Python Project-3 **by- Phanindra Bhushan Chaturvedi**

In this project we are provided with retail market data. Data consist of three csv files giving data of-

1.Customer details

2.Transaction details

3.Product category details

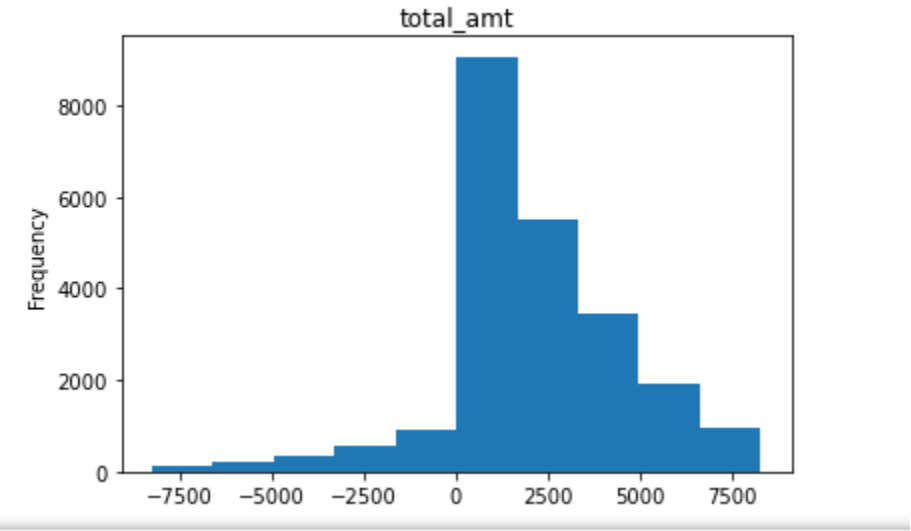
On working with the above csv files I analysed below mentioned points-

Business Insights

1.Data available is in the time period of- 02/01/2011 to 02/12/2014

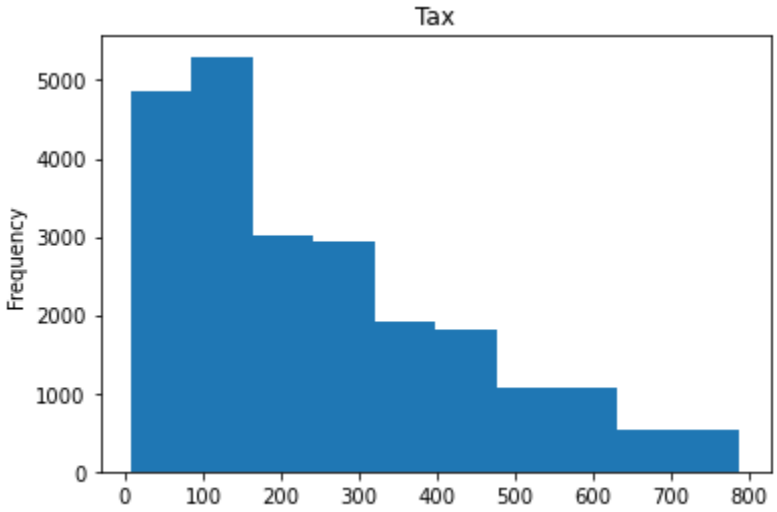
2.On creating Histogram I analysed

a.



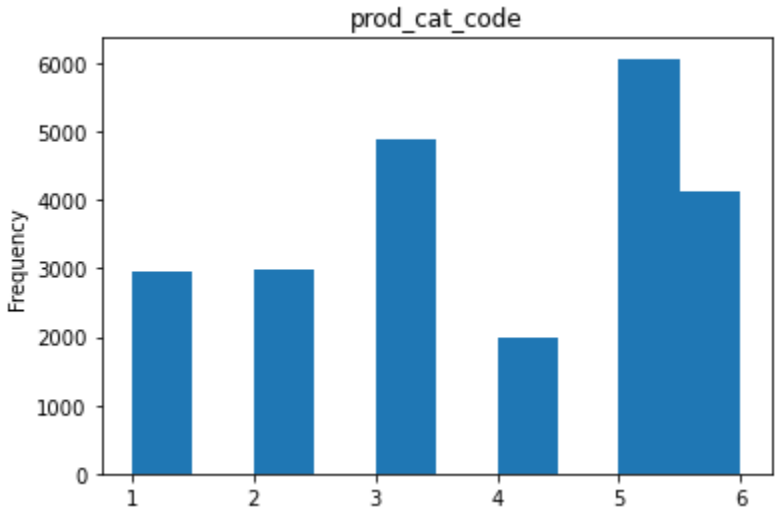
The frequency of total amount spent by customer is more in the range of 0-2500,also there are negative transactions which may be due to damaged product replacement , Returned item before being delivered, Returned item when it was in transit etc.

b.



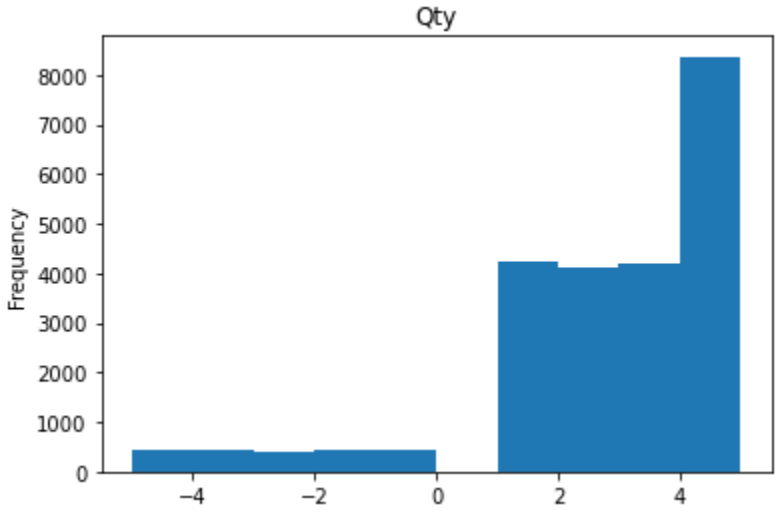
In the above histogram we can see the item which tax in between 0-200 are more frequently sold. It means low income group people more visit and daily essential objects are more sold as they are placed in low tax slab

c.



The more in demand product is of product category code 5 and least is for category code 4

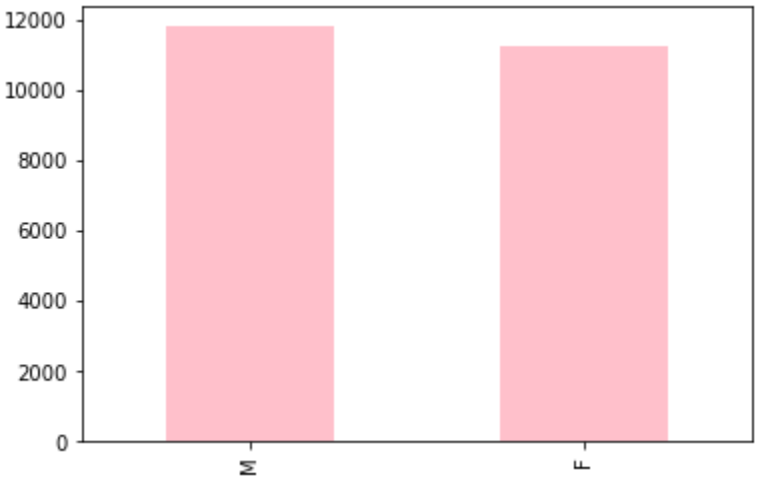
d.



From above graph we can say the frequency of buying 4 item together is highest

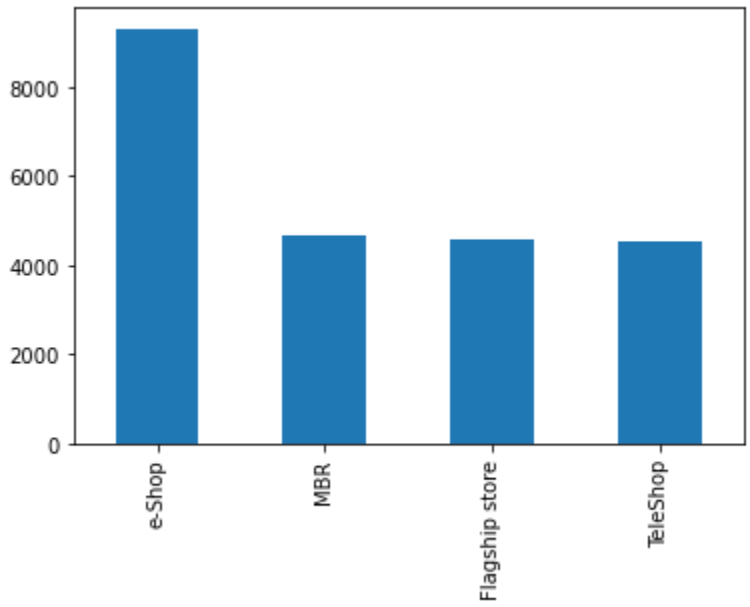
3.On creating Bar chart for Categorical data I found

a.



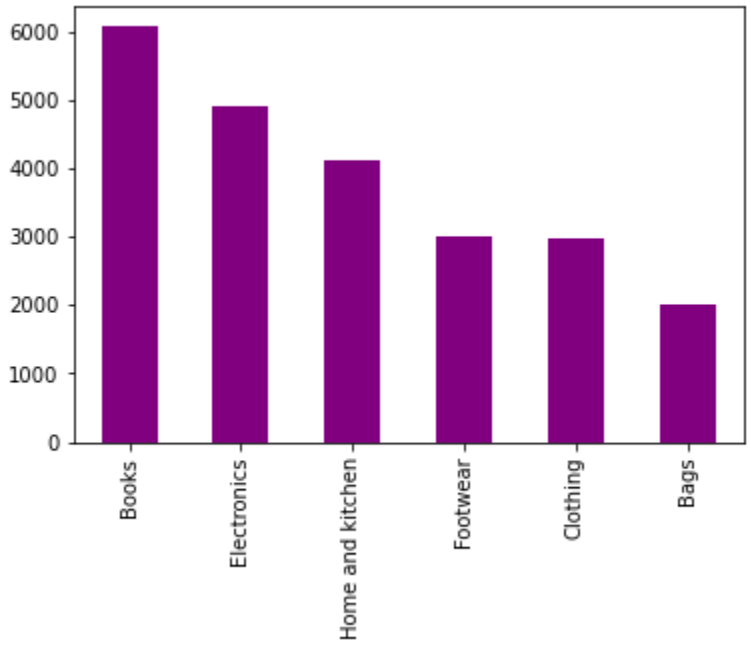
Male customers are more than female customer but there is only a slight difference between male and female customer as it can be clearly seen in above bar chart

b.



Among type of store e-shop has highest number of customers

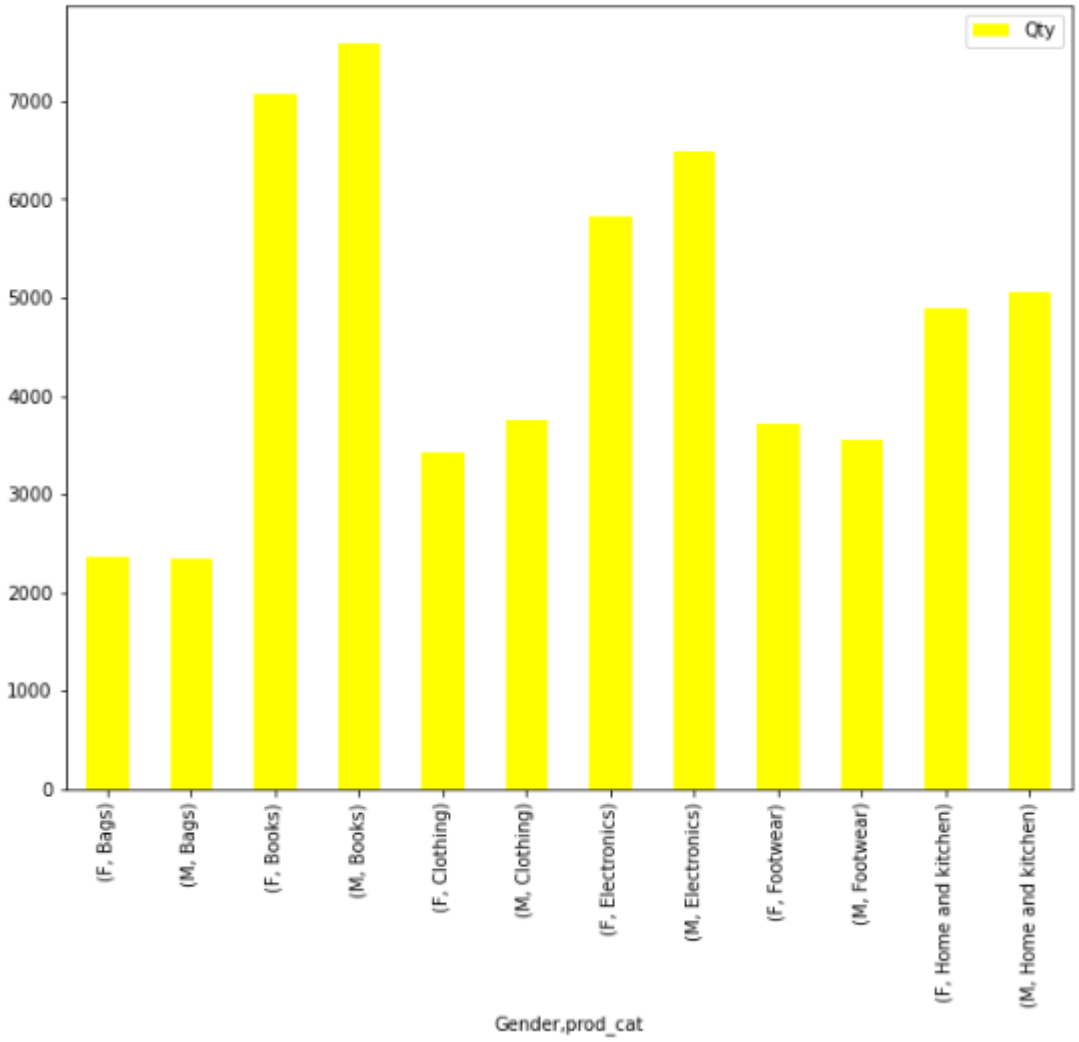
c.



Highest sold product category is of Books and least of Bags

4.On analysing I found that around 2164 transactions were negative transactions,it could be due to damaged product replacement , Returned item before being delivered, Returned item when it was in transit etc.

5.



From the above table we can say that Book is the category which highest sold among both male as well female

Also I analysed

#popular products in male are

Books

Clothing

Electronics

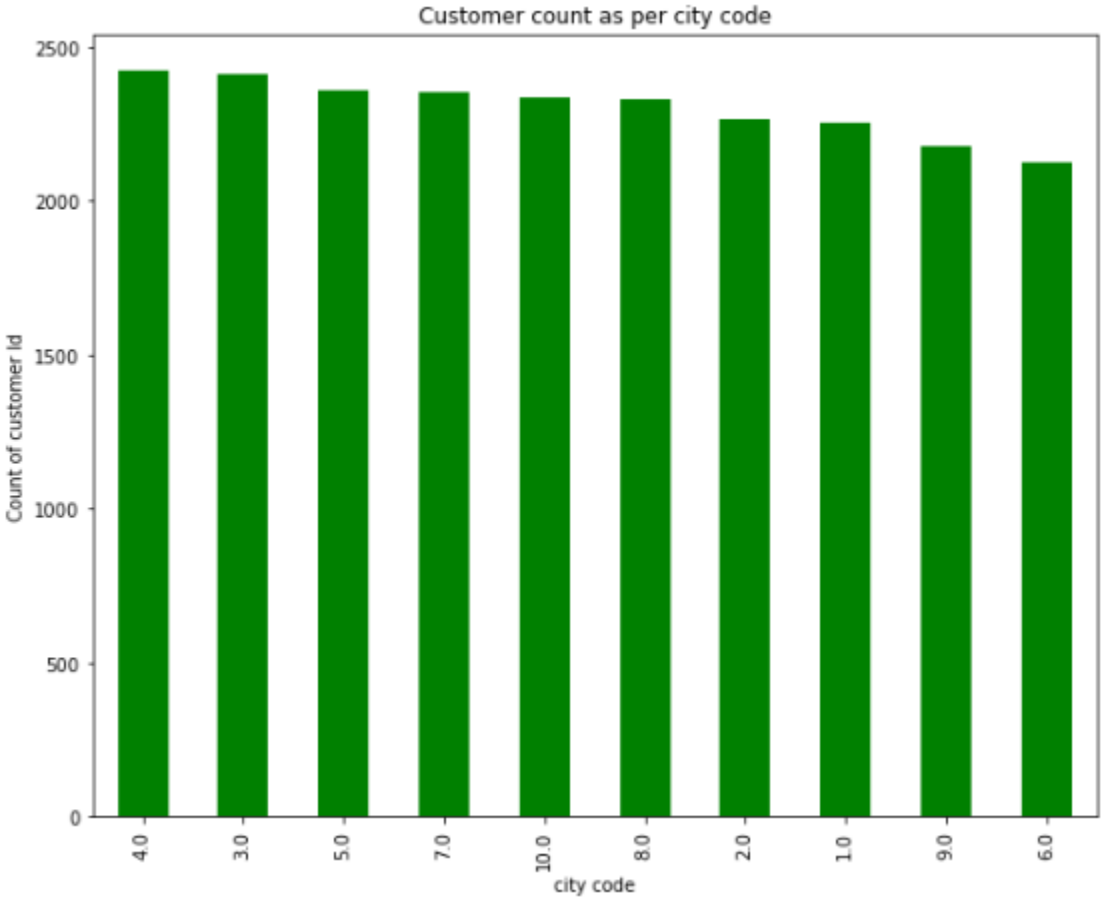
Home and Kitchen

#popular products in female are

Bags

Footwear

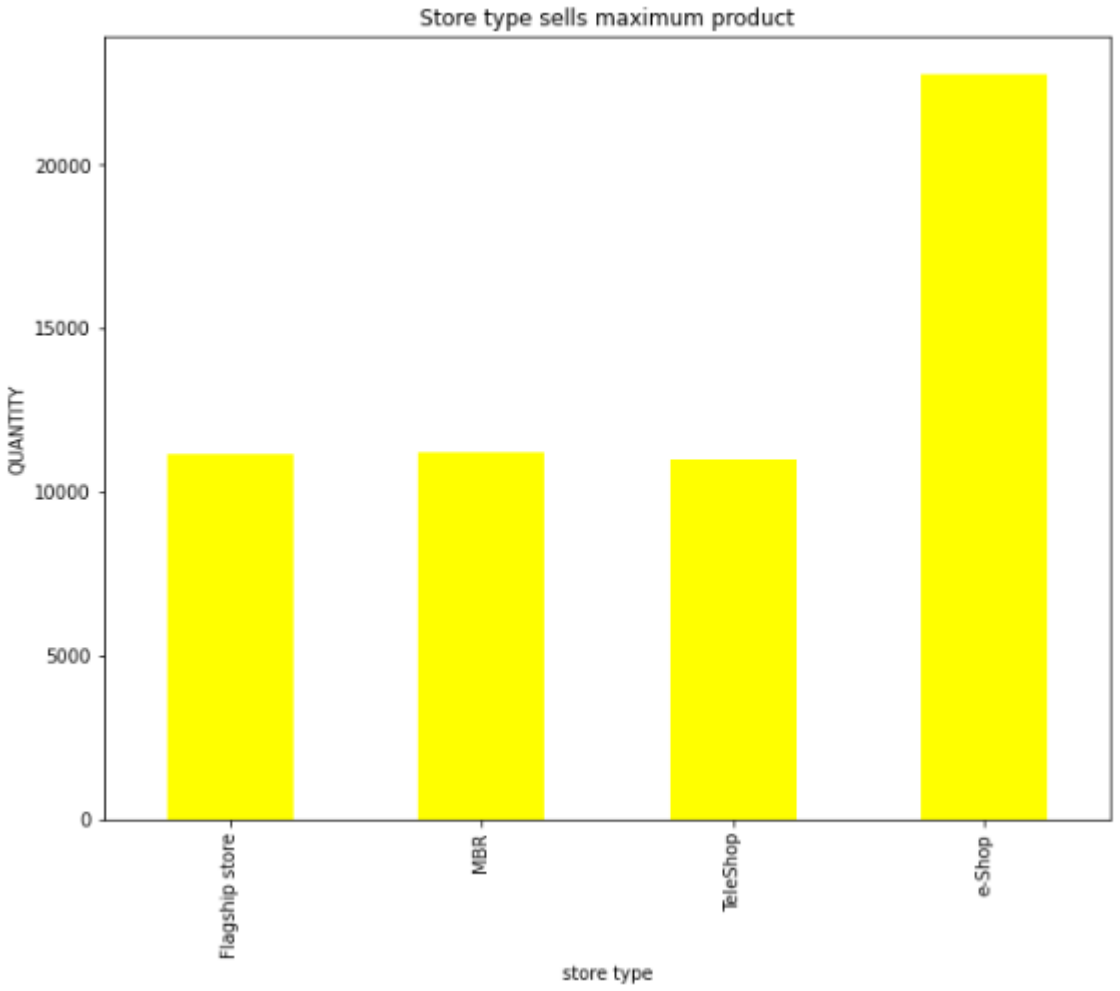
6.



From the above table we can say that number of customers are highest from the city code 4.0

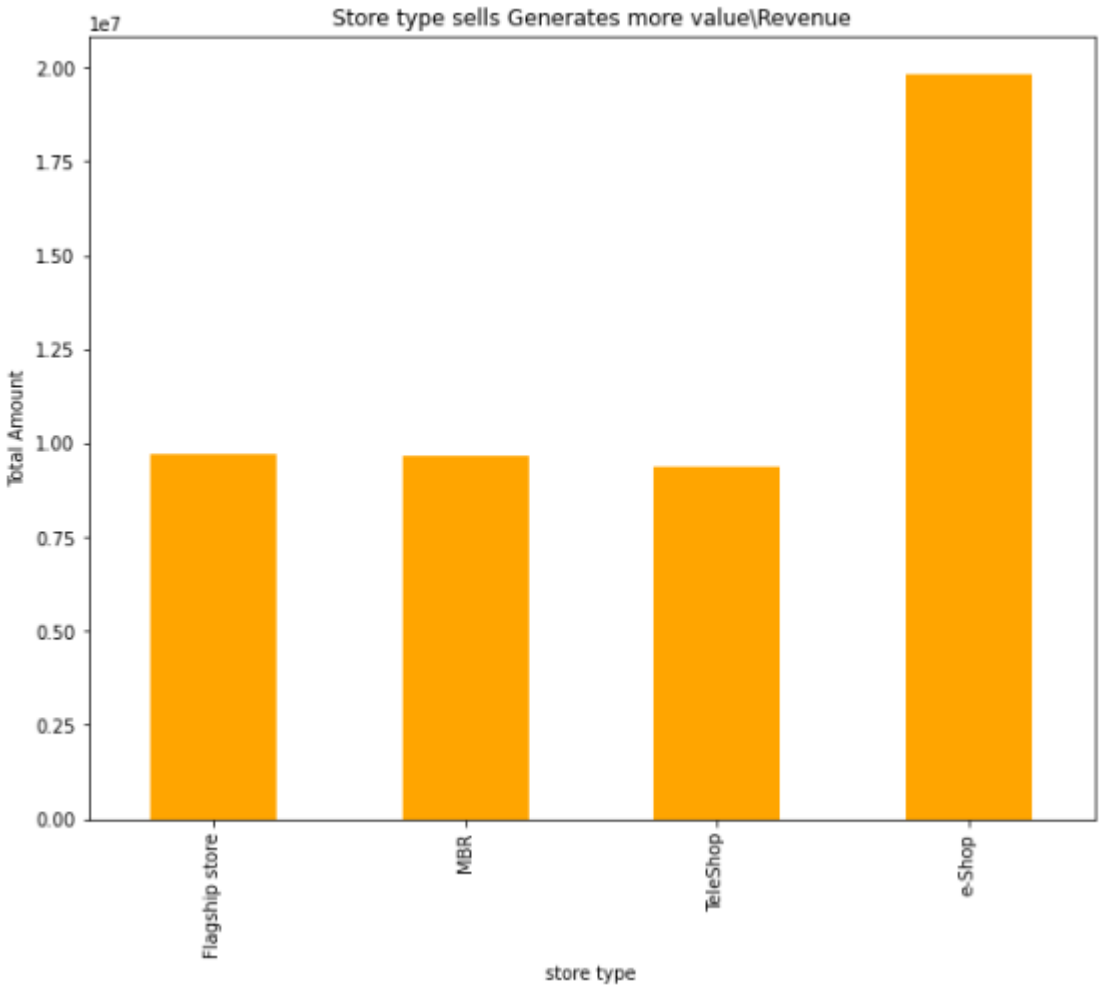
And it is least from city code 6.0

7.



e-shop sells maximum product

8.



It is the e-shop which generates highest revenue

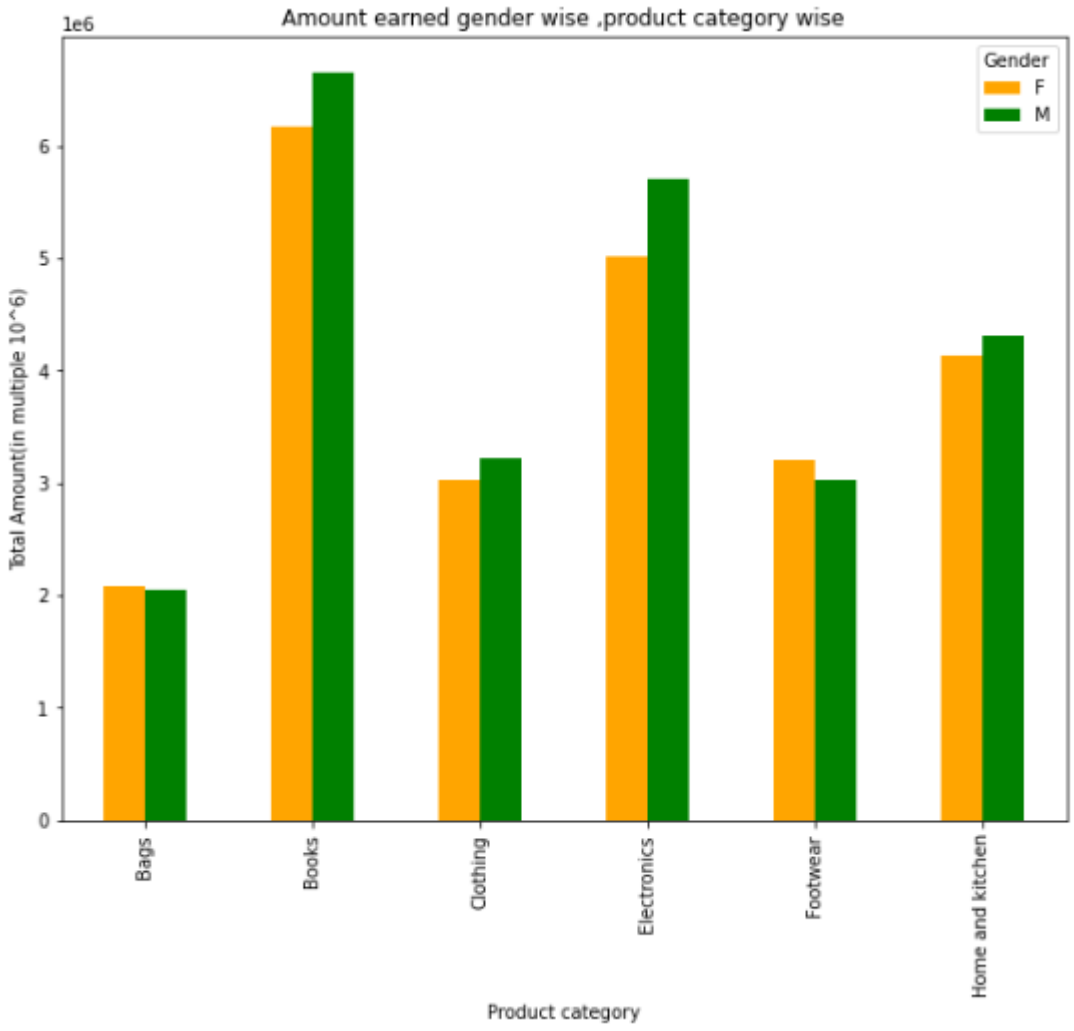
9.

Store_type prod_cat	Flagship store	MBR	TeleShop	e-Shop
Bags	870548.84	848678.68	789181.06	1617933.26
Books	2493677.81	2496039.19	2545714.47	5297161.16
Clothing	1194423.23	1287686.34	1241834.36	2527193.56
Electronics	2215136.04	2107969.83	1978457.20	4429142.77
Footwear	1234806.56	1112163.72	1235719.29	2643215.25
Home and kitchen	1713004.15	1822403.57	1581227.38	3327977.12

The above table shows the type of store, product category and Total amount generated by each product category

10.

Gender	F	M
prod_cat		
Bags	2079618.84	2046722.99
Books	6174590.82	6645972.78
Clothing	3026750.80	3224079.50
Electronics	5019354.21	5711351.62
Footwear	3203155.22	3020200.36
Home and kitchen	4133702.24	4305169.50



The above table and chart shows total amount by each product category gender wise.

Outcomes

1.e-shop is generating more revenue and value

2.number of male customers are more as compared to female customer

3.Low to medium income group are highest visitors they should be preserved(on basis of product tax I assumed)

4.Best selling item are Books and least are Bags,focus should also be made on Bag product to grow in market and generate revenue

5. Number of customers are highest from the city code 4.0 And it is least from city code 6.0

Detailed solution has been done in Jupyter notebook.

1.Merge the datasets Customers, Product Hierarchy and Transactions as Customer_Final. Ensure to keep all customers who have done transactions with us and select the join type accordingly.

Ans-

In csv files few columns were mismatching I renamed the columns to match the column names

Renamed customer_id to cust_id in customer table

Renamed prod_subcat_code as prod_sub_cat_code in transaction table

Next I joined the csv files available through inner join

Later I searched for duplicate rows and removed them

2. Prepare a summary report for the merged data set.

a. Get the column names and their corresponding data types

ans-

```
transaction_id          int64
cust_id                 int64
tran_date              object
prod_sub_cat_code       int64
prod_cat_code           int64
Qty                     int64
Rate                    int64
Tax                   float64
total_amt             float64
Store_type             object
DOB                    object
Gender                 object
city_code             float64
prod_cat               object
prod_subcat            object
dtype: object
```

b. Top/Bottom 10 observations

ans-

In [22]:
```
#top 10
cust_final.head(10)
```

Out[22]:

	transaction_id	cust_id	tran_date	prod_sub_cat_code	prod_cat_code	Qty	Rate	Tax	total_amt	Store_type	DOB	Gender	city_code	prod_cat	prod_s
0	80712190438	270351	28-02-2014	1	1	-5	-772	405.300	-4265.300	e-Shop	26-09-1981	M	5.0	Clothing	W
1	80712190438	270351	20-02-2014	1	1	5	772	405.300	4265.300	e-Shop	26-09-1981	M	5.0	Clothing	W
2	18505840838	271509	16-12-2013	1	1	3	1229	387.135	4074.135	Flagship store	08-06-1981	M	3.0	Clothing	W
3	92814475704	267750	16-08-2013	1	1	-4	-284	119.280	-1255.280	Flagship store	13-10-1986	M	1.0	Clothing	W
4	92814475704	267750	7/8/2013	1	1	4	284	119.280	1255.280	Flagship store	13-10-1986	M	1.0	Clothing	W
5	4737317330	269345	29-07-2011	1	1	5	1141	599.025	6304.025	MBR	26-06-1970	F	10.0	Clothing	W
6	44425889101	274987	18-03-2012	1	1	4	897	376.740	3964.740	Flagship store	08-10-1983	M	2.0	Clothing	W
7	90501340928	271817	19-02-2012	1	1	1	1122	117.810	1239.810	TeleShop	24-12-1989	M	8.0	Clothing	W
8	99335419136	268755	13-12-2012	1	1	3	1181	372.015	3915.015	e-Shop	15-07-1984	F	8.0	Clothing	W
9	35030444164	268129	18-11-2011	1	1	5	1047	549.675	5784.675	MBR	07-08-1982	F	9.0	Clothing	W

In [23]:
```
#bottom 10
cust_final.tail(10)
```

Out[23]:

	transaction_id	cust_id	tran_date	prod_sub_cat_code	prod_cat_code	Qty	Rate	Tax	total_amt	Store_type	DOB	Gender	city_code	prod_cat	pr
23043	3387244829	269114	15-07-2011	4	4	1	388	40.740	428.740	e-Shop	22-01-1989	F	5.0	Bags	
23044	76906459516	267940	15-06-2011	4	4	2	1263	265.230	2791.230	Flagship store	09-06-1979	M	9.0	Bags	
23045	73549617163	271334	5/7/2011	4	4	5	263	138.075	1453.075	e-Shop	08-12-1983	F	10.0	Bags	
23046	75339646315	274827	2/5/2011	4	4	4	1381	580.020	6104.020	e-Shop	27-12-1988	F	8.0	Bags	
23047	6650926717	268110	5/4/2011	4	4	4	1036	435.120	4579.120	MBR	06-03-1976	M	2.0	Bags	
23048	7173864364	271157	9/4/2011	4	4	5	788	413.700	4353.700	Flagship store	15-10-1973	F	6.0	Bags	
23049	5618131425	272010	3/3/2011	4	4	2	1150	241.500	2541.500	MBR	22-12-1972	F	5.0	Bags	
23050	18727956164	267161	23-02-2011	4	4	5	668	350.700	3690.700	e-Shop	08-05-1981	M	9.0	Bags	
23051	60416814232	273281	18-02-2011	4	4	4	202	84.840	892.840	Flagship store	14-12-1988	F	9.0	Bags	
23052	83245680995	273723	26-01-2011	4	4	4	1477	620.340	6528.340	e-Shop	21-01-1984	F	4.0	Bags	

c. “Five-number summary” for continuous variables (min, Q1, median, Q3 and max) d. Frequency tables for all the categorical variables

In [24]:
```
cust_final.describe()
#25%=Q1
#75%=Q3
#50%=median
```

Out[24]:

	transaction_id	cust_id	prod_sub_cat_code	prod_cat_code	Qty	Rate	Tax	total_amt	city_code
count	2.304000e+04	23040.000000	23040.000000	23040.000000	23040.000000	23040.000000	23040.000000	23040.000000	23032.000000
mean	5.006955e+10	271021.880252	6.148785	3.763498	2.435764	637.094965	248.677488	2109.865226	5.483067
std	2.898062e+10	2431.573668	3.726197	1.677091	2.264326	621.727374	187.188311	2505.610295	2.863331
min	3.268991e+06	266783.000000	1.000000	1.000000	-5.000000	-1499.000000	7.350000	-8270.925000	1.000000
25%	2.493315e+10	268935.000000	3.000000	2.000000	1.000000	312.000000	98.280000	762.450000	3.000000
50%	5.009188e+10	270980.500000	5.000000	4.000000	3.000000	710.000000	199.080000	1756.950000	5.000000
75%	7.532632e+10	273114.250000	10.000000	5.000000	4.000000	1109.000000	365.767500	3570.255000	8.000000
max	9.998755e+10	275265.000000	12.000000	6.000000	5.000000	1500.000000	787.500000	8287.500000	10.000000

2.d. Frequency tables for all the categorical variables

Ans

To get categorical data I first changed data type of “DOB” and “transactiondate” to date time then printed the frequency of categorical variables

	Store_type	Gender	prod_cat	prod_subcat
count	23040	23031	23040	23040
unique	4	2	6	18
top	e-Shop	M	Books	Women
freq	9304	11804	6066	3046

3.Generate histograms for all continuous variables and frequency bars for categorical variables.

Ans-Shown above

Assumed continuous variables only having data type numeric and excluded transaction_id and customer_id in making histogram

4. Calculate the following information using the merged dataset :

a. Time period of the available transaction data

calculated minimum date as start date and max date as end date and then there difference

```
Start_date
```

```
Timestamp('2011-01-02 00:00:00')
```

```
duration=End_date-Start_date
```

```
duration
```

```
Timedelta('1430 days 00:00:00')
```

#Time period of available transaction data startdate=02/01/2011 endDate=02/12/2014 thus duration is 1430 days

b. Count of transactions where the total amount of transaction was negative

Ans-(looked for transation_id where total amount<0)

On analysing I found that around 2164 transactions were negative transactions,it could be due to damaged product replacement , Returned item before being delivered, Returned item when it was in transit etc.

5. Analyze which product categories are more popular among females vs male customers.

Ans-

Popular_products = cust_final.groupby(["Gender","prod_cat"])[["Qty"]].sum()

#popular products in men are

Books

Clothing

Electronics

Home and Kitchen

#popular products in women are

Bags

Footwear

6. Which City code has the maximum customers and what was the percentage of customers from that city?

Ans-10.52% customer are from city code 4

```
cty_max = cust_final.groupby(["city_code"])["cust_id"].count().sort_values(ascending = False)
```

and then rounded it upto 2 decimal

7. Which store type sells the maximum products by value and by quantity?

Ans-e-shop

```
store_max_quantity=cust_final.groupby(["Store_type"])["Qty"].sum()
```

8. What was the total amount earned from the "Electronics" and "Clothing" categories from Flagship Stores?

Ans-3409559.27

Step1-

```
flagship_amt = cust_final.groupby(["Store_type","prod_cat"])[["total_amt"]].sum()
```

step2-

```
flag_Store_group = round(cust_final.pivot_table(index = "prod_cat", columns = "Store_type", v
alues = "total_amt",  aggfunc = "sum"),2)
```

step3-

```
Total_amt = flag_Store_group.loc[["Clothing","Electronics"],"Flagship store"].sum()
```

9. What was the total amount earned from "Male" customers under the "Electronics" category?

Ans-5711351.62

Step1-

gender_earn = round(cust_final.pivot_table(index = "prod_cat", columns = "Gender", values = "total_amt", aggfunc = "sum"),2)

step2-

Male_earn = gender_earn.loc["Electronics","M"].sum()

10. How many customers have more than 10 unique transactions, after removing all transactions which have any negative amounts?

Ans-none(0 customer)

First found non negative transactions

Then unique transactions

Then unique transaction with transaction_id count>10

Result was 0

11. For all customers aged between 25 - 35, find out:

Step1-

Made separate column for age

Made sepaate column for different age group ['25-35','36-46','47-57']

a. What was the total amount spent for “Electronics” and “Books” product categories?

Ans-6675409.98
Step1-

cust_final.groupby(['Age_cat','prod_cat'])['total_amt'].sum()

step2-

Year_25_35.loc['25-35',['Books','Electronics']].sum().round(2)

b. What was the total amount spent by these customers between 1st Jan, 2014 to 1st Mar, 2014?

Ans-The total amount spent by customers aged 25-35 between 1st Jan 2014 to 1st Mar 2014 is 456079.91

Step1-

total_amount = customer_total_amount_25_35[(customer_total_amount_25_35['tran_date'] >='2014-01-01') & (customer_total_amount_25_35['tran_date'] <='2014-03-01')]

step2-

total_amount['total_amt'].sum()

and got therequired results

Few references has been taken from below websites to solve the entire things in jupyter notebook-

1. pandas.pydata.org

2.www.kaggle.com

3.stackoverflow.com