Data Analytics Assignment -2

Data Loading

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

# Step 1: Dataset Loading
pd.set_option('display.max_columns', None)
df=pd.read_csv("Mall_Customers.csv")

print("First 5 rows:")
print(df.head())
print("\nLast 5 rows:")
print(df.tail())

print("\nShape of dataset:", df.shape)
print("\nColumn names:")
print(df.columns)

print("\nDataset info:")
print(df.info())
```

Output:

```
First 5 rows:
   CustomerID   Genre  Age  Annual_Income_(k$)  Spending_Score
0           1    Male   19                  15              39
1           2    Male   21                  15              81
2           3  Female   20                  16               6
3           4  Female   23                  16              77
4           5  Female   31                  17              40


Last 5 rows:
     CustomerID   Genre  Age  Annual_Income_(k$)  Spending_Score
195         196  Female   35                 120              79
196         197  Female   45                 126              28
197         198    Male   32                 126              74
198         199    Male   32                 137              18
199         200    Male   30                 137              83
```

```
Shape of dataset: (200, 5)

Column names:
Index(['CustomerID', 'Genre', 'Age', 'Annual_Income_(k$)', 'Spending_Score'], dtype='object')
```

```
Data columns (total 5 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   CustomerID          200 non-null    int64
 1   Genre               200 non-null    object
 2   Age                 200 non-null    int64
 3   Annual_Income_(k$)  200 non-null    int64
 4   Spending_Score      200 non-null    int64
```

Data Cleaning

```
print("\nMissing values:")
print(df.isnull().sum())

print(df.drop_duplicates(inplace=True))
```

Output:

```
Missing values:
CustomerID            0
Genre                 0
Age                   0
Annual_Income_(k$)    0
Spending_Score        0
dtype: int64
None
```

EDA (Exploratory Data Analysis)

```
print("\nSummary statistics:")
print(df.describe())

cat_cols = df.select_dtypes(include='object').columns
if len(cat_cols) > 0:
    print("\nValue counts:")
    print(df[cat_cols[0]].value_counts())

corr = df.corr(numeric_only=True)
print("\nCorrelation matrix:")
print(corr)
```

Output:

	CustomerID	Age	Annual_Income_(k$)	Spending_Score
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

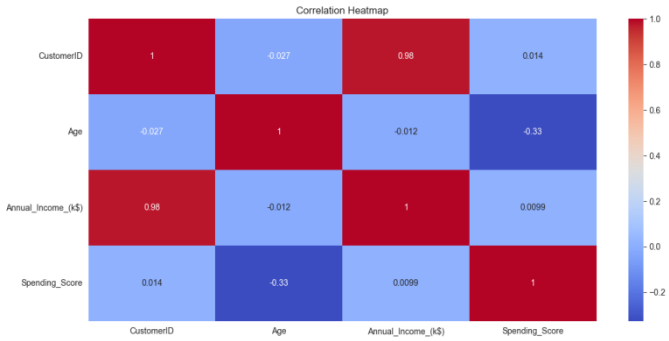
```
Value counts:
Female    112
Male       88
Name: Genre, dtype: int64
```

Correlation matrix:

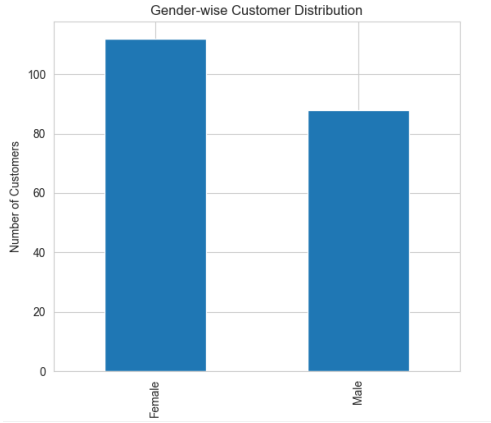
	CustomerID	Age	Annual_Income_(k$)	Spending_Score
CustomerID	1.000000	-0.026763	0.977548	0.013835
Age	-0.026763	1.000000	-0.012398	-0.327227
Annual_Income_(k$)	0.977548	-0.012398	1.000000	0.009903
Spending_Score	0.013835	-0.327227	0.009903	1.000000

Data Visualization

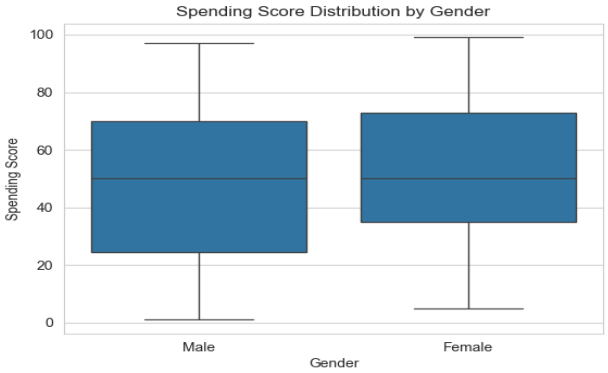
(a) Scatter Plot



(b) Heatmap



(c) Bar Chart



(d) Box Plot

Insights

- The dataset contains 200 customers with 5 attributes related to demographics and spending behaviour.
- There are more female customers than male customers in the dataset.
- Age shows a negative correlation with Spending Score, meaning younger customers tend to spend more.
- Annual Income does not strongly influence Spending Score, indicating that spending behaviour is not solely income-dependent.
- Female customers generally show slightly higher and more varied spending patterns compared to male customers.
- Customers spending behaviour varies widely even within similar income groups.