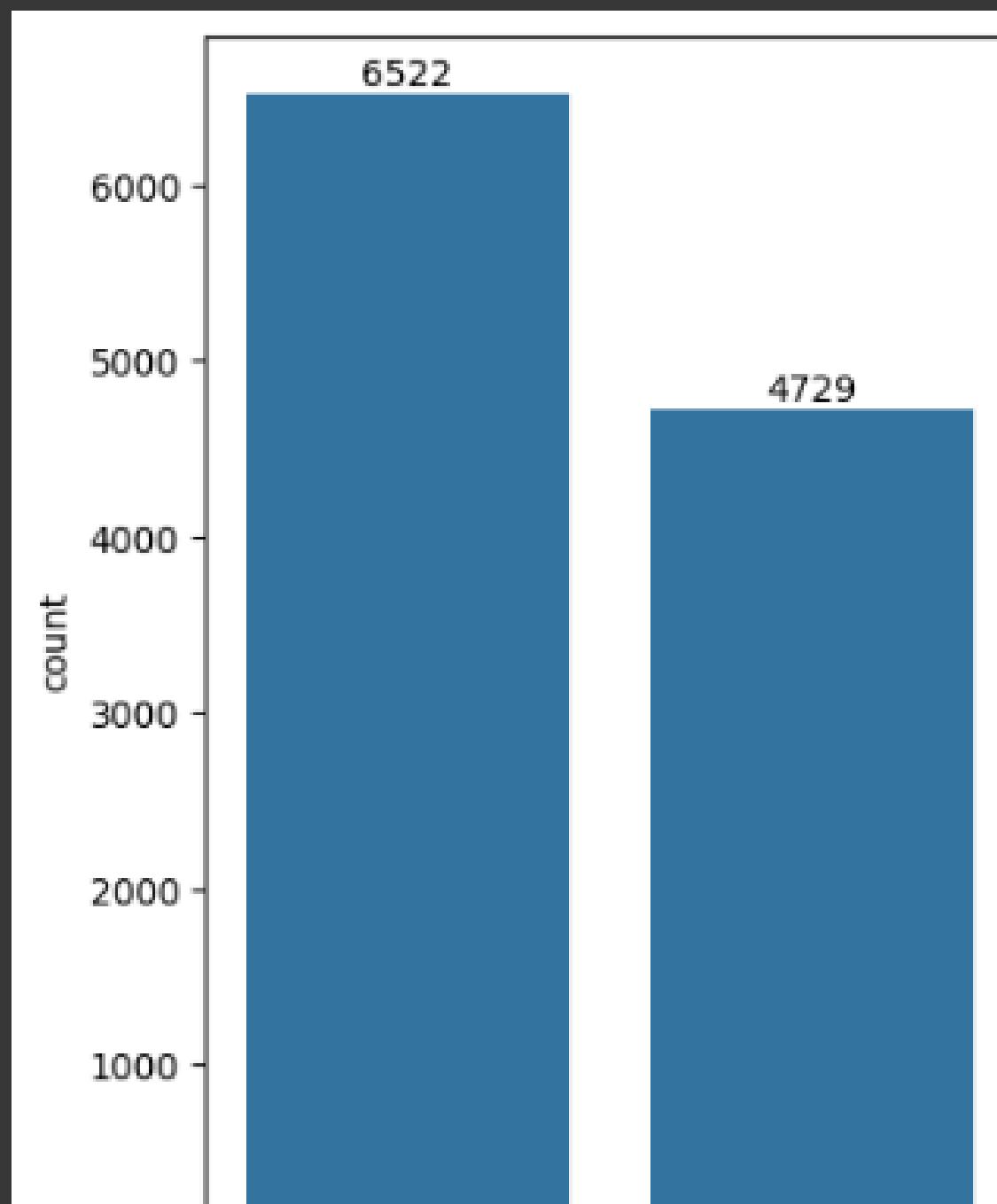
As we can see above chart.Uttar pradesh placed higher order in comparision to other states.

▼ Marital_Status

0 for married people

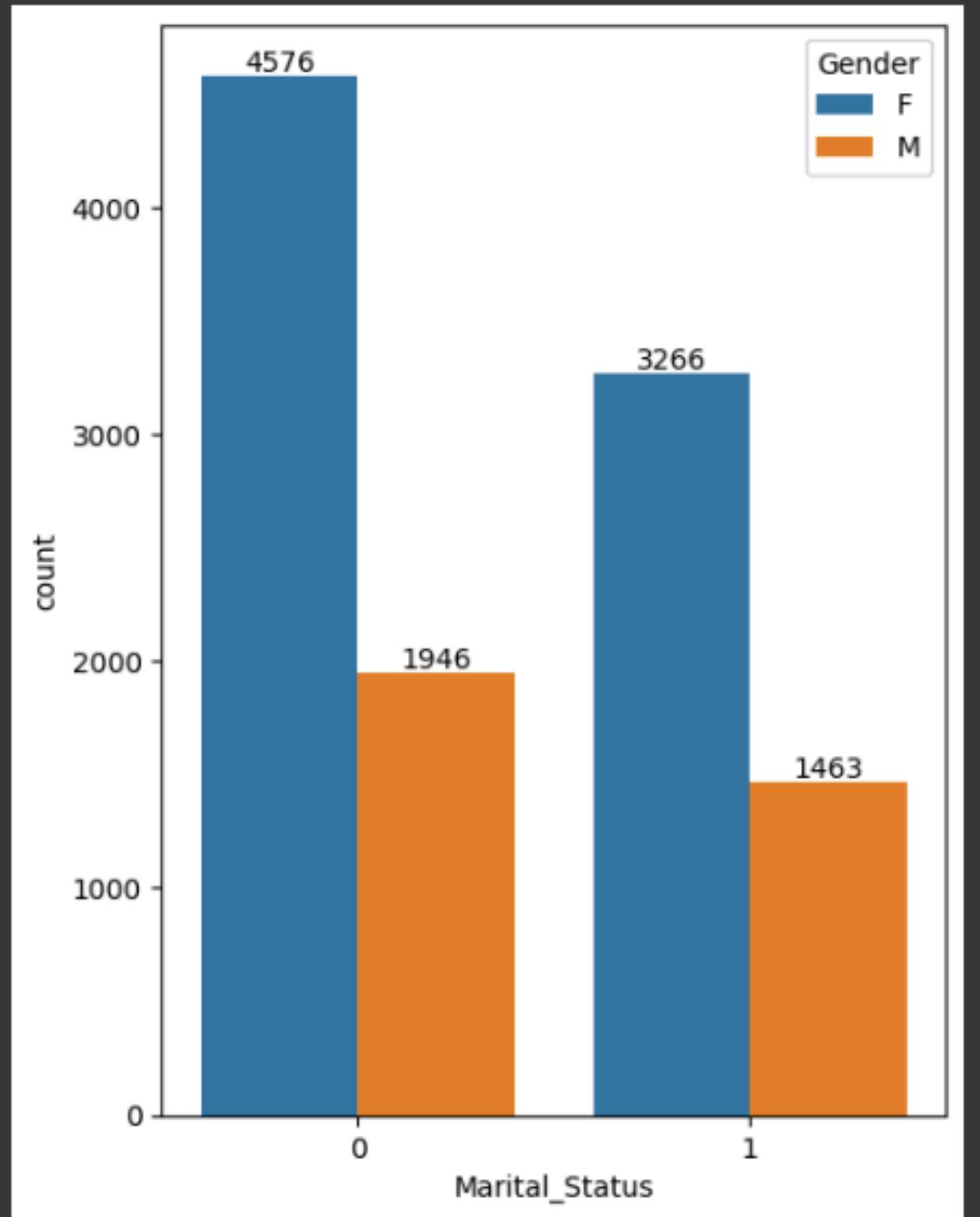
1 for unmarried people

```
plt.figure(figsize=(4,6))
ax=sns.countplot(x="Marital_Status",data=df)
for bars in ax.containers:
    ax.bar_label(bars)
```



It is clear that married people buyed more product in comparision to unmarried.

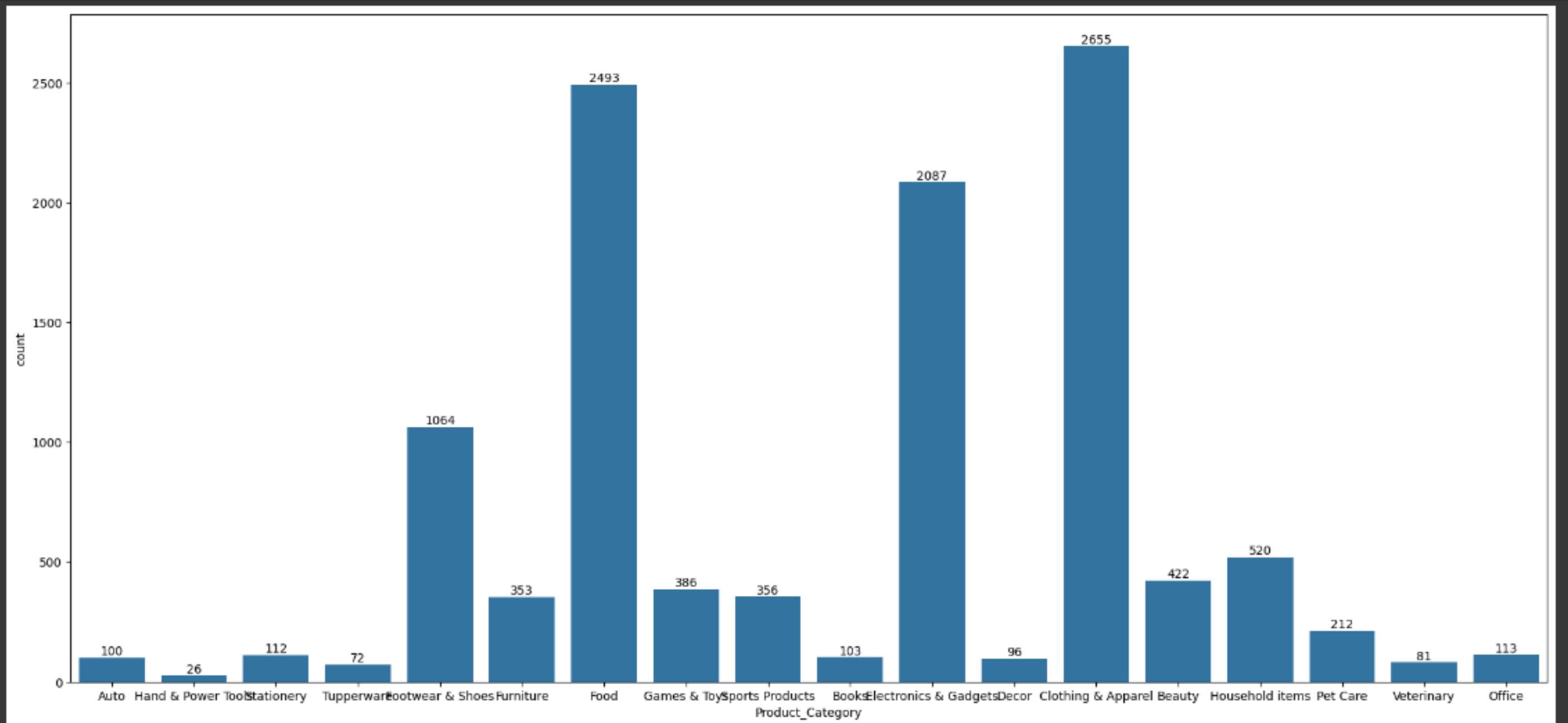
```
▶ plt.figure(figsize=(5,7))
ax=sns.countplot(x="Marital_Status",data=df,hue="Gender")
for bars in ax.containers:
    ax.bar_label(bars)
```



Married females Purchasing power is high in comparision to men.

Product_Catagory

```
plt.figure(figsize=(22,10))
ax= sns.countplot(x="Product_Catagory",data=df)
for bars in ax.containers:
    ax.bar_label(bars)
```



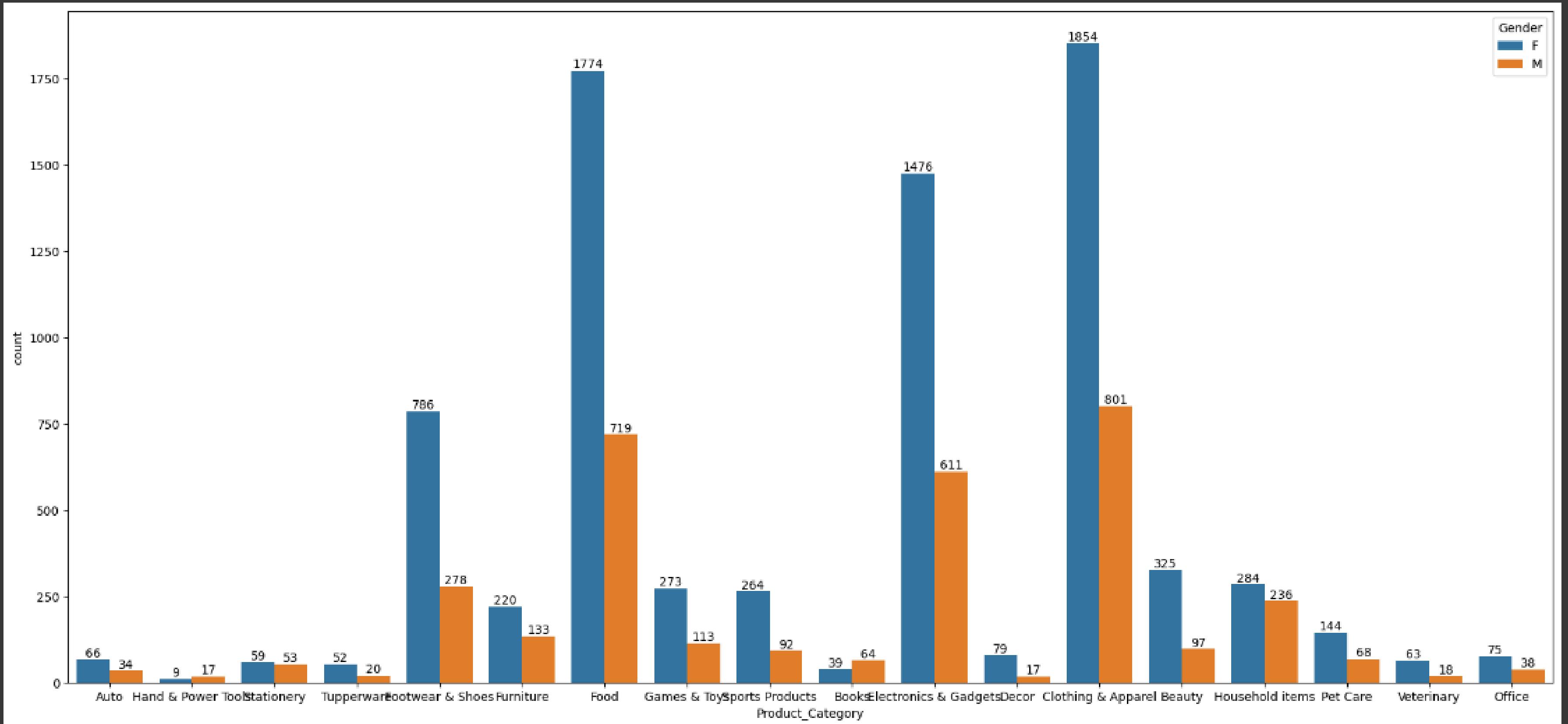
Clothing & Apparel is on first then Food and Electronic & Gadget is the highest place of Product category.

People spending there money on Clothing & Apprael, Food, Electronic & Gadget.

```

plt.figure(figsize=(22,18))
ax=sns.countplot(x="Product_Category",data=df,hue="Gender")
for bars in ax.containers:
    ax.bar_label(bars)

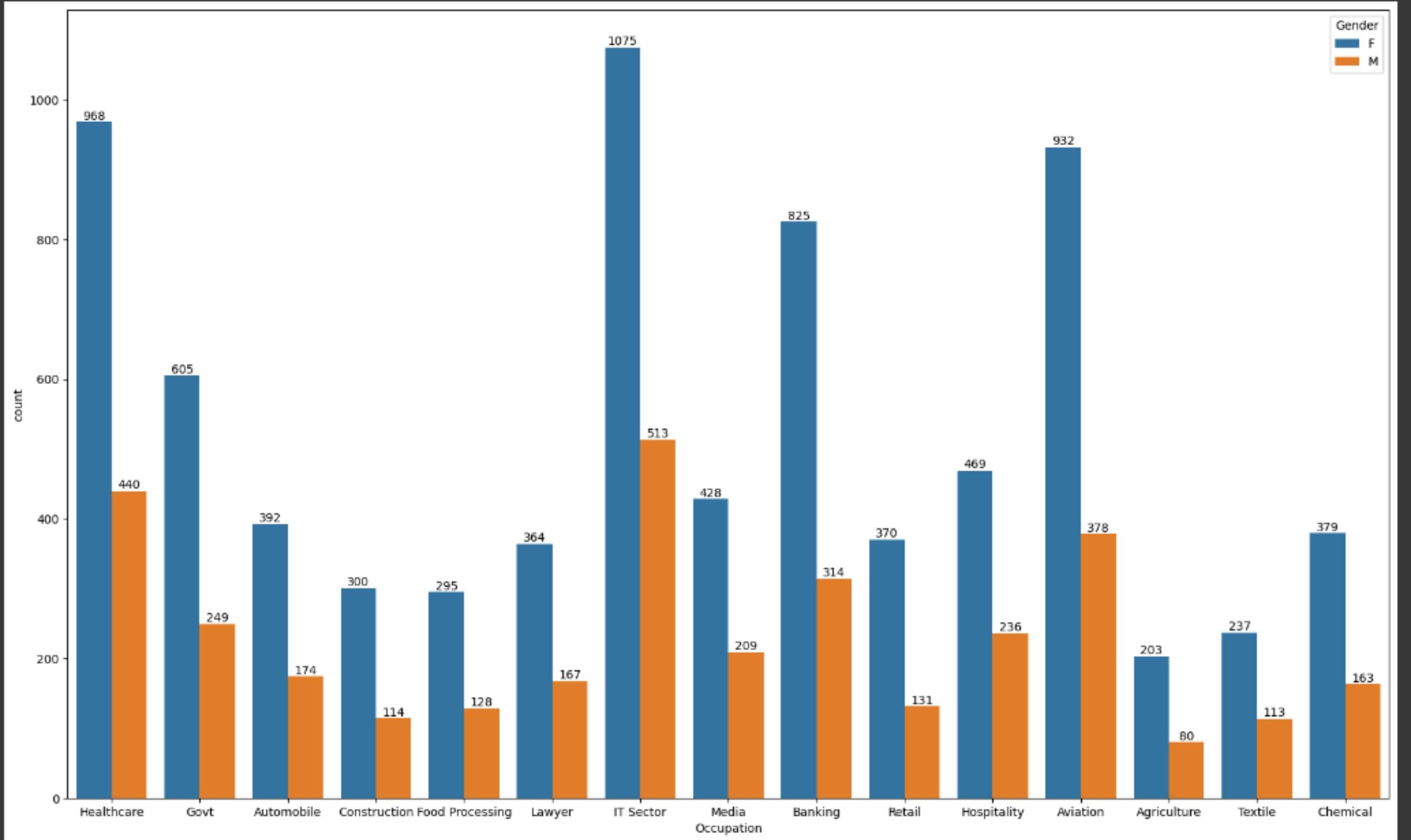
```



Women spending there more money on clothing & Apparel ,Food, Electronic & Gadget.

Occupation

```
plt.figure(figsize=(20,12))
ax=sns.countplot(x="Occupation",data=df,hue="Gender")
for bars in ax.containers:
    ax.bar_label(bars)
```



Women are Earning high salary.

They are Working in IT Sector, Healthcare, Aviation, Banking sectors.

- ✓ *Conclusion*

Married Women age group 26-35 yrs from UP, IT, Healthcare and Aviation are more likely to buy products from Clothing, Food and Electronic Category.

- ✓ ** THANK YOU**

