DIWALI SALES ANALYSIS

```
# import python libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt # visualizing data
%matplotlib inline
import seaborn as sns
# import csv file
df = pd.read csv(r"C:\Diwali Sales Data.csv",encoding=
'unicode escape')
df
       User ID
                  Cust name Product ID Gender Age Group Age
Marital Status
0
       1002903
                  Sanskriti P00125942
                                                    26-35
                                                            28
0
1
                     Kartik
       1000732
                              P00110942
                                                    26-35
                                                            35
1
2
       1001990
                      Bindu
                              P00118542
                                                    26-35
                                                            35
1
3
       1001425
                      Sudevi
                              P00237842
                                                     0-17
                                                            16
0
4
       1000588
                        Joni
                              P00057942
                                             М
                                                    26-35
                                                            28
1
       1000695
11246
                    Manning
                              P00296942
                                                    18-25
                                                            19
11247
       1004089
                Reichenbach
                              P00171342
                                                    26-35
                                                            33
11248
      1001209
                       0shin
                              P00201342
                                                    36-45
                                                            40
11249
       1004023
                     Noonan
                              P00059442
                                                    36-45
                                                            37
11250
       1002744
                    Brumley P00281742
                                                    18-25
                                                            19
                State
                            Zone
                                       Occupation Product Category
0rders
0
          Maharashtra
                         Western
                                       Healthcare
                                                               Auto
1
1
       Andhra Pradesh Southern
                                             Govt
                                                               Auto
3
2
        Uttar Pradesh
                                       Automobile
                         Central
                                                               Auto
```

3 3	Kar	nataka	Southern	Cons	struc [.]	tion		Auf	to
2	Nai	IIataka	Journern	Cons	Struc	CIOII		Au	LO
4	C	Gujarat	Western	Food Pi	oces	sing		Au	to
2									
		• • •				• • •		•	
11246	Mahar	ashtra	Western		Chem:	ical		Offic	ce
4 11247 3	ŀ	laryana	Northern	Не	ealth	care	,	Veterina	ry
11248 4	Madhya F	Pradesh	Central		Tex	tile		Offic	ce
11249	Kar	nataka	Southern	Agı	ricul	ture		Offic	ce
3 11250	Mahar	ashtra	Western	Не	ealth	care		Offic	ce
3									
0 1 2 3 4	Amount 23952.0 23934.0 23924.0 23912.0 23877.0	Status NaN NaN NaN NaN NaN	NaN NaN NaN NaN NaN						
11246 11247 11248 11249 11250	370.0 367.0 213.0 206.0 188.0	NaN NaN NaN NaN NaN	NaN NaN NaN						
[11251	rows x 1	5 colum	nsl						
df.head									
	_ID Cus	st_name	Product_ID	Gender	Age (Group	Age	Marital_	_Status
0 1002	2903 Sar	nskriti	P00125942	F	Ž	26-35	28		0
1 1000	9732	Kartik	P00110942	F	ï	26-35	35		1
2 1001	1990	Bindu	P00118542	F	;	26-35	35		1
3 1001	1425	Sudevi	P00237842	М		0-17	16		0
4 1000	9588	Joni	P00057942	М		26-35	28		1
\	Sta	ate	Zone	0ccupa	ation	Produ	ct_Ca	tegory (Orders

0	ľ	1ahar	ashtra	West	tern		Heal ⁻	thcare			Auto	1	
1	Andl	nra P	radesh	South	nern			Govt			Auto	3	
2	Uti	tar P	radesh	Cent	tral		Autor	nobile	<u> </u>		Auto	3	
3		Karı	nataka	South	nern	(Constr	uction			Auto	2	
4		G	ujarat	West	tern	Food	d Proce	essing			Auto	2	
0 1 2 3 4	2395 2395 2395 2395 2385	ount 52.0 34.0 24.0 12.0 77.0	Status NaN NaN NaN NaN NaN	unna	amed1 NaN NaN NaN NaN NaN								
αT	.tai		TD (Cuct r	2200	Drodi	ıct ID	Condo	r Ago	Croup	Λαο		
		User L_Sta	tus \	_			_	Gende	_	Group	Age		
1	246	1000		Manr			296942		M	18-25	19		
0	247	1004		Lchenk			171342		M	26-35	33		
0	248	1001			shin		201342		F	36-45	40		
11 0	249	1004	023	Noc	onan	P000	959442		М	36-45	37		
11 0	250	1002	744	Brun	nley	P002	281742		F	18-25	19		
			Sta	ate	7	one	0ccui	nation	Prod	uct Ca [.]	tegory	0rders	
	ount	\ M.								_			
	246 0.0	IMI	aharasht	Lra	west	ern	Che	emica			Office	4	
	247 7.0		Harya	ana N	North	ern	Heal [.]	thcare		Vete	rinary	3	
11	248 3.0	Madh	ya Prade	esh	Cent	ral	Te	extile	1		Office	4	
11	249		Karnata	aka S	South	ern	Agric	ulture			Office	3	
11	6.0 250 8.0	Ma	aharasht	tra	West	ern	Heal ⁻	thcare			Office	3	
11	246 247 248	Na	us unna aN aN aN	amed1 NaN NaN NaN									

```
11249
                    NaN
          NaN
11250
          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
     Cust_name
                        11251 non-null
1
                                       object
 2
     Product ID
                        11251 non-null
                                        object
 3
     Gender
                                        object
                        11251 non-null
 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
                        0 non-null
                                        float64
     Status
 14
     unnamed1
                        0 non-null
                                        float64
dtypes: float64(3), int64(4), object(8)
memory usage: 1.3+ MB
df.shape
(11251, 15)
#drop unrelated/blank columns
df.drop(['Status', 'unnamed1'], axis=1, inplace=True)
#check for null values
pd.isnull(df).sum()
User ID
                      0
Cust name
                      0
                      0
Product ID
                      0
Gender
                      0
Age Group
                      0
Age
Marital Status
                      0
                      0
State
                      0
Zone
Occupation
                      0
                      0
Product Category
                      0
0rders
```

Amount 12 dtype: int64 # drop null values df.dropna(inplace=True) df User ID Cust name Product ID Gender Age Group Age Marital_Status 1002903 Sanskriti P00125942 26-35 28 0 1 1000732 Kartik P00110942 26-35 35 1 2 1001990 Bindu P00118542 26-35 35 1 3 1001425 Sudevi P00237842 0-17 16 0 4 P00057942 1000588 Joni 26-35 28 1 . . . 11246 1000695 Manning P00296942 18-25 19 11247 1004089 Reichenbach P00171342 26-35 33 11248 1001209 0shin P00201342 36-45 40 36-45 11249 1004023 Noonan P00059442 37 11250 1002744 Brumley P00281742 18-25 19 Zone Occupation Product Category State 0rders 0 Maharashtra Western Healthcare Auto 1 1 Andhra Pradesh Southern Govt Auto 3 2 Uttar Pradesh Automobile Central Auto 3 3 Karnataka Southern Construction Auto 2 4 Gujarat Western Food Processing Auto 2 11246 Maharashtra Chemical Office Western 11247 Haryana Northern Healthcare Veterinary

```
3
11248
      Madhya Pradesh Central
                                       Textile
                                                         Office
11249
           Karnataka Southern
                                    Agriculture
                                                         Office
11250
         Maharashtra
                       Western
                                     Healthcare
                                                         Office
3
       Amount
0
      23952.0
1
      23934.0
2
      23924.0
3
      23912.0
4
      23877.0
11246
        370.0
        367.0
11247
11248
        213.0
11249
        206.0
        188.0
11250
[11239 rows x 13 columns]
# change data type
df['Amount'] = df['Amount'].astype('int')
df['Amount'].dtypes
dtype('int32')
df.columns
Index(['User ID', 'Cust name', 'Product ID', 'Gender', 'Age Group',
'Age',
       'Marital Status', 'State', 'Zone', 'Occupation',
dtype='object')
#rename column
df.rename(columns= {'Marital Status':'Shaadi'})
                 Cust name Product ID Gender Age Group Age
      User ID
Shaadi
      1002903
                 Sanskriti P00125942
                                           F
                                                26-35
                                                        28
                                                                 0
      1000732
                    Kartik P00110942
                                                26-35
                                                        35
                                                                 1
2
      1001990
                     Bindu P00118542
                                                26-35
                                                        35
                                                                 1
3
      1001425
                    Sudevi P00237842
                                          М
                                                 0-17
                                                        16
                                                                 0
```

4	1000588		Joni	P00	057942	М	26-35	28	1
11246	1000695	Ма	nning	P00	296942	М	18-25	19	1
11247	1004089	Reiche	nbach	P00	171342	М	26-35	33	0
11248	1001209		0shin	P00	201342	F	36-45	40	0
11249	1004023	N	oonan	P00	059442	М	36-45	37	0
11250	1002744	Br	umley	P00	281742	F	18-25	19	0
0rders	\	State	Z	one	0	ccupatio	n Product _.	_Catego	ry
0 1	Mahara	ashtra	West	ern	Н	ealthcar	e	Au	ito
1	Andhra P	radesh	South	ern		Gov	t	Au	ito
3 2 3 3 2 4	Uttar P	radesh	Cent	ral	А	utomobil	e	Au	to
3	Karı	nataka	South	ern	Con	structio	n	Au	ito
4	Gı	ujarat	West	ern	Food P	rocessin	g	Au	ito
2									
 11246	Mahara	ashtra	West	ern		Chemica	l	0ffi	.ce
4 11247	H:	aryana	North	ern	Н	ealthcar	e V	eterina	
3 11248	Madhya P	-	Cent			Textil		0ffi	
4	•								
11249 3		nataka	South		_	ricultur		0ffi	
11250 3	Mahara	ashtra	West	ern	Н	ealthcar	e	0ffi	.ce
	Amount								
0	23952 23934								
1 2 3	23924 23912								
4	23877								
11246 11247	370 367								
11271	307								

```
11248 213
11249 206
11250 188
```

[11239 rows x 13 columns]

describe() method returns description of the data in the DataFrame
(i.e. count, mean, std, etc)
df.describe()

	User ID	Age	Marital Status	0rders				
Amount	_	_	_					
count	1.123900e+04	11239.000000	11239.000000	11239.000000				
11239.	000000							
mean	1.003004e+06	35.410357	0.420055	2.489634				
9453.6	10553							
std	1.716039e+03	12.753866	0.493589	1.114967				
5222.3	55168							
min	1.000001e+06	12.000000	0.000000	1.000000				
188.00	0000							
25%	1.001492e+06	27.000000	0.000000	2.000000				
5443.0	5443.000000							
50%	1.003064e+06	33.000000	0.000000	2.000000				
8109.0	00000							
75%	1.004426e+06	43.000000	1.000000	3.000000				
12675.000000								
max	1.006040e+06	92.000000	1.000000	4.000000				
23952.	000000							

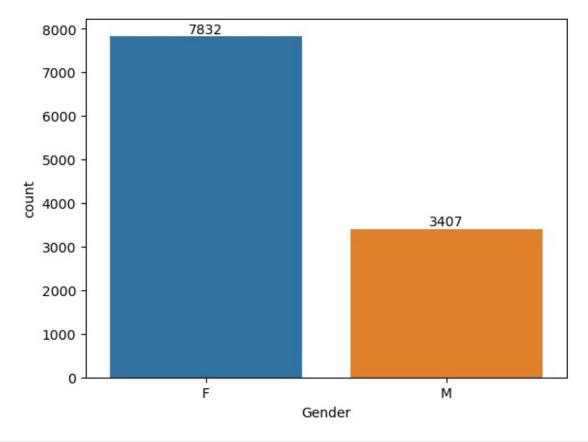
use describe() for specific columns
df[['Age', 'Orders', 'Amount']].describe()

	Age	0rders	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

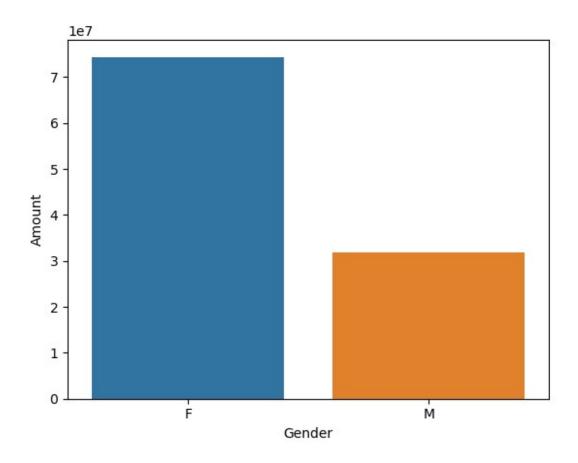
EXPLORATORY DATA ANALYSIS

GENDER

```
# plotting a bar chart for Gender and it's count
ax = sns.countplot(x = 'Gender', data = df)
for bars in ax.containers:
    ax.bar_label(bars)
```



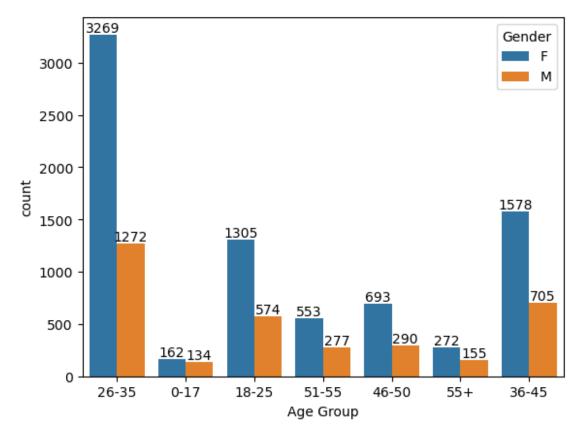
```
# plotting a bar chart for gender vs total amount
sales_gender = df.groupby(['Gender'], as_index=False)
['Amount'].sum().sort_values(by='Amount', ascending=False)
sns.barplot(x = 'Gender',y= 'Amount' ,data = sales_gender)
<Axes: xlabel='Gender', ylabel='Amount'>
```



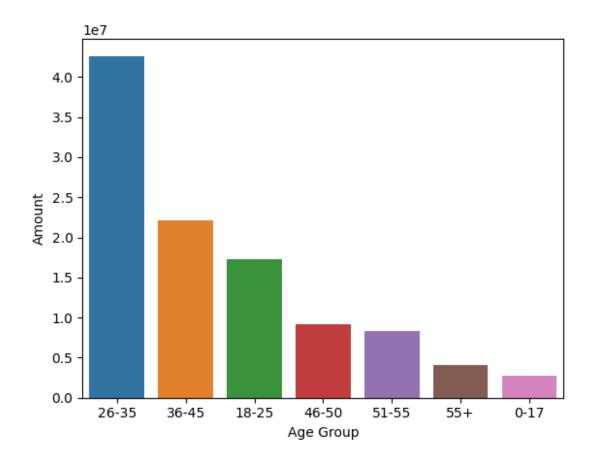
From above graphs we can see that most of the buyers are females and even the purchasing power of females are greater than men

AGE

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



```
# Total Amount vs Age Group
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)
<Axes: xlabel='Age Group', ylabel='Amount'>
```

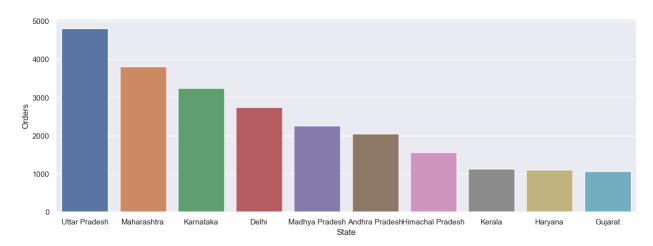


From above graphs we can see that most of the buyers are of age group between 26-35 yrs female

STATE

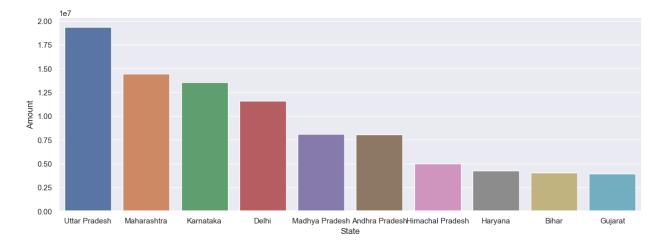
```
# total number of orders from top 10 states
sales_state = df.groupby(['State'], as_index=False)
['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)
sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(data = sales_state, x = 'State',y= 'Orders')

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



```
# total amount/sales from top 10 states
sales_state = df.groupby(['State'], as_index=False)
['Amount'].sum().sort_values(by='Amount', ascending=False).head(10)
sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(data = sales_state, x = 'State',y= 'Amount')

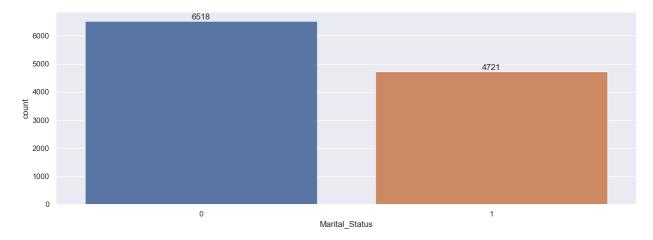
<Axes: xlabel='State', ylabel='Amount'>
```



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

MARITAL STATUS

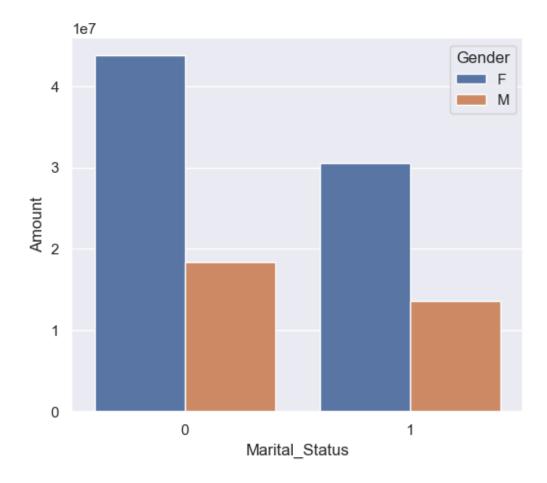
```
ax = sns.countplot(data = df, x = 'Marital_Status')
sns.set(rc={'figure.figsize':(7,5)})
for bars in ax.containers:
    ax.bar_label(bars)
```



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

sns.set(rc={'figure.figsize':(6,5)})
sns.barplot(data = sales_state, x = 'Marital_Status',y= 'Amount', hue='Gender')

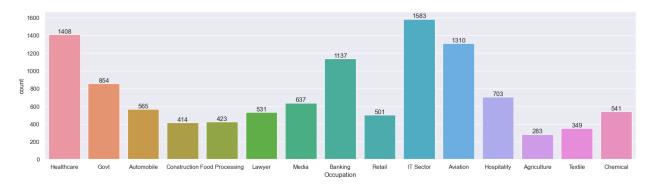
<Axes: xlabel='Marital_Status', ylabel='Amount'>
```



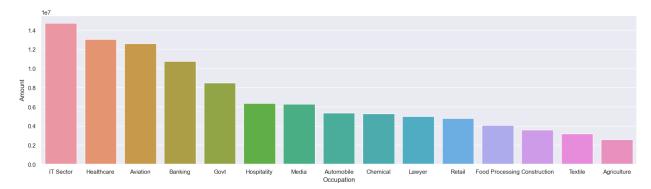
From above graphs we can see that most of the buyers are married (women) and they have high purchasing power

OCCUPATION

```
sns.set(rc={'figure.figsize':(20,5)})
ax = sns.countplot(data = df, x = 'Occupation')
for bars in ax.containers:
    ax.bar_label(bars)
```



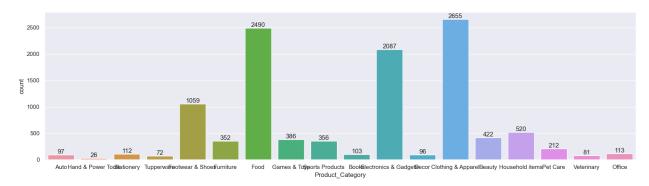
```
sales_state = df.groupby(['Occupation'], as_index=False)
['Amount'].sum().sort_values(by='Amount', ascending=False)
sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Occupation',y= 'Amount')
<Axes: xlabel='Occupation', ylabel='Amount'>
```



From above graphs we can see that most of the buyers are working in IT, Healthcare and Aviation sector

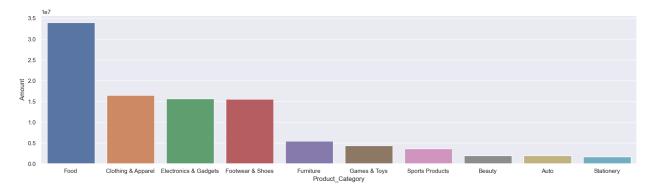
PRODUCT CATEGORY

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



```
sales_state = df.groupby(['Product_Category'], as_index=False)
['Amount'].sum().sort_values(by='Amount', ascending=False).head(10)
sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Product_Category',y= 'Amount')

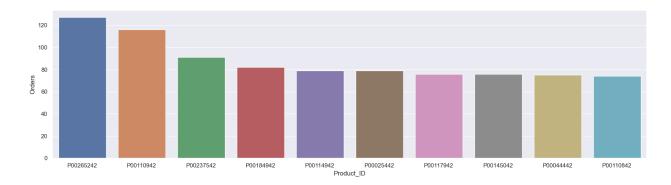
<Axes: xlabel='Product_Category', ylabel='Amount'>
```



From above graphs we can see that most of the sold products are from Food, Clothing and Electronics category

```
sales_state = df.groupby(['Product_ID'], as_index=False)
['Orders'].sum().sort_values(by='Orders', ascending=False).head(10)
sns.set(rc={'figure.figsize':(20,5)})
sns.barplot(data = sales_state, x = 'Product_ID',y= 'Orders')

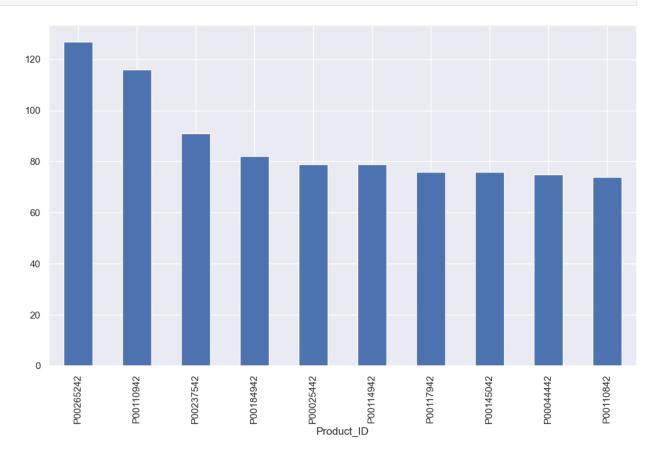
<Axes: xlabel='Product_ID', ylabel='Orders'>
```



top 10 most sold products

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

<Axes: xlabel='Product_ID'>



CONCLUSION:

Married women age group 26-35 yrs from UP, Maharastra and Karnataka working in IT, Healthcare and Aviation are more likely to buy products from Food, Clothing and Electronics category