About Walmart

Walmart is an American multinational retail corporation that operates a chain of supercenters, discount departmental stores, and grocery stores from the United States. Walmart has more than 100 million customers worldwide.

Business Problem

The Management team at Walmart Inc. wants to analyze the customer purchase behavior (specifically, purchase amount) against the customer's gender and the various other factors to help the business make better decisions. They want to understand if the spending habits differ between male and female customers: Do women spend more on Black Friday than men? (Assume 50 million customers are male and 50 million are female).

Importing Libraries

In [29]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Reading Data

In [5]:
```
df = pd.read_csv('https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/293/original/walmart_data.
df.head()
```

Out[5]:

	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	A	2	0	3	8370
1	1000001	P00248942	F	0-17	10	A	2	0	1	15200
2	1000001	P00087842	F	0-17	10	A	2	0	12	1422
3	1000001	P00085442	F	0-17	10	A	2	0	12	1057
4	1000002	P00285442	M	55+	16	C	4+	0	8	7969

1. Data Analysis

I. Dimension of data

In [6]:
```
df.shape
```

Out[6]:
```
(550068, 10)
```

II. Datatype of all atributes

In [7]:
```
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
```

III. Statstical Summary of Data

In [8]:
```
df.describe()
```

Out[8]:

	User_ID	Occupation	Marital_Status	Product_Category	Purchase
count	5.500680e+05	550068.000000	550068.000000	550068.000000	550068.000000
mean	1.003029e+06	8.076707	0.409653	5.404270	9263.968713
std	1.727592e+03	6.522660	0.491770	3.936211	5023.065394
min	1.000001e+06	0.000000	0.000000	1.000000	12.000000
25%	1.001516e+06	2.000000	0.000000	1.000000	5823.000000
50%	1.003077e+06	7.000000	0.000000	5.000000	8047.000000
75%	1.004478e+06	14.000000	1.000000	8.000000	12054.000000
max	1.006040e+06	20.000000	1.000000	20.000000	23961.000000

In [9]:
```
df.describe(include="object")
```

Out[9]:

	Product_ID	Gender	Age	City_Category	Stay_In_Current_City_Years
count	550068	550068	550068	550068	550068
unique	3631	2	7	3	5
top	P00265242	M	26-35	B	1
freq	1880	414259	219587	231173	193821

IV. Missing Value Detection

In [10]:
```
df.isna().sum()
```

Out[10]:
```
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
```

V. Converting Categorical attributes

In [14]:
```
df['City_Category'] = pd.Categorical(df['City_Category'])
df['Product_Category'] = pd.Categorical(df['Product_Category'])
df['Marital_Status'] = pd.Categorical(df['Marital_Status'])
df['Marital_Status'] = df['Marital_Status'].cat.rename_categories({0:"Unmarried",1:"Married"})
df['Gender'] = pd.Categorical(df['Gender'])
df['Gender'] = df['Gender'].cat.rename_categories({"F":"Female","M":"Male"})
df.head()
```

Out[14]:

	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category	Purchase
0	1000001	P00069042	Female	0-17	10	A	2	Unmarried	3	8370
1	1000001	P00248942	Female	0-17	10	A	2	Unmarried	1	15200
2	1000001	P00087842	Female	0-17	10	A	2	Unmarried	12	1422
3	1000001	P00085442	Female	0-17	10	A	2	Unmarried	12	1057
4	1000002	P00285442	Male	55+	16	C	4+	Unmarried	8	7969

In [16]:
```
df['City_Category'].unique()
```

Out[16]:
```
['A', 'C', 'B']
Categories (3, object): ['A', 'B', 'C']
```

In [17]:
```
df['Product_Category'].unique()
```

Out[17]:
```
[3, 1, 12, 8, 5, ..., 10, 17, 9, 20, 19]
Length: 20
Categories (20, int64): [1, 2, 3, 4, ..., 17, 18, 19, 20]
```

In [15]:
```
df['Marital_Status'].unique()
```

Out[15]:
```
['Unmarried', 'Married']
Categories (2, object): ['Unmarried', 'Married']
```

In [18]:
```
df['Gender'].unique()
```

Out[18]:
```
['Female', 'Male']
Categories (2, object): ['Female', 'Male']
```

2. Non- Graphical Analysis

I. Value Counts

In [19]:
```
df['City_Category'].value_counts()
```

Out[19]:
```
B    231173
C    171175
A    147720
Name: City_Category, dtype: int64
```

In [20]:
```
df['Product_Category'].value_counts()
```

Out[20]:
```
5     150933
1     140378
8     113925
11     24287
2      23864
6      20466
3      20213
4      11753
16      9828
15      6290
13      5549
10      5125
12      3947
7       3721
18      3125
20      2550
19      1603
14      1523
17       578
9        410
Name: Product_Category, dtype: int64
```

In [21]:
```
df['Marital_Status'].value_counts()
```

Out[21]:
```
Unmarried    324731
Married      225337
Name: Marital_Status, dtype: int64
```

In [22]:
```
df['Gender'].value_counts()
```

Out[22]:
```
Male      414259
Female    135809
Name: Gender, dtype: int64
```

In [23]:
```
df['Age'].value_counts()
```

Out[23]:
```
26-35    219587
36-45    110013
18-25     99660
46-50     45701
51-55     38501
55+       21504
0-17      15102
Name: Age, dtype: int64
```

II. Unique Attributes

In [24]:
```
df.nunique()
```

Out[24]:
```
User_ID                        5891
Product_ID                     3631
Gender                            2
Age                               7
Occupation                       21
City_Category                     3
Stay_In_Current_City_Years        5
Marital_Status                    2
Product_Category                 20
Purchase                      18105
dtype: int64
```

In [25]:
```
df['Age'].unique()
```

Out[25]:
```
array(['0-17', '55+', '26-35', '46-50', '51-55', '36-45', '18-25'],
      dtype=object)
```

In [26]:
```
df['Occupation'].unique()
```

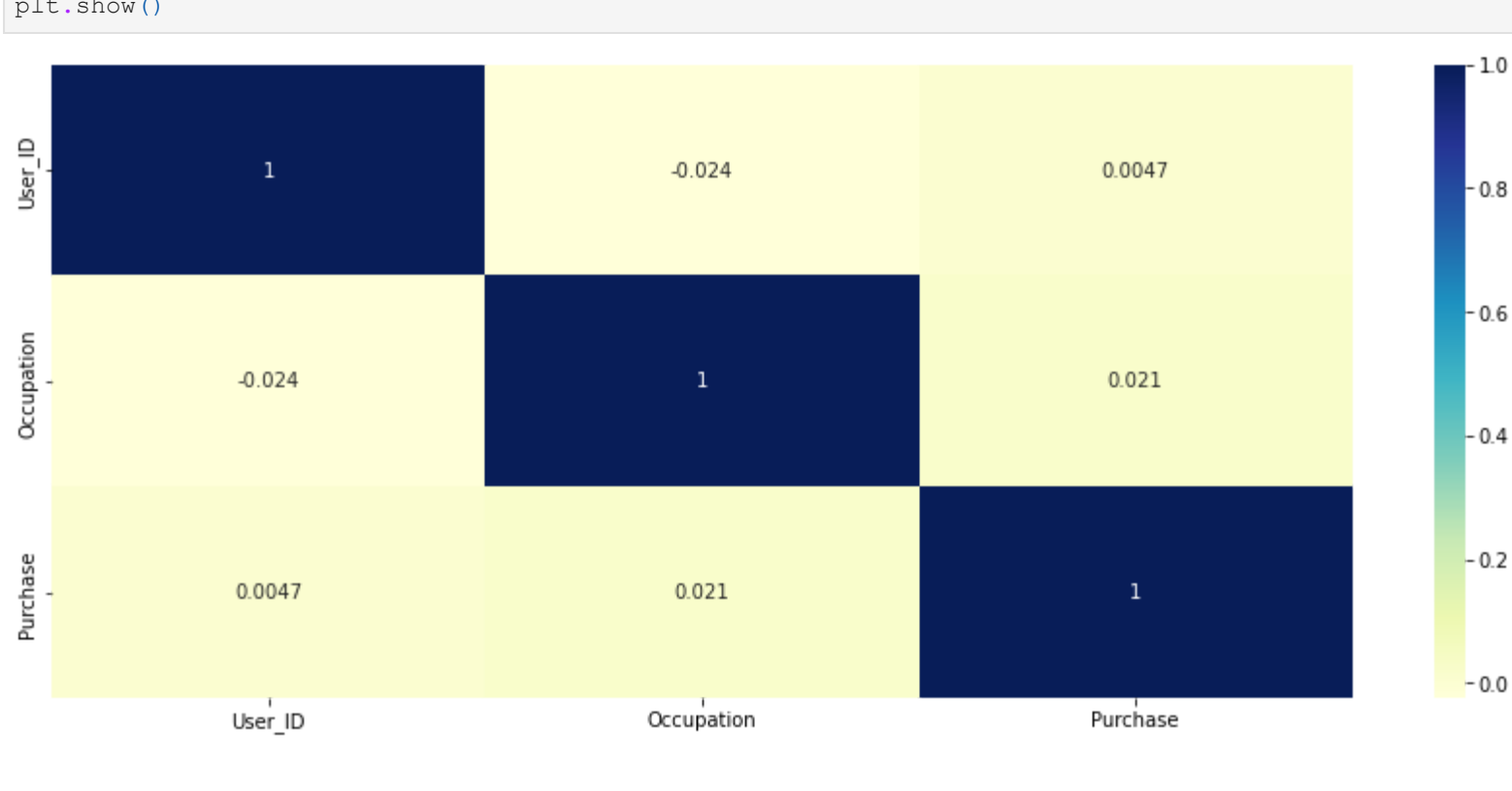
Out[26]:
```
array([10, 16, 15,  7, 20,  9,  1, 12, 17,  0,  3,  4, 11,  8, 19,  2, 18,
        5, 14, 13,  6], dtype=int64)
```

In [27]:
```
df['Stay_In_Current_City_Years'].unique()
```

Out[27]:
```
array(['2', '4+', '3', '1', '0'], dtype=object)
```

3. Visual Analysis

In [38]:
```
df_corr = df.corr()
plt.figure(figsize=(15,6))
sns.heatmap(df_corr, cmap="YlGnBu", annot=True)
plt.show()
```



1. Very weak relation between the features observed from the above plot
2. No strong influence of the features on Purchase

In [31]:
```
df[["Purchase"]].agg(['mean','median', 'sum']).rename(index = {"mean":"Mean_Purchase","median":"Median_Purchase
                                                               "sum":"Total_Purchase"})
```

Out[31]:

	Purchase
Mean_Purchase	9.263969e+03
Median_Purchase	8.047000e+03
Total_Purchase	5.095813e+09

In [32]:
```
plt.figure(figsize=(16,6))
ax = sns.boxplot(data=df,x='Purchase')
plt.title("Distribution of Purchase amount")
ax.set_xticks(np.arange(12,24000,1000))
plt.xticks(rotation=45)
plt.show()
```

1. The median purchase amount on Black Friday lies around 8012 dollars.
2. The interquartile range of purchase amount lies between 5762 dollars to 12012 dollars.
3. There many outliers present at the right side of the boxplot denoting that some people spending large amount on purchasing.

Removal of Outliers

In [33]:
```
Q1 = np.percentile(df['Purchase'], 25, interpolation = 'midpoint')
Q3 = np.percentile(df['Purchase'], 75, interpolation = 'midpoint')
IQR = Q3 - Q1
df_new = df[(df['Purchase'] <= (Q3+1.5*IQR)) & (df['Purchase'] >= (Q1-1.5*IQR))]
df_new.shape
```

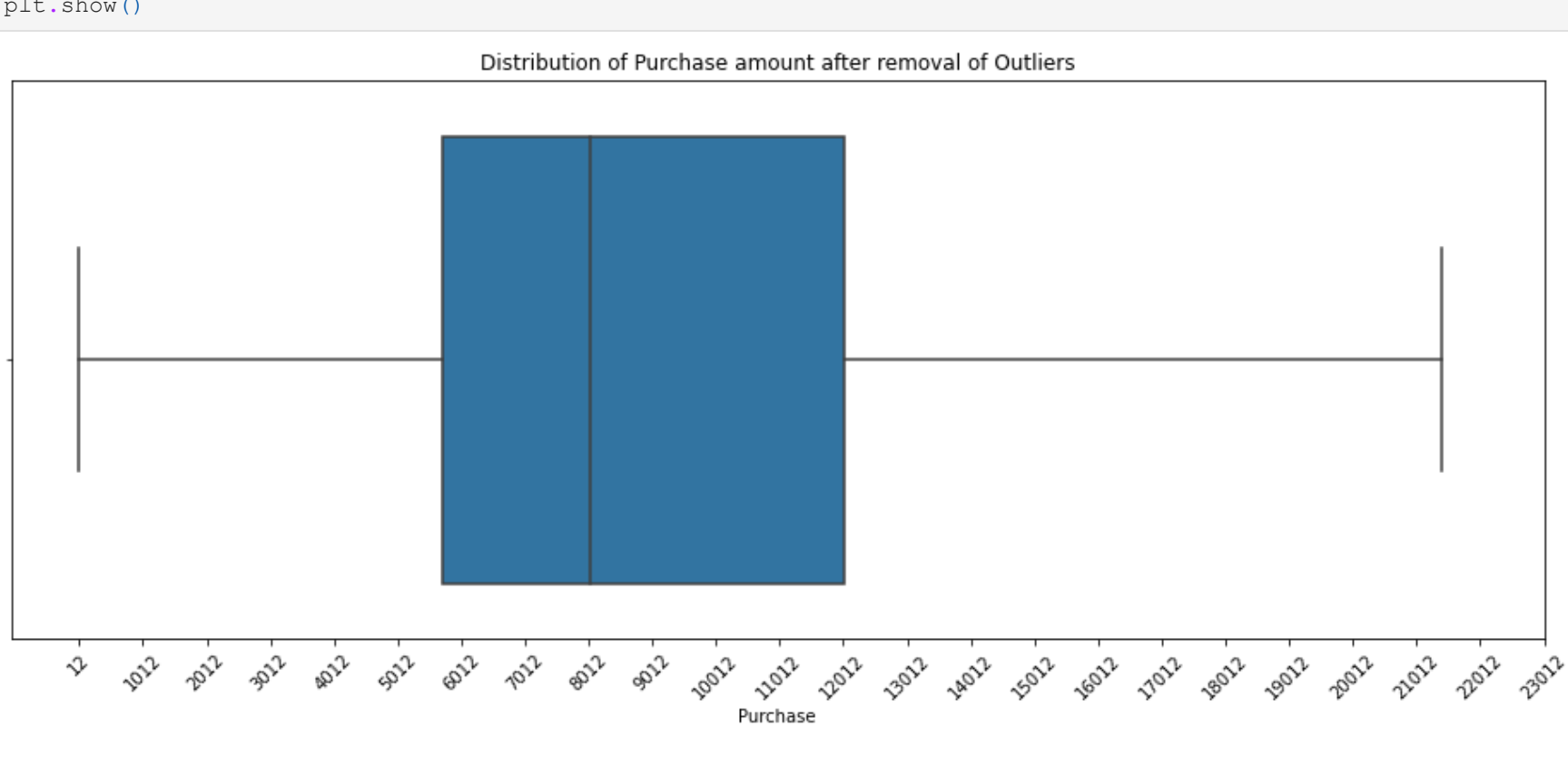
Out[33]:
```
(547391, 10)
```

In [34]:
```
df_new[["Purchase"]].agg(['mean','median', 'sum']).rename(index = {"mean":"Mean_Purchase","median":"Median_Purc
                                                                   "sum":"Total_Purchase"})
```

Out[34]:

	Purchase
Mean_Purchase	9.195627e+03
Median_Purchase	8.038000e+03
Total_Purchase	5.033604e+09

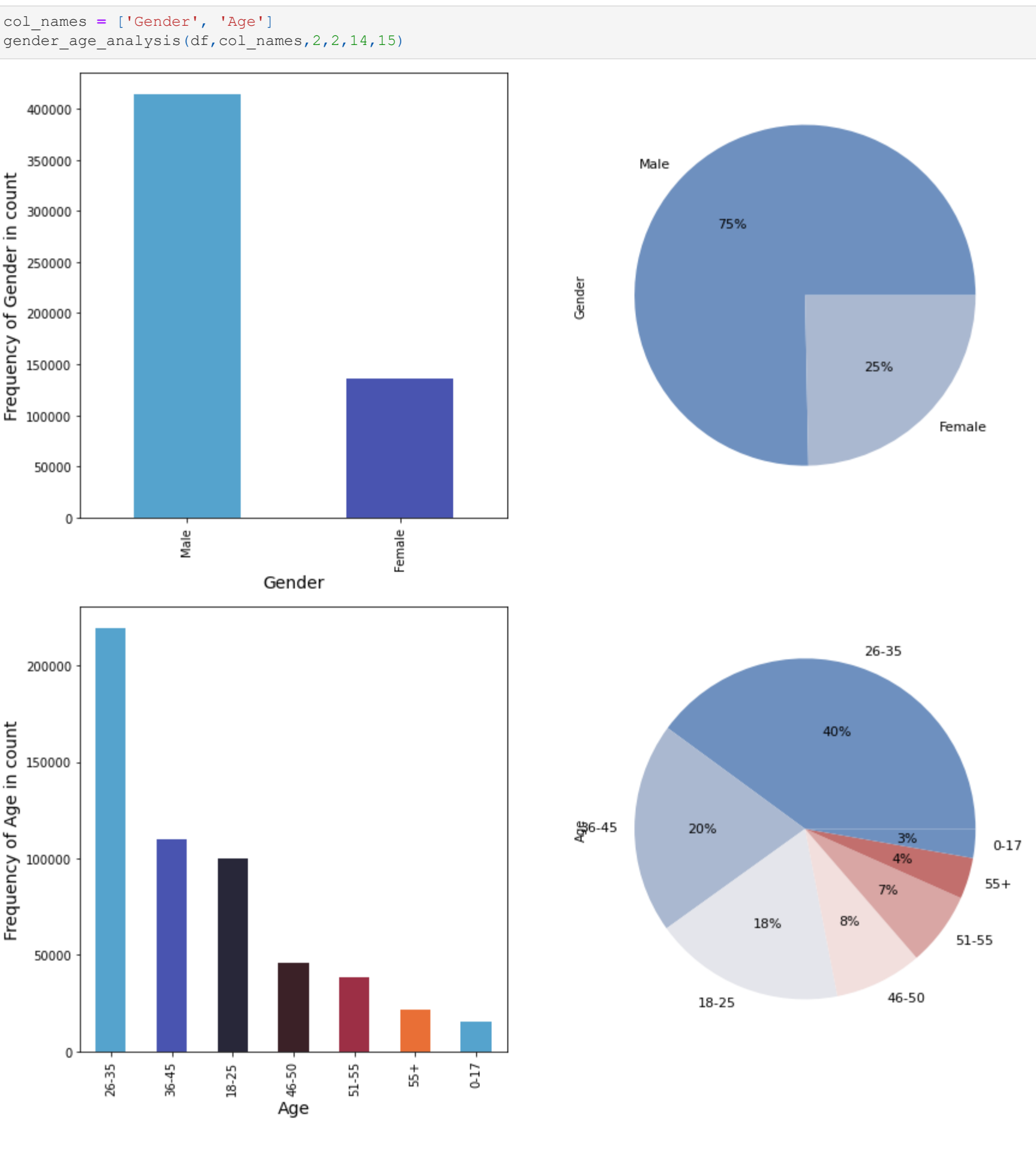
In [35]:
```
plt.figure(figsize=(16,6))
ax = sns.boxplot(data=df_new,x='Purchase')
plt.title("Distribution of Purchase amount after removal of Outliers")
ax.set_xticks(np.arange(12,24000,1000))
plt.xticks(rotation=45)
plt.show()
```



1. The median and interquartile range almost remain same after the removal of outliers.
2. There is also very slight decrease in mean purchase amount after removing the outliers

In [52]:
```
def gender_age_analysis(df, colnames, nrows=2,mcols=2,width=15,height=10, sortbyindex=False):
    fig , ax = plt.subplots(nrows,mcols,figsize=(width,height))
    fig.set_facecolor(color = 'white')
    string = "Frequency of "
    rows = 0
    for colname in colnames:
        count = (df[colname].value_counts(normalize=True)*100)
        count1 = (df[colname].value_counts())
        string += colname + ' in count'
        if sortbyindex:
            count = count.sort_index()
            count1 = count1.sort_index()
        count1.plot.bar(color=sns.color_palette("icefire"),ax=ax[rows][0])
        ax[rows][0].set_ylabel(string, fontsize=14,family = "DejaVu Sans")
        ax[rows][0].set_xlabel(colname, fontsize=14,family = "DejaVu Sans")
        count.plot.pie(colors = sns.color_palette("vlag"),autopct='%0.0f%%',
                       textprops={'fontsize': 11,'family':"DejaVu Sans"},ax=ax[rows][1])
        string = "Frequency of "
        rows += 1
```

In [53]:
```
col_names = ['Gender', 'Age']
gender_age_analysis(df,col_names,2,2,14,15)
```



1. 75% of the purchase transactions are done by men and 25% of the purchase transactions are done by women
2. 40% of the transactions are done by age group 26-35, 20% are done by age group 36-45, and 18% are done by age group 18-25

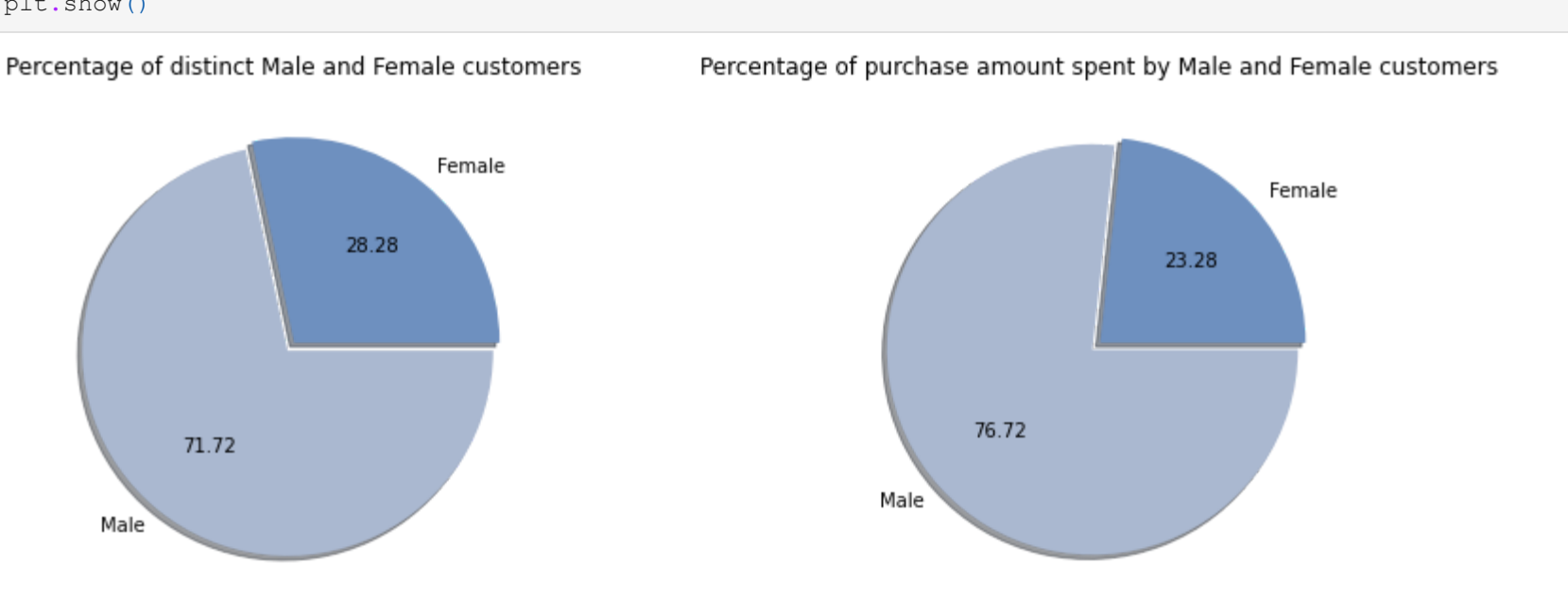
Analysis on Gender

In [58]:
```
gender=df.groupby('Gender').agg({'User_ID':'nunique','Purchase':'sum'}).reset_index().rename(columns={"User_ID"
gender.head()
```

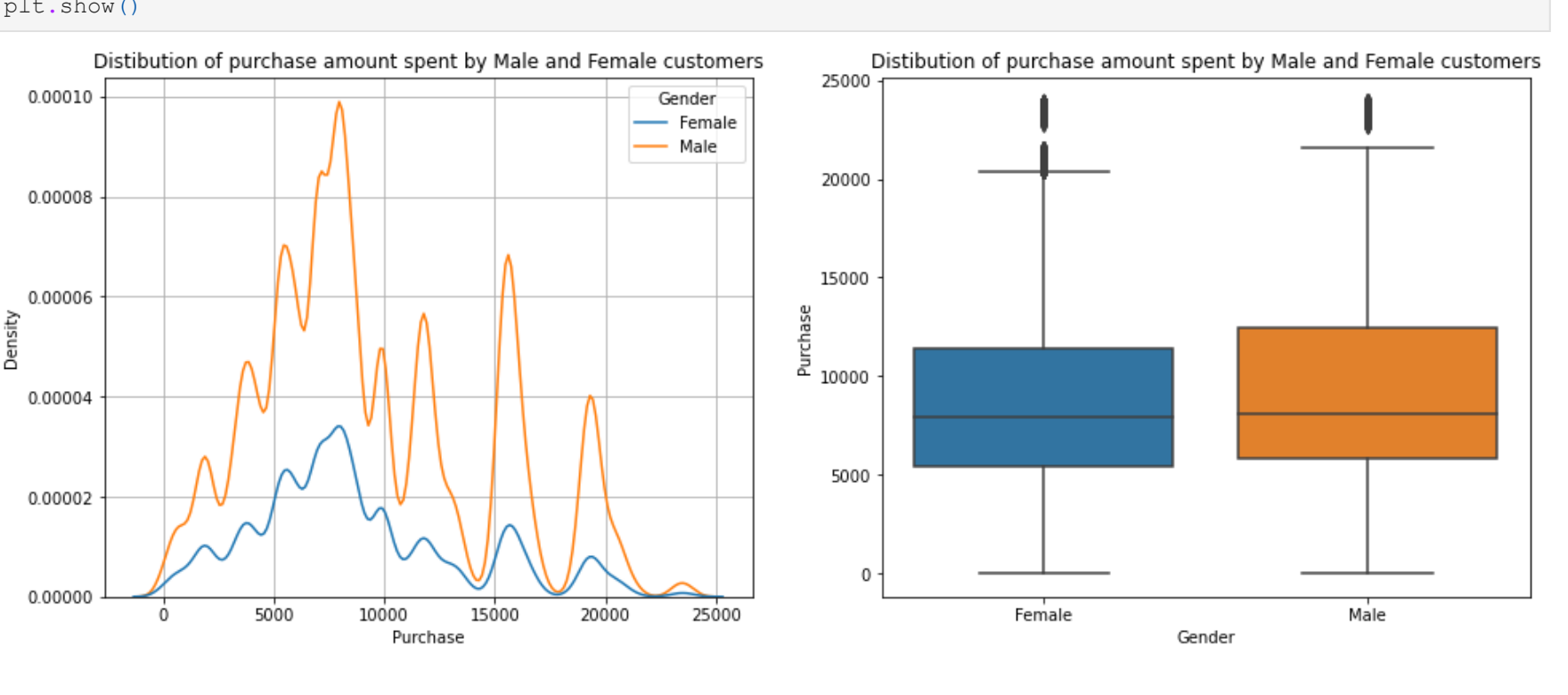
Out[58]:

	Gender	Number of people	Total_Purchase_amount
0	Female	1666	1186232642
1	Male	4225	3909580100

In [62]:
```
plt.figure(figsize=(14,5))
plt.subplot(1,2,1)
colors = sns.color_palette('vlag')[0:2]
plt.pie(x= gender['Number of people'], labels= gender['Gender'], colors= colors, autopct= "%0.2f",
explode= (0,0.05), shadow= True)
plt.title("Percentage of distinct Male and Female customers")
plt.subplot(1,2,2)
colors = sns.color_palette('vlag')[0:2]
plt.pie(x= gender['Total_Purchase_amount'], labels= gender['Gender'], colors= colors, autopct= "%0.2f",
explode= (0,0.05), shadow= True)
plt.title("Percentage of purchase amount spent by Male and Female customers")
plt.show()
```



In [63]:
```
plt.figure(figsize=(16,6))
plt.subplot(1,2,1)
sns.kdeplot(data=df, x="Purchase", hue='Gender')
plt.title("Distibution of purchase amount spent by Male and Female customers")
plt.grid()
plt.subplot(1,2,2)
sns.boxplot(data=df, x='Gender', y='Purchase')
plt.title("Distibution of purchase amount spent by Male and Female customers")
plt.show()
```



1. In the kdeplot also, we can clearly see that the density of men spending money on purchasing is always higher than the density of women.
2. The distribution of purchase amount spent by men are almost similar to the distribution of purchase amount spent by women, only the density of men are more than women that's why distribution of purchase amount spent by men is higher than that of women.
3. The median purchase amount spent by both men and women are comparable but the interquartile range of the purchase amount spent by men are wider than that of women.
4. The number of outliers are more in the distribution of purchase amount spent by women than that of men.

In [64]:
```
df.groupby('Gender')["Purchase"].agg(['mean','median', 'sum']).reset_index().rename(columns={"mean":"Mean_Purch
```

Out[64]:

	Gender	Mean_Purchase	Median_Purchase	Total_Purchase
0	Female	8734.565765	7914.0	1186232642
1	Male	9437.526040	8098.0	3909580100

From the above data frame, we can see that the median purchase amount spent by both men and women are comparable but there is significant difference between the mean purchase amount spent by men and women.

Confidence Interval on Mean Purchase by Male and Female customers

In [65]:
```
df_F = df[df['Gender']=='Female']
df_M = df[df['Gender']=='Male']
print(df_F.shape)
print(df_M.shape)
```

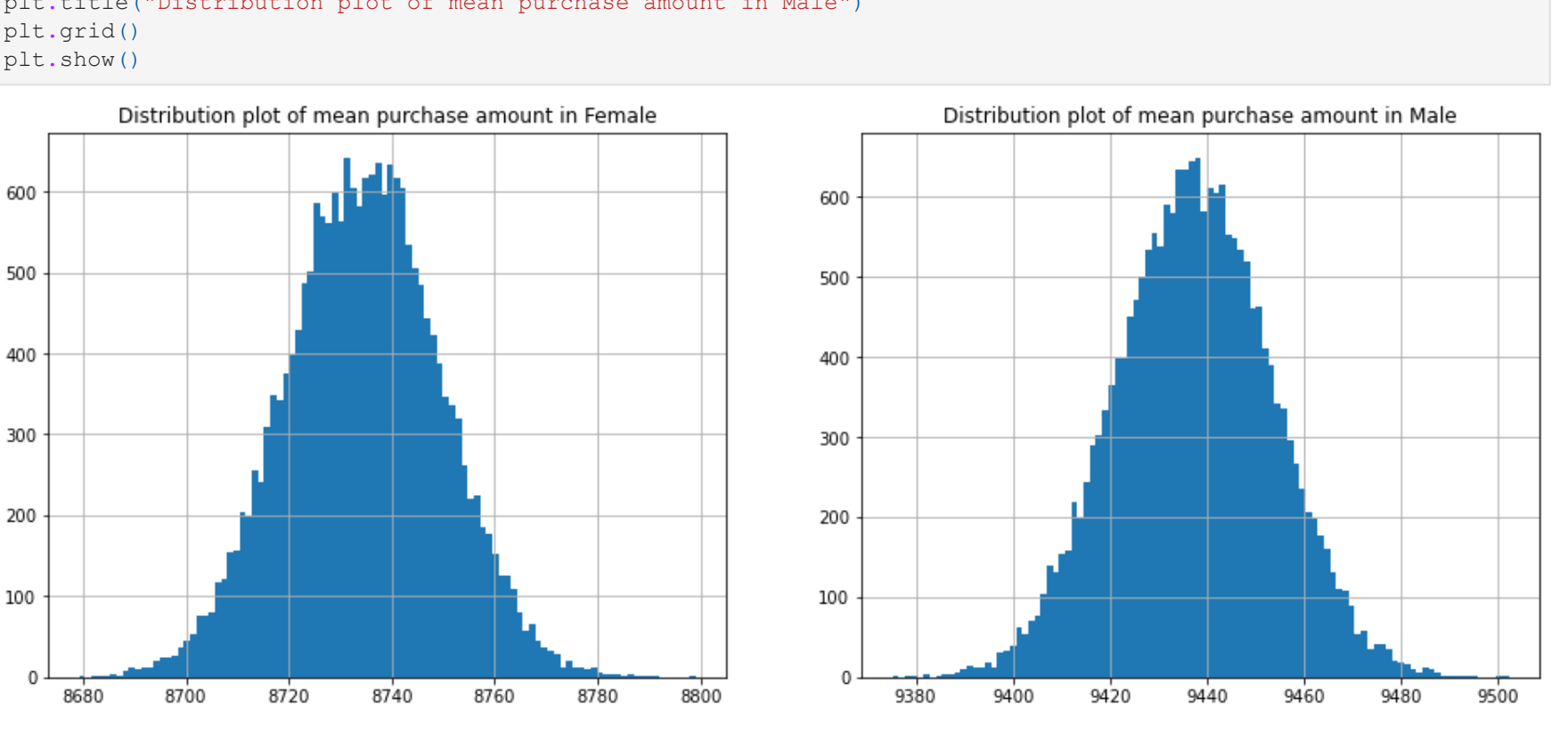
```
(135809, 10)
(414259, 10)
```

Here we take the sample size = 100k for both men and women to calculate the confidence interval of average spend by both 50 millions male and female customers. For computational feasibility we take the r=20000.

In [70]:
```
r = 20000
size = 100000
bs_means_F = np.empty(r)
bs_means_M = np.empty(r)
for i in range(r):
    bs_sample_F = np.random.choice(df_F['Purchase'],size=size)
    bs_sample_M = np.random.choice(df_M['Purchase'],size=size)
    bs_means_F[i] = np.mean(bs_sample_F)
    bs_means_M[i] = np.mean(bs_sample_M)
```

```
CPU times: total: 1min 12s
Wall time: 1min 12s
```

In [69]:
```
plt.figure(figsize=(16,6))
plt.subplot(1,2,1)
plt.hist(bs_means_F,bins=100)
plt.title("Distribution plot of mean purchase amount in Female")
plt.grid()
plt.subplot(1,2,2)
plt.hist(bs_means_M,bins=100)
plt.title("Distribution plot of mean purchase amount in Male")
plt.grid()
plt.show()
```



1. After implementing the bootstrap sampling the distribution of average spend of both male and female which we get is almost Normal distribution(which we also check below using QQ-plot too).
2. For most of the women, we can see their average spend lies between 8700 dollars to 8760 dollars.
3. Similarly, for most of the men, their average spend lies between 9400 dollars to 9460 dollars.

In [73]:
```
CI_F = [np.round(np.percentile(bs_means_F,5),2), np.round(np.percentile(bs_means_F,95),2)]
CI_M = [np.round(np.percentile(bs_means_M,5),2), np.round(np.percentile(bs_means_M,95),2)]
print("The 90% Confidence Interval of mean purchase amount spent by Female customers is ",CI_F)
print("The 90% Confidence Interval of mean purchase amount spent by Male customers is ",CI_M)
print()
CI_F = [np.round(np.percentile(bs_means_F,2.5),2), np.round(np.percentile(bs_means_F,97.5),2)]
CI_M = [np.round(np.percentile(bs_means_M,2.5),2), np.round(np.percentile(bs_means_M,97.5),2)]
print("The 95% Confidence Interval of mean purchase amount spent by Female customers is ",CI_F)
print("The 95% Confidence Interval of mean purchase amount spent by Male customers is ",CI_M)
print()
CI_F = [np.round(np.percentile(bs_means_F,0.5),2), np.round(np.percentile(bs_means_F,99.5),2)]
CI_M = [np.round(np.percentile(bs_means_M,0.5),2), np.round(np.percentile(bs_means_M,99.5),2)]
print("The 99% Confidence Interval of mean purchase amount spent by Female customers is ",CI_F)
print("The 99% Confidence Interval of mean purchase amount spent by Male customers is ",CI_M)
```

```
The 90% Confidence Interval of mean purchase amount spent by Female customers is  [8709.63, 8759.4]
The 90% Confidence Interval of mean purchase amount spent by Male customers is  [9411.02, 9463.89]

The 95% Confidence Interval of mean purchase amount spent by Female customers is  [8705.13, 8764.36]
The 95% Confidence Interval of mean purchase amount spent by Male customers is  [9405.95, 9469.15]

The 99% Confidence Interval of mean purchase amount spent by Female customers is  [8695.55, 8773.12]
The 99% Confidence Interval of mean purchase amount spent by Male customers is  [9395.75, 9478.78]
```

We have calculate the 90%, 95% and 99% confidence interval on average spends for both men and women above.

1. This confidence intervals tell about the average spend by all the 50 millions of each men and women population by just using a sample size of 100k of each.
2. For all the 3 cases, the confidence intervals of average spend by men and women are not overlapping. The confidence intervals of average spend for women is lower than that of men.
3. As the increase the percentage the width of the confidence intervals of average spend by both men and women increases.

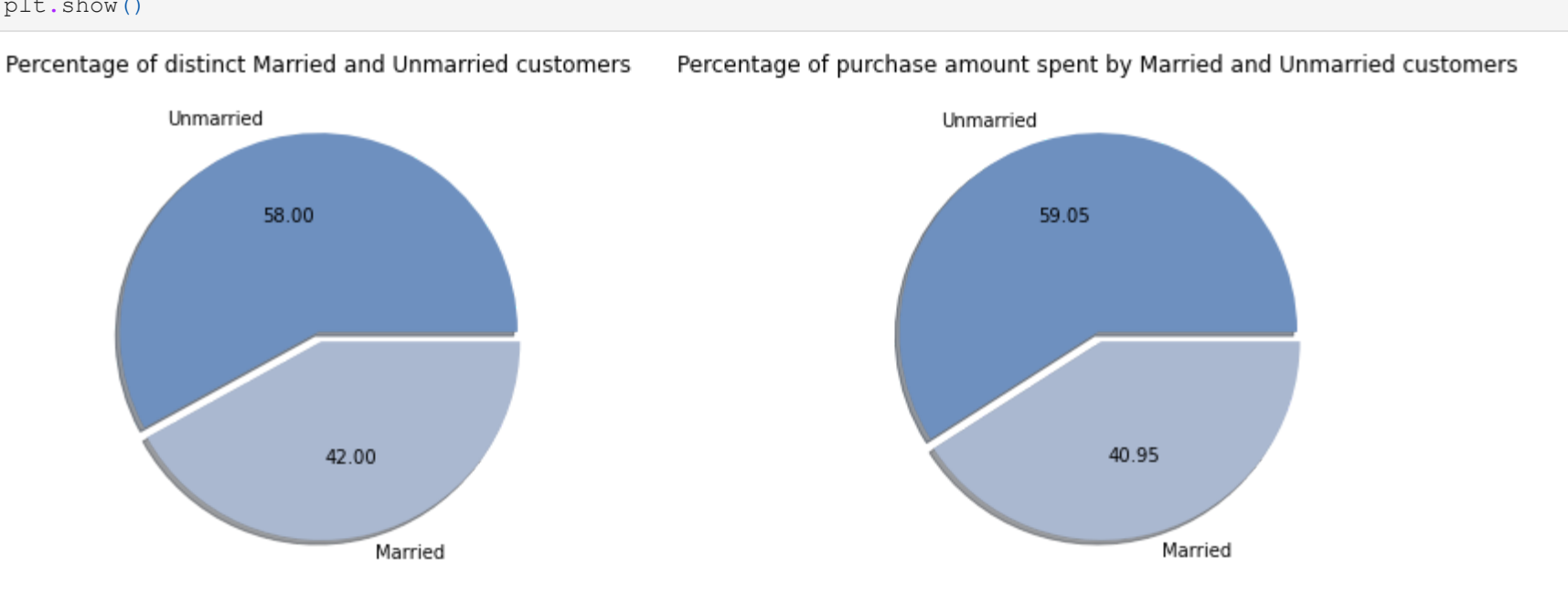
Analysis on Marital Status

In [74]:
```
marital_status=df.groupby('Marital_Status').agg({'User_ID':'nunique','Purchase':'sum'}).reset_index().rename(co
marital_status.head()
```

Out[74]:

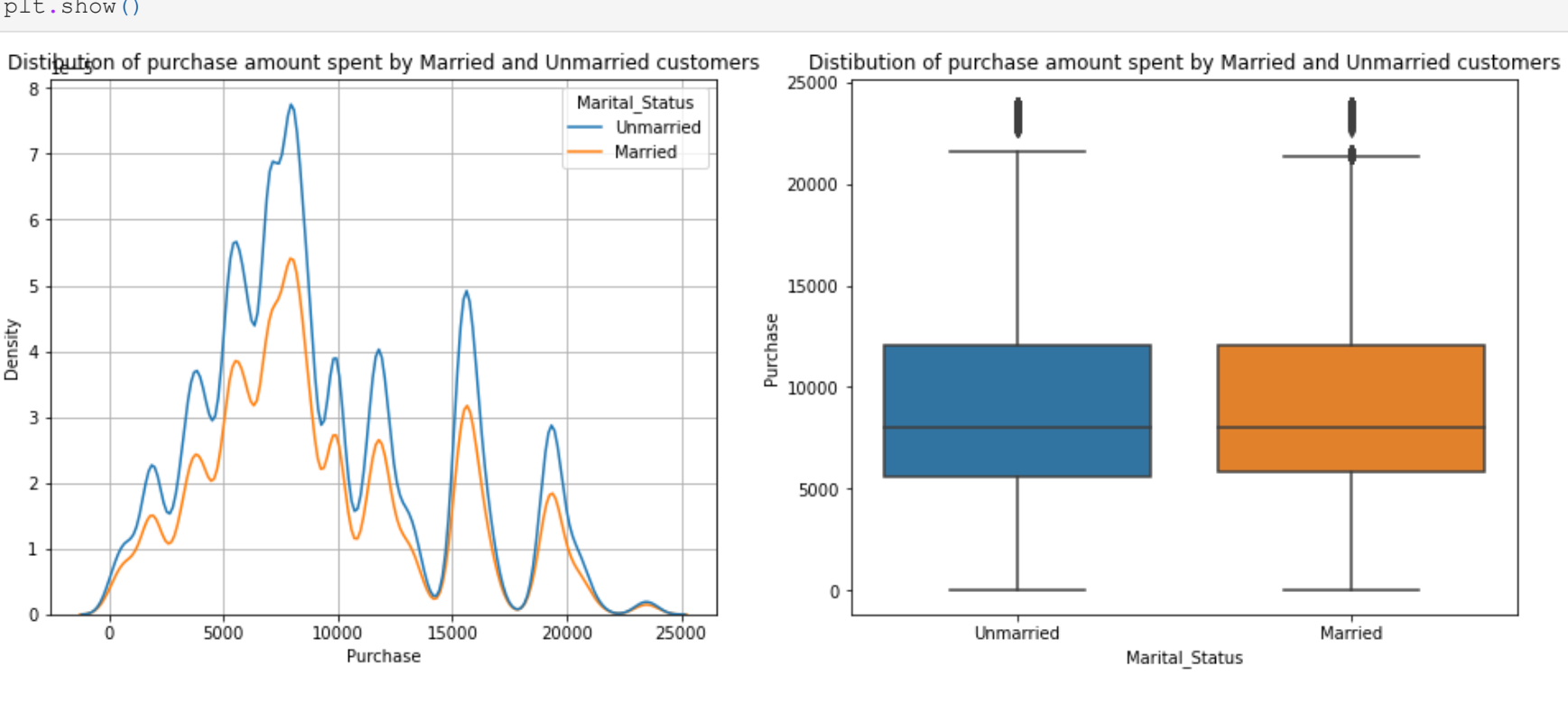
	Marital_Status	Number of people	Total_Purchase_amount
0	Unmarried	3417	3008927447
1	Married	2474	2086885295

In [76]:
```
plt.figure(figsize=(14,5))
plt.subplot(1,2,1)
colors = sns.color_palette('vlag')[0:2]
plt.pie(x= marital_status['Number of people'], labels= marital_status['Marital_Status'], colors= colors, autopc
explode= (0,0.05), shadow= True)
plt.title("Percentage of distinct Married and Unmarried customers")
plt.subplot(1,2,2)
colors = sns.color_palette('vlag')[0:2]
plt.pie(x= marital_status['Total_Purchase_amount'], labels= marital_status['Marital_Status'], colors= colors,
autopct= "%0.2f", explode= (0,0.05), shadow= True)
plt.title("Percentage of purchase amount spent by Married and Unmarried customers")
plt.show()
```



1. As in this pie chart, we can see that 58% of the customers who purchased something from Walmart on Good Friday are Unmarried and rest 42% are Married.
2. Similarly, around 59% of total purchase amount are spent by Unmarried and 41% spent by Married.

In [77]:
```
plt.figure(figsize=(16,6))
plt.subplot(1,2,1)
sns.kdeplot(data=df, x="Purchase", hue='Marital_Status')
plt.grid()
plt.title("Distibution of purchase amount spent by Married and Unmarried customers")
plt.subplot(1,2,2)
sns.boxplot(data=df, x='Marital_Status', y='Purchase')
plt.title("Distibution of purchase amount spent by Married and Unmarried customers")
plt.show()
```



1. In the kdeplot also, we can clearly see that the density of Unmarried spending money on purchasing is slightly higher than the density of married always.
2. The distribution of purchase amount spent by Unmarried are almost similar to the distribution of purchase amount spent by Married.
3. The median purchase amount spent by both Unmarried and Married are comparable and the interquartile range of the purchase amount spent by Unmarried and Married are also comparable.
4. The number of outliers are more in the distribution of purchase amount spent by Married than that of Unmarried.

In [78]:
```
df.groupby('Marital_Status')["Purchase"].agg(['mean','median', 'sum']).reset_index().rename(columns={"mean":"Me
```

Out[78]:

	Marital_Status	Mean_Purchase	Median_Purchase	Total_Purchase
0	Unmarried	9265.907619	8044.0	3008927447
1	Married	9261.174574	8051.0	2086885295

From the above data frame, we can see that both the median and mean purchase amount spent by both married and unmarried are comparable.

Confidence Interval on Mean Purchase by Unmarried and Married customers

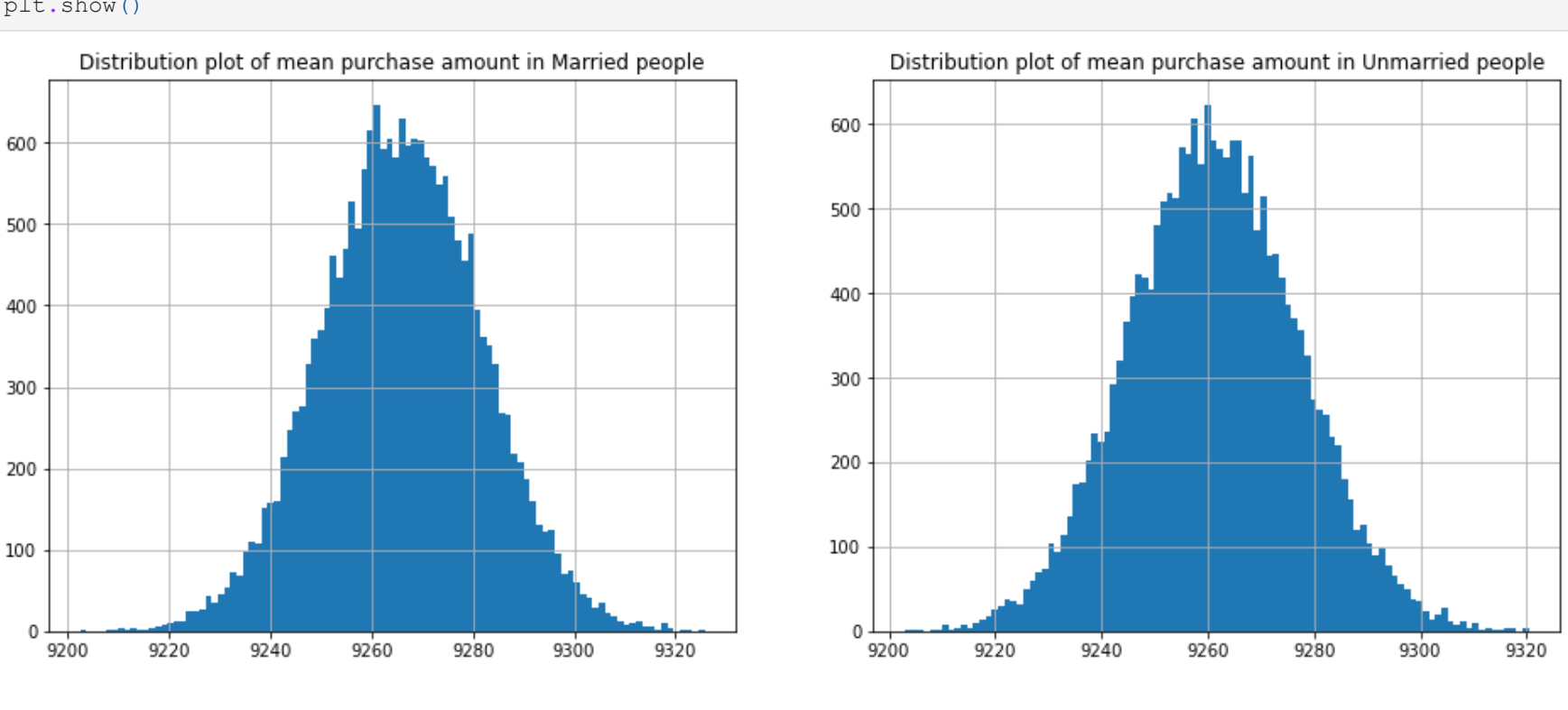
In [79]:
```
df_unmarried = df[df['Marital_Status']=='Unmarried']
df_married = df[df['Marital_Status']=='Married']
print(df_unmarried.shape)
print(df_married.shape)
```

```
(324731, 10)
(225337, 10)
```

Here we take the sample size = 100k for both men and women to calculate the confidence interval of average spend by both 50 millions male and female customers. For computational feasibility we take the r=20000.

In [80]:
```
r = 20000
size = 100000
bs_means_mar = np.empty(r)
bs_means_unmar = np.empty(r)
for i in range(r):
    bs_sample_mar = np.random.choice(df_unmarried['Purchase'],size=size)
    bs_sample_unmar = np.random.choice(df_married['Purchase'],size=size)
    bs_means_mar[i] = np.mean(bs_sample_mar)
    bs_means_unmar[i] = np.mean(bs_sample_unmar)
```

In [81]:
```
plt.figure(figsize=(16,6))
plt.subplot(1,2,1)
plt.hist(bs_means_mar,bins=100)
plt.title("Distribution plot of mean purchase amount in Married people")
plt.grid()
plt.subplot(1,2,2)
plt.hist(bs_means_unmar,bins=100)
plt.title("Distribution plot of mean purchase amount in Unmarried people")
plt.grid()
plt.show()
```



1. After implementing the bootstrap sampling the distribution of average spend of both married and unmarried which we get is almost Normal distribution(which we also check below using QQ-plot too).
2. For most of the married, we can see their average spend lies between 9230 dollars to 9300 dollars.
3. Similarly, for most of the unmarried, their average spend lies between 9225 dollars to 9290 dollars.

In [83]:
```
CI_married = [np.round(np.percentile(bs_means_mar,5),2), np.round(np.percentile(bs_means_mar,95),2)]
CI_unmarried = [np.round(np.percentile(bs_means_unmar,5),2), np.round(np.percentile(bs_means_unmar,95),2)]
print("The 90% Confidence Interval of mean purchase amount spent by Married customer is ",CI_married)
print("The 90% Confidence Interval of mean purchase amount spent by Unmarried customer is ",CI_unmarried)
print()
CI_married = [np.round(np.percentile(bs_means_mar,2.5),2), np.round(np.percentile(bs_means_mar,97.5),2)]
CI_unmarried = [np.round(np.percentile(bs_means_unmar,2.5),2), np.round(np.percentile(bs_means_unmar,97.5),2)]
print("The 95% Confidence Interval of mean purchase amount spent by Married customer is ",CI_married)
print("The 95% Confidence Interval of mean purchase amount spent by Unmarried customer is ",CI_unmarried)
print()
CI_married = [np.round(np.percentile(bs_means_mar,0.5),2), np.round(np.percentile(bs_means_mar,99.5),2)]
CI_unmarried = [np.round(np.percentile(bs_means_unmar,0.5),2), np.round(np.percentile(bs_means_unmar,99.5),2)]
print("The 99% Confidence Interval of mean purchase amount spent by Married customer is ",CI_married)
print("The 99% Confidence Interval of mean purchase amount spent by Unmarried customer is ",CI_unmarried)
```

```
The 90% Confidence Interval of mean purchase amount spent by Married customer is  [9240.06, 9292.0]
The 90% Confidence Interval of mean purchase amount spent by Unmarried customer is  [9235.12, 9287.52]

The 95% Confidence Interval of mean purchase amount spent by Married customer is  [9235.06, 9296.92]
The 95% Confidence Interval of mean purchase amount spent by Unmarried customer is  [9230.09, 9292.88]

The 99% Confidence Interval of mean purchase amount spent by Married customer is  [9225.34, 9306.62]
The 99% Confidence Interval of mean purchase amount spent by Unmarried customer is  [9220.85, 9303.13]
```

We have calculate the 90%, 95% and 99% confidence interval on average spends for both married and unmarried above.

1. This confidence intervals tell about the average spend by entire population of married and unmarried by just using a sample size of 100k each.
2. For all the 3 cases, the confidence intervals of average spend by married and unmarried people are overlapping and are quite comparable.
3. As the increase the percentage the width of the confidence intervals of average spend by both married and unmarried people increases.

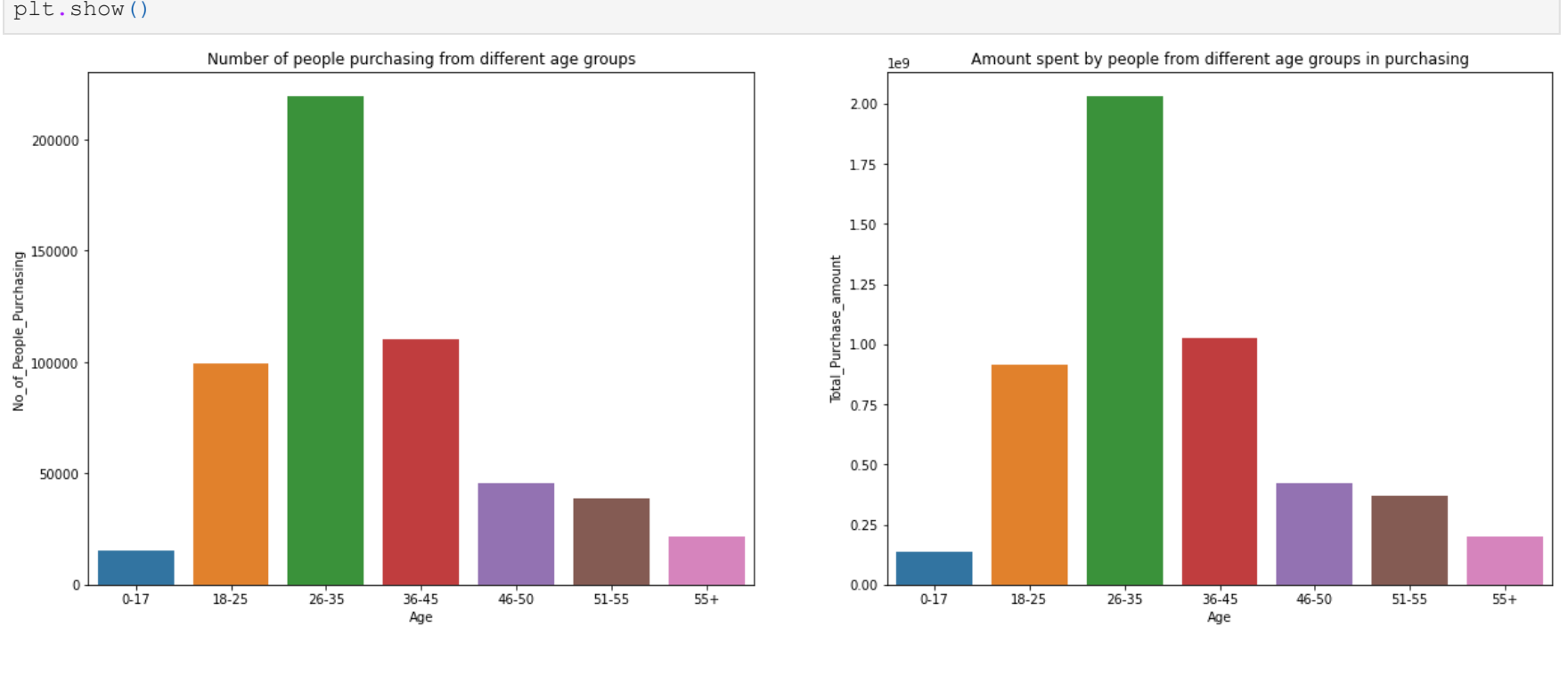
Analysis on Age

In [84]:
```
agewise = df.groupby('Age').agg({'User_ID':'count','Purchase':'sum'}).reset_index().rename(columns={"User_ID":
agewise
```

Out[84]:

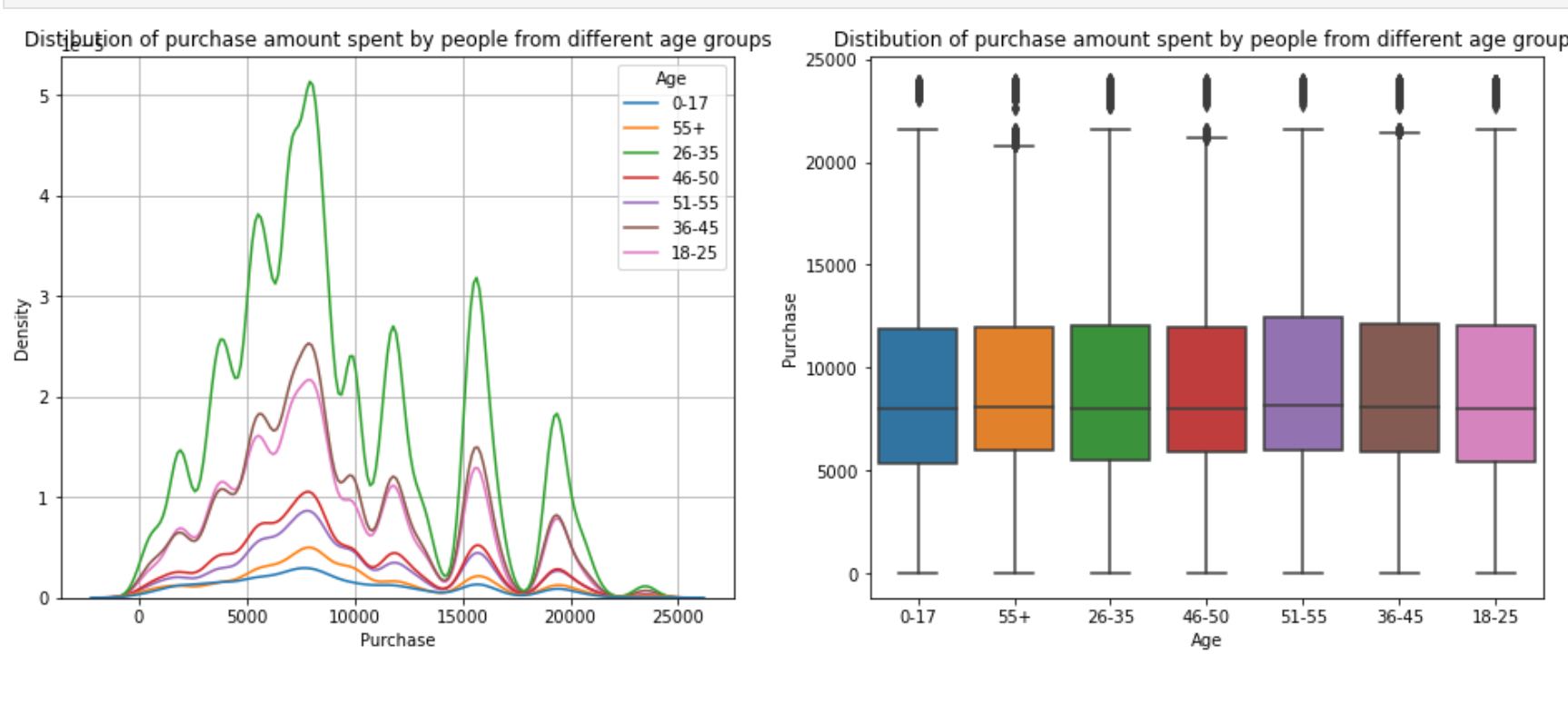
	Age	No_of_People_Purchasing	Total_Purchase_amount
0	0-17	15102	134913183
1	18-25	99660	913848675
2	26-35	219587	2031770578
3	36-45	110013	1026569884
4	46-50	45701	420843403
5	51-55	38501	367099644
6	55+	21504	200767375

In [85]:
```
plt.figure(figsize=(20,7))
plt.subplot(1,2,1)
sns.barplot(data=agewise,x="Age",y="No_of_People_Purchasing")
plt.title("Number of people purchasing from different age groups")
plt.subplot(1,2,2)
sns.barplot(data=agewise,x="Age",y="Total_Purchase_amount")
plt.title("Amount spent by people from different age groups in purchasing")
plt.subplots_adjust(left=0.1,
bottom=0.1,
right=0.9,
top=0.9,
wspace=0.2,
hspace=0.5)
plt.show()
```



1. The maximum number of people purchasing from Walmart on Good Friday are from age group of 26-35 years.
2. The number of people in age group of 0-17 years is lowest.
3. The number of people in age group of 18-25 years and 36-45 years purchasing from Walmart on Good Friday are also good.
4. Similarly, the total amount spend by people belonging to any age group is directly proportional to the number of customers belong to that age group.

In [86]:
```
plt.figure(figsize=(16,6))
plt.subplot(1,2,1)
sns.kdeplot(data=df, x="Purchase", hue='Age')
plt.title("Distibution of purchase amount spent by people from different age groups")
plt.grid()
plt.subplot(1,2,2)
sns.boxplot(data=df, x='Age', y='Purchase')
plt.title("Distibution of purchase amount spent by people from different age groups")
plt.show()
```



1. In the kdeplot also, we can clearly see that the density of People from age group 26-35 years spending money on purchasing is highest compared to any other age-group people.
2. The density of People from age group 36-45 years and 18-25 years spending money on purchasing are quite comparable.
3. Similarly, the density of People from age group 46-50 years and 51-55 years spending money on purchasing are quite comparable.
4. The density of people belonging to very young age group (0-17 years) and very old age group (55+ years) spending money on purchasing are very low.
5. The median purchase amount spent by all age groups people are almost same and the interquartile range of the purchase amount spent by all age groups are also little high or low (i.e, comparable).
6. Outliers of people spending large amount on shopping are present in all age groups.

In [87]:
```
agewise = df.groupby('Age')["Purchase"].agg(['mean','median', 'sum']).reset_index().rename(columns={"mean":"Mea
agewise
```

Out[87]:

	Age	Mean_Purchase	Median_Purchase	Total_Purchase
0	0-17	8933.464640	7986.0	134913183
1	18-25	9169.663606	8027.0	913848675
2	26-35	9252.690633	8030.0	2031770578
3	36-45	9331.350695	8061.0	1026569884
4	46-50	9208.625697	8036.0	420843403
5	51-55	9534.808031	8130.0	367099644
6	55+	9336.280459	8105.5	200767375

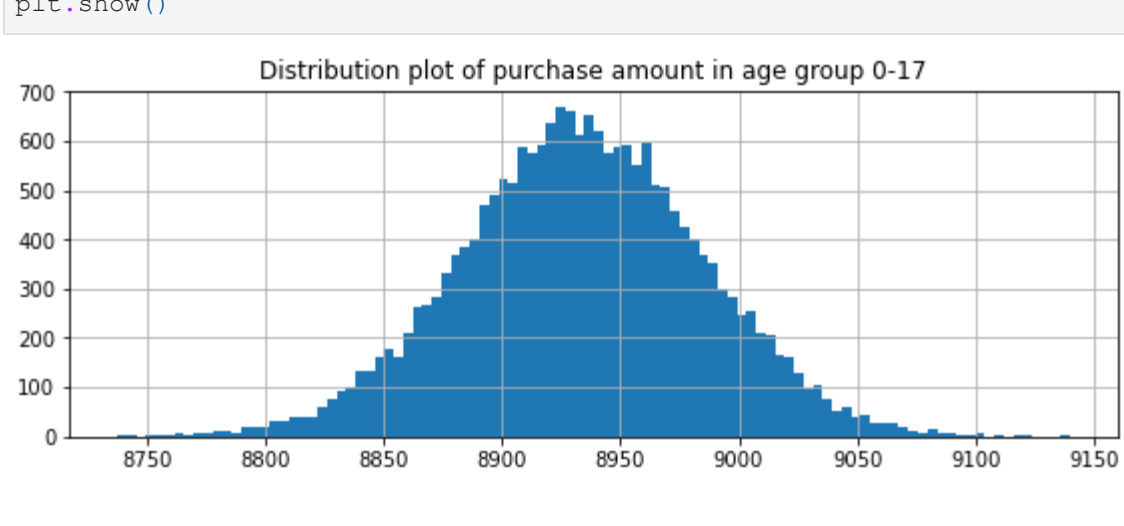
Confidence Interval on Mean Purchase by customers of different Age groups

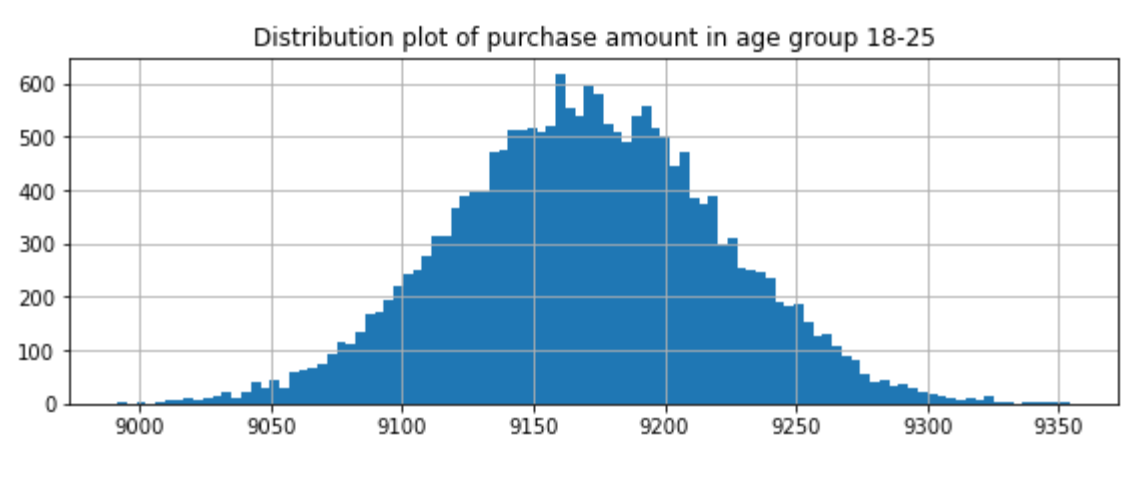
Here we take the sample size = 10k for all age groups people to calculate the confidence interval of average spend by population from each age group. For computational feasibility we take the r=20000.

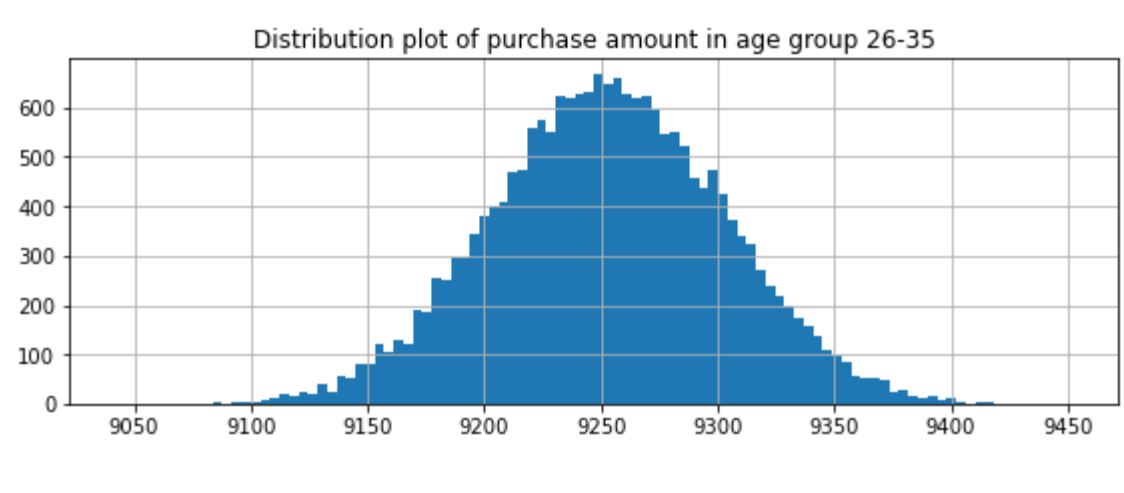
In [88]:
```
bs_means_age = []
count = 1
index = 0
age_group = list(agewise['Age'])
for i in age_group:
    df_subset = df[df['Age']==i]
    print(i," ",df_subset.shape)
    r = 20000
    size = 10000
    bs_means = np.empty(r)
    for j in range(r):
        bs_sample = np.random.choice(df_subset['Purchase'],size=size)
        bs_means[j] = np.mean(bs_sample)
    bs_means_age.append(bs_means)
```

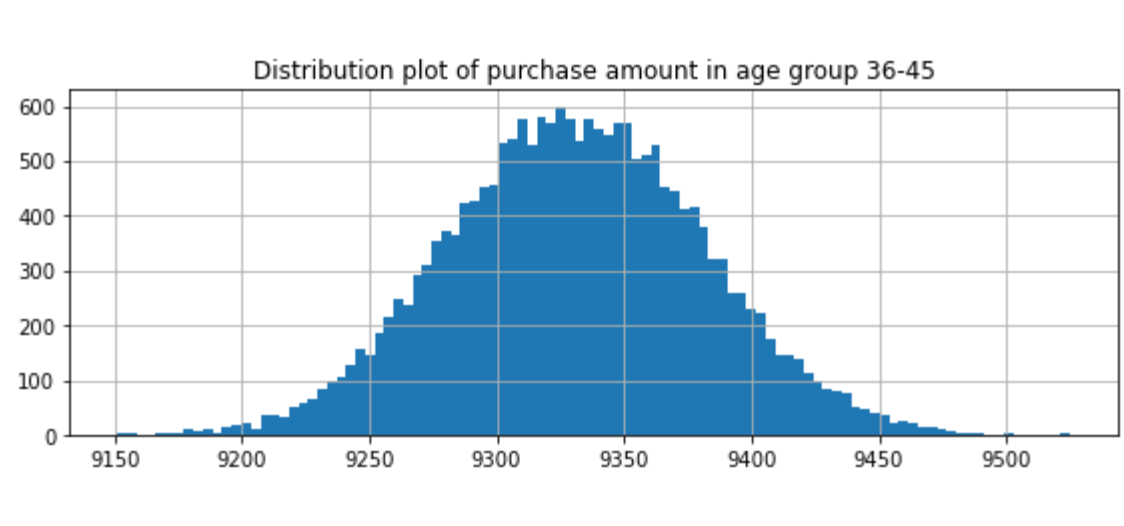
```
0-17   (15102, 10)
18-25   (99660, 10)
26-35   (219587, 10)
36-45   (110013, 10)
46-50   (45701, 10)
51-55   (38501, 10)
55+   (21504, 10)
```

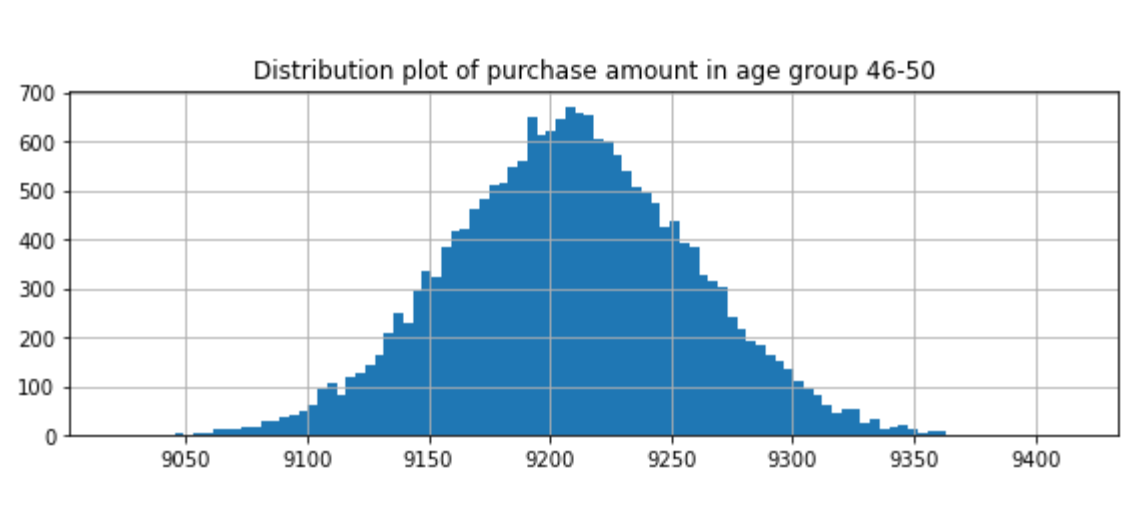
In [89]:
```
plt.figure(figsize=(20,30))
count_1= 1
count_2 = 2
for i in range(len(age_group)):
    plt.subplot(7,2,count_1)
    plt.hist(bs_means_age[i],bins=100)
    plt.title("Distribution plot of purchase amount in age group "+age_group[i])
    plt.grid()

    count_1 += 2
    count_2 += 2
    plt.subplots_adjust(left=0.1,
    bottom=0.1,
    right=0.9,
    top=0.9,
    wspace=0.2,
    hspace=0.5)
plt.show()
```

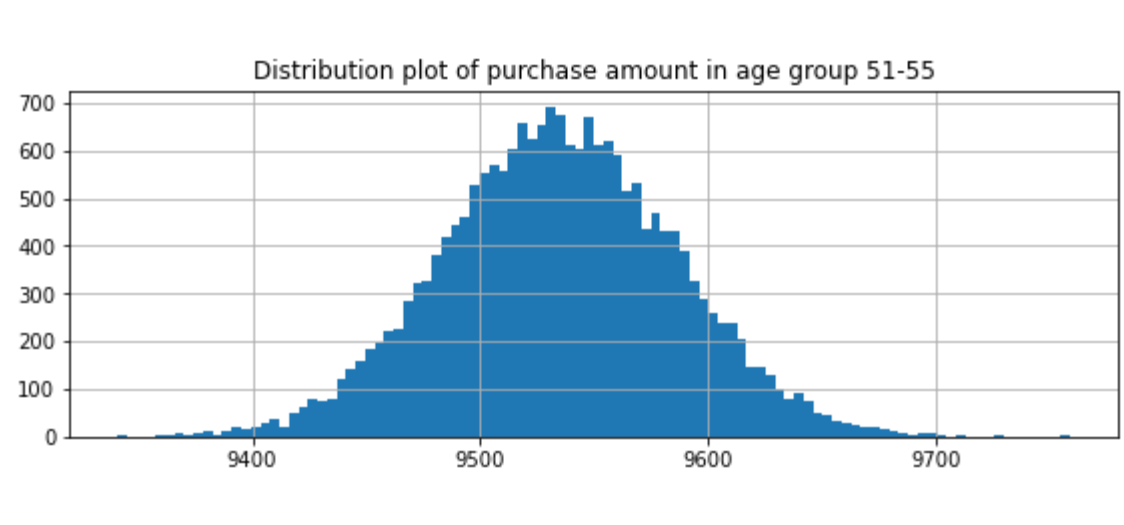


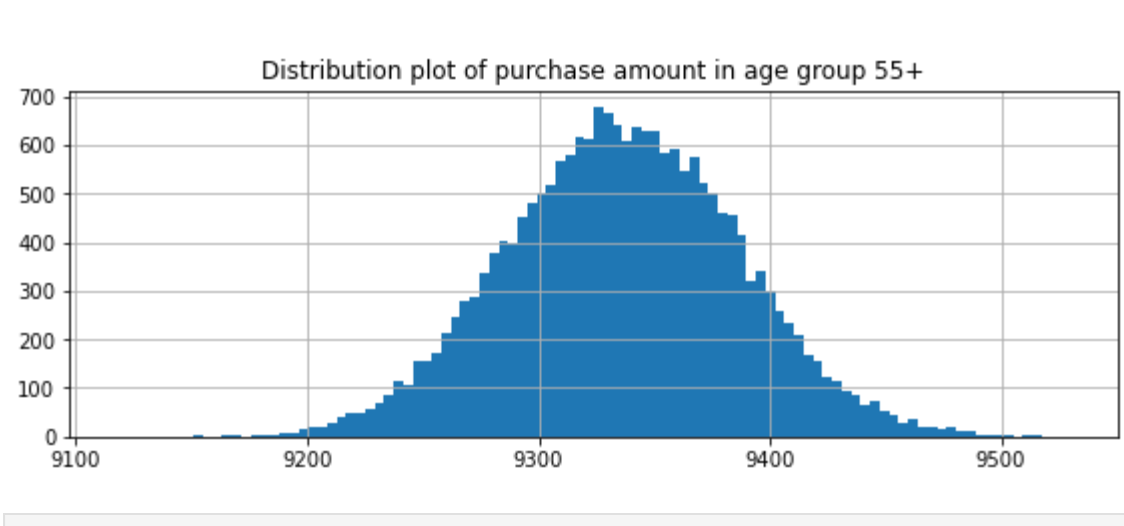












In [91]:
```
for i in range(df['Age'].nunique()):
    CI_age = [np.round(np.percentile(bs_means_age[i],5),2), np.round(np.percentile(bs_means_age[i],95),2)]
    print("The 90% Confidence Interval of mean purchase of people with age ",df['Age'].unique()[i]," is ",CI_ag
print()
for i in range(df['Age'].nunique()):
    CI_age = [np.round(np.percentile(bs_means_age[i],2.5),2), np.round(np.percentile(bs_means_age[i],97.5),2)]
    print("The 95% Confidence Interval of mean purchase of people with age ",df['Age'].unique()[i]," is ",CI_ag
print()
for i in range(df['Age'].nunique()):
    CI_age = [np.round(np.percentile(bs_means_age[i],0.5),2), np.round(np.percentile(bs_means_age[i],99.5),2)]
    print("The 99% Confidence Interval of mean purchase of people with age ",df['Age'].unique()[i]," is ",CI_ag
```

```
The 90% Confidence Interval of mean purchase of people with age  0-17  is  [8849.16, 9017.05]
The 90% Confidence Interval of mean purchase of people with age  55+  is  [9086.44, 9253.87]
The 90% Confidence Interval of mean purchase of people with age  26-35  is  [9170.73, 9335.1]
The 90% Confidence Interval of mean purchase of people with age  46-50  is  [9248.74, 9414.54]
The 90% Confidence Interval of mean purchase of people with age  51-55  is  [9127.09, 9291.8]
The 90% Confidence Interval of mean purchase of people with age  36-45  is  [9451.02, 9617.77]
The 90% Confidence Interval of mean purchase of people with age  18-25  is  [9253.82, 9418.45]

The 95% Confidence Interval of mean purchase of people with age  0-17  is  [8833.41, 9032.21]
The 95% Confidence Interval of mean purchase of people with age  55+  is  [9070.65, 9269.25]
The 95% Confidence Interval of mean purchase of people with age  26-35  is  [9154.42, 9350.26]
The 95% Confidence Interval of mean purchase of people with age  46-50  is  [9233.41, 9430.6]
The 95% Confidence Interval of mean purchase of people with age  51-55  is  [9109.96, 9306.86]
The 95% Confidence Interval of mean purchase of people with age  36-45  is  [9436.32, 9634.42]
The 95% Confidence Interval of mean purchase of people with age  18-25  is  [9238.29, 9435.47]

The 99% Confidence Interval of mean purchase of people with age  0-17  is  [8800.2, 9064.39]
The 99% Confidence Interval of mean purchase of people with age  55+  is  [9039.4, 9300.36]
The 99% Confidence Interval of mean purchase of people with age  26-35  is  [9123.99, 9380.34]
The 99% Confidence Interval of mean purchase of people with age  46-50  is  [9201.18, 9461.25]
The 99% Confidence Interval of mean purchase of people with age  51-55  is  [9079.44, 9335.62]
The 99% Confidence Interval of mean purchase of people with age  36-45  is  [9405.62, 9666.63]
The 99% Confidence Interval of mean purchase of people with age  18-25  is  [9208.87, 9468.28]
```

We have calculate the 90%, 95% and 99% confidence interval on average spends for both men and women above.

1. This confidence intervals tell about the average spend by entire population belonging to different age groups by just using a sample size of 10k each.
2. For all the 3 cases, apart from the confidence intervals of average spend by 0-17 years of age group people which is lowest, all other confidence intervals of average spend by people of different age groups are overlapping.
3. As the increase the percentage the width of the confidence intervals of average spend by both married and unmarried people increases.

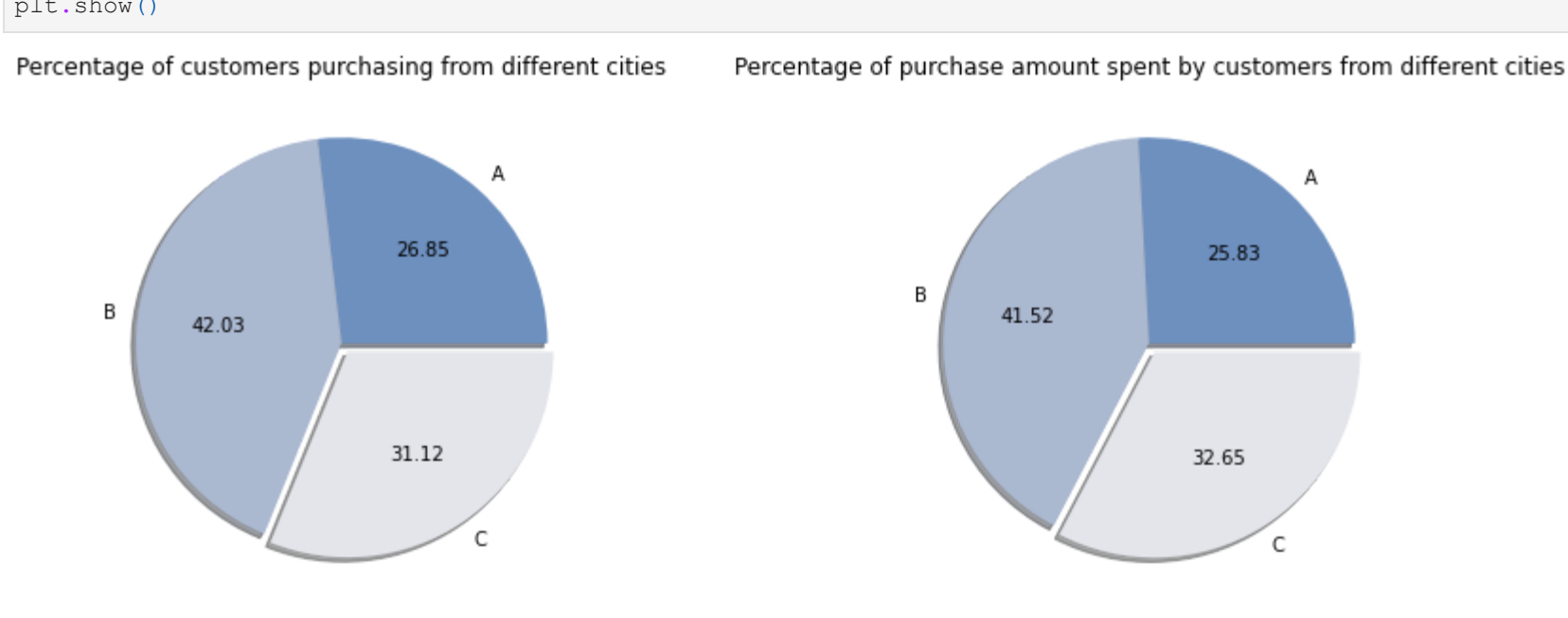
Analysis on City Category

In [92]:
```
city = df.groupby(['City_Category']).agg({'User_ID':'count','Purchase':'sum'}).reset_index().rename(columns={"U
city
```

Out[92]:

	City_Category	Number of people	Total_Purchase_amount
0	A	147720	1316471661
1	B	231173	2115533605
2	C	171175	1663807476

In [93]:
```
plt.figure(figsize=(14,5))
plt.subplot(1,2,1)
colors = sns.color_palette('vlag')[0:3]
plt.pie(x= city['Number of people'], labels= city['City_Category'], colors= colors, autopct= "%0.2f",
explode= (0,0,0.05), shadow= True)
plt.title("Percentage of customers purchasing from different cities")
plt.subplot(1,2,2)
colors = sns.color_palette('vlag')[0:3]
plt.pie(x= city['Total_Purchase_amount'], labels= city['City_Category'], colors= colors, autopct= "%0.2f",
explode= (0,0,0.05), shadow= True)
plt.title("Percentage of purchase amount spent by customers from different cities")
plt.show()
```



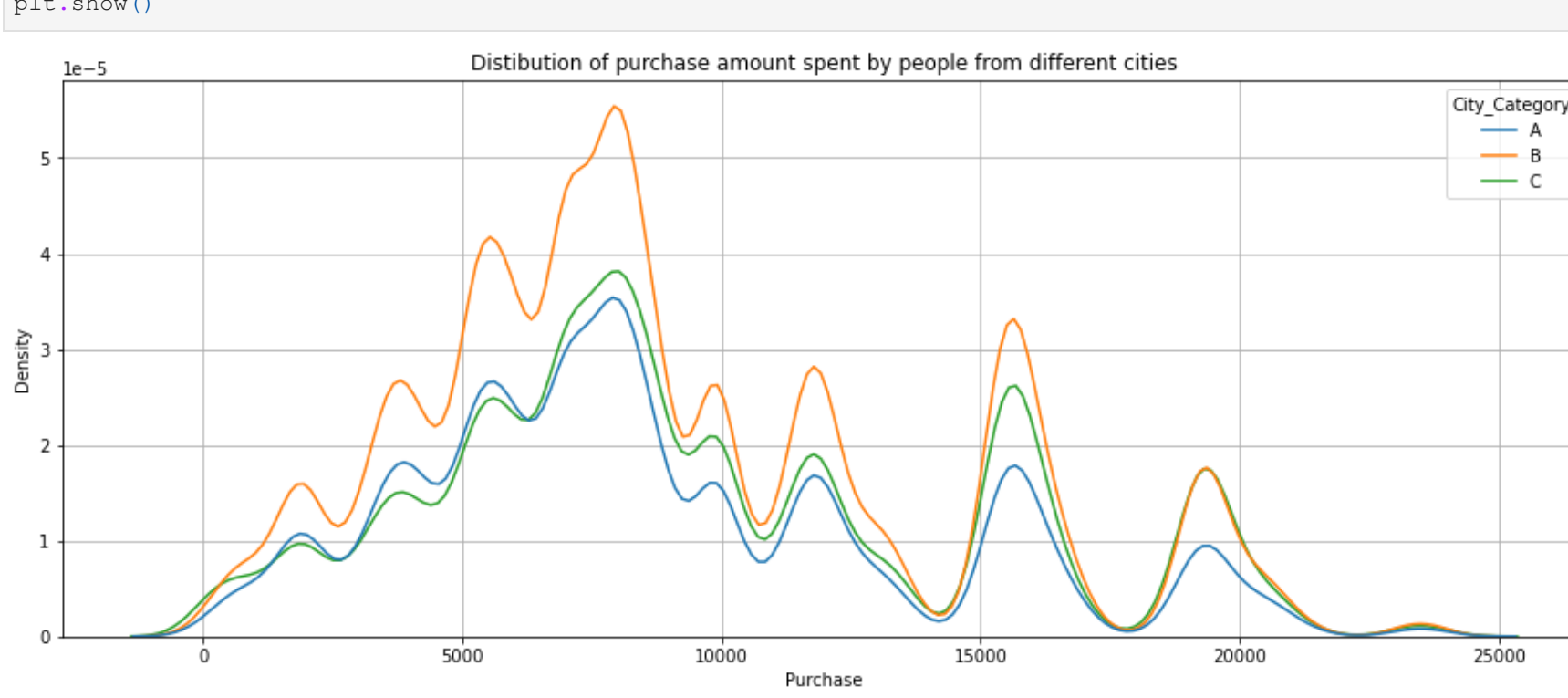
1. As in this pie chart, we can see that 42% of the customers who purchased something from Walmart on Good Friday are from city B, 31% are from city C and rest 27% are from city A.
2. Similarly, around 41.5% of total purchase amount are spent by people from city B, 32.5% are from city C and 26% spent by people from city A.

In [94]:
```
citywise = df.groupby(['City_Category','Gender'])["Purchase"].agg(['mean','median', 'sum']).reset_index().renam
citywise
```

Out[94]:

	City_Category	Gender	Mean_Purchase	Median_Purchase	Total_Purchase
0	A	Female	8579.708576	7847.0	306329915
1	A	Male	9017.834470	7963.0	1010141746
2	B	Female	8540.677694	7839.0	493617008
3	B	Male	9354.854433	8065.0	1621916597
4	C	Female	9130.107518	8077.0	386285719
5	C	Male	9913.567248	8655.0	1277521757

In [95]:
```
plt.figure(figsize=(16,6))
sns.kdeplot(data=df, x="Purchase", hue='City_Category')
plt.title("Distibution of purchase amount spent by people from different cities")
plt.grid()
plt.show()
```



1. In the kdeplot also, we can clearly see that the density of People from city C spending money on purchasing is highest compared to people from city A and C.
2. The density of People from city A and C spending money on purchasing are quite comparable.

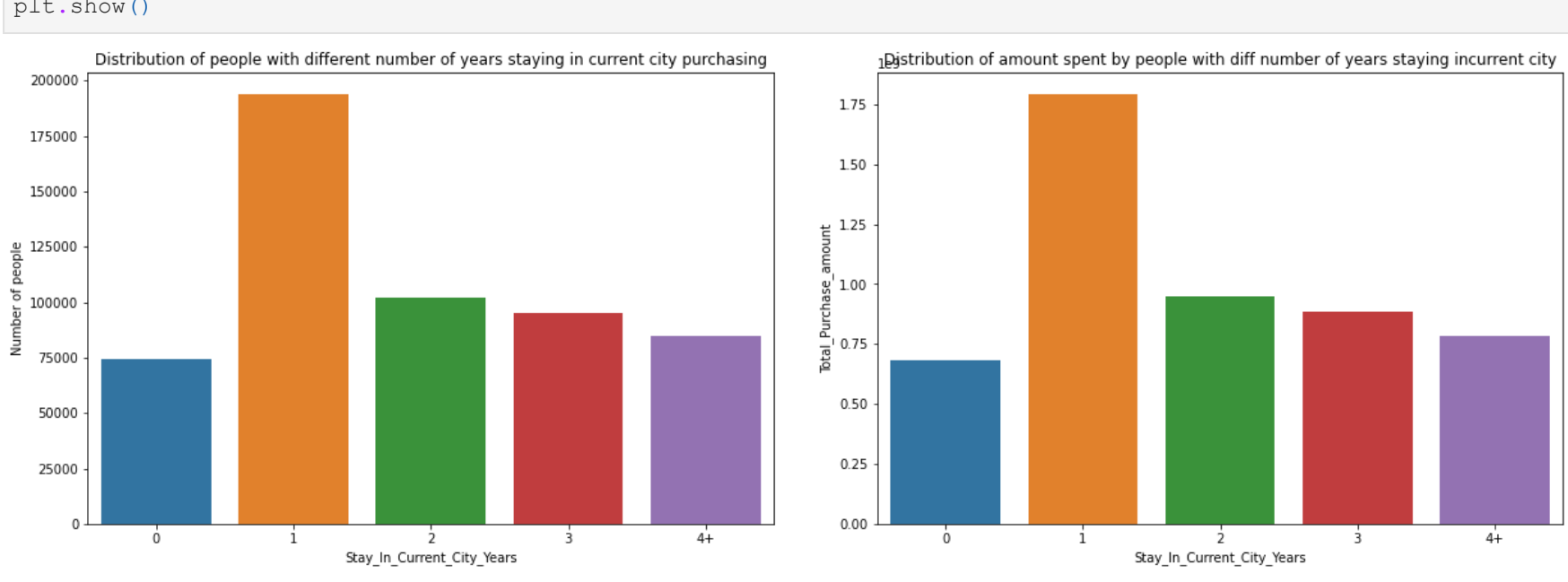
Analysis on Stay In Current City Years

In [96]:
```
city_yr = df.groupby('Stay_In_Current_City_Years').agg({'User_ID':'count','Purchase':'sum'}).reset_index().renai
city_yr.head()
```

Out[96]:

	Stay_In_Current_City_Years	Number of people	Total_Purchase_amount
0	0	74398	682979229
1	1	193821	1792872533
2	2	101838	949173931
3	3	95285	884902659
4	4+	84726	785884390

In [98]:
```
plt.figure(figsize=(20,7))
plt.subplot(1,2,1)
sns.barplot(data=city_yr,x="Stay_In_Current_City_Years",y="Number of people")
plt.title("Distribution of people with different number of years staying in current city purchasing")
plt.subplot(1,2,2)
sns.barplot(data=city_yr,x="Stay_In_Current_City_Years",y="Total_Purchase_amount")
plt.title("Distribution of amount spent by people with diff number of years staying incurrent city")
plt.subplots_adjust(left=0.1,
bottom=0.2,
right=0.9,
top=0.9,
wspace=0.15,
hspace=0.5)
plt.show()
```



Distribution of people with different number of years staying in current city purchasing
Distribution of amount spent by people with diff number of years staying incurrent city
Number of people
Total_Purchase_amount
Stay_In_Current_City_Years

1. The maximum number of people purchasing from Walmart on Good Friday are staying for 1 year in their current city.
2. The number of people staying for 2,3 or 4+ years in their current city, purchasing from Walmart on Good Friday are also good and comparable.
3. Similarly, the total amount spend by people based on number of years they are staying in their current city is directly proportional to the number of customers belong to the group of years they are staying in their current city.

In [99]:
```
cityyr = df.groupby('Stay_In_Current_City_Years')["Purchase"].agg(['mean','median', 'sum']).reset_index().renam
cityyr
```

Out[99]:

	Stay_In_Current_City_Years	Mean_Purchase	Median_Purchase	Total_Purchase
0	0	9180.075123	8025.0	682979229
1	1	9250.145923	8041.0	1792872533
2	2	9320.429810	8072.0	949173931
3	3	9286.904119	8047.0	884902659
4	4+	9275.598872	8052.0	785884390

In [101…
```
plt.figure(figsize=(16,6))
sns.kdeplot(data=df, x="Purchase", hue='Stay_In_Current_City_Years')
plt.title("Distibution of purchase amount spent by people with diff number of years staying in current city")
plt.grid()
plt.show()
```

Business Insights

1. 75% of the purchase transactions are done by men and 25% of the purchase transactions are done by women.
2. 40% of the transactions are done by age group 26-35, 20% are done by age group 36-45, and 18% are done by age group 18-25.
3. 58% of the customers who purchased something from Walmart on Good Friday are Unmarried and rest 42% are Married.
4. Around 59% of total purchase amount are spent by Unmarried and 41% spent by Married.
5. The maximum number of people purchasing from Walmart on Good Friday are from age group of 26-35 years.
6. The number of people in age group of 0-17 years is lowest.
7. The number of people in age group of 18-25 years and 36-45 years purchasing from Walmart on Good Friday are also good.
8. Similarly, the total amount spend by people belonging to any age group is directly proportional to the number of customers belong to that age group.
9. For most of the women, we can see their average spend lies between 8700 dollars to 8760 dollars. Similarly, for most of the men, their average spend lies between 9400 dollars to 9460 dollars.
10. Very weak co-relation exists between the features when observed with respect to purchase.

Recommendations

1. The average amount spend by women is lower compared to men in Walmart on Good Friday. So, inorder to increase their average amount spend Walmart can provide some discounts on certain products exclusive for women on different occassions like Good Friday to improve that.
2. We noticed that the number of married people are less than unmarried who are purchasing on Good Friday and the average amount spend by them is slightly less than unmarried people. Walmart can provide a couple membership cards or provide offers on the products exclusive for married folks in different festive occassions to increase their numbers as well as the amount spend by them.
3. Most of the customers who purchased from Walmart on Good Friday are from age group 18-45.So, in that scenario to increase the revenue Walmart can provide some sort of membership card with exclusive or prime features to those who spends a minimum threshold amount to make them as their customers for long term.
4. For older people with age 55+ years, the reason why their number is less might be they are unable to go to stores due to their old age. For them Walmart can provide home delivery to them so that, they can purchase the items they want without any difficulty.
5. For different products in their store, If the total unit sold for a particular category is less then maybe Walmart can go for store assorment optimization and can remove some products under that category whose demand is quite low inorder to save money without comprising with the revenue generation.
6. The number of customers are compartively less from city A and C who are purchasing. So, Walmart can do some analysis on this cities and can open new stores in those regions where the it is highly populated and can sell those products which has high demand in those region.