Problem Statement

- **Walmart** is an American multinational retail corporation that operates a chain of supercenters, discount department stores, and grocery stores in the United States.
  Walmart serves **more than 100 million customers worldwide**.
- The **Management Team at Walmart Inc.** wants to analyze customer purchase behavior—specifically **purchase amount**—in relation to **customer gender** and various other factors to support better business decisions.
- They want to understand whether spending habits differ between male and female customers:
  Do women spend more on Black Friday than men?

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt


import warnings
warnings.filterwarnings("ignore")
```

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

```
df.shape
```

```
(550068, 10)
```

```
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
```

```
df.describe()
```

	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

```
df.describe(include="object")
```

	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

```
df[df.duplicated()].shape
```

```
(0, 10)
```

Analyzing the gender column

```
df["Gender"].value_counts()
```

Gender	count
M	414259
F	135809

dtype: int64

```
df["Gender"].value_counts(normalize=True)
```

Gender	proportion
M	0.753105
F	0.246895

dtype: float64

```
df.groupby("Gender")["User_ID"].nunique()
```

Gender	User_ID
F	1666
M	4225

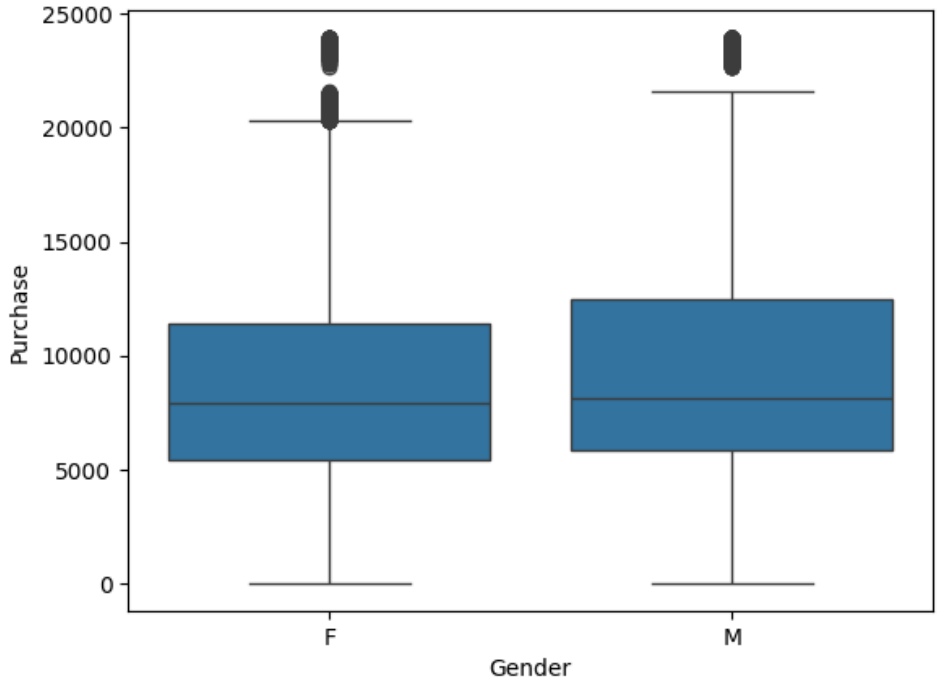
dtype: int64

```
df.groupby("Gender")["Purchase"].describe().T
```

Gender	F	M
count	135809.000000	414259.00000
mean	8734.565765	9437.52604
std	4767.233289	5092.18621
min	12.000000	12.00000
25%	5433.000000	5863.00000
50%	7914.000000	8098.00000
75%	11400.000000	12454.00000
max	23959.000000	23961.00000

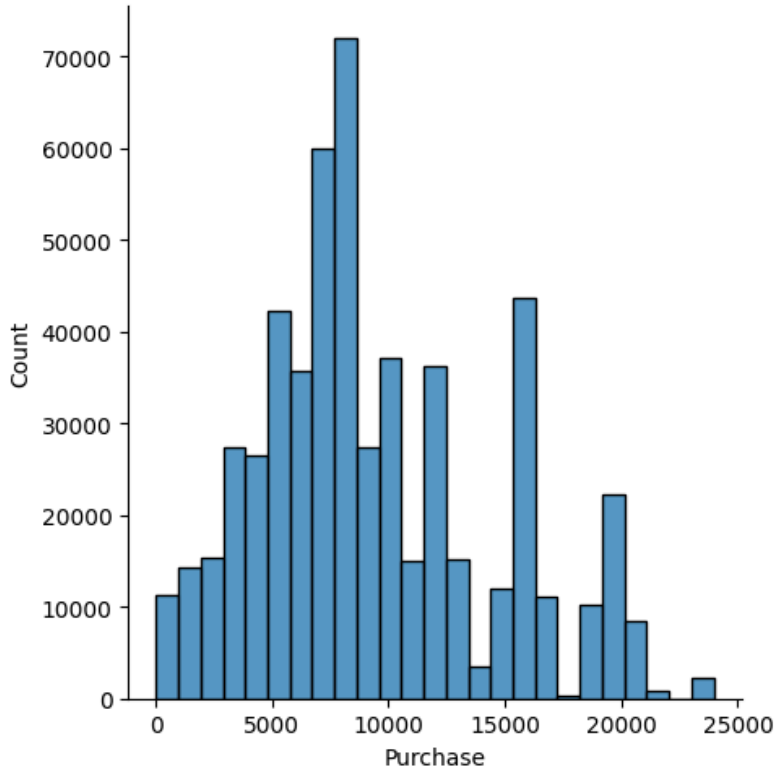
```
sns.boxplot(x='Gender', y='Purchase', data=df)
```

```
<Axes: xlabel='Gender', ylabel='Purchase'>
```



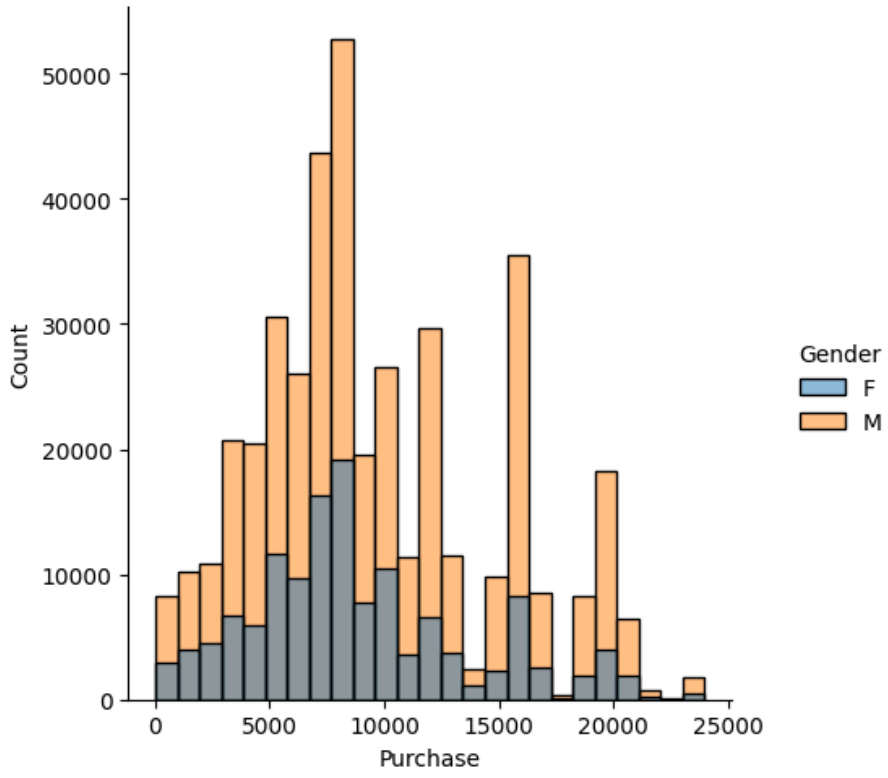

```
sns.displot(x="Purchase",data=df,bins=25)
```

```
<seaborn.axisgrid.FacetGrid at 0x7c141bd8c080>
```




```
sns.displot(x='Purchase', data=df,bins=25,hue="Gender")
```

```
<seaborn.axisgrid.FacetGrid at 0x7c141b48a2d0>
```



CLT

```
male_sample_means=[df[df["Gender"]== "M"]["Purchase"].sample(3000).mean() for i in range(
```

```
female_sample_means=[df[df["Gender"]== "F"]["Purchase"].sample(3000).mean() for i in rang
```

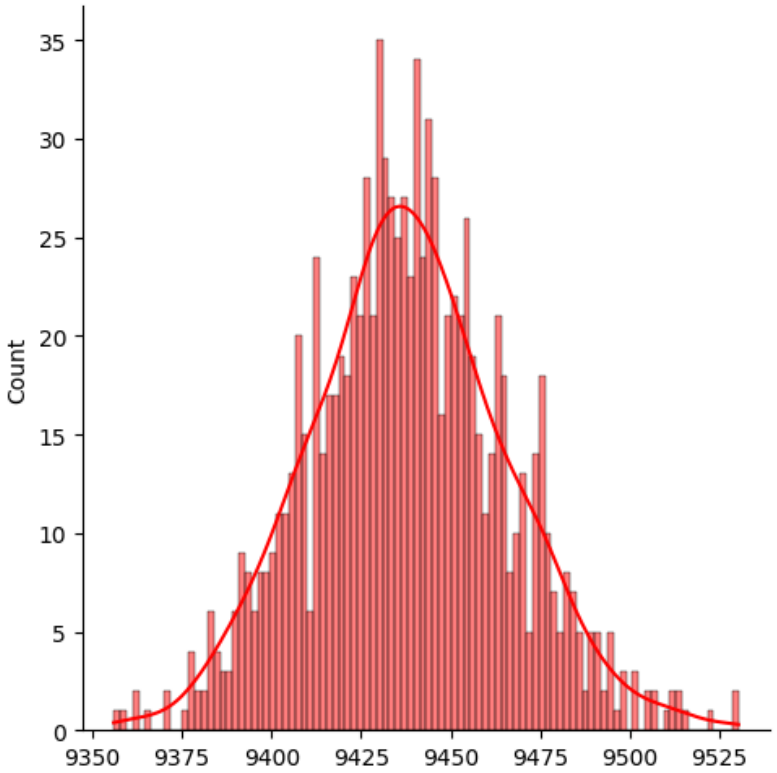
```
male_sample_means=[df[df["Gender"]== "M"]["Purchase"].sample(30000).mean() for i in range
female_sample_means=[df[df["Gender"]== "F"]["Purchase"].sample(30000).mean() for i in ran
```

```
df[df["Gender"] == "M"]["Purchase"].sample(30000).mean()
```

```
np.float64(9405.373133333333)
```

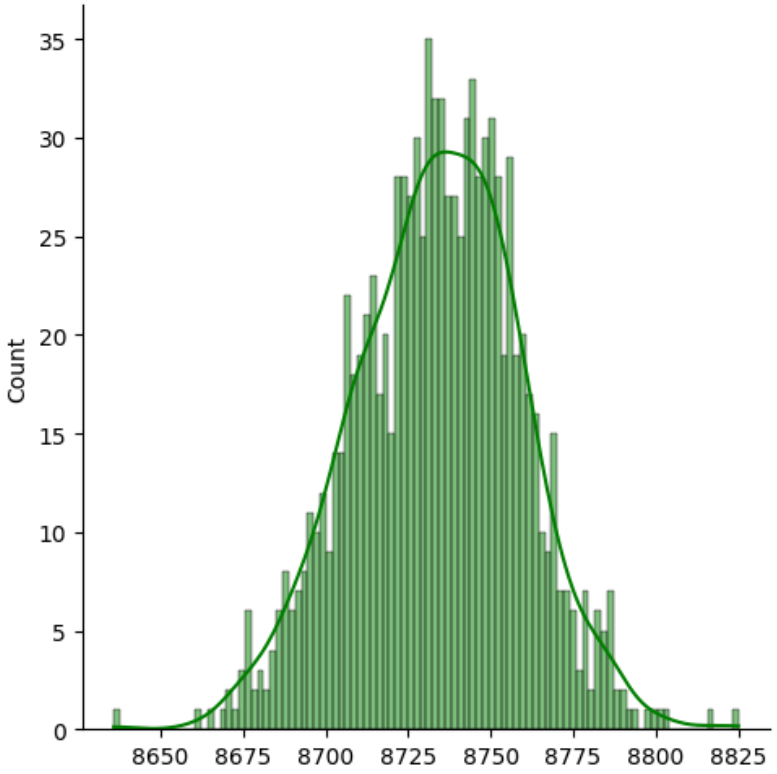
```
sns.displot(male_sample_means,kde=True,bins=100,color='r')
```

```
<seaborn.axisgrid.FacetGrid at 0x7c141bfa15b0>
```




```
sns.displot(female_sample_means,kde=True,bins=100,color='g')
```

```
<seaborn.axisgrid.FacetGrid at 0x7c141b336c00>
```



```
np.mean(male_sample_means), np.mean(female_sample_means)
```

```
(np.float64(9438.088883100001), np.float64(8733.974285533335))
```

Confidence Interval

```
## 95% confidence interval

lower_limit_males=np.mean(male_sample_means)-(1.96*np.std(male_sample_means))
lower_limit_males
```

```
np.float64(9384.522679715055)
```

```
upper_limit_males=np.mean(male_sample_means)+(1.96*np.std(male_sample_means))
upper_limit_males
```

```
np.float64(9491.655086484947)
```

```
(lower_limit_males, upper_limit_males)
```

```
(np.float64(9384.522679715055), np.float64(9491.655086484947))
```

```
lower_limit_females= np.mean(female_sample_means) - (1.96 * np.std(female_sample_means))
upper_limit_females= np.mean(female_sample_means) + (1.96 * np.std(female_sample_means))
```

```
(lower_limit_females,upper_limit_females)
```

```
(np.float64(8684.953223775901), np.float64(8782.99534729077))
```

```
plt.figure(figsize=(10, 6))
sns.histplot(male_sample_means, kde=True, bins=100, color='r')
sns.histplot(female_sample_means, kde=True, bins=100, color='g')
plt.plot()
```

```
[]
```