In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

SANITY NO 1

In [3]:
```
df=pd.read_csv("Credit Banking - 3.csv")
```

In [4]:
```
df.isnull().sum()
```

Out[4]:
```
Credit_card          5
Product_ID           0
P_CATEGORY           0
CONDTION             0
Brand                0
Price                0
Selling_price        0
Coupon_ID           20
Date                 0
Time                 0
GTIN                 0
MPN                  0
Merchant_name        0
M_ID                 0
Payment Method       0
Transaction ID       0
Return_ind           0
Return_date       8467
dtype: int64
```

In [5]:
```
df['Credit_card'].fillna(9999,inplace=True)  ## sanity check 1
```

In [6]:
```
df.isnull().sum()
```

Out[6]:
```
Credit_card          0
Product_ID           0
P_CATEGORY           0
CONDTION             0
Brand                0
Price                0
Selling_price        0
Coupon_ID           20
Date                 0
Time                 0
GTIN                 0
MPN                  0
Merchant_name        0
M_ID                 0
Payment Method       0
Transaction ID       0
Return_ind           0
Return_date       8467
dtype: int64
```

SANITY NO 2

In [24]:
```
d=ts[ts['Price ']==ts['Selling_price']]
```

In [25]:
```
d.shape
```

Out[25]:
```
(41, 19)
```

In [26]:
```
d[d['Coupon_ID'].notnull()]
```

Out[26]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_pric
180	4852.0	765	DECOR	New	ZACKV	$4,557.18	$4,557.1
485	7979.0	534	SHOES	New	RKFCM	$1,484.74	$1,484.7
496	9305.0	838	BEDDING	Refurbished	CJUEU	$1,980.83	$1,980.8
511	3768.0	912	APPLIANCES	New	AVMAI	$3,332.40	$3,332.4
512	5008.0	535	KITCHEN & DINING	New	HLICV	$4,990.97	$4,990.9
524	7848.0	145	CLOTHING	New	ZVYTC	$3,631.77	$3,631.7

In [13]:
```
d['Coupon_ID'].isnull().sum()
```

Out[13]:
```
0
```

In [27]:
```
d['Selling_price']=d['Selling_price'].str.replace('$','')
```

In [28]:
```
d['Selling_price']=d['Selling_price'].str.replace(',','')
```

In [29]:
```
d['Price ']=d['Price '].str.replace('$','')
```

In [30]:
```
d['Price ']=d['Price '].str.replace(' ','')
```

In [31]:
```
d['Price ']=d['Price '].str.replace(',','')
```

In [32]:
```
d['Selling_price']=d['Selling_price'].astype('float64')
```

In [33]:
```
d['Price ']=d['Price '].astype('float64')
```

In [35]:
```
d['Discount']=d['Price ']*(5/100)
```

In [36]:
```
d.insert(loc=8,column='Dis',value=d['Discount'])
```

In [37]:
```
d.drop(columns='Discount',inplace=True)
```

In [38]:
```
d.rename(columns={'Dis':'Discount'},inplace=True)
```

In [41]:
```
d #### sanity check no.2
```

Out[41]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price	C
180	4852.0	765	DECOR	New	ZACKV	4557.18	4557.18	
485	7979.0	534	SHOES	New	RKFCM	1484.74	1484.74	
496	9305.0	838	BEDDING	Refurbished	CJUEU	1980.83	1980.83	
511	3768.0	912	APPLIANCES	New	AVMAI	3332.40	3332.40	
512	5008.0	535	KITCHEN & DINING	New	HLICV	4990.97	4990.97	
524	7848.0	145	CLOTHING	New	ZVYTC	3631.77	3631.77	
525	2024.0	111	GAMES	New	EITXF	4258.22	4258.22	
526	3500.0	644	DECOR	Used	ZACKV	4514.96	4514.96	
559	1174.0	931	BEDDING	New	CJUEU	4083.10	4083.10	
846	7173.0	594	CLOTHING	New	ONKHS	3715.86	3715.86	
847	9563.0	295	COMPUTERS	Used	DCJRW	622.19	622.19	
882	2377.0	119	COMPUTERS	New	YXQFM	901.26	901.26	
883	8299.0	635	APPLIANCES	Refurbished	AVMAI	2294.19	2294.19	
884	7173.0	956	COMPUTERS	New	YEJZI	3158.59	3158.59	
885	7631.0	539	ELECTRONICS	New	OULOW	668.11	668.11	
886	3199.0	912	SHOES	New	VKKEA	2230.61	2230.61	
931	2412.0	662	BABY CLOTHING	New	CTAOI	4658.93	4658.93	
932	2637.0	752	OFFICE SUPPLIES	Refurbished	EVWJY	4802.17	4802.17	
933	3970.0	381	GAMES	New	HWAVW	751.39	751.39	
935	4238.0	670	GAMES	Used	KJOGG	2461.84	2461.84	

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price	C
16	3833.0	763	DECOR	Refurbished	LBVIW	4943.44	4943.44	
27	4879.0	552	SHOES	New	RKFCM	1257.44	1257.44	
77	6286.0	960	ELECTRONICS	New	CWTLA	3994.10	3994.10	
127	3613.0	484	OFFICE SUPPLIES	Used	WELZA	2205.71	2205.71	
177	3768.0	838	LUGGAGE	Used	IWPJG	379.56	379.56	
178	2304.0	956	SHOES	Used	DYHNA	2800.18	2800.18	
179	3040.0	830	BABY CLOTHING	Used	TPDLE	4489.47	4489.47	
189	8067.0	912	SHOES	New	VKKEA	2230.61	2230.61	
190	3966.0	775	ELECTRONICS	Used	BEFJD	3061.88	3061.88	
204	2172.0	849	ELECTRONICS	Refurbished	FMBZC	3697.22	3697.22	
205	7856.0	816	DECOR	New	DGWZG	1050.51	1050.51	
206	9262.0	484	OFFICE SUPPLIES	Used	WELZA	2205.71	2205.71	
220	7075.0	554	SHOES	New	RKFCM	3461.72	3461.72	
221	9103.0	470	APPLIANCES	New	TTNTN	1187.85	1187.85	
222	2024.0	722	GAMES	New	KJOGG	1680.33	1680.33	
411	4139.0	657	BABY CLOTHING	Used	LATFL	3437.43	3437.43	
412	4878.0	762	KITCHEN & DINING	Refurbished	HLICV	335.54	335.54	
413	3598.0	715	BABY CLOTHING	Used	TPDLE	2638.33	2638.33	
806	7496.0	616	OFFICE SUPPLIES	Used	EVWJY	1011.90	1011.90	
807	8914.0	616	BABY TOYS	Refurbished	BGYXR	378.28	378.28	

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price	C
808	5008.0	435	COMPUTERS	Refurbished	YXQFM	716.53	716.53	

SANITY NO 3

In [7]:
```
q=df[~df.duplicated('Transaction ID')]
```

In [8]:
```
q.shape
```

Out[8]: (1419, 18)

In [22]:
```
r=ts[ts['Return_date'].notnull()]
```

In [23]:
```
r['Return_date']=pd.to_datetime(r['Return_date'])
```

In [9]:
```
q['Date']=pd.to_datetime(q['Date'])
```

In [10]:
```
qq=q[q['Return_date'].notnull()]
```

In [11]:
```
qq.shape
```

Out[11]: (871, 18)

In [12]:
```
rr=q[q['Return_date'].isna()]
```

In [13]:
```
rr.shape
```

Out[13]: (548, 18)

In [14]:
```
qq['op']=qq['Date']<qq['Return_date']
```

In [15]:
```
ss=pd.concat([qq,rr],axis=0)
```

In [16]:
```
tp=qq[qq['op']==True]
```

In [17]:
```
st=ss[ss['op']==True]
```

In [18]:
```
ts=pd.concat([st,rr],axis=0)
```

In [21]:
```
ts ##3
```

Out[21]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
9	5974.0	470	LUGGAGE	Used	TNVON	$4,783.25	$4,764.25
28	7979.0	389	CLOTHING	New	GTFFL	$3,747.69	$3,704.69
73	6502.0	662	BABY CLOTHING	New	CTAOI	$4,658.93	$4,609.93
126	3199.0	484	OFFICE SUPPLIES	Used	WELZA	$2,205.71	$2,171.71
147	2744.0	702	APPLIANCES	Refurbished	BUDLO	$4,042.54	$4,027.54
...	...	...	...	...	...	...	...
828	9723.0	722	ELECTRONICS	New	OULOW	$3,469.59	$3,422.59
829	4852.0	396	BABY CLOTHING	Refurbished	LATFL	$2,043.07	$1,999.07
830	8639.0	552	SHOES	New	RKFCM	$1,257.44	$1,227.44
831	1960.0	622	BABY TOYS	New	GOZLI	$2,603.49	$2,563.49
832	2141.0	960	BEDDING	New	RKCPD	$2,882.99	$2,843.99

1175 rows × 19 columns

SANITY NO 4

In [303]:
```
a=ts[ts['Coupon_ID'].isna()]
```

In [304]:
```
a.shape
```

Out[304]:
```
(20, 19)
```

In [305]:
```
a['selling_price']=a['Price ']
```

In [306]:
```
a[['Price ','selling_price']]
```

Out[306]:

	Price	selling_price
377	$2,132.96	$2,132.96
445	$2,230.61	$2,230.61
505	$4,802.17	$4,802.17
5	$3,788.97	$3,788.97
23	$3,697.22	$3,697.22
37	$379.49	$379.49
61	$623.05	$623.05
85	$2,213.77	$2,213.77
105	$2,280.16	$2,280.16
134	$635.56	$635.56
166	$1,050.51	$1,050.51
201	$3,205.26	$3,205.26
237	$2,455.58	$2,455.58
272	$3,631.77	$3,631.77
298	$1,680.33	$1,680.33
323	$1,363.08	$1,363.08
353	$379.56	$379.56
399	$298.03	$298.03
417	$2,868.97	$2,868.97
418	$4,641.79	$4,641.79

In [307]:
```
a.drop(columns={'Selling_price'},inplace=True)
```

In [308]:
```
a.insert(loc=6,column='sel',value=a['selling_price'])
```

In [309]:
```
a.drop(columns='selling_price',inplace=True)
```

In [310]:
```
a.rename(columns={"sel":"Selling_price"},inplace=True)
```

In [311]:
```
a.insert(loc=7,column='Discount',value=0)
```

In [312]:
```
a['Discount'].replace(0,'-',inplace=True)
```

In [313]:
```
a     ### sanity check no 4
```

Out[313]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
377	4284	324	ELECTRONICS	New	AGSDS	$2,132.96	$2,132.96
445	7486	912	SHOES	New	VKKEA	$2,230.61	$2,230.61
505	7412	752	OFFICE SUPPLIES	Refurbished	EVWJY	$4,802.17	$4,802.17
5	4305	922	LUGGAGE	New	IWPJG	$3,788.97	$3,788.97
23	8532	849	ELECTRONICS	Refurbished	FMBZC	$3,697.22	$3,697.22
37	9275	775	BABY TOYS	Refurbished	CTAOI	$379.49	$379.49
61	6740	644	CLOTHING	Used	KXUWE	$623.05	$623.05
85	7412	827	KITCHEN & DINING	New	NSILF	$2,213.77	$2,213.77
105	7861	781	SHOES	New	VKKEA	$2,280.16	$2,280.16
134	6105	970	LUGGAGE	Refurbished	CENQJ	$635.56	$635.56
166	7001	816	DECOR	New	DGWZG	$1,050.51	$1,050.51
201	6105	986	COMPUTERS	New	YXQFM	$3,205.26	$3,205.26
237	7496	300	COMPUTERS	Used	YXQFM	$2,455.58	$2,455.58
272	8523	145	CLOTHING	New	ZVYTC	$3,631.77	$3,631.77
298	3723	722	GAMES	New	KJOGG	$1,680.33	$1,680.33
323	1093	151	COMPUTERS	New	DCJRW	$1,363.08	$1,363.08
353	3958	838	LUGGAGE	Used	IWPJG	$379.56	$379.56
399	9725	773	SHOES	Used	DYHNA	$298.03	$298.03
417	7735	295	COMPUTERS	New	YEJZI	$2,868.97	$2,868.97
418	8523	195	COMPUTERS	Refurbished	GCVNE	$4,641.79	$4,641.79

SANITY NO 5

In [42]:
```
df1=pd.read_csv('Credit Banking - 3part1.csv')
```

In [43]:
```
df1[df1['Age']<=18]
```

Out[43]:

	C_ID	Email	Name	Mobile_number	Gender	Age	City
28	7173.0	marie@hotmail.com	MARIE	9.936858e+09	F	18.0	Kansas City
38	9063.0	alan@gmail.com	ALAN	9.983136e+09	F	15.0	Dallas
48	3175.0	christine@hotmail.com	CHRISTINE	9.445081e+09	F	17.0	San Jose
52	6143.0	rebecca@yahoomail.com	REBECCA	9.236432e+09	F	17.0	San Francisco
54	9587.0	karen@hotmail.com	KAREN	9.446573e+09	F	15.0	San Francisco
63	6071.0	craig@hotmail.com	CRAIG	8.946727e+09	M	15.0	Phoenix
76	7122.0	shirley@yahoomail.com	SHIRLEY	9.222061e+09	F	15.0	Houston
105	5157.0	lisa@hotmail.com	LISA	9.804086e+09	F	17.0	San Diego
117	5974.0	judith@gmail.com	JUDITH	9.743625e+09	M	16.0	Austin
150	5069.0	philip@hotmail.com	PHILIP	9.155965e+09	F	15.0	Louisville
159	8772.0	don@hotmail.com	DON	9.834172e+09	M	17.0	Boston
170	7856.0	julie@yahoomail.com	JULIE	9.840073e+09	M	17.0	Las Vegas
197	9532.0	katherine@hotmail.com	KATHERINE	9.033151e+09	M	18.0	San Jose

In [44]:
```
df1['Age']=np.where(df1['Age']<=18,np.nan,df1['Age'])
```

In [45]:
```
df1[df1['Age']<=18]
```

Out[45]:

C_ID	Email	Name	Mobile_number	Gender	Age	City	State	Address

In [46]:
```
q=df1['Age'].mean(skipna=True)
```

In [47]:
```
q
```

Out[47]:
```
60.65405405405406
```

In [48]:
```
df1['Age'].fillna(q,inplace=True)
```

In [49]:
```
df1[df1['Age']==q]
```

Out[49]:

	C_ID	Email	Name	Mobile_number	Gender	Age	
28	7173.0	marie@hotmail.com	MARIE	9.936858e+09	F	60.654054	Kan
38	9063.0	alan@gmail.com	ALAN	9.983136e+09	F	60.654054	Da
48	3175.0	christine@hotmail.com	CHRISTINE	9.445081e+09	F	60.654054	San J
52	6143.0	rebecca@yahoomail.com	REBECCA	9.236432e+09	F	60.654054	Franci
54	9587.0	karen@hotmail.com	KAREN	9.446573e+09	F	60.654054	Franci
63	6071.0	craig@hotmail.com	CRAIG	8.946727e+09	M	60.654054	Phoe
76	7122.0	shirley@yahoomail.com	SHIRLEY	9.222061e+09	F	60.654054	Hous
105	5157.0	lisa@hotmail.com	LISA	9.804086e+09	F	60.654054	Di
117	5974.0	judith@gmail.com	JUDITH	9.743625e+09	M	60.654054	Au
150	5069.0	philip@hotmail.com	PHILIP	9.155965e+09	F	60.654054	Louis
159	8772.0	don@hotmail.com	DON	9.834172e+09	M	60.654054	Bos
170	7856.0	julie@yahoomail.com	JULIE	9.840073e+09	M	60.654054	Ve
197	9532.0	katherine@hotmail.com	KATHERINE	9.033151e+09	M	60.654054	San J
198	NaN	NaN	NaN	NaN	NaN	60.654054	N
199	NaN	NaN	NaN	NaN	NaN	60.654054	N
200	NaN	NaN	NaN	NaN	NaN	60.654054	N
201	NaN	NaN	NaN	NaN	NaN	60.654054	N

In [50]:
```
df1 #### sanity check no 5
```

Out[50]:

	C_ID	Email	Name	Mobile_number	Gender	Age	C
0	3768.0	eddie@yahoomail.com	EDDIE	9.045258e+09	M	83.000000	Louisv
1	4852.0	rose@hotmail.com	ROSE	8.834789e+09	F	87.000000	Kans C
2	1174.0	amy@yahoomail.com	AMY	9.557690e+09	F	31.000000	Sea
3	4807.0	clarence@gmail.com	CLARENCE	9.394398e+09	M	37.000000	Sea
4	9131.0	johnny@hotmail.com	JOHNNY	9.976624e+09	F	80.000000	Columb
...	...	...	...	...	...	...	
197	9532.0	katherine@hotmail.com	KATHERINE	9.033151e+09	M	60.654054	San Jo
198	NaN	NaN	NaN	NaN	NaN	60.654054	N
199	NaN	NaN	NaN	NaN	NaN	60.654054	N
200	NaN	NaN	NaN	NaN	NaN	60.654054	N
201	NaN	NaN	NaN	NaN	NaN	60.654054	N

202 rows × 9 columns

SANITY NO 6

In [52]:
```
df[df['Transaction ID'].duplicated()]
```

Out[52]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
564	8639.0	138	LUGGAGE	Refurbished	CENQJ	$3,968.70	$3,956.70
623	9174.0	145	BEDDING	New	PIUEM	$3,272.59	$3,243.59
954	2637.0	931	LUGGAGE	Used	CENQJ	$334.55	$288.55
1043	2637.0	119	OFFICE SUPPLIES	Refurbished	EVWJY	$3,130.29	$3,104.29
1118	3040.0	324	LUGGAGE	New	IWPJG	$4,178.16	$4,163.16
...	...	...	...	...	...	...	...
9994	8609.0	470	APPLIANCES	New	TTNTN	$1,187.85	$1,187.85
9995	9725.0	406	CLOTHING	Used	ONKHS	$2,202.20	$2,202.20
9996	1548.0	694	ELECTRONICS	New	AGSDS	$3,123.80	$3,079.80
9997	4878.0	552	SHOES	New	RKFCM	$1,257.44	$1,227.44
9998	8532.0	395	OFFICE SUPPLIES	New	WELZA	$2,909.68	$2,883.68

8580 rows × 18 columns

In [27]:
```
q=df[~df.duplicated('Transaction ID')]
```

In [55]:
```
ts ###sanity no 6
```

Out[55]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
9	5974.0	470	LUGGAGE	Used	TNVON	$4,783.25	$4,764.25
28	7979.0	389	CLOTHING	New	GTFFL	$3,747.69	$3,704.69
73	6502.0	662	BABY CLOTHING	New	CTAOI	$4,658.93	$4,609.93
126	3199.0	484	OFFICE SUPPLIES	Used	WELZA	$2,205.71	$2,171.71
147	2744.0	702	APPLIANCES	Refurbished	BUDLO	$4,042.54	$4,027.54
...	...	...	...	...	...	...	...
828	9723.0	722	ELECTRONICS	New	OULOW	$3,469.59	$3,422.59
829	4852.0	396	BABY CLOTHING	Refurbished	LATFL	$2,043.07	$1,999.07
830	8639.0	552	SHOES	New	RKFCM	$1,257.44	$1,227.44
831	1960.0	622	BABY TOYS	New	GOZLI	$2,603.49	$2,563.49
832	2141.0	960	BEDDING	New	RKCPD	$2,882.99	$2,843.99

1175 rows × 19 columns

TASK 1

In [56]:
```
ts['Selling_price']=ts['Selling_price'].str.replace('$','')
```

In [57]:
```
ts['Selling_price']=ts['Selling_price'].str.replace(',','')
```

In [59]:
```
ts[ts['Selling_price']=='(1.87)']=np.nan
```

In [60]:
```
ts['Selling_price']=ts['Selling_price'].astype('float64')
```

In [61]:
```
ts['Selling_price'].dtypes
```

Out[61]:
```
dtype('float64')
```

In [62]:
```
a=df1[(df1['Gender']=='M')]
```

In [63]:
```
a.shape
```

Out[63]: (93, 9)

In [64]:
```
def agee(val):
    if 17<val<=30:
        return 'Young Males'
    if 31<=val<=60:
        return 'Mid age Males'
    elif val>60:
        return 'Old Males'
```

In [65]:
```
a['Catg']=a['Age'].map(agee)
```

In [66]:
```
c=df1[(df1['Gender']=='F')]
```

In [67]:
```
c.shape
```

Out[67]: (105, 9)

In [68]:
```
def agee(val):
    if 17<val<=30:
        return 'Young Females'
    if 31<=val<=60:
        return 'Mid age Females'
    elif val>60:
        return 'Old Females'
```

In [69]:
```
c['category']=c['Age'].map(agee)
```

In [70]:
```
pdd=pd.concat([a,c],axis=0)
```

In [71]:
```
pdd.shape
```

Out[71]: (198, 11)

In [72]:
```
pdd['cattg']=pdd['Catg'].fillna('')+pdd['category'].fillna('')
```

In [73]:
```
pdd.isnull().sum()
```

Out[73]:
```
C_ID              0
Email             0
Name              0
Mobile_number     0
Gender            0
Age               0
City              0
State             0
Address           0
Catg            105
category         93
cattg             0
dtype: int64
```

In [74]:
```
pdd.rename(columns={'cattg':'Age_catg'},inplace=True)
```

In [75]:
```
pdd.drop(columns={'Catg','category'},inplace=True)
```

In [76]:
```
pdd ### TASK 1
```

Out[76]:

	C_ID	Email	Name	Mobile_number	Gender	Age	City	
0	3768.0	eddie@yahoomail.com	EDDIE	9.045258e+09	M	83.0	Louisville	
3	4807.0	clarence@gmail.com	CLARENCE	9.394398e+09	M	37.0	Seattle	Wa
5	3496.0	clifford@yahoomail.com	CLIFFORD	9.591889e+09	M	35.0	Seattle	Wa
9	2637.0	ann@hotmail.com	ANN	9.506618e+09	M	65.0	Louisville	
11	9896.0	troy@yahoomail.com	TROY	9.197303e+09	M	39.0	Kansas City	
...	...	...	...	...	...	...	...	
189	5229.0	helen@gmail.com	HELEN	9.064066e+09	F	60.0	Seattle	Wa
191	7496.0	chris@hotmail.com	CHRIS	9.626130e+09	F	72.0	Seattle	Wa
193	4575.0	tina@yahoomail.com	TINA	8.819353e+09	F	56.0	Austin	
195	8609.0	jennifer@hotmail.com	JENNIFER	9.935043e+09	F	93.0	Seattle	Wa
196	4542.0	michelle@hotmail.com	MICHELLE	9.730382e+09	F	60.0	Louisville	

198 rows × 10 columns

TASK 2

In [77]:
```
df2=pd.concat([ts,df1],axis=1)
```

In [78]:
```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1178 entries, 9 to 199
Data columns (total 28 columns):
 #   Column          Non-Null Count  Dtype         
---  ------          --------------  -----         
 0   Credit_card     1174 non-null   float64       
 1   Product_ID      1174 non-null   float64       
 2   P_CATEGORY      1174 non-null   object        
 3   CONDTION        1174 non-null   object        
 4   Brand           1174 non-null   object        
 5   Price           1174 non-null   object        
 6   Selling_price   1174 non-null   float64       
 7   Coupon_ID       1154 non-null   object        
 8   Date            1174 non-null   datetime64[ns]
 9   Time            1174 non-null   object        
 10  GTIN            1174 non-null   float64       
 11  MPN             1174 non-null   float64       
 12  Merchant_name   1174 non-null   object        
 13  M_ID            1174 non-null   object        
 14  Payment Method  1174 non-null   object        
 15  Transaction ID  1174 non-null   float64       
 16  Return_ind      1174 non-null   float64       
 17  Return_date     626 non-null    object        
 18  op              626 non-null    object        
 19  C_ID            198 non-null    float64       
 20  Email           198 non-null    object        
 21  Name            198 non-null    object        
 22  Mobile_number   198 non-null    float64       
 23  Gender          198 non-null    object        
 24  Age             202 non-null    float64       
 25  City            198 non-null    object        
 26  State           198 non-null    object        
 27  Address         198 non-null    object        
dtypes: datetime64[ns](1), float64(10), object(17)
memory usage: 266.9+ KB
```

In [79]:
```
q=df2.groupby(['P_CATEGORY','State','Payment Method'])['Selling_price'].
```

In [80]:
```
pq=q.reset_index()
```

In [81]:
```
pq.rename(columns={'Selling_price':'Total_spend'},inplace=True)
```

In [82]:
```
pq ####2
```

Out[82]:

	P_CATEGORY	State	Payment Method	Total_spend
0	APPLIANCES	California	Credit card	4891.58
1	APPLIANCES	California	Mobile carrier Billing	4966.87
2	APPLIANCES	California	Prepaid card	3196.65
3	APPLIANCES	Massachusetts	Credit card	2028.24
4	APPLIANCES	Massachusetts	Prepaid card	2109.85
...	...	...	...	...
133	SHOES	Texas	Credit card	4592.91
134	SHOES	Texas	Mobile carrier Billing	6232.90
135	SHOES	Texas	Prepaid card	2751.18
136	SHOES	Washington	Credit card	4284.02
137	SHOES	Washington	Mobile carrier Billing	2230.61

138 rows × 4 columns

TASK 3

In [83]:
```
df3=pd.concat([ts,pdd],axis=1)
```

In [84]:
```
df3.head(6)
```

Out[84]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price	C
9	5974.0	470.0	LUGGAGE	Used	TNVON	$4,783.25	4764.25	
28	7979.0	389.0	CLOTHING	New	GTFFL	$3,747.69	3704.69	
73	6502.0	662.0	BABY CLOTHING	New	CTAOI	$4,658.93	4609.93	
126	3199.0	484.0	OFFICE SUPPLIES	Used	WELZA	$2,205.71	2171.71	
147	2744.0	702.0	APPLIANCES	Refurbished	BUDLO	$4,042.54	4027.54	
164	1256.0	198.0	KITCHEN & DINING	Refurbished	RGMIK	$3,643.75	3607.75	

6 rows × 29 columns

In [86]:
```
q=df3.groupby(['Age_catg'])['Selling_price'].sum()
```

In [87]:
```
qq=q.reset_index()
```

In [88]:
```
qq.nlargest(5,'Selling_price')###3
```

Out[88]:

	Age_catg	Selling_price
2	Old Females	138406.78
3	Old Males	135582.36
0	Mid age Females	89337.38
1	Mid age Males	58639.10
4	Young Females	35911.55

TASK 4

In [112]:
```
r=ts[ts['Return_date'].notnull()]
```

In [113]:
```
r.isnull().sum()
```

Out[113]:
```
Credit_card       0
Product_ID        0
P_CATEGORY        0
CONDTION          0
Brand             0
Price             0
Selling_price     0
Coupon_ID         3
Date              0
Time              0
GTIN              0
MPN               0
Merchant_name     0
M_ID              0
Payment Method    0
Transaction ID    0
Return_ind        0
Return_date       0
op                0
dtype: int64
```

In [114]:
```
r
```

Out[114]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_pr
9	5974.0	470.0	LUGGAGE	Used	TNVON	$4,783.25	4764
28	7979.0	389.0	CLOTHING	New	GTFFL	$3,747.69	3704
73	6502.0	662.0	BABY CLOTHING	New	CTAOI	$4,658.93	4609
126	3199.0	484.0	OFFICE SUPPLIES	Used	WELZA	$2,205.71	2171
147	2744.0	702.0	APPLIANCES	Refurbished	BUDLO	$4,042.54	4027
164	1256.0	198.0	KITCHEN &	Refurbished	BCMIK	$3,643.75	3607

In [115]:
```
r[r['Return_date'].notnull()]
```

Out[115]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_pr
9	5974.0	470.0	LUGGAGE	Used	TNVON	$4,783.25	4764
28	7979.0	389.0	CLOTHING	New	GTFFL	$3,747.69	3704
73	6502.0	662.0	BABY CLOTHING	New	CTAOI	$4,658.93	4609
126	3199.0	484.0	OFFICE SUPPLIES	Used	WELZA	$2,205.71	2171
147	2744.0	702.0	APPLIANCES	Refurbished	BUDLO	$4,042.54	4027
164	1256.0	198.0	KITCHEN &	Refurbished	BCMIK	$3,643.75	3607

In [116]:
```
tt=pd.concat([r,pdd],axis=1)
```

In [117]:
```
p=tt[['P_CATEGORY','CONDTION','Return_date','Age_catg','State']]
```

In [118]:
```
p.columns
```

Out[118]:
```
Index(['P_CATEGORY', 'CONDTION', 'Return_date', 'Age_catg', 'State'], d
type='object')
```

In [119]:
```
p.rename(columns={"Return_date":"Return_Date"},inplace=True)
```

In [120]:
```
a=r[['Brand','Price ','Return_date']]
```

In [121]:
```
pa=pd.concat([p,a],axis=1)
```

In [122]:
```
pa[pa['Return_date'].notnull()]
```

Out[122]:

	P_CATEGORY	CONDTION	Return_Date	Age_catg	State	Brand	Price
9	LUGGAGE	Used	30-09-2014	Old Males	Kentucky	TNVON	$4,783.25
28	CLOTHING	New	24-01-2014	Old Females	Missouri	GTFFL	$3,747.69
73	BABY CLOTHING	New	10-10-2014	Old Females	Arizona	CTAOI	$4,658.93
126	OFFICE SUPPLIES	Used	23-02-2014	Young Males	California	WELZA	$2,205.71
147	APPLIANCES	Refurbished	11-02-2014	Old Males	California	BUDLO	$4,042.54
164	KITCHEN & DINING	Refurbished	11-02-2014	Old Males	Illinois	RGMIK	$3,643.75
180	DECOR	New	14-11-2014	Old Males	Kentucky	ZACKV	$4,557.18

In [123]:
```
e=pa[pa['Return_date'].notnull()]
```

In [124]:
```
e.shape
```

Out[124]:
```
(626, 8)
```

In [125]:
```
e.drop(columns='Return_Date',inplace=True)
```

In [126]:
```
e.drop(columns={'Price '},inplace=True)
```

In [131]:
```
e ### 4
```

Out[131]:

	P_CATEGORY	CONDTION	Age_catg	State	Brand	Return_date
9	LUGGAGE	Used	Old Males	Kentucky	TNVON	30-09-2014
28	CLOTHING	New	Old Females	Missouri	GTFFL	24-01-2014
73	BABY CLOTHING	New	Old Females	Arizona	CTAOI	10-10-2014
126	OFFICE SUPPLIES	Used	Young Males	California	WELZA	23-02-2014
147	APPLIANCES	Refurbished	Old Males	California	BUDLO	11-02-2014
164	KITCHEN & DINING	Refurbished	Old Males	Illinois	RGMIK	11-02-2014
180	DECOR	New	Old Males	Kentucky	ZACKV	14-11-2014
217	LUGGAGE	Used	NaN	NaN	CENQJ	04-02-2014
236	GAMES	New	NaN	NaN	KJOGG	09-02-2014
255	ELECTRONICS	Used	NaN	NaN	FMBZC	09-04-2014
285	KITCHEN & DINING	Used	NaN	NaN	ZGHYW	06-05-2014

TASK 5

In [89]:
```
ts['Time']=pd.to_datetime(ts['Time'])
```

In [90]:
```
ts['HR']=ts['Time'].dt.hour
```

In [91]:
```
ts['HR'].unique()
```

Out[91]:
```
array([ 5., 13., 21., 23., 22.,  4.,  2., 11.,  1., 16., 14.,  3.,  7.,
       17., 19.,  6.,  0.,  9., 10., 18., 12.,  8., 15., 20., nan])
```

In [92]:
```
ts['HR'].fillna(0,inplace=True)
```

In [93]:
```
ts['HR'].dtypes
```

Out[93]:
```
dtype('float64')
```

In [94]:
```
ts['HR']=ts['HR'].astype('int64')
```

In [96]:
```
ts['HR'].value_counts()
```

Out[96]:
```
11    68
3     59
7     57
8     57
17    56
12    56
18    54
0     54
23    52
22    52
16    51
14    51
21    49
20    48
13    46
1     46
9     44
5     44
10    43
2     39
19    38
4     38
15    37
6     36
Name: HR, dtype: int64
```

In [97]:
```
def tim(val):
    if 0<=val<=11:
        return 'Morning_customers'
    if 12<=val<=18:
        return 'Afternoon_customers'
    if 18<val<=23:
        return 'Night_customers'
```

In [98]:
```
ts['HR']=ts['HR'].map(tim)
```

In [99]:
```
ts['HR'].shape
```

Out[99]:
```
(1175,)
```

In [100]:
```
ts['HR'].value_counts()
```

Out[100]:
```
Morning_customers      585
Afternoon_customers    351
Night_customers        239
Name: HR, dtype: int64
```

In [101]:
```
ts.rename(columns={'HR':'Cust_Profile'},inplace=True)
```

In [104]:
```
ts ### TASK 5
```

Out[104]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
9	5974.0	470.0	LUGGAGE	Used	TNVON	$4,783.25	4764.25
28	7979.0	389.0	CLOTHING	New	GTFFL	$3,747.69	3704.69
73	6502.0	662.0	BABY CLOTHING	New	CTAOI	$4,658.93	4609.93
126	3199.0	484.0	OFFICE SUPPLIES	Used	WELZA	$2,205.71	2171.71
147	2744.0	702.0	APPLIANCES	Refurbished	BUDLO	$4,042.54	4027.54
...	...	...	...	...	...	...	...
828	9723.0	722.0	ELECTRONICS	New	OULOW	$3,469.59	3422.59
829	4852.0	396.0	BABY CLOTHING	Refurbished	LATFL	$2,043.07	1999.07
830	8639.0	552.0	SHOES	New	RKFCM	$1,257.44	1227.44
831	1960.0	622.0	BABY TOYS	New	GOZLI	$2,603.49	2563.49
832	2141.0	960.0	BEDDING	New	RKCPD	$2,882.99	2843.99

1175 rows × 14 columns

In [103]:
```
ts.drop(columns={'Time','GTIN','MPN','Merchant_name','M_ID','Return_ind'
```

TASK 6

In [105]:
```
o=d.groupby(['Payment Method'])['Discount'].sum().nlargest(1)
```

In [106]:
```
qu=o.reset_index()
```

In [107]:
```
qu  ### TASK
```

Out[107]:

	Payment Method	Discount
0	Mobile carrier Billing	2619.0285

TASK 7

In [109]:
```
ts['Price ']=ts['Price '].str.replace('$','')
```

In [110]:
```
ts['Price ']=ts['Price '].str.replace(',','')
```

In [111]:
```
ts['Price ']=ts['Price '].str.replace(' ','')
```

In [112]:
```
ts['Price ']=ts['Price '].astype('float64')
```

In [113]:
```
def price(val):
    if  val>3000 :
        return 'high_value_items'
    elif val <= 3000:
        return 'low_value_items'
```

In [114]:
```
ts['Price_val']=ts['Price '].map(price)
```

In [115]:
```
ts['Price_val'].value_counts()  ####7
```

Out[115]:
```
low_value_items     640
high_value_items    534
Name: Price_val, dtype: int64
```

In [ ]:
```
## NO OF ORDERS IS HIGH FOR LOW_VALUE_ITEMS
```

TASK 8

In [116]:
```
import matplotlib.pyplot as plt
```

In [117]:
```
d
```

Out[117]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
180	4852.0	765	DECOR	New	ZACKV	4557.18	4557.18
485	7979.0	534	SHOES	New	RKFCM	1484.74	1484.74
496	9305.0	838	BEDDING	Refurbished	CJUEU	1980.83	1980.83
511	3768.0	912	APPLIANCES	New	AVMAI	3332.40	3332.40
512	5008.0	535	KITCHEN & DINING	New	HLICV	4990.97	4990.97
524	7848.0	145	CLOTHING	New	ZVYTC	3631.77	3631.77

In [122]:
```
d['Count']=np.arange(d.shape[0])
```

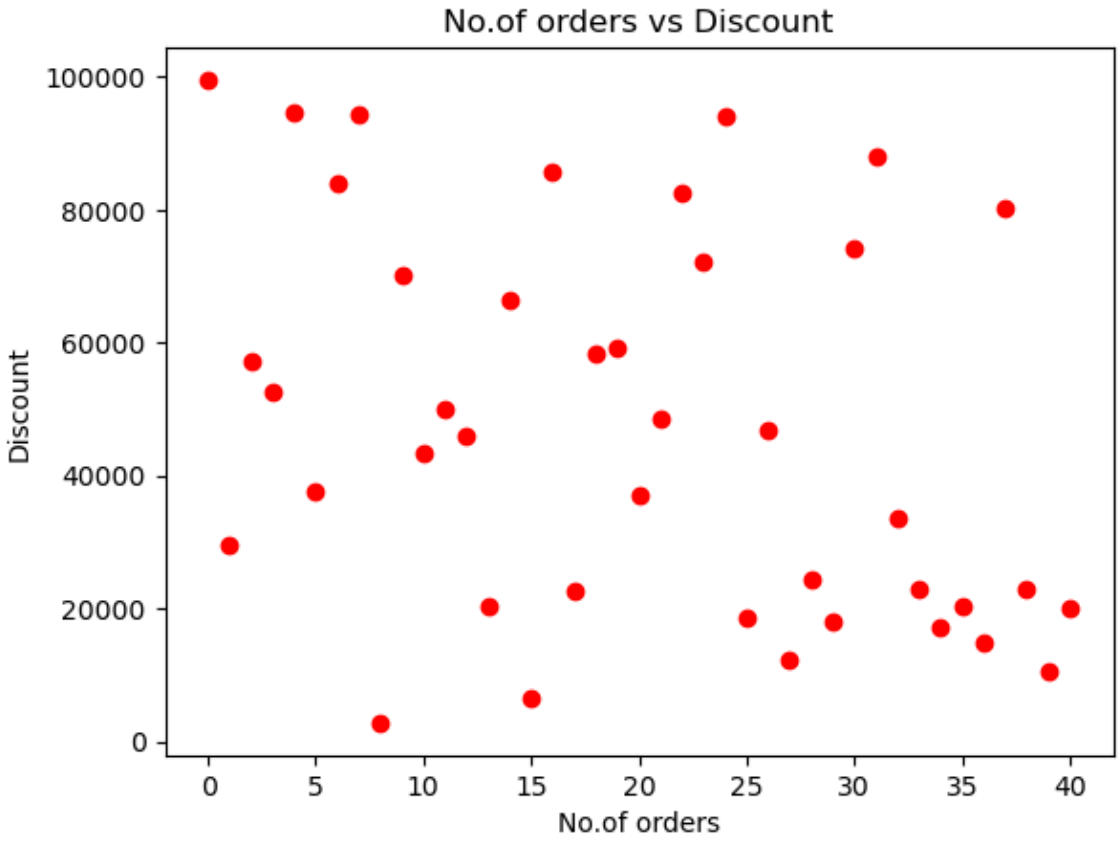
In [128]:
```
d
```

Out[128]:

	Credit_card	Product_ID	P_CATEGORY	CONDTION	Brand	Price	Selling_price
180	4852.0	765	DECOR	New	ZACKV	4557.18	4557.18
485	7979.0	534	SHOES	New	RKFCM	1484.74	1484.74
496	9305.0	838	BEDDING	Refurbished	CJUEU	1980.83	1980.83
511	3768.0	912	APPLIANCES	New	AVMAI	3332.40	3332.40
512	5008.0	535	KITCHEN & DINING	New	HLICV	4990.97	4990.97
524	7848.0	145	CLOTHING	New	ZVYTC	3631.77	3631.77

In [129]:
```
x=d['Count']
y=d['Transaction ID']
```

In [130]:
```
plt.scatter(x,y,c='r')
plt.xlabel('No.of orders')
plt.ylabel('Discount')
plt.title('No.of orders vs Discount')
plt.show()
```



In [ ]:
```
## As there is a random distribution between the axes so, we can say tha
```