<pre>importing the dataset f_train=pd.read_csv('train f_train User_ID Product_ID General 0 1000001 P00069042</pre>	der Age Occupation	0 A	Stay_In_Current_C	2	0	3	Category_2 Product_Ca	NaN 8370	
1 1000001 P00248942 2 1000001 P00087842 3 1000001 P00085442 4 1000002 P00285442 50063 1006033 P00372445 50064 1006035 P00375436 50065 1006036 P00375436 50066 1006038 P00375436 50067 1006039 P00371644 0068 rows × 12 columns	F 26-35 15 F 55+ 1	0 A 0 A 6 C 3 B 1 C		2 2 4+ 1 3 4+ 2 4+	0 0 0 1 0 1	1 12 12 8 20 20 20 20 20	6.0 NaN 14.0 NaN NaN NaN NaN NaN NaN	14.0 15200 NaN 1422 NaN 1057 NaN 7969 NaN 368 NaN 371 NaN 137 NaN 365 NaN 490	
#import thetest data df_test=pd.read_csv('test.cdf_test User_ID Product_ID General Description	M 46-50 7 M 26-35 17 F 36-45 1	7 В	Stay_In_Current_C	ity_Years Marit 2 0 4+ 4+ 1	tal_Status Product_Ca 1 0 1 1 0	tegory_1 Product_C 1 3 5 4	Category_2 Product_Ca 11.0 5.0 14.0 9.0 5.0	ntegory_3 NaN NaN NaN NaN NaN 12.0	
	F 26-35 15 F 26-35 15 F 26-35 15 F 46-50 1 F 46-50 0	5 B			 1 1 1 0 1	 8 5 1 10 4	 NaN 8.0 5.0 16.0 5.0	NaN NaN 12.0 NaN NaN	
### User_ID Product_ID General Control of Test User_ID Product_I	der Age Occupation F 0-17 10 F 0-17 10 F 0-17 10 F 0-17 10 M 55+ 16	0 A 0 A 0 A 0 A 0 A 0 C 5 B	Stay_In_Current_C	ity_Years Marit 2 2 2 2 4+ 4+ 4+	Product_Ca 0 0 0 0 0 1	tegory_1 Product_0 3 1 12 12 8 8	NaN 6.0 NaN 14.0 NaN NaN 8.0	NaN 8370.0 14.0 15200.0 NaN 1422.0 NaN 1057.0 NaN 7969.0 NaN NaN NaN	
33596 1006036 P00031842 33597 1006037 P00124742 33598 1006039 P00316642 33667 rows × 12 columns df.info() class 'pandas.core.frame.Dant64Index: 783667 entries, ata columns (total 12 colum# Column	F 26-35 15 F 46-50 1 F 46-50 0 ataFrame'> 0 to 233598 mns): Non-Null Cou	5 B 1 C 0 B		4+ 4+ 4+	1 0 1	1 10 4	5.0 16.0 5.0	12.0 NaN NaN NaN NaN NaN	
<pre>0 User_ID 1 Product_ID 2 Gender 3 Age 4 Occupation 5 City_Category 6 Stay_In_Current_City_Ye 7 Marital_Status 8 Product_Category_1 9 Product_Category_2 10 Product_Category_3 11 Purchase types: float64(3), int64(4) emory usage: 77.7+ MB</pre>	783667 non-n 783667 non-n 537685 non-n 237858 non-n 550068 non-n	null object null object null object null int64 null object null object null int64 null int64 null int64 null float64							
User_ID Occupation ount 7.836670e+05 783667.000000 nean 1.003029e+06 8.079300 std 1.727267e+03 6.522200 min 1.000001e+06 0.0000000 25% 1.001519e+06 2.0000000 50% 1.003075e+06 7.0000000 75% 1.004478e+06 14.0000000 max 1.006040e+06 20.0000000000000000000000000000000000	0 0.409777 6 0.491793 0 0.000000 0 0.000000 0 0.000000 0 1.000000	duct_Category_1 P 783667.000000 5.366196 3.878160 1.000000 1.000000 5.000000 8.000000 20.000000	537685.00000 9.84450 5.08909 2.00000 5.00000 9.00000 15.00000	0 237858 6 12 3 4 0 3 0 9 0 14 0 16	egory_3 Purchas .000000 550068.00000 .668605 9263.96871 .125510 5023.06539 .000000 12.00000 .000000 5823.00000 .000000 8047.00000 .000000 12054.00000 .000000 23961.00000	0 3 4 0 0 0			
df.drop('User_ID',axis=1,indhead() Product_ID Gender Age Occupo069042 F 0-17 P00248942 F 0-17 P00087842 F 0-17	upation City_Category 10 A 10 A 10 A	\ \	2 2 2	0 0 0	3 1 12	NaN 6.0 NaN	NaN 83 14.0 152 NaN 14	70.0 00.0 22.0	
P00085442 F 0-17 P00285442 M 55+ df.info() class 'pandas.core.frame.Dant64Index: 783667 entries, ata columns (total 11 column	0 to 233598	unt Dtype null object null object null object	2 4+	0	12 8	14.0 NaN		57.0	
5 Stay_In_Current_City_Ye 6 Marital_Status 7 Product_Category_1 8 Product_Category_2 9 Product_Category_3 10 Purchase types: float64(3), int64(3) emory usage: 71.7+ MB od.get_dummies(df['Gender']) F M 0 1 0 1 1 0	ears 783667 non-n 783667 non-n 783667 non-n 537685 non-n 237858 non-n 550068 non-n), object(5)	null object null int64 null int64 null float64 null float64							
2 1 0 3 1 0 4 0 1 33594 1 0 33595 1 0 33596 1 0 33597 1 0 33598 1 0									
#handling the categorical f df['Gender']=df['Gender'].m df.head() Product_ID Gender Age Occu P00069042 0 0-17 P00248942 0 0-17 P00087842 0 0-17	nap({'F':0,'M':1})	Stay_In_Current_0	City_Years Marita 2 2 2 2	al_Status Produ 0 0	act_Category_1 Production 3 1 1 12	ct_Category_2 Prod NaN 6.0 NaN	NaN 83 14.0 152	70.0	
P00085442 0 0-17 P00285442 1 55+ #handling categorical featured df['Age'].unique() rray(['0-17', '55+', '26-35', dtype=object) #df['Age']=df['Age'].map({'df['Age']=df['Age'].map({'0}, df['Age']=df['Age'].map({'0}, df['Age']=df['Age']].map({'0}, df['Age']=df['Age'].map({'0}, df['Age']=df['Age']].map({'0}, df['Age']=df['Age']=df['Age']].map({'0}, df['Age']=df['Age']=df['Age']].map({'0}, df['Age']=df['Age']=df['Age']=df['Age']].map({'0}, df['Age']=df['Age'	5', '46-50', '51-5 '0-17':1,'18-25':2	55', '36-45', '1 2,'26-35':3,'36-	45':4,'46-50'			14.0 NaN		57.0	
df['Age']=df['Age'].map({'0 df['Age'] df['Age'] 1 1 1 7 33594 3 33595 3 33596 3 33597 5 33598 5 ame: Age, Length: 783667, c	0-17':1,'18-25':2,								
df.info() class 'pandas.core.frame.Da nt64Index: 783667 entries, ata columns (total 11 colum # Column 0 Product_ID 1 Gender 2 Age 3 Occupation 4 City_Category 5 Stay_In_Current_City_Ye 6 Marital_Status 7 Product_Category_1	ataFrame'> 0 to 233598 mns): Non-Null Cou 783667 non-n	null object null int64 null int64 null int64 null object null object null int64 null int64							
7 Product_Category_1 8 Product_Category_2 9 Product_Category_3 10 Purchase types: float64(3), int64(5) emory usage: 71.7+ MB df.head() Product_ID Gender Age Occu P00069042 0 1 P00248942 0 1 P00087842 0 1	783667 non-n 537685 non-n 237858 non-n 550068 non-n), object(3)	null int64 null float64 null float64 null float64 null float64	City_Years Marita 2 2 2 2 2	I_Status Produ 0 0 0 0	ct_Category_1 Product 3 1 12 12	Ct_Category_2 Prod NaN 6.0 NaN 14.0	NaN 83 14.0 1520 NaN 142	70.0	
P00285442 1 7 df.info() class 'pandas.core.frame.Da nt64Index: 783667 entries, ata columns (total 11 colum # Column 0 Product_ID 1 Gender 2 Age 3 Occupation 4 City_Category	16 C ataFrame'> 0 to 233598 nns): Non-Null Cou 783667 non-n 783667 non-n 783667 non-n 783667 non-n 783667 non-n	unt Dtype null object null int64 null int64 null int64	2 4+	0	12 8	14.0 NaN		57.0	
	783667 non-nears 783667 non-n 783667 non-n 783667 non-n 537685 non-n 237858 non-n 550068 non-n), object(3)	null object null object null int64 null int64 null float64 null float64 null float64 null float64	City_Years Marita 2 2	I_Status Produ 0 0	ct_Category_1 Produc 3 1	ct_Category_2 Prode NaN 6.0		70.0	
P00087842	10 A 10 A 10 C		2 2 4+	0 0 0 0	1 12 12 8	6.0 NaN 14.0 NaN	NaN 142 NaN 109	00.0 22.0 57.0 69.0	
df_city=pd.get_dummies(df[' df_city.head() B C 0 0 0 0 0 0 0 0 0 1	CITY_Category'], di	υp_first =True)							
#drop city category Product_ID Gender Age Occu P00069042 0 1 P00248942 0 1 P00087842 0 1 P00285442 1 7	10 A 10 A 10 A 10 A 10 A 10 A		City_Years Marita 2 2 2 2 2 4+	I_Status Produ 0 0 0 0 0	ct_Category_1 Product 3 1 12 12 8	NaN 6.0 NaN 14.0 NaN	NaN 83 14.0 1520 NaN 142 NaN 109	70.0	
#df.drop('City_Category',ax #df.head() df.isnull().sum() roduct_ID ender ge ccupation ity_Category	o 0 0 0 0								
tay_In_Current_City_Years arital_Status roduct_Category_1 roduct_Category_2 roduct_Category_3 urchase type: int64 #focus on replacing the mis df['Product_Category_2'].un rray([nan, 6., 14., 2.,	nique() 8., 15., 16., 11. 18.]) nique()								
rray([3, 1, 12, 8, 5, 9, 20, 19], dtype=ir df['Product_Category_2'].va .0 91317 4.0 78834 .0 70498 6.0 61687 5.0 54114 .0 37165 .0 36705 .0 23575 1.0 20230 7.0 19104	4, 2, 6, 14, 11 nt64)	1, 13, 15, 7, 1	6, 18, 10, 17						
7.0 19104 3.0 15054 .0 8177 2.0 7801 0.0 4420 .0 4123 8.0 4027 .0 854 ame: Product_Category_2, dt #replacing the missing valu df['Product_Category_2']=df	ues with null	y_2'].fillna(df	['Product_Cat	egory_2'].mo	de()[0])				
.0 df.isnull().sum() roduct_ID ender ge ccupation ity_Category tay_In_Current_City_Years arital_Status roduct_Category_1 roduct_Category_2 roduct_Category_3	0 0 0 0 0 0 0 0 0								
roduct_Category_3 urchase type: int64 df['Product_Category_3'].is	233599 snull()								
33598 True ame: Product_Category_3, Le df['Product_Category_3'].va 6.0		/pe: bool							
8.0 6621 .0 2691 1.0 2585 9.0 2501 .0 878 ame: Product_Category_3, dt df['Product_Category_3'].mo	ode()[0] df['Product_Categor	ry_3'].fillna(d	f['Product_Ca	tegory_3'].m	ode()[0])				
df['Product_Category_3'].is df.head() Product_ID Gender Age Occu P00069042 0 1 P00248942 0 1 P00087842 0 1 P00085442 0 1 P00285442 1 7	upation City_Category 10 A 10 A 10 A 10 A 10 A		City_Years Marita 2 2 2 2 2 4+	I_Status Produ 0 0 0 0 0	ct_Category_1 Product 3 1 12 12 12 8	8.0 8.0 6.0 8.0 14.0 8.0	16.0 83° 14.0 1520	70.0 00.0 22.0 57.0	
df.shape 783667, 11) df.info() class 'pandas.core.frame.Dant64Index: 783667 entries, ata columns (total 11 colum # Column 0 Product_ID	ataFrame'> 0 to 233598 mns): Non-Null Cou 783667 non-n	int Dtype 	4+	0	8	8.0	16.0 796		
1 Gender 2 Age 3 Occupation 4 City_Category 5 Stay_In_Current_City_Ye 6 Marital_Status 7 Product_Category_1 8 Product_Category_2 9 Product_Category_3 10 Purchase types: float64(3), int64(5) emory usage: 71.7+ MB	783667 non-n 550068 non-n (), object(3)	null int64 null int64 null int64 null object null object null int64 null int64 null float64 null float64 null float64							
e treated as literal string df['Stay_In_Current_City_Yodf.head() Product_ID Gender Age Occu P00069042 0 1 P00248942 0 1	ears']=df['Stay_In_emp/ipykernel_9276 gs when regex=True Years']=df['Stay_I upation City_Category 10 A 10 A	Current_City_Y 6/2063355665.py: cn_Current_City_ Stay_In_Current_C	1: FutureWarn Years'].str.r City_Years Marita 2 2	ing: The def eplace('+',' I_Status Produ 0 0	ault value of reg ') ct_Category_1 Product 3 1	ct_Category_2 Prod 8.0 6.0	uct_Category_3 Purch 16.0 83 14.0 1520	70.0 00.0	acter regular exp
P00087842 0 1 P00085442 0 1 P00285442 1 7 #convert object into integer of ['Stay_In_Current_City_Yeddf.info()] class 'pandas.core.frame.Dant64Index: 783667 entries, ata columns (total 11 columns)	10 A 10 A 10 C er ears']=df['Stay_In_ ataFrame'> 0 to 233598 mns):	_Current_City_Y	2 2 4	0 0 0	1 12 12 8	6.0 8.0 14.0 8.0	16.0 142 16.0 109	00.0 22.0 57.0 59.0	
ata columns (total 11 columns Columns Columns Columns Columns Product_ID Gender Age City_Category Stay_In_Current_City_Yes Marital_Status Product_Category_1 Product_Category_2 Product_Category_3 Product_Category_3 Purchase Product_Status Purchase Product_Status Product_Category_3 Product_Category_3 Product_Category_3 Purchase Product_Category_3 Purchase Purchas	Non-Null Cou 	null object null int64 null int64 null int64 null object null int32 null int64 null int64 null int64 null float64 null float64 null float64							
Product_ID Gender Age Occu P00069042 0 1 P00248942 0 1 P00087842 0 1 P00085442 0 1 P00285442 1 7	10 A 10 A 10 A 10 A 10 A 10 A		City_Years Marita 2 2 2 2 4	I_Status Produ 0 0 0 0 0	ct_Category_1 Product 3 1 12 12 8	8.0 6.0 8.0 14.0 8.0	16.0 83 14.0 1520 16.0 142 16.0 109	70.0	
her arguments without an ex	e',hue='Gender',da ite-packages\seabo xplicit keyword wi	orn_decorators. .ll result in an	py:36: Future error or mis	Warning: Pas interpretati	s the following v on.	ariables as keyv	vord args: x, y. F	rom version 0.12, the only valid positional ar	gument will be `da
warnings.warn(AxesSubplot:xlabel='Age', y		7							
AxesSubplot:xlabel='Age', y	4 5 6 Age f men is higher tha	an women							gument will be `da
AxesSubplot:xlabel='Age', y 10000 8000 6000 2000 Gender 0 1 2 2 2 4 4 4 4 4 4 4 4 4 5 6 6 6 7 7 8 6 6 7 8 7 8 8 8 8 8 8 8 8 8 8	eion Purchase', hue='Geno ite-packages\seabo xplicit keyword wi	der',data=df) orn_decorators. ill result in an	py:36: Future error or mis	Warning: Pas interpretati	s the following v on.	<mark>ariables as key</mark> v	vord args: x, y. F	rom version 0.12, the only valid positional ar	
AxesSubplot:xlabel='Age', y 10000 8000 4000 2000 Gender 2000 4000 2000 Wizualization with occupate Sns.barplot('Occupation', 'P :\Users\Hp\anaconda3\lib\si her arguments without an ex warnings.warn(AxesSubplot:xlabel='Occupat 10000 8000 8000	Purchase', hue='Gendite-packages\seaboxplicit keyword wition', ylabel='Purburburburburburburburburburburburburbu	der', data=df) orn_decorators. ill result in an chase'> hue='Gender', da orn_decorators.	ta=df) py:36: Future	interpretati	s the following v			rom version 0.12, the only valid positional ar	
AxesSubplot:xlabel='Age', y 10000 8000 4000 2000 Gender 4000 2000 Gender 4vizualization purchasing of 4vizualization with occupate sns.barplot('Occupation', 'P :\Users\Hp\anaconda3\lib\si her arguments without an ex warnings.warn(AxesSubplot:xlabel='Occupate 10000 8000 6000	ender o la	der', data=df) orn_decorators. Il result in an chase'> hue='Gender', da orn_decorators. Il result in an	ta=df) py:36: Future error or mis	interpretati	s the following v				