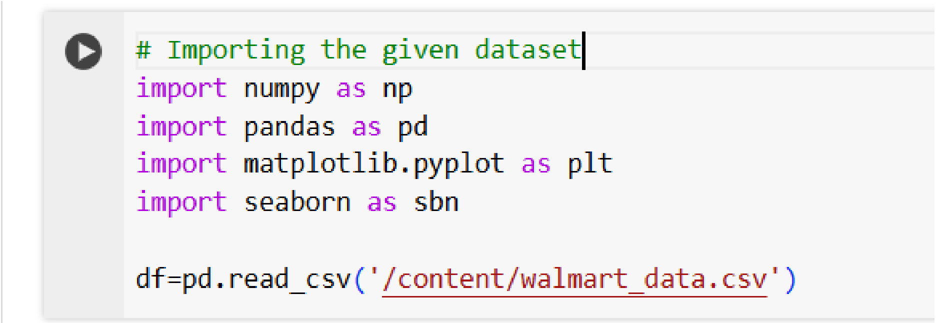
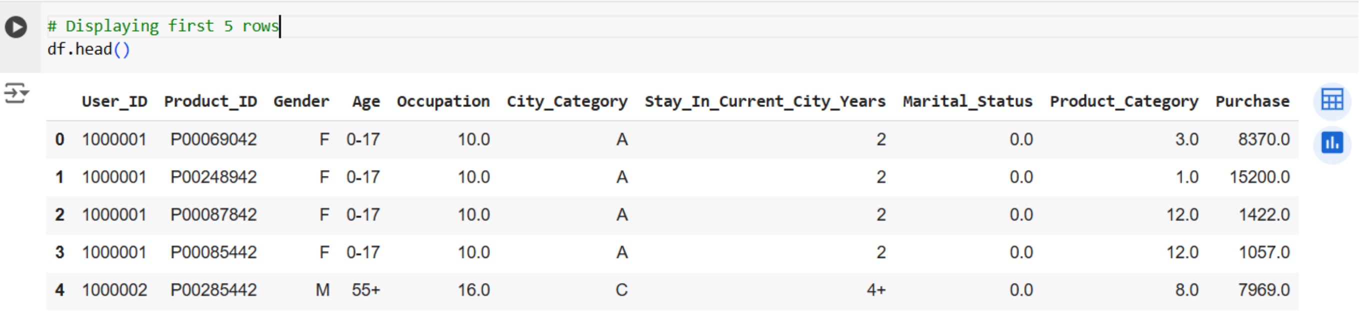
Walmart Business case study

1. Import the dataset and do usual data analysis steps like checking the structure & characteristics of the dataset.



```
# Importing the given dataset
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sbn

df=pd.read_csv('/content/walmart_data.csv')
```

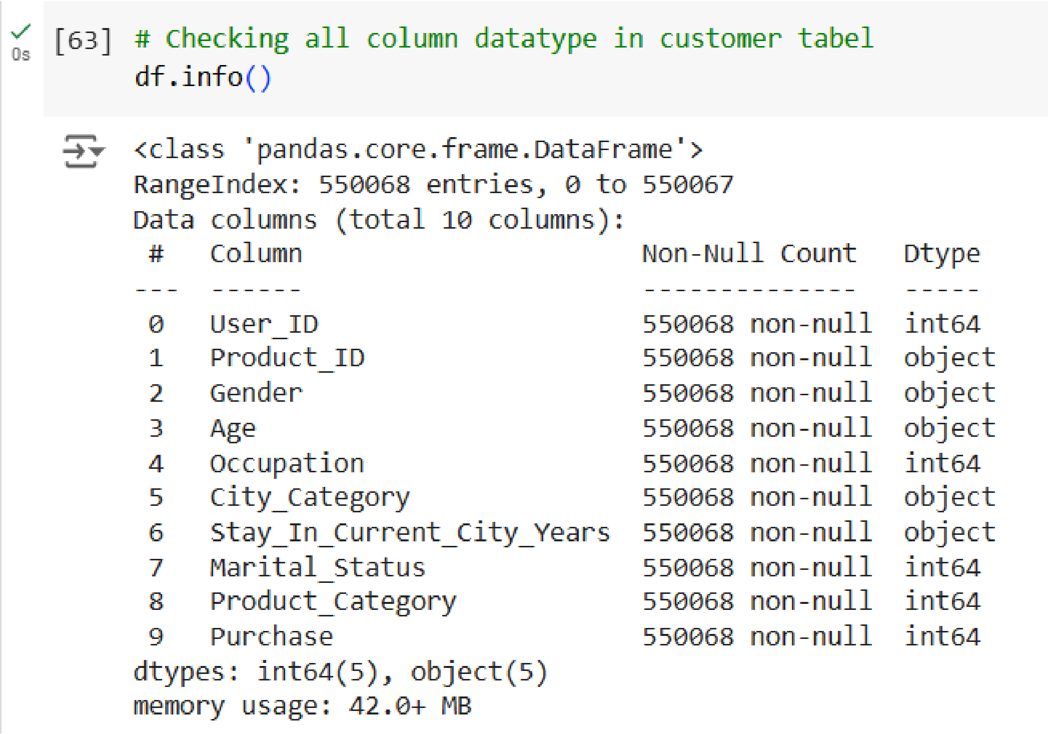


```
# Displaying first 5 rows
df.head()
```

	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category	Purchase
0	1000001	P00069042	F	0-17	10.0	A	2	0.0	3.0	8370.0
1	1000001	P00248942	F	0-17	10.0	A	2	0.0	1.0	15200.0
2	1000001	P00087842	F	0-17	10.0	A	2	0.0	12.0	1422.0
3	1000001	P00085442	F	0-17	10.0	A	2	0.0	12.0	1057.0
4	1000002	P00285442	M	55+	16.0	C	4+	0.0	8.0	7969.0

a. The data type of all columns in the “customers” table.

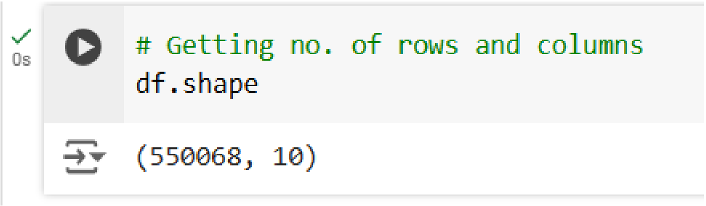
Three types are data are there: 1. Int64 2. Object 3. float64



```
[63] # Checking all column datatype in customer tabel
     df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 550068 entries, 0 to 550067
Data columns (total 10 columns):
 #   Column                      Non-Null Count   Dtype 
---  ------                      --------------   ----- 
 0   User_ID                     550068 non-null  int64 
 1   Product_ID                  550068 non-null  object
 2   Gender                      550068 non-null  object
 3   Age                         550068 non-null  object
 4   Occupation                  550068 non-null  int64 
 5   City_Category               550068 non-null  object
 6   Stay_In_Current_City_Years  550068 non-null  object
 7   Marital_Status              550068 non-null  int64 
 8   Product_Category            550068 non-null  int64 
 9   Purchase                    550068 non-null  int64 
dtypes: int64(5), object(5)
memory usage: 42.0+ MB
```

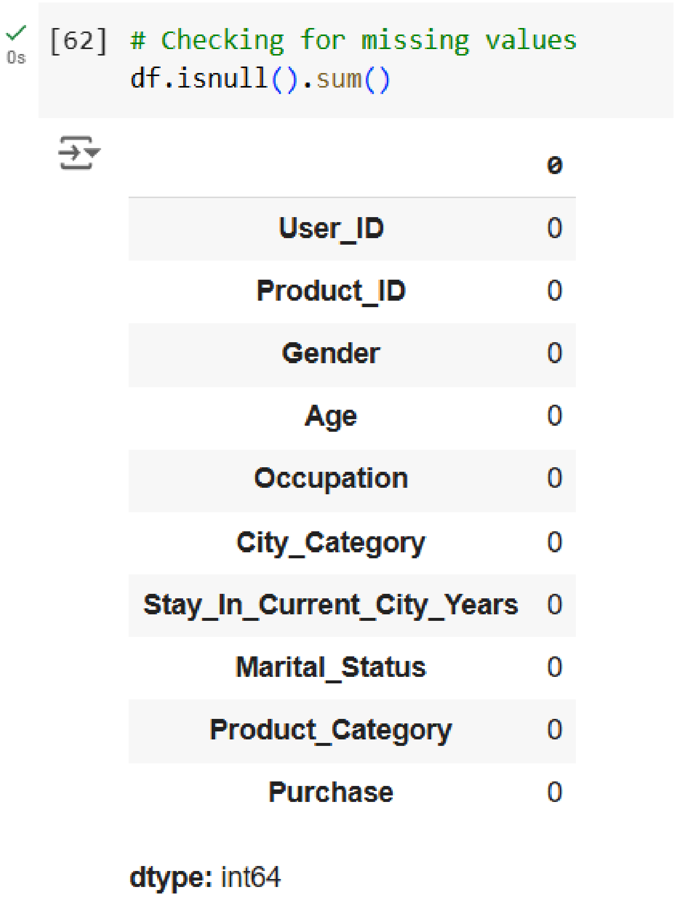
b. You can find the number of rows and columns given in the dataset.
Rows=550068, Columns=10



```
# Getting no. of rows and columns
df.shape
```

```
(550068, 10)
```

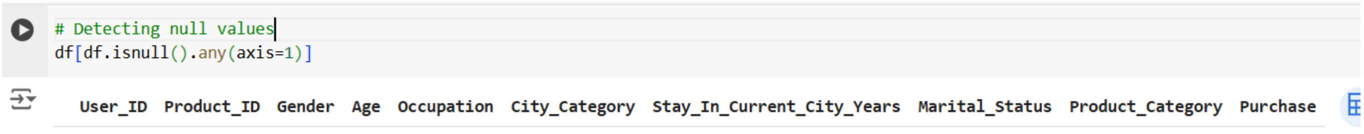
c. Check for the missing values and find the number of missing values in each Column.
There are no null values in the given dataset.



```
[62] # Checking for missing values
     df.isnull().sum()
```

```
                                0
                 User_ID        0
              Product_ID        0
                  Gender        0
                     Age        0
              Occupation        0
           City_Category        0
Stay_In_Current_City_Years      0
          Marital_Status        0
        Product_Category        0
                Purchase        0

dtype: int64
```

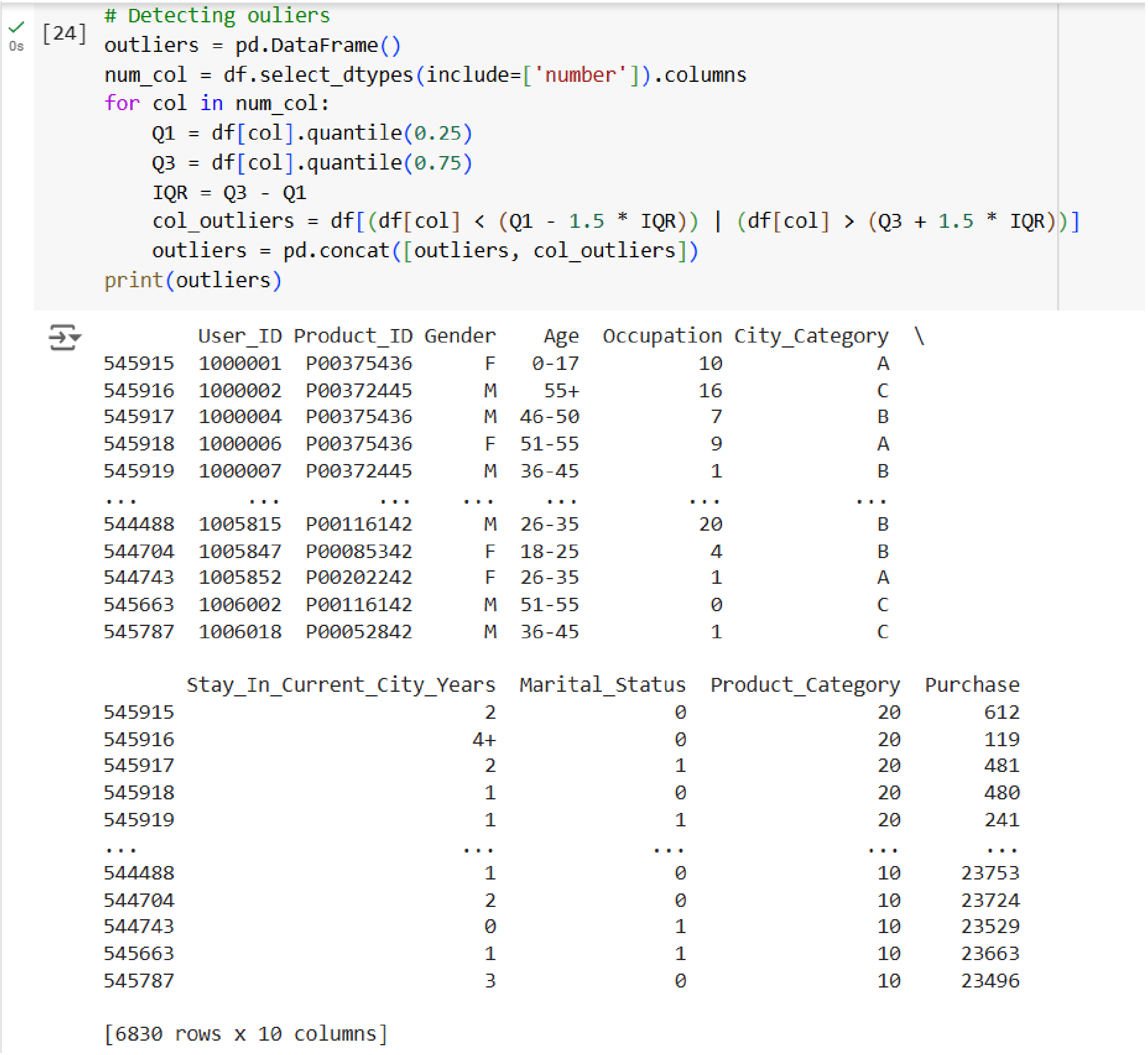
2. Detect Null values and outliers.



```
# Detecting null values
df[df.isnull().any(axis=1)]
```

```
User_ID  Product_ID  Gender  Age  Occupation  City_Category  Stay_In_Current_City_Years  Marital_Status  Product_Category  Purchase
```

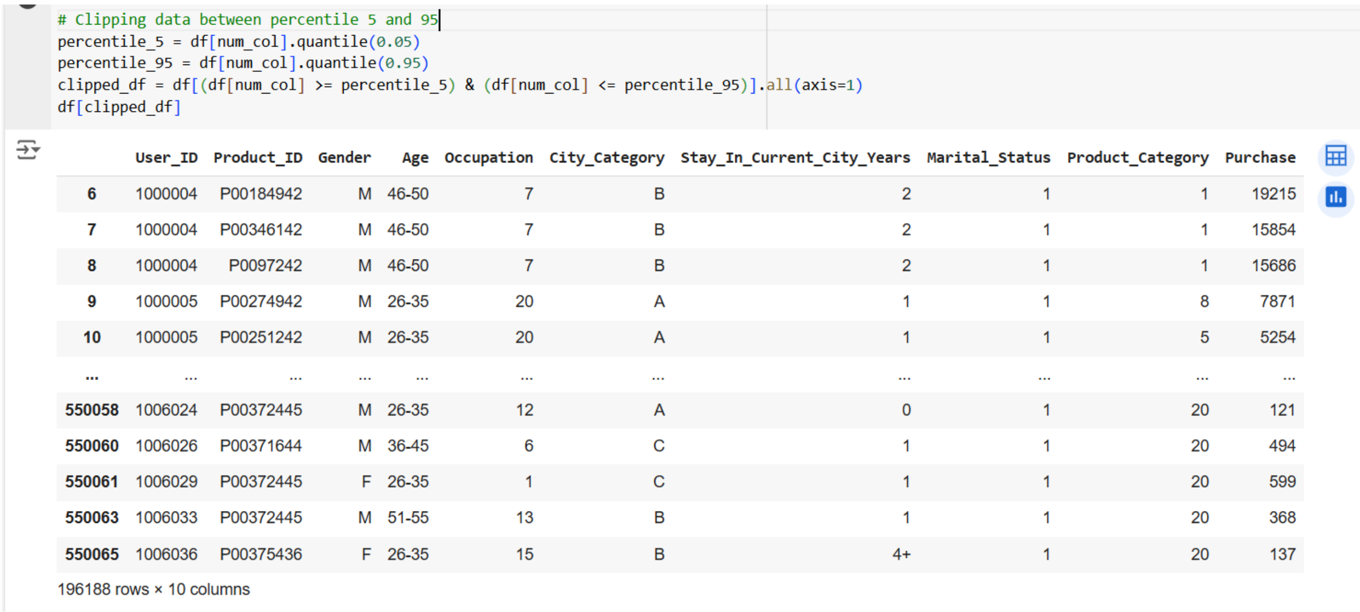
a. Find the outliers for every continuous variable in the dataset.



```
[24]
# Detecting ouliers
outliers = pd.DataFrame()
num_col = df.select_dtypes(include=['number']).columns
for col in num_col:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    col_outliers = df[(df[col] < (Q1 - 1.5 * IQR)) | (df[col] > (Q3 + 1.5 * IQR))]
    outliers = pd.concat([outliers, col_outliers])
print(outliers)
```

```
        User_ID Product_ID Gender    Age  Occupation City_Category  \
545915  1000001  P00375436      F   0-17          10             A
545916  1000002  P00372445      M    55+          16             C
545917  1000004  P00375436      M  46-50           7             B
545918  1000006  P00375436      F  51-55           9             A
545919  1000007  P00372445      M  36-45           1             B
...         ...        ...    ...    ...         ...           ...
544488  1005815  P00116142      M  26-35          20             B
544704  1005847  P00085342      F  18-25           4             B
544743  1005852  P00202242      F  26-35           1             A
545663  1006002  P00116142      M  51-55           0             C
545787  1006018  P00052842      M  36-45           1             C

       Stay_In_Current_City_Years  Marital_Status  Product_Category  Purchase
545915                          2               0                20       612
545916                         4+               0                20       119
545917                          2               1                20       481
545918                          1               0                20       480
545919                          1               1                20       241
...                           ...             ...               ...       ...
544488                          1               0                10     23753
544704                          2               0                10     23724
544743                          0               1                10     23529
545663                          1               1                10     23663
545787                          3               0                10     23496

[6830 rows x 10 columns]
```

b. Remove/clip the data between the 5 percentile and 95 percentile.



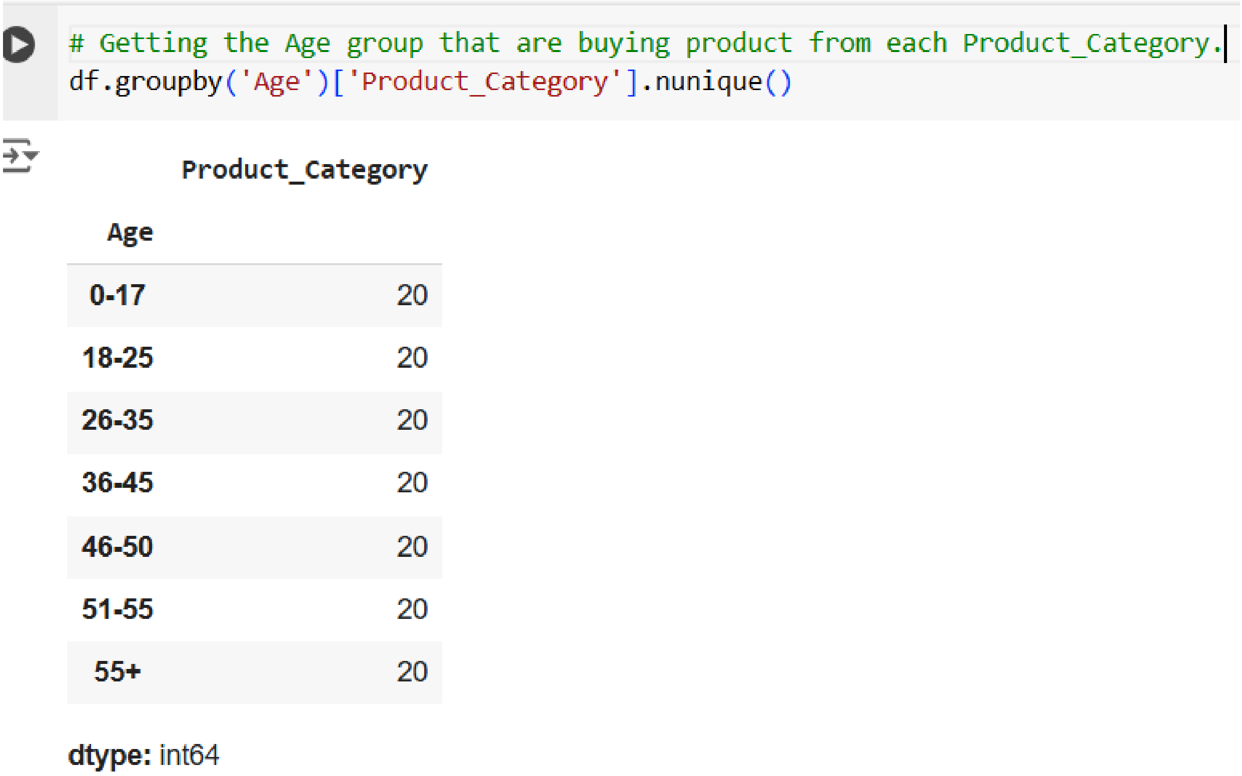
```python
# Clipping data between percentile 5 and 95
percentile_5 = df[num_col].quantile(0.05)
percentile_95 = df[num_col].quantile(0.95)
clipped_df = df[(df[num_col] >= percentile_5) & (df[num_col] <= percentile_95)].all(axis=1)
df[clipped_df]
```

	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category	Purchase
6	1000004	P00184942	M	46-50	7	B	2	1	1	19215
7	1000004	P00346142	M	46-50	7	B	2	1	1	15854
8	1000004	P0097242	M	46-50	7	B	2	1	1	15686
9	1000005	P00274942	M	26-35	20	A	1	1	8	7871
10	1000005	P00251242	M	26-35	20	A	1	1	5	5254
...	...	...	...	...	...	...	...	...	...	...
550058	1006024	P00372445	M	26-35	12	A	0	1	20	121
550060	1006026	P00371644	M	36-45	6	C	1	1	20	494
550061	1006029	P00372445	F	26-35	1	C	1	1	20	599
550063	1006033	P00372445	M	51-55	13	B	1	1	20	368
550065	1006036	P00375436	F	26-35	15	B	4+	1	20	137

196188 rows × 10 columns

3. Data Exploration:

a. What products are different age groups buying?



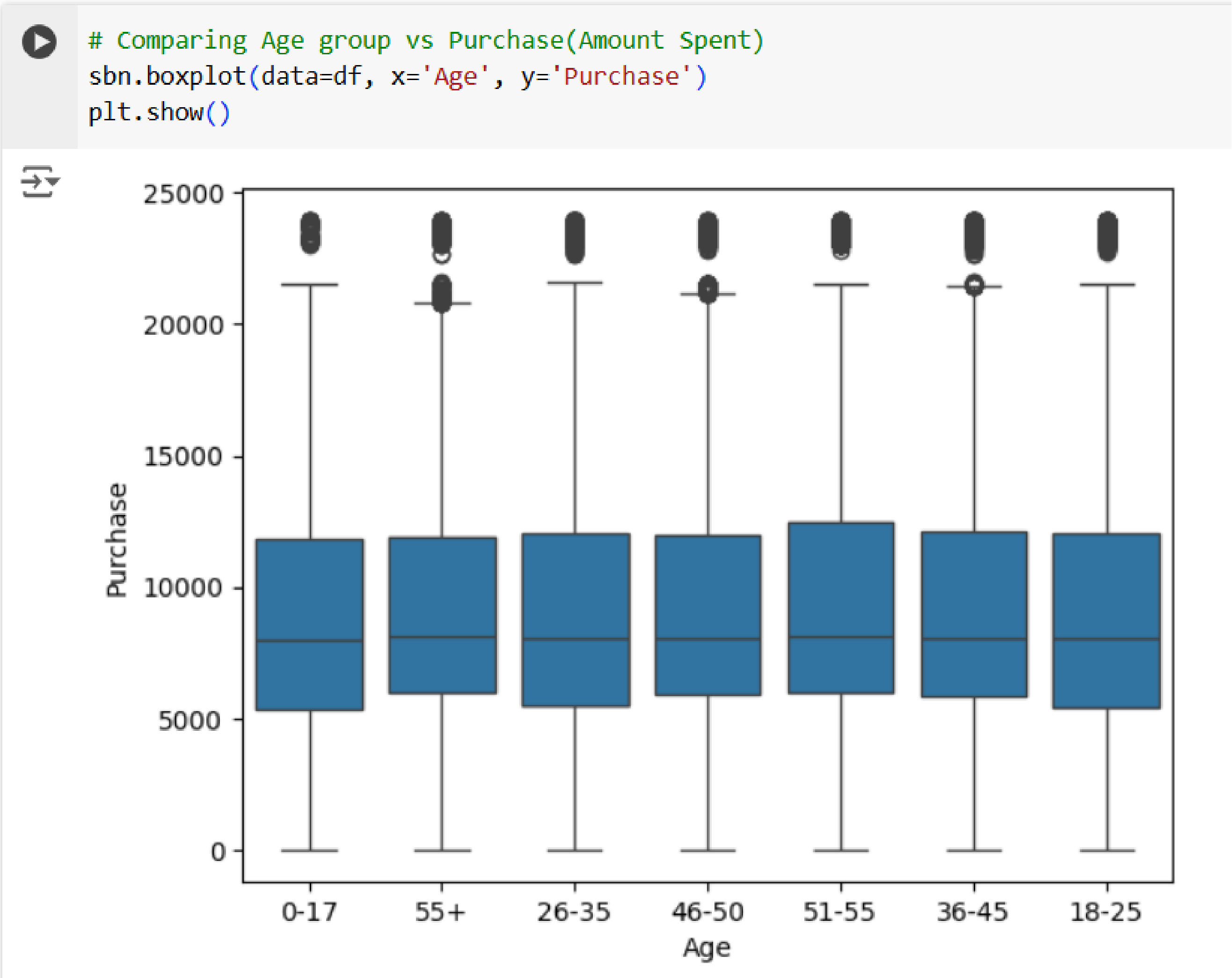
```python
# Getting the Age group that are buying product from each Product_Category.
df.groupby('Age')['Product_Category'].nunique()
```

	Product_Category
Age	
0-17	20
18-25	20
26-35	20
36-45	20
46-50	20
51-55	20
55+	20

dtype: int64

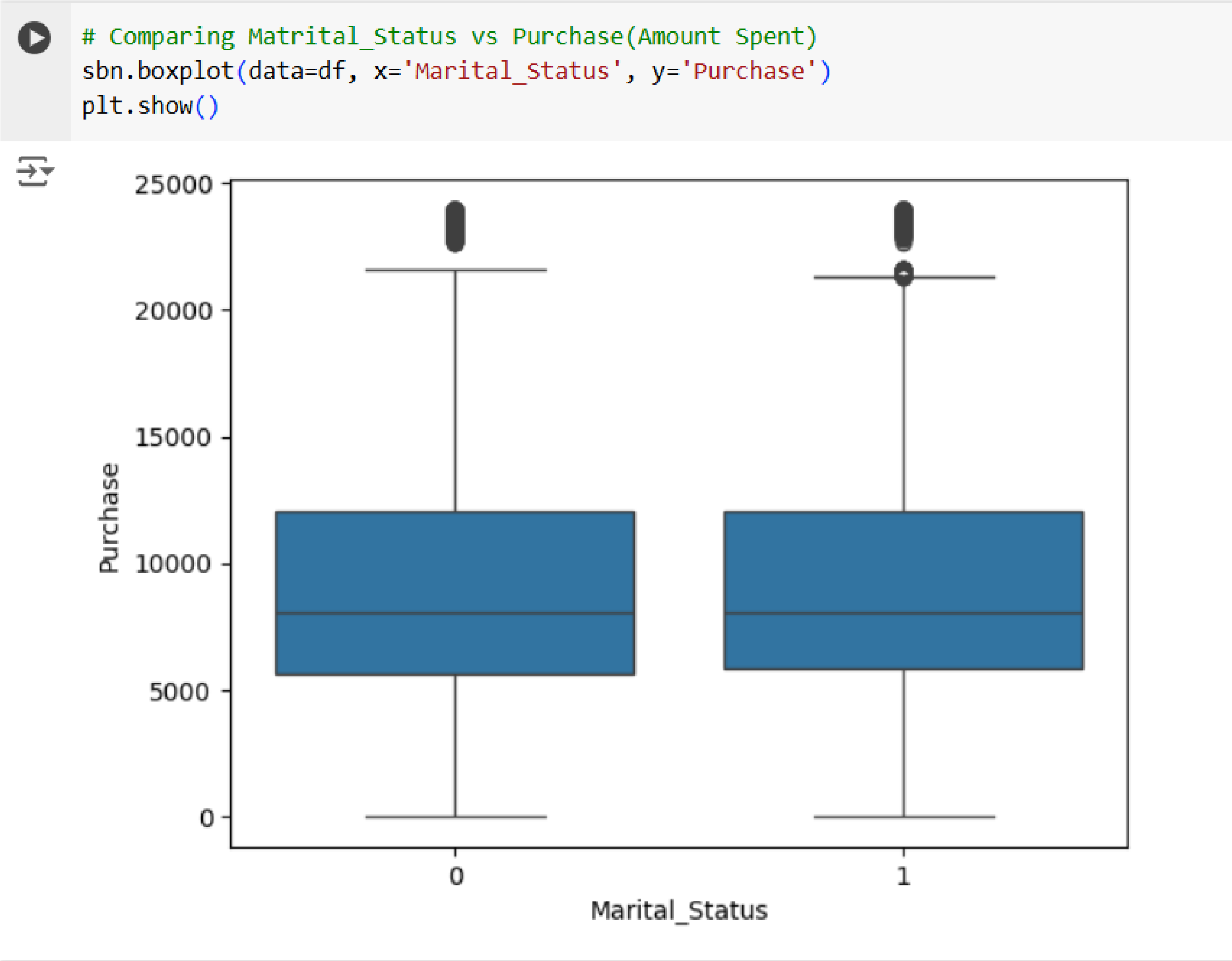
b. Is there a relationship between age, marital status, and the amount spent?

- Median Amount spent in each Age group is almost equal.



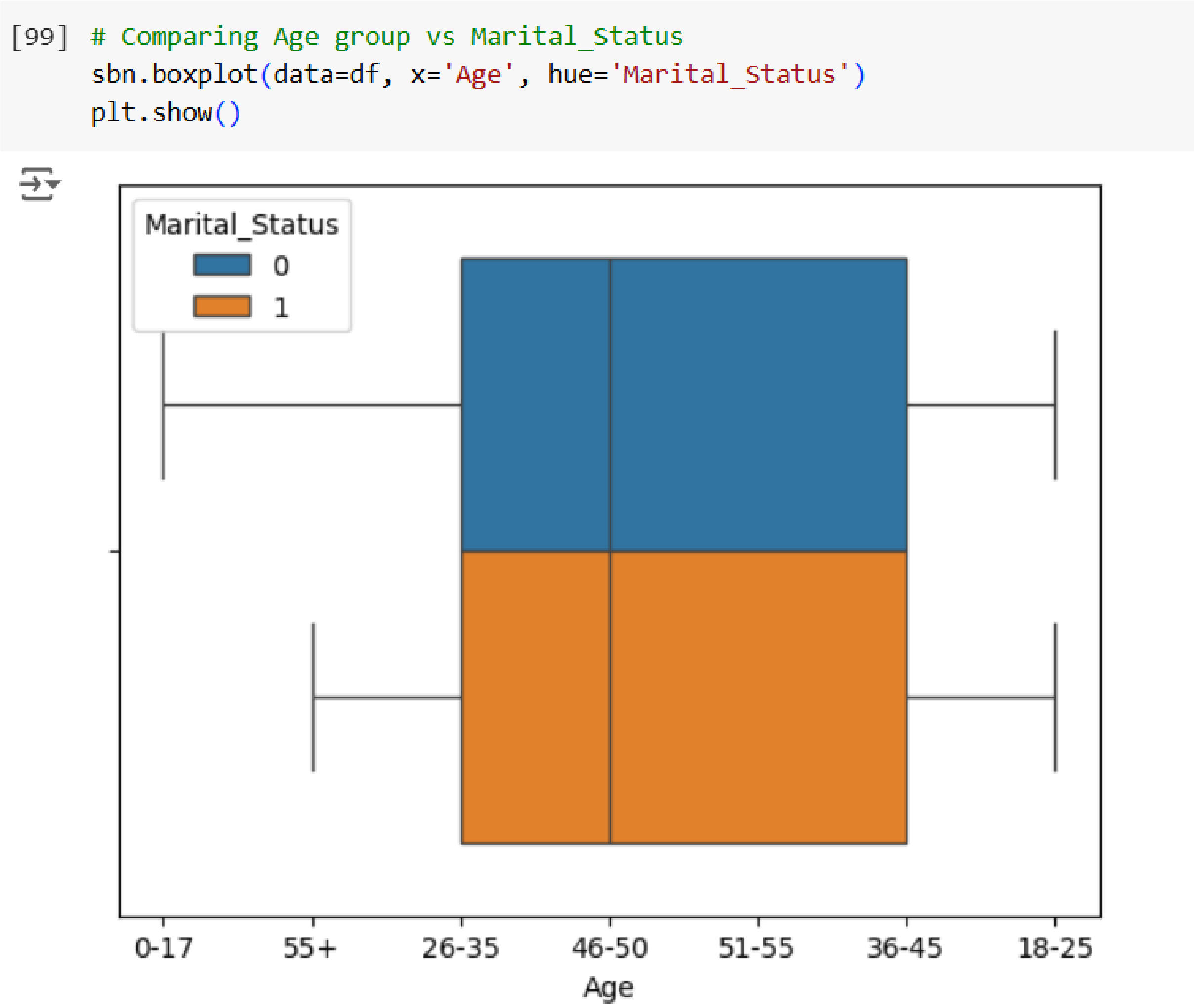
```
# Comparing Age group vs Purchase(Amount Spent)
sbn.boxplot(data=df, x='Age', y='Purchase')
plt.show()
```

- Median Amount spent is equal for single and married people both.



```
# Comparing Matrital_Status vs Purchase(Amount Spent)
sbn.boxplot(data=df, x='Marital_Status', y='Purchase')
plt.show()
```

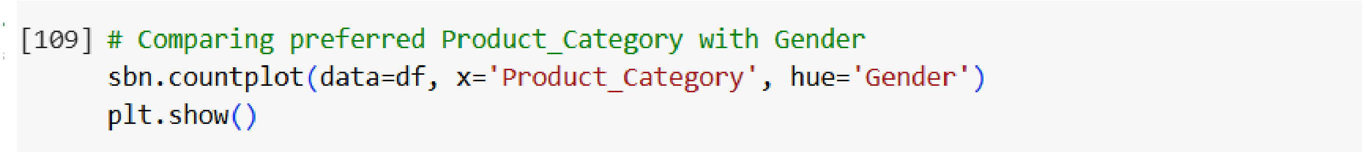
- Median Age for single and married people is same.



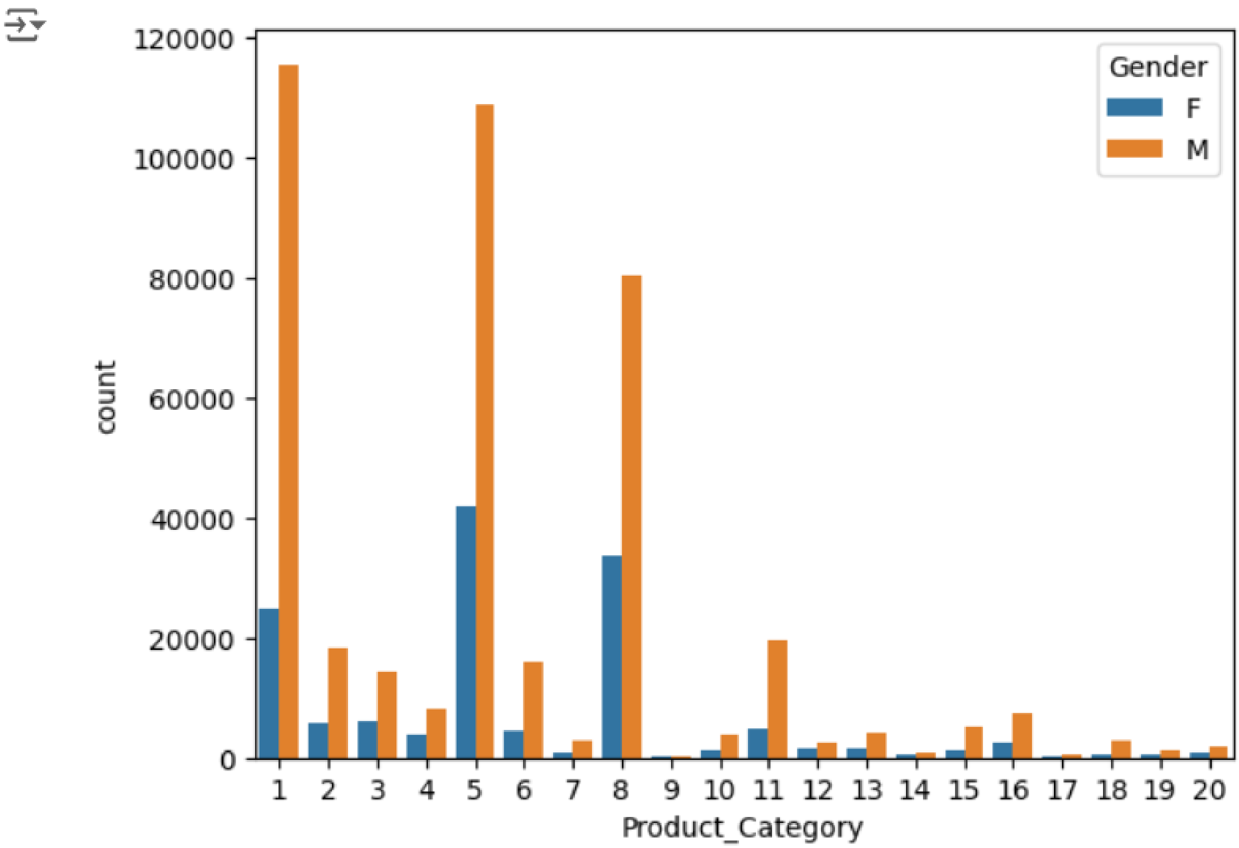
```
[99] # Comparing Age group vs Marital_Status
     sbn.boxplot(data=df, x='Age', hue='Marital_Status')
     plt.show()
```

d. Are there preferred product categories for different genders?

- Top five Product Categories preferred by male people are 1,5,8,11 and 16 in descending order.
- Similarly top 5 Categories preferred by female people are 5,8,1,11 and 16 in descending order.

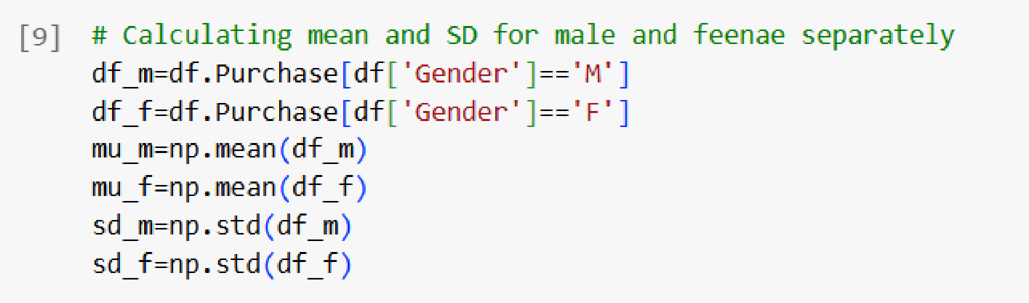


```
[109] # Comparing preferred Product_Category with Gender
      sbn.countplot(data=df, x='Product_Category', hue='Gender')
      plt.show()
```

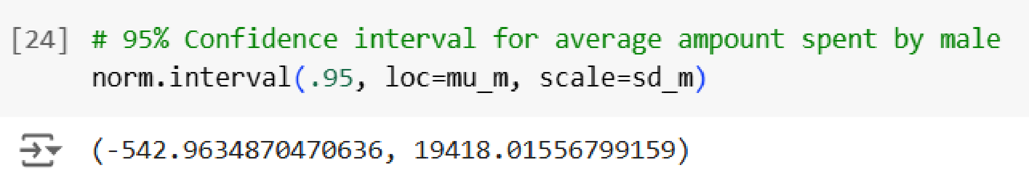


4. How does gender affect the amount spent?

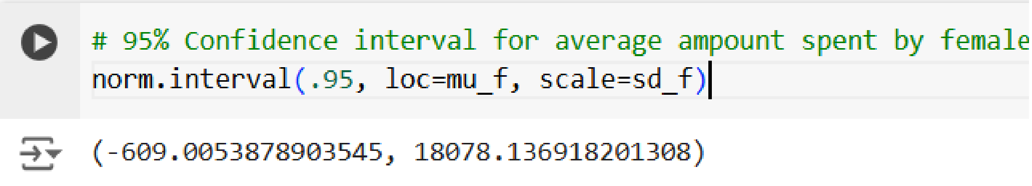
95% confidence interval for average amount spent is given below for male and female separately.



```
[9]  # Calculating mean and SD for male and feenae separately
     df_m=df.Purchase[df['Gender']=='M']
     df_f=df.Purchase[df['Gender']=='F']
     mu_m=np.mean(df_m)
     mu_f=np.mean(df_f)
     sd_m=np.std(df_m)
     sd_f=np.std(df_f)
```



```
[24] # 95% Confidence interval for average ampount spent by male
     norm.interval(.95, loc=mu_m, scale=sd_m)

     (-542.9634870470636, 19418.01556799159)
```



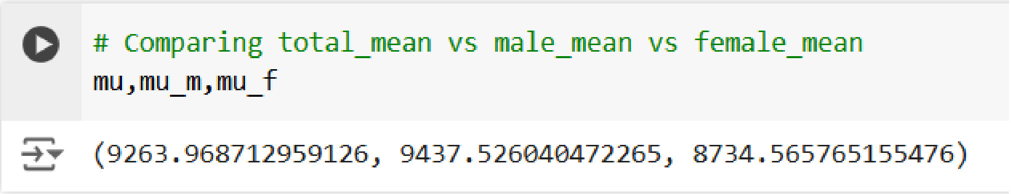
```
     # 95% Confidence interval for average ampount spent by female
     norm.interval(.95, loc=mu_f, scale=sd_f)

     (-609.0053878903545, 18078.136918201308)
```

a. From the above calculated CLT answer the following questions.

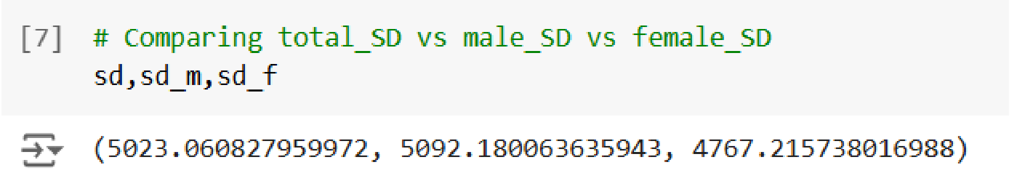
i. Is the confidence interval computed using the entire dataset wider for one of the genders? Why is this the case?

Ans. No, Confidence interval for entire dataset in not wider because for entire dataset, n (sample size) is increased and then standard error decrease, consequently value (MU+SE) or (MU-SE) in decreased. In case of male category, it is less wide due increased sample size.

But in case of female category, confidence interval for entire dataset is observed wider as compared to female purchase because of female purchase is less dispersed, resulting value of standard error less.



```
     # Comparing total_mean vs male_mean vs female_mean
     mu,mu_m,mu_f

     (9263.968712959126, 9437.526040472265, 8734.565765155476)
```



```
[7]  # Comparing total_SD vs male_SD vs female_SD
     sd,sd_m,sd_f

     (5023.060827959972, 5092.180063635943, 4767.215738016988)
```

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

ii. How is the width of the confidence interval affected by the sample size?

Ans. Width of the confidence interval in inversely proportional with square root of sample size, Large sample size reduces the standard error, leading to a more precise estimate and narrower interval.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iii. Do the confidence intervals for different sample sizes overlap?

Ans. Yes, the confidence interval for different sample sizes can overlap because each sample, regardless of size, may produce similar estimates of the population mean, especially if they're taken from the same population. Smaller samples have wider intervals to account for higher variability, while larger samples have narrower intervals due to increased precision.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iv. How does the sample size affect the shape of the distributions of the means?

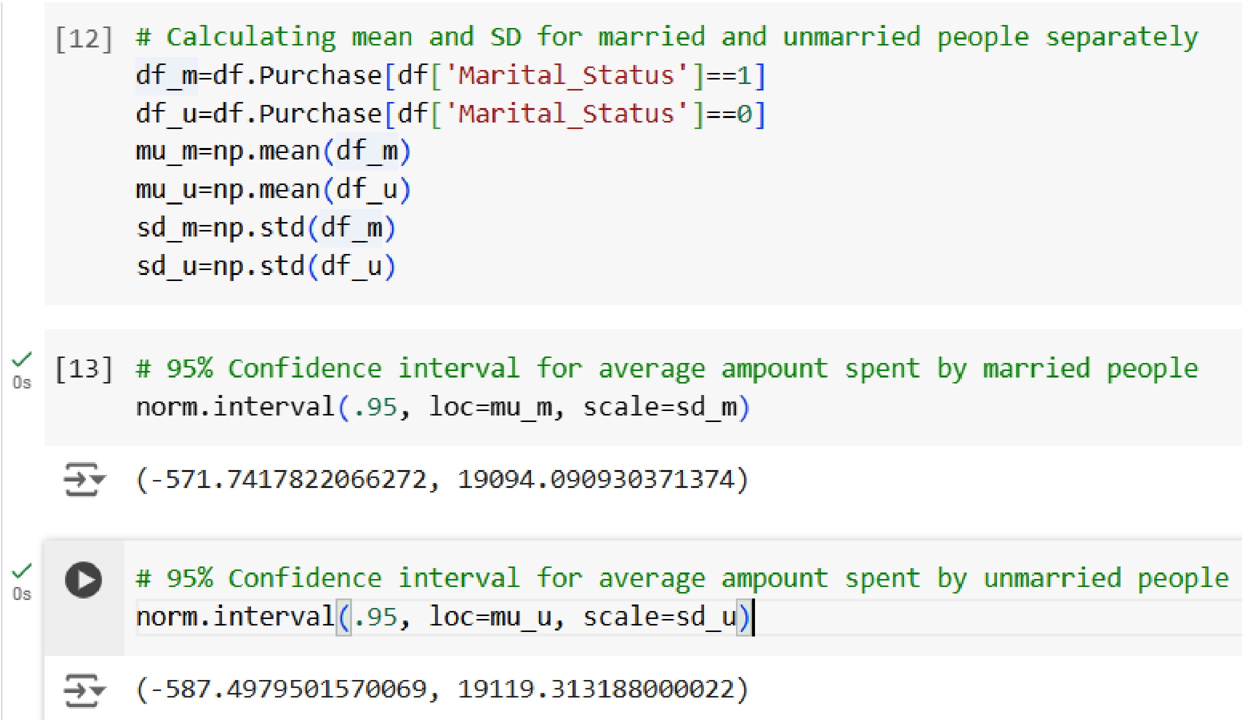
Ans. As sample size increases, the distribution of sample means becomes more normal and narrower, with less variability, providing a more precise estimate of the population mean,

And if the sample size decreases the distribution of sample means becomes more flatter with more variability and provides less precise estimate of the population mean.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

5. How does Marital_Status affect the amount spent?

Ans. 95% confidence interval for average amount spent is given below for married and unmarried people separately.



```
[12] # Calculating mean and SD for married and unmarried people separately
     df_m=df.Purchase[df['Marital_Status']==1]
     df_u=df.Purchase[df['Marital_Status']==0]
     mu_m=np.mean(df_m)
     mu_u=np.mean(df_u)
     sd_m=np.std(df_m)
     sd_u=np.std(df_u)
```

```
[13] # 95% Confidence interval for average ampount spent by married people
     norm.interval(.95, loc=mu_m, scale=sd_m)
```

```
(-571.7417822066272, 19094.090930371374)
```

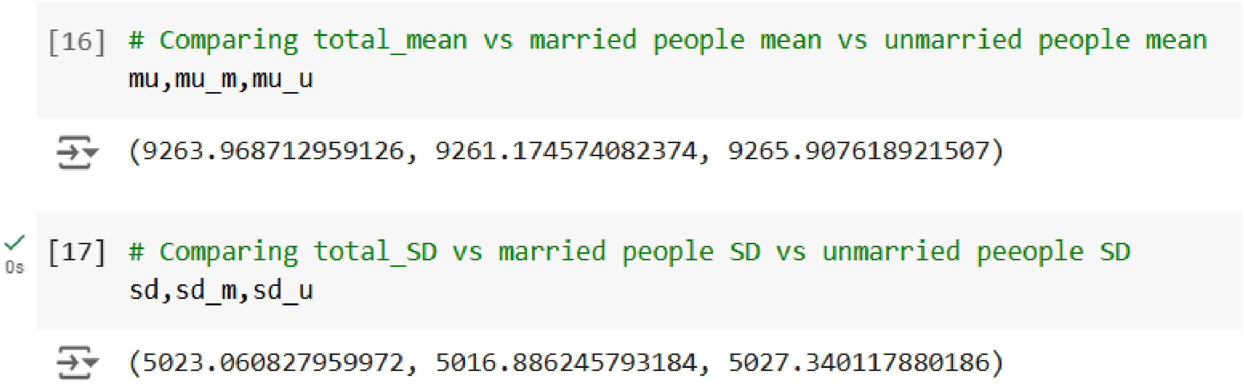
```
# 95% Confidence interval for average ampount spent by unmarried people
norm.interval(.95, loc=mu_u, scale=sd_u)
```

```
(-587.4979501570069, 19119.313188000022)
```

a. From the above calculated CLT answer the following questions.

i. Is the confidence interval computed using the entire dataset wider for one of the genders? Why is this the case?

Ans. Yes, Confidence interval for entire dataset in wider than married category because dispersion in entire dataset is significantly higher than married category. But in case of unmarried category, confidence interval for entire dataset is observed narrower as compared to unmarried people because dispersion observed in entire dataset is significantly lower as compared to unmarried people.

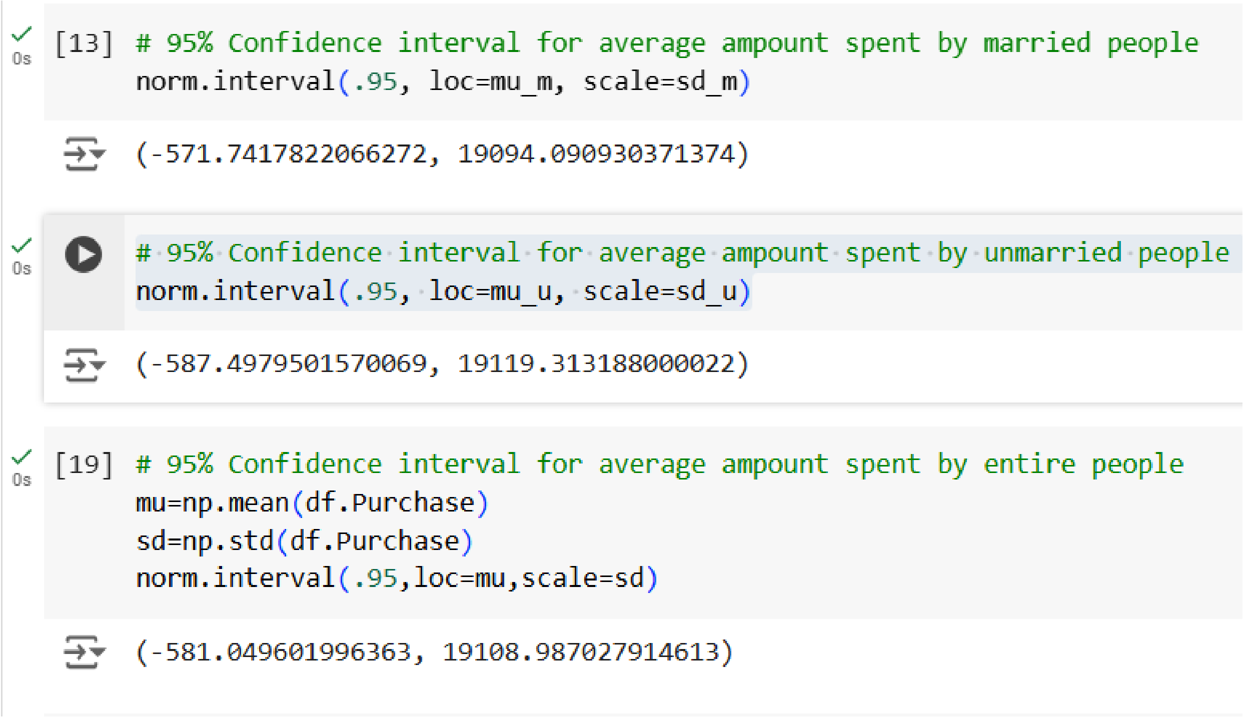


```
[16] # Comparing total_mean vs married people mean vs unmarried people mean
     mu,mu_m,mu_u
```

```
(9263.968712959126, 9261.174574082374, 9265.907618921507)
```

```
[17] # Comparing total_SD vs married people SD vs unmarried peeople SD
     sd,sd_m,sd_u
```

```
(5023.060827959972, 5016.886245793184, 5027.340117880186)
```



```
[13] # 95% Confidence interval for average ampount spent by married people
     norm.interval(.95, loc=mu_m, scale=sd_m)
```

```
(-571.7417822066272, 19094.090930371374)
```

```
# 95% Confidence interval for average ampount spent by unmarried people
norm.interval(.95, loc=mu_u, scale=sd_u)
```

```
(-587.4979501570069, 19119.313188000022)
```

```
[19] # 95% Confidence interval for average ampount spent by entire people
     mu=np.mean(df.Purchase)
     sd=np.std(df.Purchase)
     norm.interval(.95,loc=mu,scale=sd)
```

```
(-581.049601996363, 19108.987027914613)
```

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

ii. How is the width of the confidence interval affected by the sample size?

Ans. Width of the confidence interval in inversely proportional with square root of sample size, large sample size reduces the standard error, leading to a more precise estimate and narrower interval.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iii. Do the confidence intervals for different sample sizes overlap?

Ans. Yes, confidence intervals for different sample sizes can overlap because each sample, regardless of size, may produce similar estimates of the population mean. Smaller samples have wider intervals due to higher variability, while larger samples have narrower intervals with greater precision.

Overlap indicates consistency in the estimates across sample sizes.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iv. How does the sample size affect the shape of the distributions of the means?

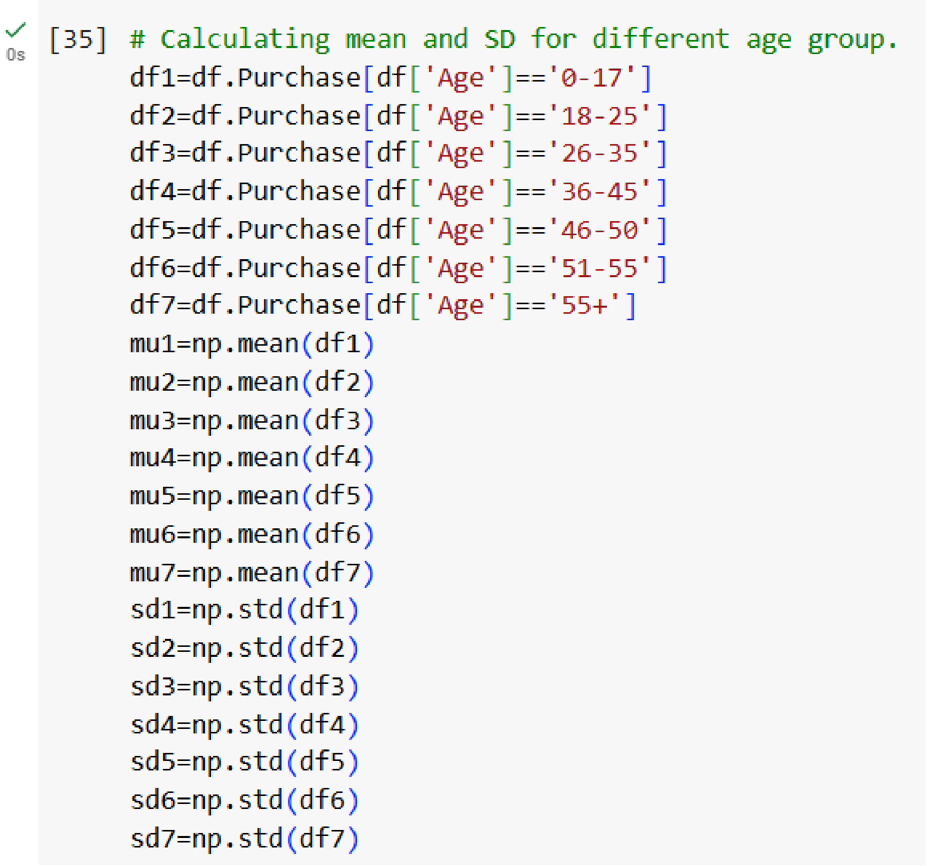
Ans. As sample size increases, the distribution of sample means becomes more normal and narrower, with less variability, providing a more precise estimate of the population mean,

And if the sample size decreases the distribution of sample means becomes more flatter with more variability and provides less precise estimate of the population mean.

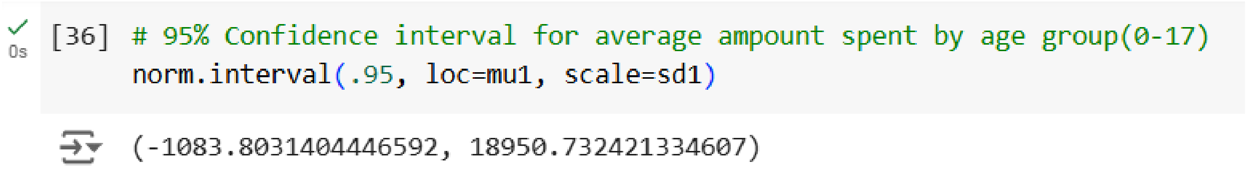
$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

6. How does Age affect the amount spent?

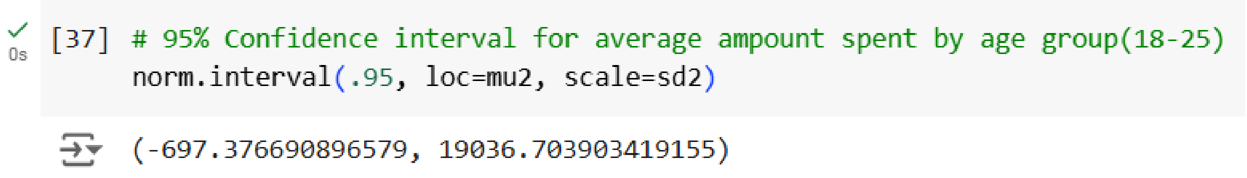
Ans. 95% confidence interval for average amount spent is given below for different age groups.



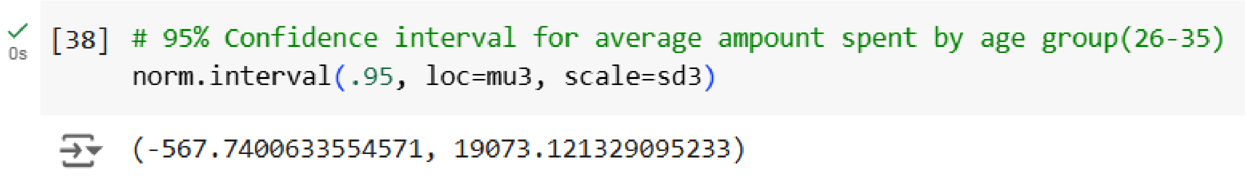
```
[35] # Calculating mean and SD for different age group.
     df1=df.Purchase[df['Age']=='0-17']
     df2=df.Purchase[df['Age']=='18-25']
     df3=df.Purchase[df['Age']=='26-35']
     df4=df.Purchase[df['Age']=='36-45']
     df5=df.Purchase[df['Age']=='46-50']
     df6=df.Purchase[df['Age']=='51-55']
     df7=df.Purchase[df['Age']=='55+']
     mu1=np.mean(df1)
     mu2=np.mean(df2)
     mu3=np.mean(df3)
     mu4=np.mean(df4)
     mu5=np.mean(df5)
     mu6=np.mean(df6)
     mu7=np.mean(df7)
     sd1=np.std(df1)
     sd2=np.std(df2)
     sd3=np.std(df3)
     sd4=np.std(df4)
     sd5=np.std(df5)
     sd6=np.std(df6)
     sd7=np.std(df7)
```



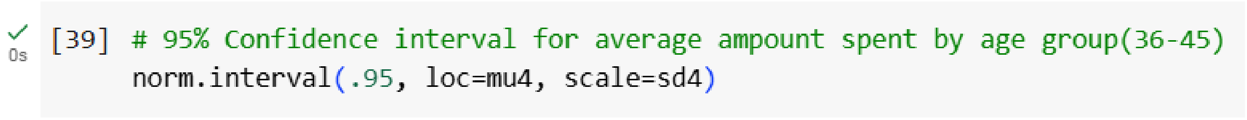
```
[36] # 95% Confidence interval for average ampount spent by age group(0-17)
     norm.interval(.95, loc=mu1, scale=sd1)

     (-1083.8031404446592, 18950.732421334607)
```



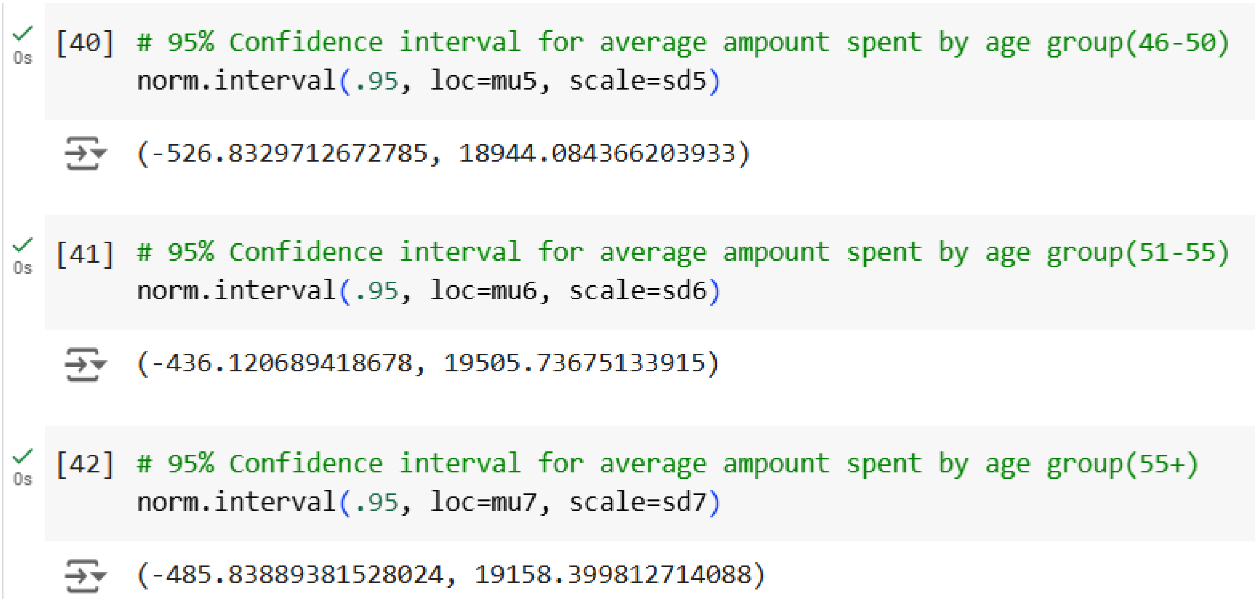
```
[37] # 95% Confidence interval for average ampount spent by age group(18-25)
     norm.interval(.95, loc=mu2, scale=sd2)

     (-697.376690896579, 19036.703903419155)
```



```
[38] # 95% Confidence interval for average ampount spent by age group(26-35)
     norm.interval(.95, loc=mu3, scale=sd3)

     (-567.7400633554571, 19073.121329095233)
```



```
[39] # 95% Confidence interval for average ampount spent by age group(36-45)
     norm.interval(.95, loc=mu4, scale=sd4)
```

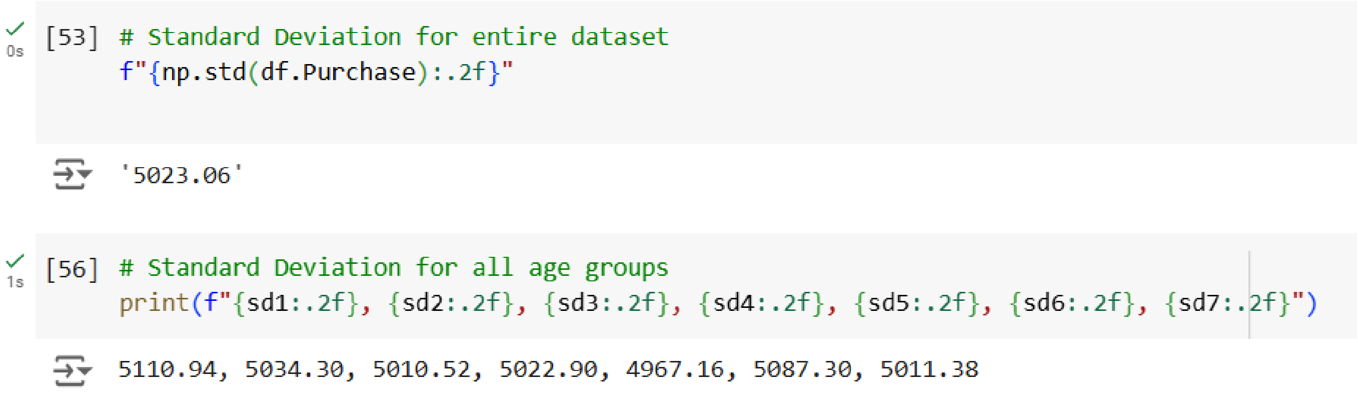


```
[40] # 95% Confidence interval for average ampount spent by age group(46-50)
     norm.interval(.95, loc=mu5, scale=sd5)

     (-526.8329712672785, 18944.084366203933)

[41] # 95% Confidence interval for average ampount spent by age group(51-55)
     norm.interval(.95, loc=mu6, scale=sd6)

     (-436.120689418678, 19505.73675133915)

[42] # 95% Confidence interval for average ampount spent by age group(55+)
     norm.interval(.95, loc=mu7, scale=sd7)

     (-485.83889381528024, 19158.399812714088)
```

a. From the above calculated CLT answer the following questions.

i. Is the confidence interval computed using the entire dataset wider for one of the genders? Why is this the case?

Ans. Yes, Confidence interval for entire dataset in wider than the confidence interval of three age groups (26-35, 46-50 & 55+) as standard deviation of the entire dataset is significantly higher than these groups.
But rest age group (except '36-45') have confidence interval wider than the entire dataset as standard deviation of these groups is higher than the standard deviation of the entire dataset.



```
[53] # Standard Deviation for entire dataset
     f"{np.std(df.Purchase):.2f}"

     '5023.06'

[56] # Standard Deviation for all age groups
     print(f"{sd1:.2f}, {sd2:.2f}, {sd3:.2f}, {sd4:.2f}, {sd5:.2f}, {sd6:.2f}, {sd7:.2f}")

     5110.94, 5034.30, 5010.52, 5022.90, 4967.16, 5087.30, 5011.38
```

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

ii. How is the width of the confidence interval affected by the sample size?

Ans. Width of the confidence interval in inversely proportional with square root of sample size, large sample size reduces the standard error, leading to a more precise estimate and narrower interval.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iii. Do the confidence intervals for different sample sizes overlap?

Ans. Yes, confidence intervals for different sample sizes can overlap because each sample, regardless of size, may produce similar estimates of the population mean. Smaller samples have wider intervals due to higher variability, while larger samples have narrower intervals with greater precision.

Overlap indicates consistency in the estimates across sample sizes.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

iv. How does the sample size affect the shape of the distributions of the

means?

Ans. As sample size increases, the distribution of sample means becomes more normal and narrower, with less variability, providing a more precise estimate of the population mean,

And if the sample size decreases the distribution of sample means becomes more flatter with more variability and provides less precise estimate of the population mean.

$$\left(\bar{x} - z \cdot \frac{\sigma}{\sqrt{n}}, \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}\right)$$

7. Create a report:

a. Report whether the confidence intervals for the average amount spent by males and females (computed using all the data) overlap. How can Walmart leverage this conclusion to make changes or improvements?

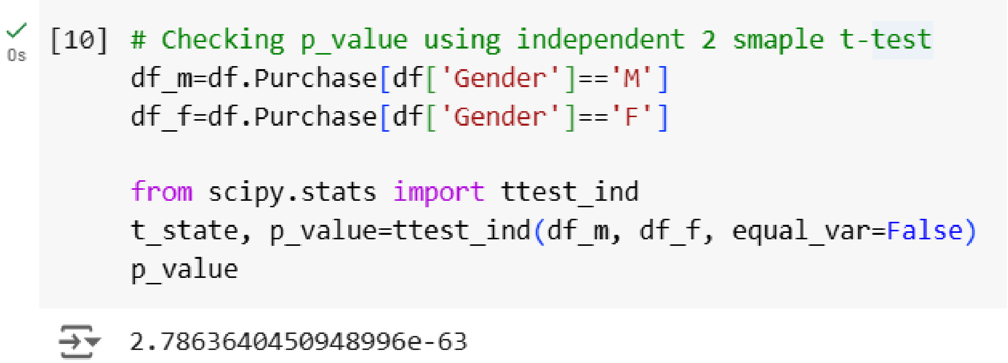
Ans. Determining overlap of confidence interval of Purchase for males & females.

If the interval overlap, there may not be significant difference between in average spending between males & females.

Checking for difference in average spending across Genders using t-test:

Assumption: Ho-Average spending of males & females is similar.

Ha-Average spending of males & females is significantly different.



```
[10] # Checking p_value using independent 2 smaple t-test
     df_m=df.Purchase[df['Gender']=='M']
     df_f=df.Purchase[df['Gender']=='F']

     from scipy.stats import ttest_ind
     t_state, p_value=ttest_ind(df_m, df_f, equal_var=False)
     p_value
```

```
2.7863640450948996e-63
```

For 95% confidence level, alpha=1-0.95, i.e. 0.05

Now it is clear, p-value < alpha

Hence, we reject the null hypothesis, i.e. average spending of males & females is significantly different. Concluding that there is a potential difference in spending behaviour across Genders.

Consequently, the confidence intervals for the average amount spent by males and females do not overlap.

Since the intervals do not overlap, this indicates a significant difference in spending habits. Businesses can leverage this by using marketing strategies to each gender.

As males spend more on average, marketing could focus on products or promotions that appeal more to male consumers.

Offer gender-specific promotions or discounts to maximize revenue based on identified spending behaviour.

b. Report whether the confidence intervals for the average amount spent by married and unmarried (computed using all the data) overlap. How can Walmart leverage this conclusion to make changes or improvements?

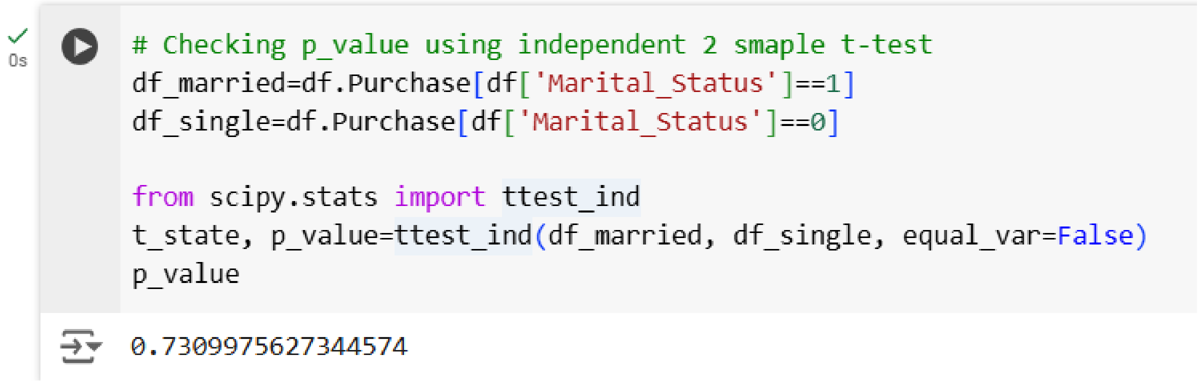
Ans. Determining overlap of confidence interval of Purchase for married & single people.

If the interval overlap, there may not be significant difference between in average spending between married & single people.

Checking for difference in average spending across Marital Status using t-test:

Assumption: Ho-Average spending of married & single people is similar.

Ha-Average spending of married & single people is significantly different.



```
# Checking p_value using independent 2 smaple t-test
df_married=df.Purchase[df['Marital_Status']==1]
df_single=df.Purchase[df['Marital_Status']==0]

from scipy.stats import ttest_ind
t_state, p_value=ttest_ind(df_married, df_single, equal_var=False)
p_value
```

```
0.7309975627344574
```

For 95% confidence level, alpha=1-0.95, i.e. 0.05

Now it is clear, p-value > alpha

Hence, we fail to reject the null hypothesis, i.e. average spending of married & single people is similar. Concluding that there is no significant difference in spending behaviour of married people & single people.

Consequently, the confidence intervals for the average amount spent by married & single people are overlapping.

This suggests that the average spending is similar across male & female people. Marketing strategies can target both male & female similarly, focusing on factors that influence spending universally rather than Marital status specific appeals.

Offer universal promotions or discounts to maximize revenue based on identified spending behaviour.

c. Report whether the confidence intervals for the average amount spent by different age groups (computed using all the data) overlap. How can Walmart leverage this conclusion to make changes or improvements?

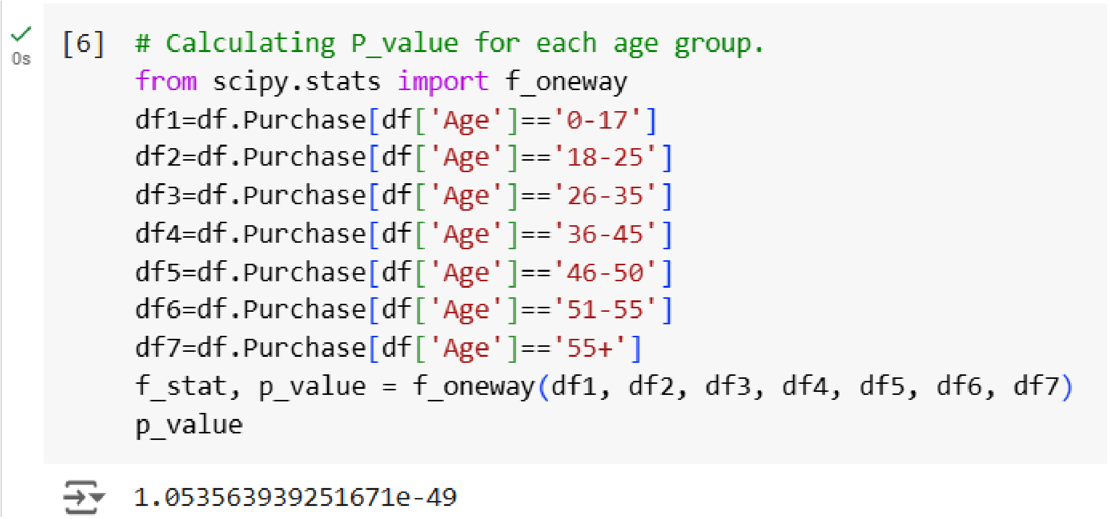
Ans. Determining overlap of confidence interval of Purchase for each age groups.

If the interval overlap, there may not be significant difference between in average spending across each age groups.

Checking for difference in average spending across each age group using ANOVA.

Assumption: Ho-Average spending by each age group is similar.

Ha-Average spending by each age group is significantly different.



```
[6] # Calculating P_value for each age group.
    from scipy.stats import f_oneway
    df1=df.Purchase[df['Age']=='0-17']
    df2=df.Purchase[df['Age']=='18-25']
    df3=df.Purchase[df['Age']=='26-35']
    df4=df.Purchase[df['Age']=='36-45']
    df5=df.Purchase[df['Age']=='46-50']
    df6=df.Purchase[df['Age']=='51-55']
    df7=df.Purchase[df['Age']=='55+']
    f_stat, p_value = f_oneway(df1, df2, df3, df4, df5, df6, df7)
    p_value
```

```
1.053563939251671e-49
```

For 95% confidence level, alpha=1-0.95, i.e. 0.05

Now it is clear, p-value < alpha

Hence, we reject the null hypothesis, i.e. average spending by each age group is significantly different. Concluding that there is a potential difference in spending behaviour across age groups.

Consequently, the confidence intervals for the average amount spent by each age group do not overlap.

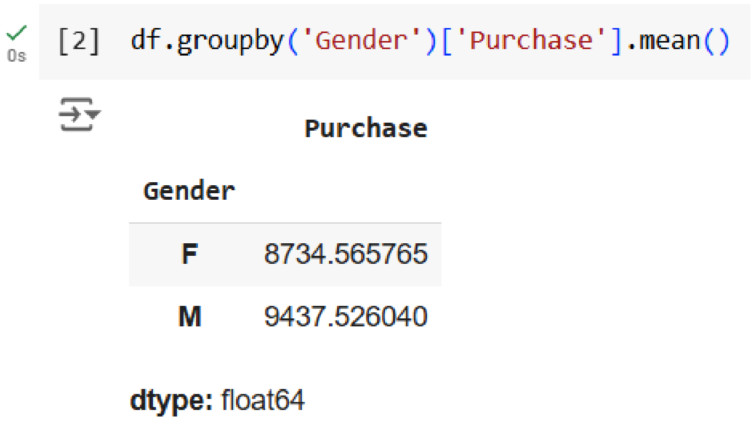
Since the intervals do not overlap, this indicates a significant difference in spending habits. Businesses can leverage this by using marketing strategies to each age group.

Offer age group-specific promotions or discounts to maximize revenue based on identified spending behaviour.

8. Recommendations:

a. Write a detailed recommendation from the analysis that you have done.

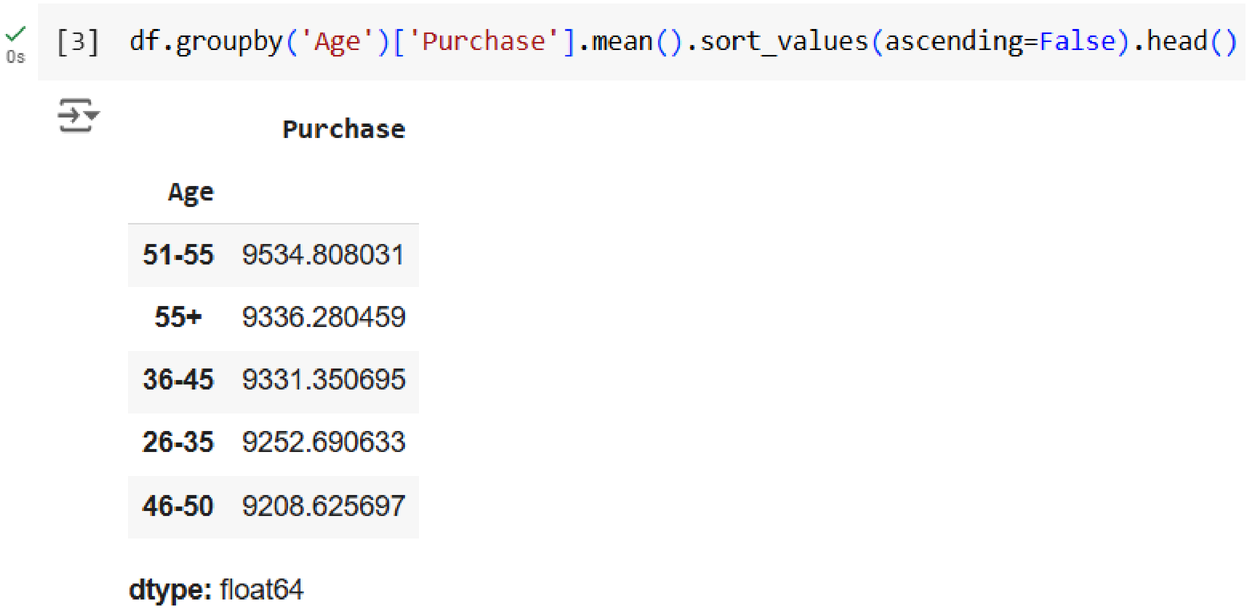
- As males spend more on average, marketing could focus on products or promotions that appeal more to male consumers.



```
[2] df.groupby('Gender')['Purchase'].mean()
```

```
        Purchase
Gender
     F  8734.565765
     M  9437.526040

dtype: float64
```

- Offer gender-specific promotions or discounts to maximize revenue based on identified spending behaviour.
- As the people in age group (51-55) are purchasing more, some promotional offer or discount should be given to then to increase the business.

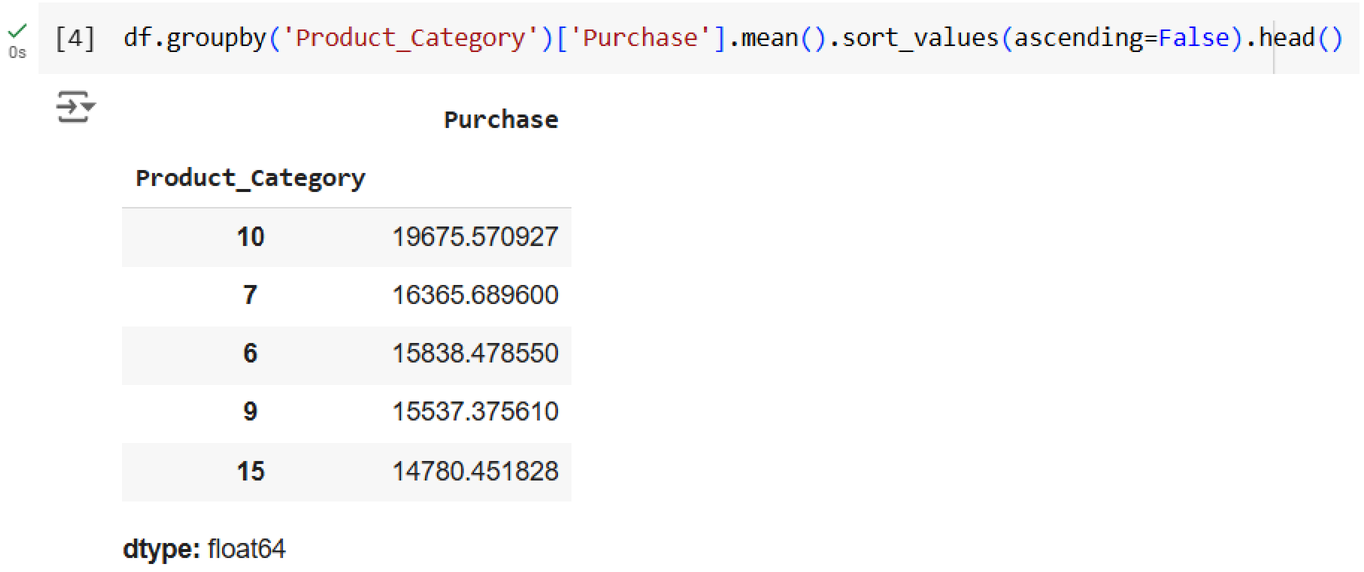


```
[3] df.groupby('Age')['Purchase'].mean().sort_values(ascending=False).head()
```

```
       Purchase
  Age
51-55  9534.808031
  55+  9336.280459
36-45  9331.350695
26-35  9252.690633
46-50  9208.625697

dtype: float64
```

- As people having age >26, are purchasing more therefore some promotional offer (age group specific) should be given to them to maximize revenue.

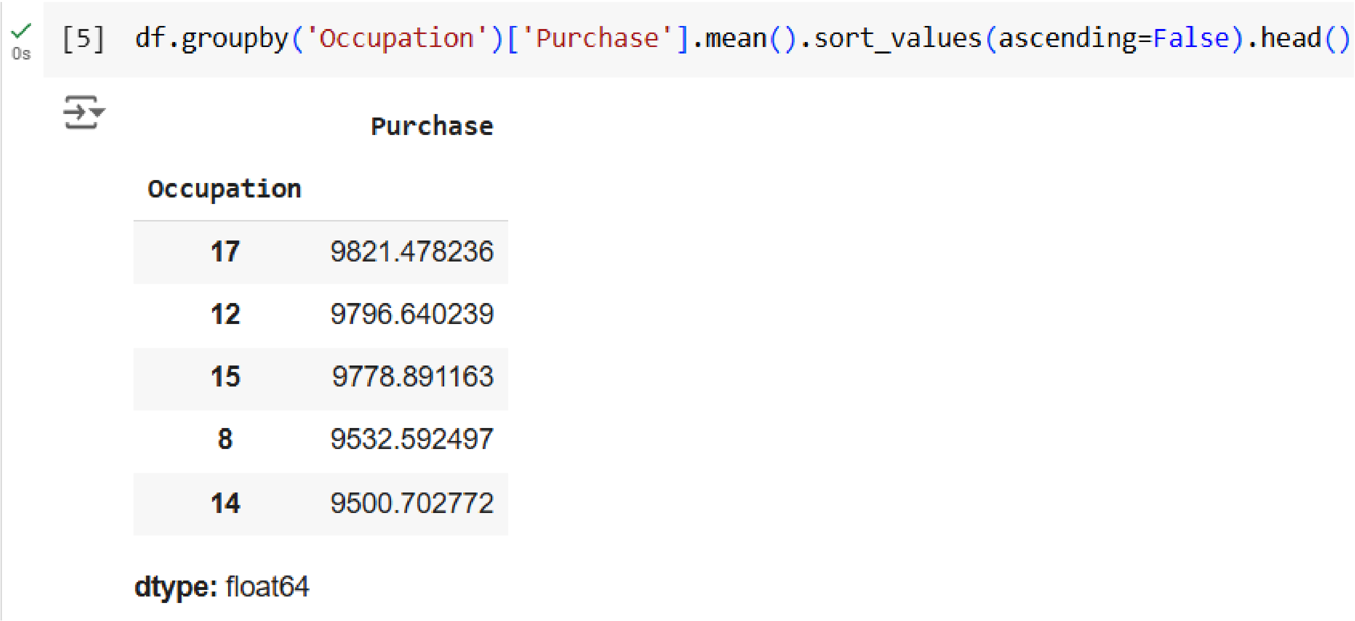
- Product Categories (6,7,9,10,15) are getting purchased more, so some offers must be there to promote product specific revenue.



```
df.groupby('Product_Category')['Purchase'].mean().sort_values(ascending=False).head()
```

```
                    Purchase
Product_Category
              10  19675.570927
               7  16365.689600
               6  15838.478550
               9  15537.375610
              15  14780.451828

dtype: float64
```

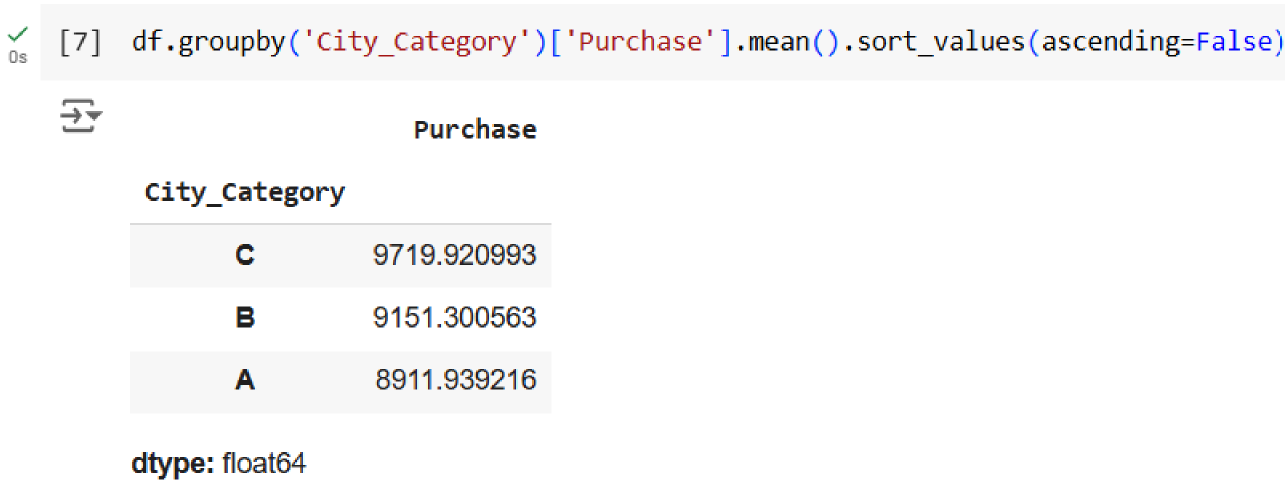
- People having occupation (8,12,14,15,17) are purchasing more, so some occupation-specific offers/discount should be given to maximize revenue.



```
df.groupby('Occupation')['Purchase'].mean().sort_values(ascending=False).head()
```

```
                Purchase
Occupation
        17   9821.478236
        12   9796.640239
        15   9778.891163
         8   9532.592497
        14   9500.702772

dtype: float64
```

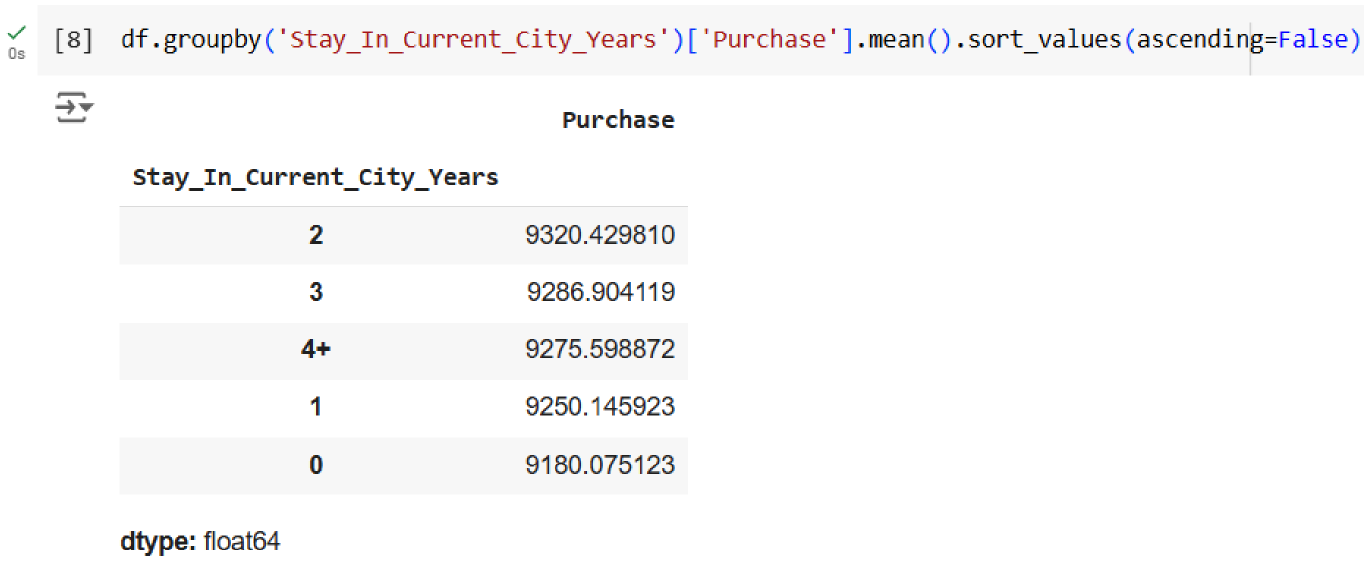
- People belonging to ‘C’ city category are purchasing more therefore some discount can be given to them to increase business.



```
[7] df.groupby('City_Category')['Purchase'].mean().sort_values(ascending=False)
```

```
                      Purchase
City_Category
      C            9719.920993
      B            9151.300563
      A            8911.939216

dtype: float64
```

- People staying in current city for 2 or 3 years are purchasing more therefore some promotional offer/discount should be given to them to maximize revenue.



```
[8] df.groupby('Stay_In_Current_City_Years')['Purchase'].mean().sort_values(ascending=False)
```

```
                                       Purchase
Stay_In_Current_City_Years
              2                     9320.429810
              3                     9286.904119
              4+                    9275.598872
              1                     9250.145923
              0                     9180.075123

dtype: float64
```

- Develop loyalty programs or incentives that cater to the spending patterns of each Gender, Age, Product category etc. enhancing customer satisfaction and retention.