In [75]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

In [76]:
```
df = pd.read_csv(r'C:\Users\HP\OneDrive\Documents\new certificates\Google Data Analy
```

In [77]:
```
df.shape
```

Out[77]:
```
(11251, 15)
```

In [78]:
```
df.head(10)
```

Out[78]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western
5	1000588	Joni	P00057942	M	26-35	28	1	Himachal Pradesh	Northern
6	1001132	Balk	P00018042	F	18-25	25	1	Uttar Pradesh	Central
7	1002092	Shivangi	P00273442	F	55+	61	0	Maharashtra	Western
8	1003224	Kushal	P00205642	M	26-35	35	0	Uttar Pradesh	Central
9	1003650	Ginny	P00031142	F	26-35	26	1	Andhra Pradesh	Southern

In [79]:
```
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
```

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

In [82]:
```
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
```

In [84]:
```
pd.isnull(df)
```

Out[84]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupat
0	False	False	False	False	False	False	False	False	False	Fa
1	False	False	False	False	False	False	False	False	False	Fa
2	False	False	False	False	False	False	False	False	False	Fa
3	False	False	False	False	False	False	False	False	False	Fa
4	False	False	False	False	False	False	False	False	False	Fa
...	...	...	...	...	...	...	...	...	...	
11246	False	False	False	False	False	False	False	False	False	Fa
11247	False	False	False	False	False	False	False	False	False	Fa
11248	False	False	False	False	False	False	False	False	False	Fa
11249	False	False	False	False	False	False	False	False	False	Fa
11250	False	False	False	False	False	False	False	False	False	Fa

11251 rows × 13 columns

In [85]:
```
pd.isnull(df).sum()
```

Out[85]:
```
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 [86]:
```
df.dropna(inplace=True)
```

In [87]:
```
pd.isnull(df).sum()
```

Out[87]:
```
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
```

In [88]:
```
df['Amount'] = df['Amount'].astype('int')
```

In [89]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 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  int32 
dtypes: int32(1), int64(4), object(8)
memory usage: 1.2+ MB
```

In [90]:
```
df['Amount'].dtypes
```

Out[90]:
```
dtype('int32')
```

In [91]:
```
df.columns
```

Out[91]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [92]:
```
df.rename(columns= {'Marital_Status':'Shaadi'})
```

Out[92]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Shaadi	State	Zone	C
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	A
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	C
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	
...	...	...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	

11239 rows × 13 columns

In [93]:
```
df.describe()
```

Out[93]:

	User_ID	Age	Marital_Status	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

In [94]:
```
df[['Age', 'Orders', 'Amount']].describe()
```

Out[94]:

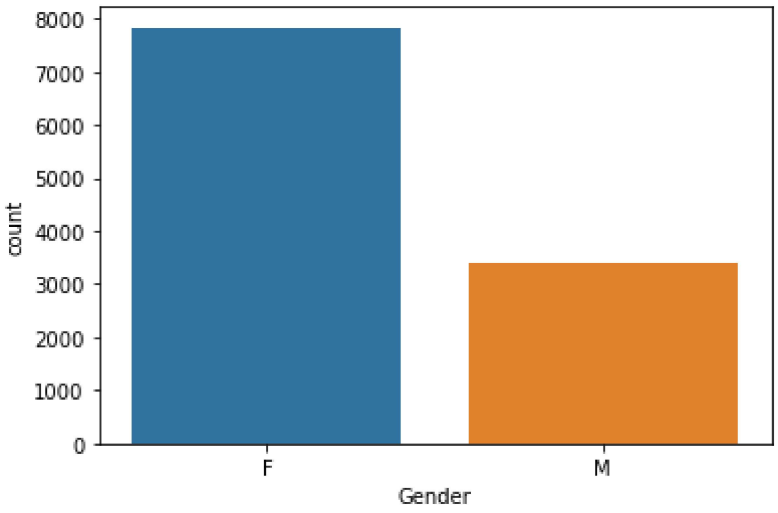
	Age	Orders	Amount
count	11239.000000	11239.000000	11239.000000
mean	35.410357	2.489634	9453.610553
std	12.753866	1.114967	5222.355168
min	12.000000	1.000000	188.000000
25%	27.000000	2.000000	5443.000000
50%	33.000000	2.000000	8109.000000
75%	43.000000	3.000000	12675.000000
max	92.000000	4.000000	23952.000000

In [95]:
```
df.columns
```

Out[95]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

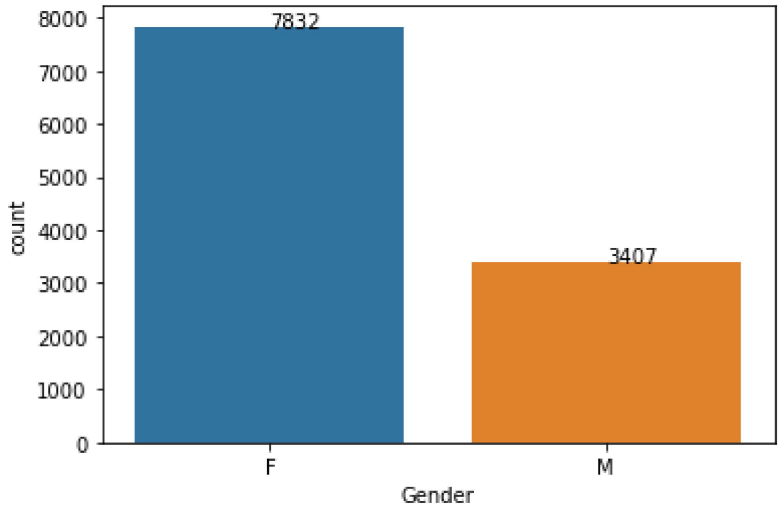
In [97]:
```
sns.countplot(x = 'Gender',data = df)
```

Out[97]:
```
<AxesSubplot:xlabel='Gender', ylabel='count'>
```



In [115…
```
ax = sns.countplot(x='Gender', data=df)

for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
    (p.get_x() + p.get_width() / 2., p.get_height()))
```



In [120…
```
df.groupby(['Gender'], as_index = False)['Amount'].sum().sort_values(by = 'Amount',
```

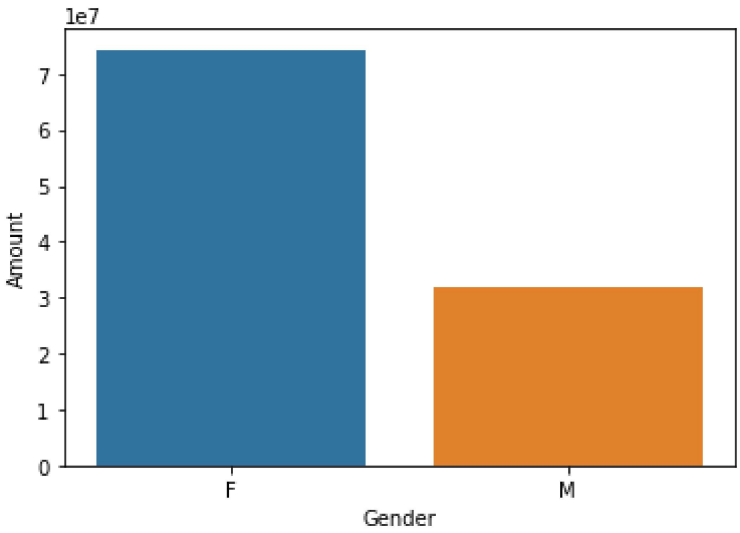
Out[120…

	Gender	Amount
0	F	74335853
1	M	31913276

In [121…
```
sales_gen = df.groupby(['Gender'], as_index = False)['Amount'].sum().sort_values(by

sns.barplot(x= 'Gender', y = 'Amount', data = sales_gen)
```

Out[121…
```
<AxesSubplot:xlabel='Gender', ylabel='Amount'>
```

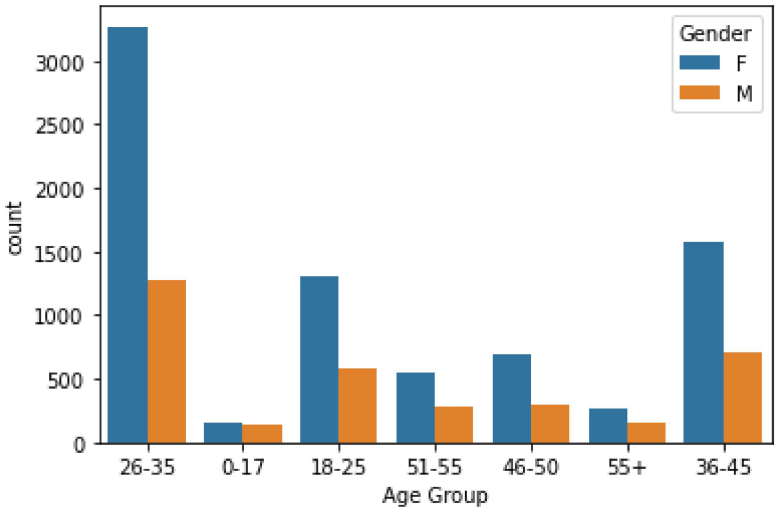


In [122…
```
df.columns
```

Out[122…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

In [124…
```
sns.countplot(x='Age Group',hue = 'Gender', data=df)
```

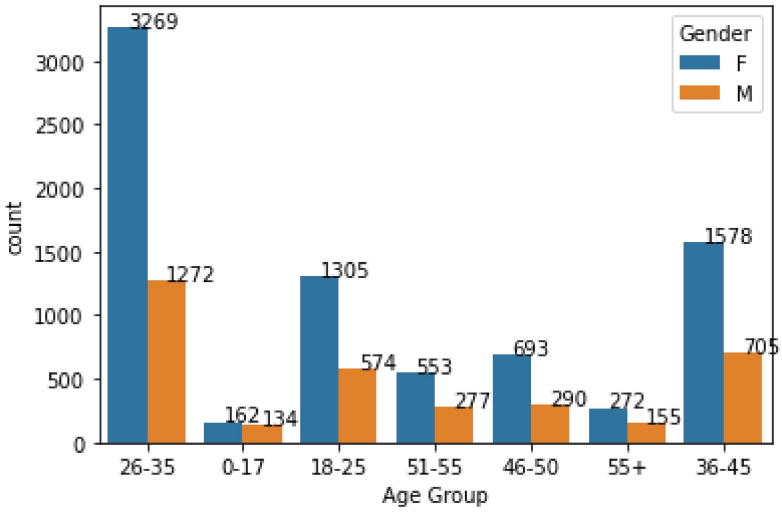
Out[124…
```
<AxesSubplot:xlabel='Age Group', ylabel='count'>
```



In [132…

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

for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
    (p.get_x() + p.get_width() / 2., p.get_height()))
```

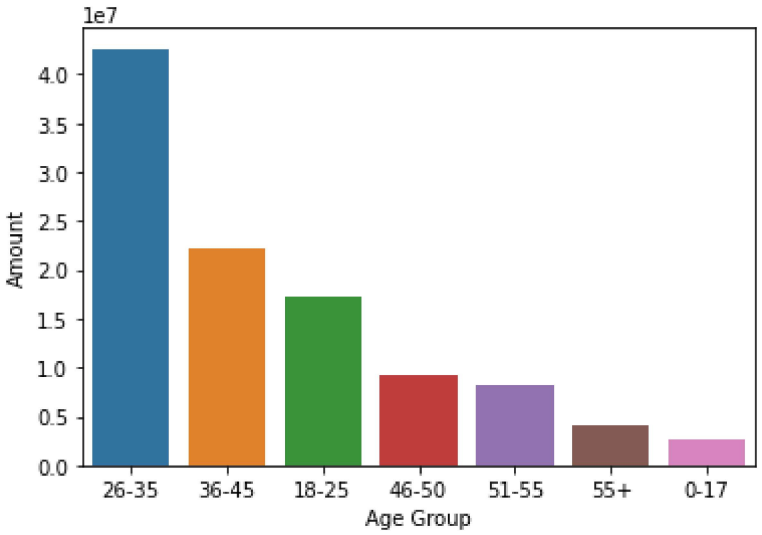


In [134…

```
sales_age= df.groupby(['Age Group'], as_index = False)['Amount'].sum().sort_values(b

sns.barplot(x= 'Age Group', y = 'Amount', data = sales_age)
```

Out[134… <AxesSubplot:xlabel='Age Group', ylabel='Amount'>

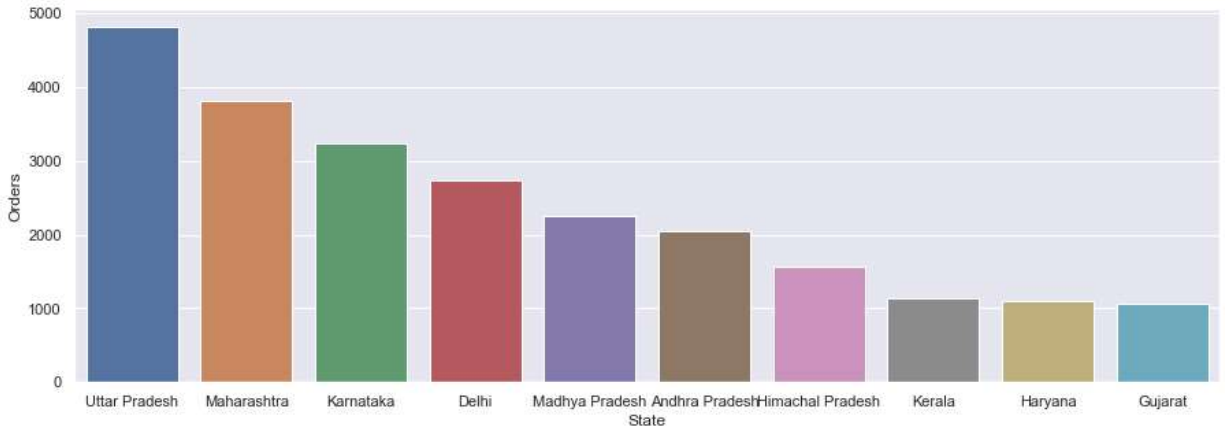


In [153…

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

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x= 'State', y = 'Orders', data = sales_state)
```

Out[153… <AxesSubplot:xlabel='State', ylabel='Orders'>

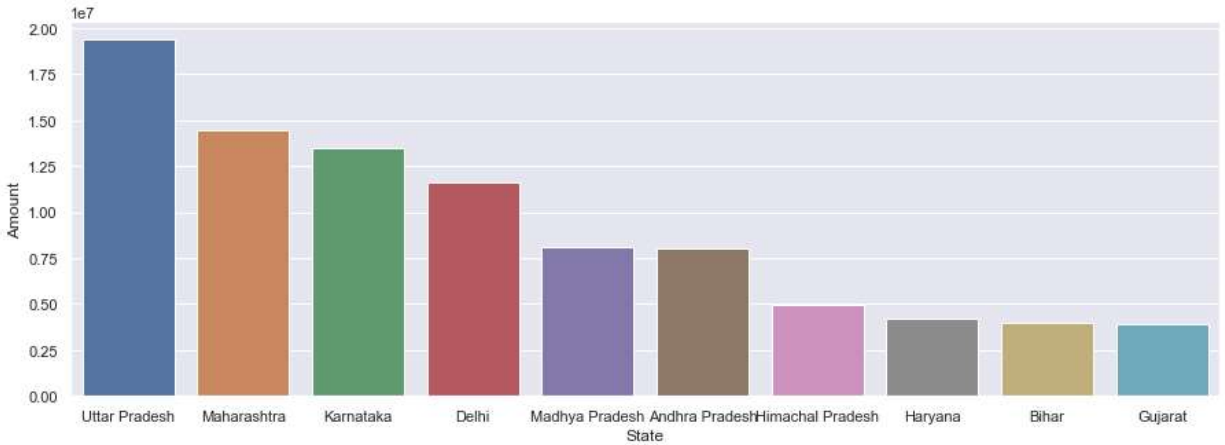


In [154…

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

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x= 'State', y = 'Amount', data = sales_state)
```

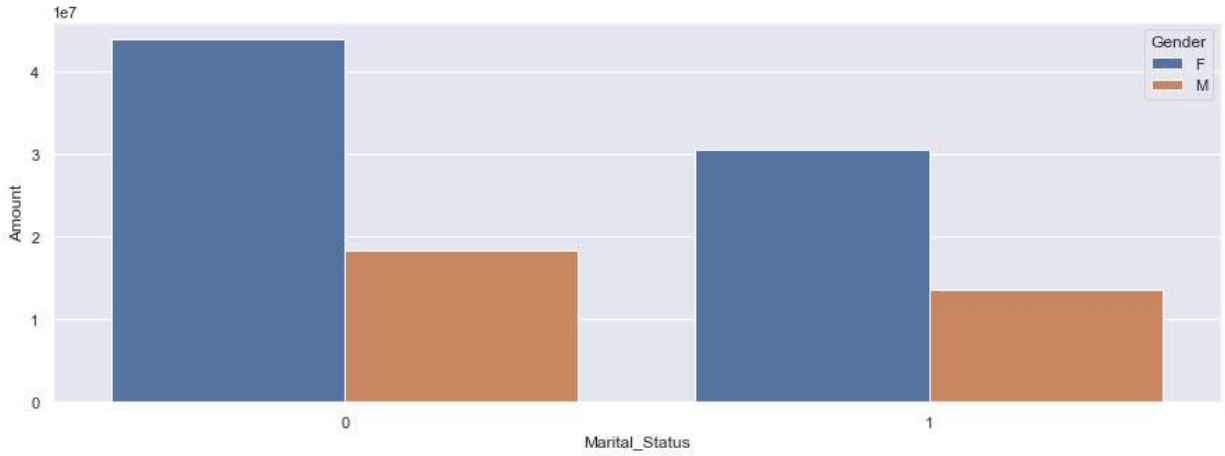
Out[154… <AxesSubplot:xlabel='State', ylabel='Amount'>



In [161…
```
sales_state= df.groupby(['Marital_Status', 'Gender'], as_index = False)['Amount'].su

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x = 'Marital_Status',y='Amount' ,hue='Gender',data = sales_state)
```

Out[161…
```
<AxesSubplot:xlabel='Marital_Status', ylabel='Amount'>
```

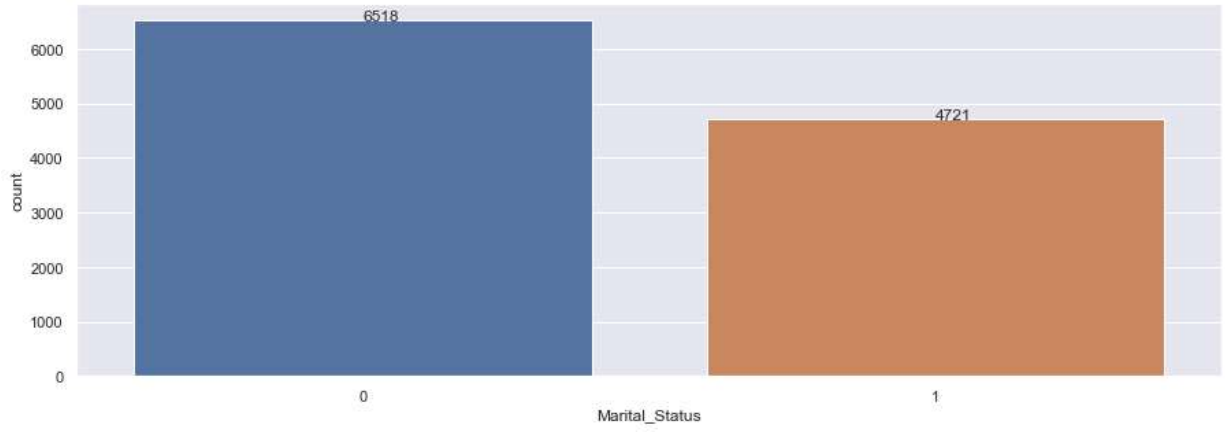


In [155…
```
df.columns
```

Out[155…
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

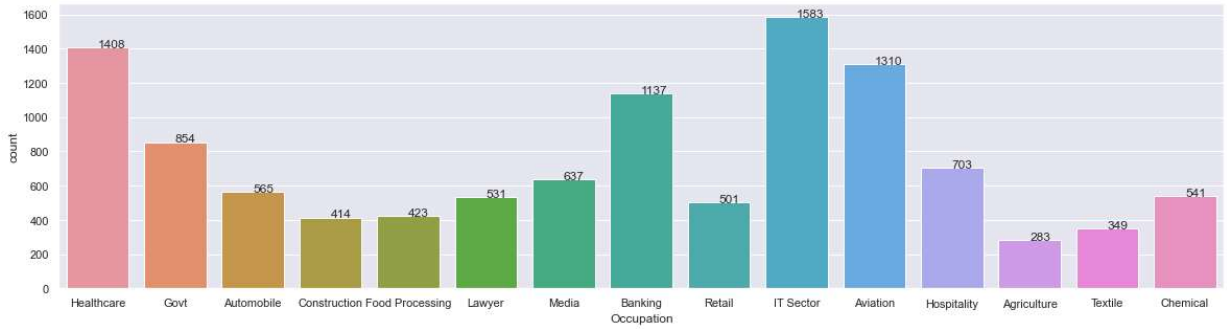
In [162…
```
ax = sns.countplot(x='Marital_Status', data=df)

for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
    (p.get_x() + p.get_width() / 2., p.get_height()))
```



In [167…
```
sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(x='Occupation', data=df)

for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
    (p.get_x() + p.get_width() / 2., p.get_height()))
```

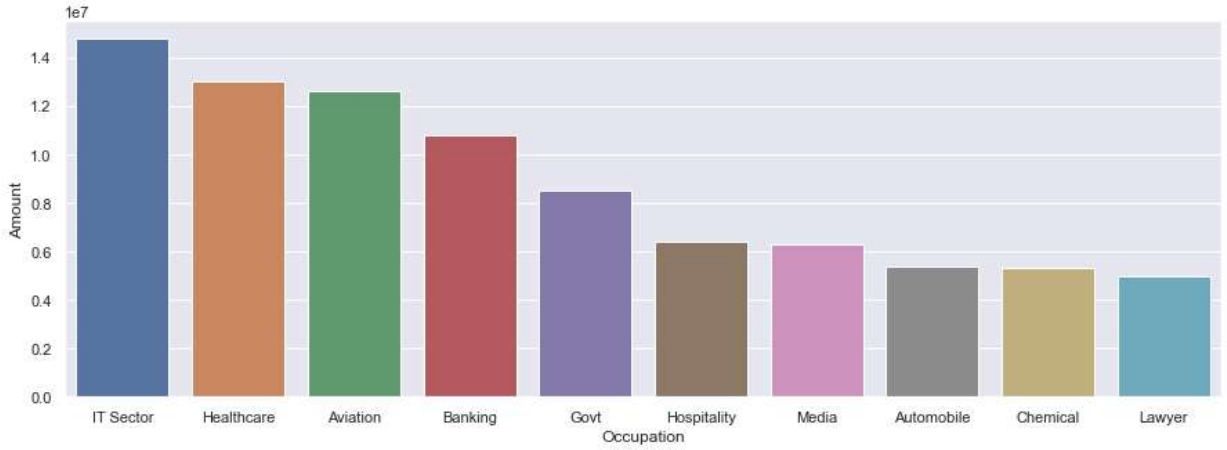


In [168…

```
sales_state= df.groupby(['Occupation'], as_index = False)['Amount'].sum().sort_value

sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x = 'Occupation',y='Amount',data = sales_state)
```

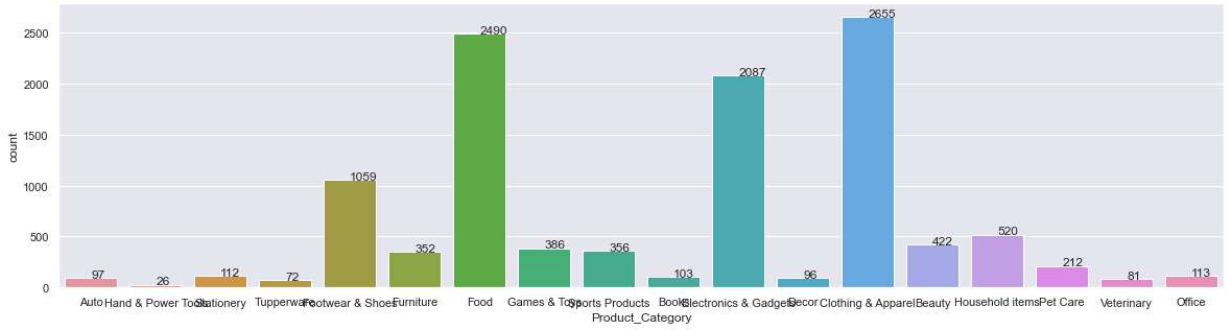
Out[168… <AxesSubplot:xlabel='Occupation', ylabel='Amount'>



In [169…

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

for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
    (p.get_x() + p.get_width() / 2., p.get_height()))
```

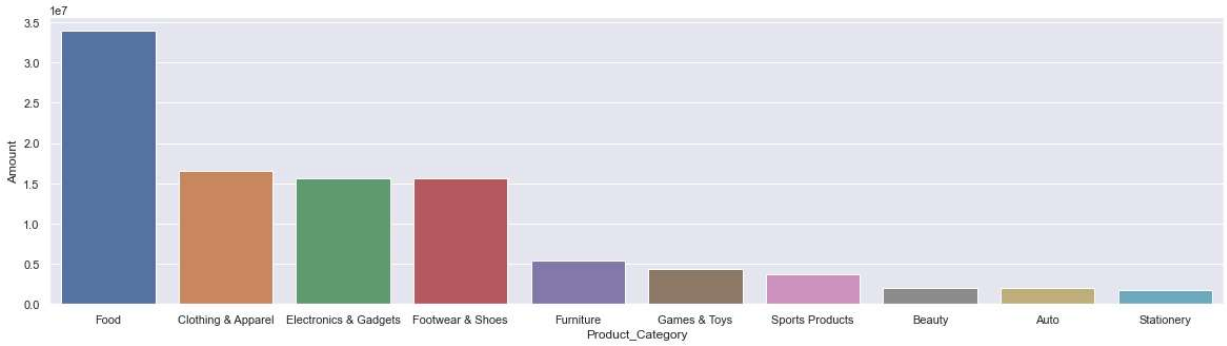


In [173…

```
sales_state= df.groupby(['Product_Category'], as_index = False)['Amount'].sum().sort

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

Out[173… <AxesSubplot:xlabel='Product_Category', ylabel='Amount'>

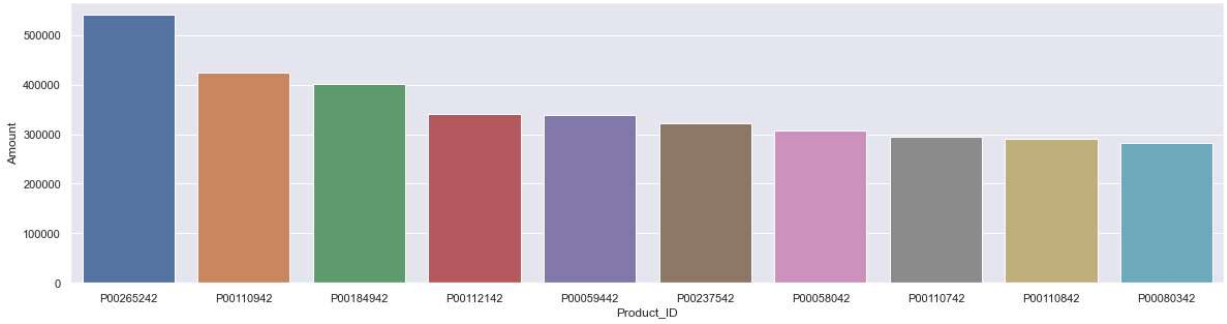


In [174…

```
sales_state= df.groupby(['Product_ID'], as_index = False)['Amount'].sum().sort_value

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

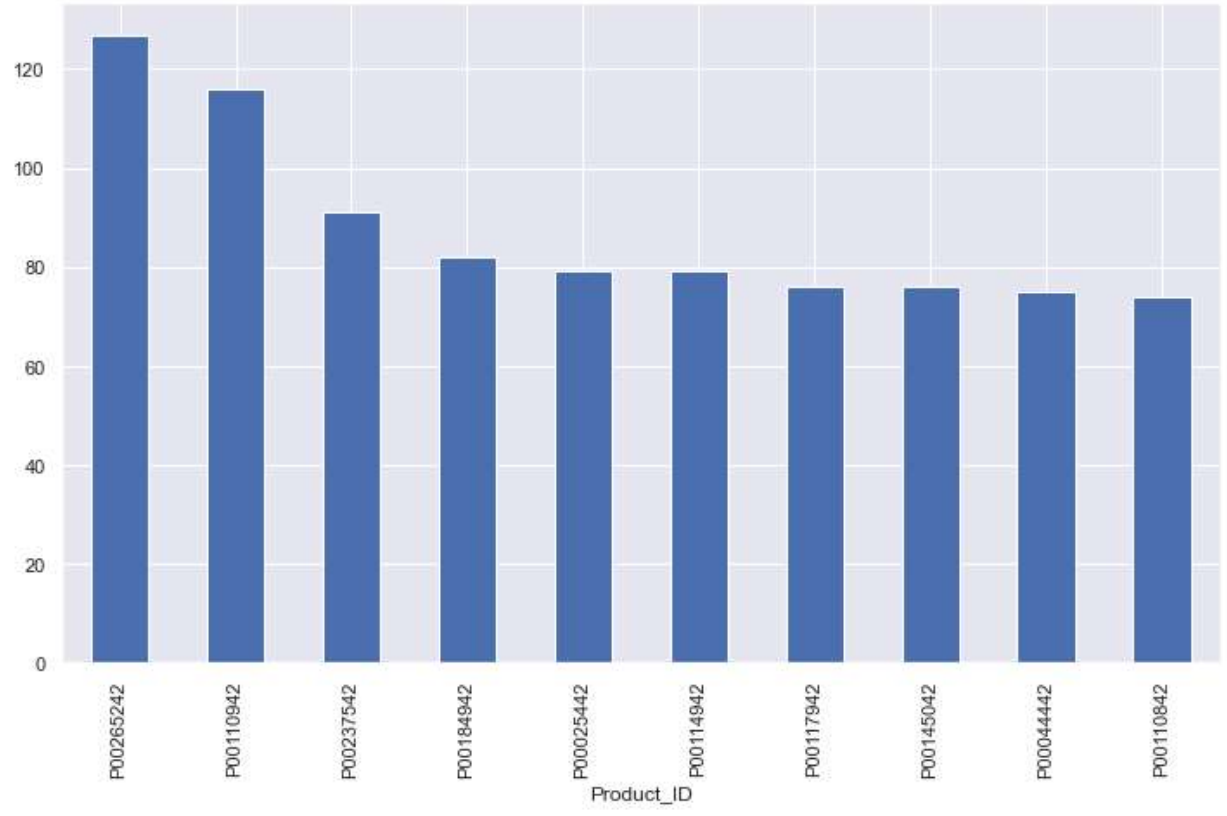
Out[174… <AxesSubplot:xlabel='Product_ID', ylabel='Amount'>



In [176…

```
fig1, ax1 = plt.subplots(figsize=(12,7))
df.groupby('Product_ID')['Orders'].sum().nlargest(10).sort_values(ascending=False).p
```

Out[176… <AxesSubplot:xlabel='Product_ID'>



In [ ]: