In [1]:
```
'https://www.kaggle.com/aungpyaeap/supermarket-sales'
```

Out[1]:
```
'https://www.kaggle.com/aungpyaeap/supermarket-sales'
```

In [2]:
```
import pandas as pd
import numpy as np
import seaborn as sb
import matplotlib.pyplot as plt
%matplotlib inline
```

In [3]:
```
# Load the data
df = pd.read_csv('/Users/ahmedalshaibani/Desktop/super-market-back-up/supermarket_sales.csv')
```

In [4]:
```
df.head()
```

Out[4]:

	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1415	5
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.8200	8
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	3
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.2880	4
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	6

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 17 columns):
Invoice ID                 1000 non-null object
Branch                     1000 non-null object
City                       1000 non-null object
Customer type              1000 non-null object
Gender                     1000 non-null object
Product line               1000 non-null object
Unit price                 1000 non-null float64
Quantity                   1000 non-null int64
Tax 5%                     1000 non-null float64
Total                      1000 non-null float64
Date                       1000 non-null object
Time                       1000 non-null object
Payment                    1000 non-null object
cogs                       1000 non-null float64
gross margin percentage    1000 non-null float64
gross income               1000 non-null float64
Rating                     1000 non-null float64
dtypes: float64(7), int64(1), object(9)
memory usage: 132.9+ KB
```

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

Out[6]:
```
Invoice ID                 0
Branch                     0
City                       0
Customer type              0
Gender                     0
Product line               0
Unit price                 0
Quantity                   0
Tax 5%                     0
Total                      0
Date                       0
Time                       0
Payment                    0
cogs                       0
gross margin percentage    0
gross income               0
Rating                     0
dtype: int64
```

In [7]:
```
df.duplicated().sum()
```

Out[7]:
```
0
```

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

Out[8]:

	Unit price	Quantity	Tax 5%	Total	cogs	gross margin percentage	
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.00000	1.000000e+03	1
mean	55.672130	5.510000	15.379369	322.966749	307.58738	4.761905e+00	1
std	26.494628	2.923431	11.708825	245.885335	234.17651	6.220360e-14	1
min	10.080000	1.000000	0.508500	10.678500	10.17000	4.761905e+00	0
25%	32.875000	3.000000	5.924875	124.422375	118.49750	4.761905e+00	5
50%	55.230000	5.000000	12.088000	253.848000	241.76000	4.761905e+00	1
75%	77.935000	8.000000	22.445250	471.350250	448.90500	4.761905e+00	2
max	99.960000	10.000000	49.650000	1042.650000	993.00000	4.761905e+00	4

In [9]:
```
df.columns
```

Out[9]:
```
Index(['Invoice ID', 'Branch', 'City', 'Customer type', 'Gender',
       'Product line', 'Unit price', 'Quantity', 'Tax 5%', 'Total', 'Da
te',
       'Time', 'Payment', 'cogs', 'gross margin percentage', 'gross inc
ome',
       'Rating'],
      dtype='object')
```

In [10]:
```
df.rename(columns = {'Customer type':'Customer_type'}, inplace = True)
```

In [11]:
```
df['Customer_type'].value_counts()
```

Out[11]:
```
Member    501
Normal    499
Name: Customer_type, dtype: int64
```

In [12]:
```
df['Gender'].value_counts()
```

Out[12]:
```
Female    501
Male      499
Name: Gender, dtype: int64
```

In [13]:
```
pd.crosstab(df.Gender, df.Customer_type)
```

Out[13]:

Customer_type	Member	Normal
Gender		
Female	261	240
Male	240	259

In [14]:
```
pd.crosstab(df.Gender, df.Branch)
```

Out[14]:

Branch	A	B	C
Gender			
Female	161	162	178
Male	179	170	150

In [15]:
```
df['Branch'].value_counts()
```

Out[15]:
```
A    340
B    332
C    328
Name: Branch, dtype: int64
```

In [16]:
```
group_df = df.groupby("Gender")
mean_df = group_df.sum()
```

In [17]:
```
mean_df = mean_df.reset_index()
```

In [18]:
```
print(mean_df)
```

```
   Gender  Unit price  Quantity    Tax 5%       Total       cogs  \
0  Female    27687.24      2869  7994.425  167882.925  159888.50
1    Male    27984.89      2641  7384.944  155083.824  147698.88

   gross margin percentage  gross income  Rating
0              2385.714286      7994.425  3489.2
1              2376.190476      7384.944  3483.5
```

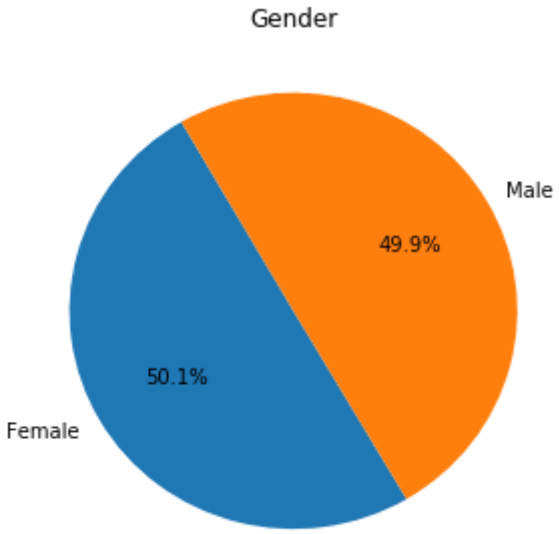
In [19]:
```
df['Rating'].mean()
```

Out[19]:
```
6.972700000000003
```

In [20]:
```
labels = df['Gender'].value_counts().index
values = df['Gender'].value_counts().values
```

In [21]:
```
plt.figure(figsize=(5,5))
plt.pie(values, labels=labels, autopct='%1.1f%%', startangle = 120)
plt.title('Gender')
plt.show
```

Out[21]: <function matplotlib.pyplot.show>



In [22]:
```
plt.figure(figsize=(10,5))
sb.countplot(x=df['Branch'],hue=df['Gender']);
```