Mall CUSTOMER SEGMENTATION ANALYSIS WITH PYTHON

INTRODUCTION

Customer Segmentation

Customer segmentation is the practice of dividing a company's customers into groups that reflect similarity among customers in each group. The goal of segmenting customers is to decide how to relate to customers in each segment in order to maximize the value of each customer to the business.

Customer Segmentation Analysis

Customer segmentation analysis is the process performed when looking to discover insights that define specific segments of customers. Marketers and brands leverage this process to determine what campaigns, offers, or products to leverage when communicating with specific segments. Customer segmentation analysis is the process performed when looking to discover insights that define specific segments of customers. Marketers and brands leverage this process to determine what campaigns, offers, or products to leverage when communicating with specific segments.

Objective

Mall customer analysis and segmentation

Questions

1. Analyse the demographic and economic information of customers that vists the mall
2. Is there correlation between spending score and customer income
3. investigate the type of customers that visits the mall based on spending score and income

METHDOLOGY

This project uses common cluster analysis method known as k-means cluster analysis, sometimes referred to as scientific segmentation. The clusters that result assist in better customer modeling and predictive analytics, and are also are used to target customers with offers and incentives personalized to their wants, needs and preferences.the process is not based on any predetermined thresholds or rules. Rather, the data itself reveals the customer prototypes that inherently exist within the population of customers

K-Means is the most popular clustering algorithm. It uses an iterative technique to group unlabeled data into K clusters based on cluster centers (centroids). The data in each cluster are chosen such that their average distance to their respective centroid is minimized.

1. Randomly place K centroids for the initial clusters.
2. Assign each data point to their nearest centroid.
3. Update centroid locations based on the locations of the data points. Repeat Steps 2 and 3 until points don't move between clusters and centroids stabilize.

RESULTS

RESULTS

Import necessary libraries and load data

In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
df=pd.read_csv("C:/Users/Dell i5/OneDrive - Cape Peninsula University of Technology/Deskt
op/Portfolio projects/Customer segmentation/Mall_Customers.csv")
```

Univariate analysis

In [3]:

```
df.head()
```

Out[3]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
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

In [4]:

```
df.describe()
```

Out[4]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
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

The annual income of customers ranges from a low of 15 thousand dollar to a high of 137 thousand dollars with an average of 60 thousand dollars.The median anual income suggest that half of customers earn less that 61.5 thousand dollars while the other half earns more than the amount. The customers who visits the mall have an average age of 38 years, withe youngest customer being at 18 years and the oldest customer being at 70 years old.

In [20]:

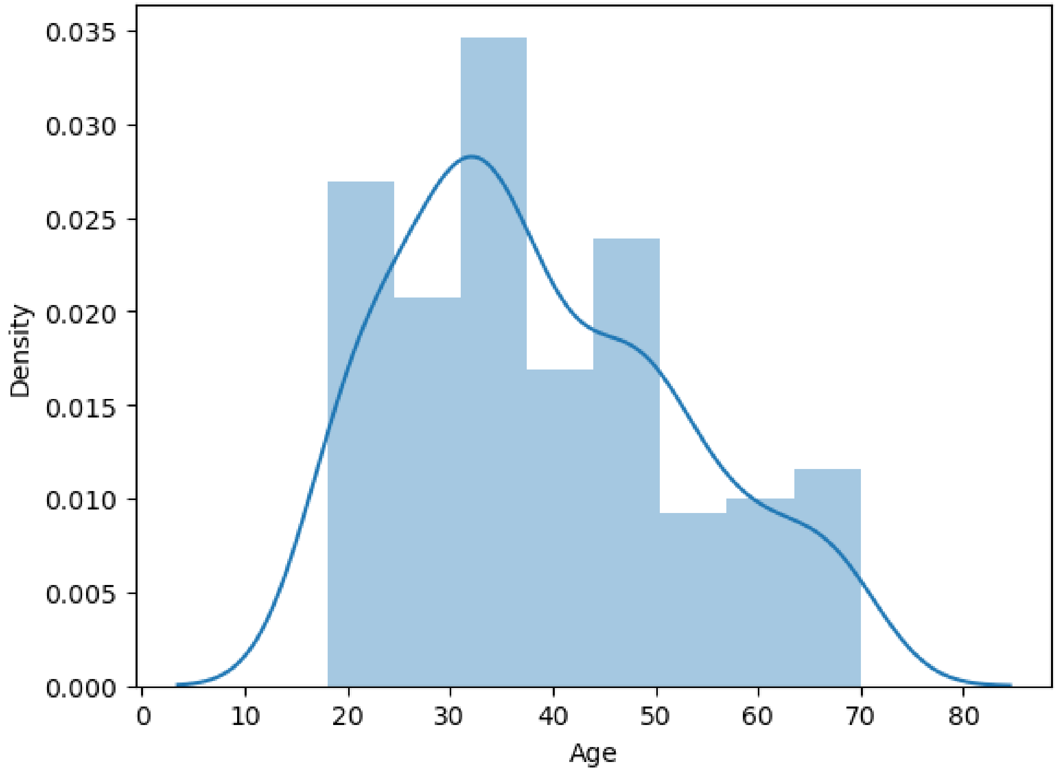
```
df.columns
```

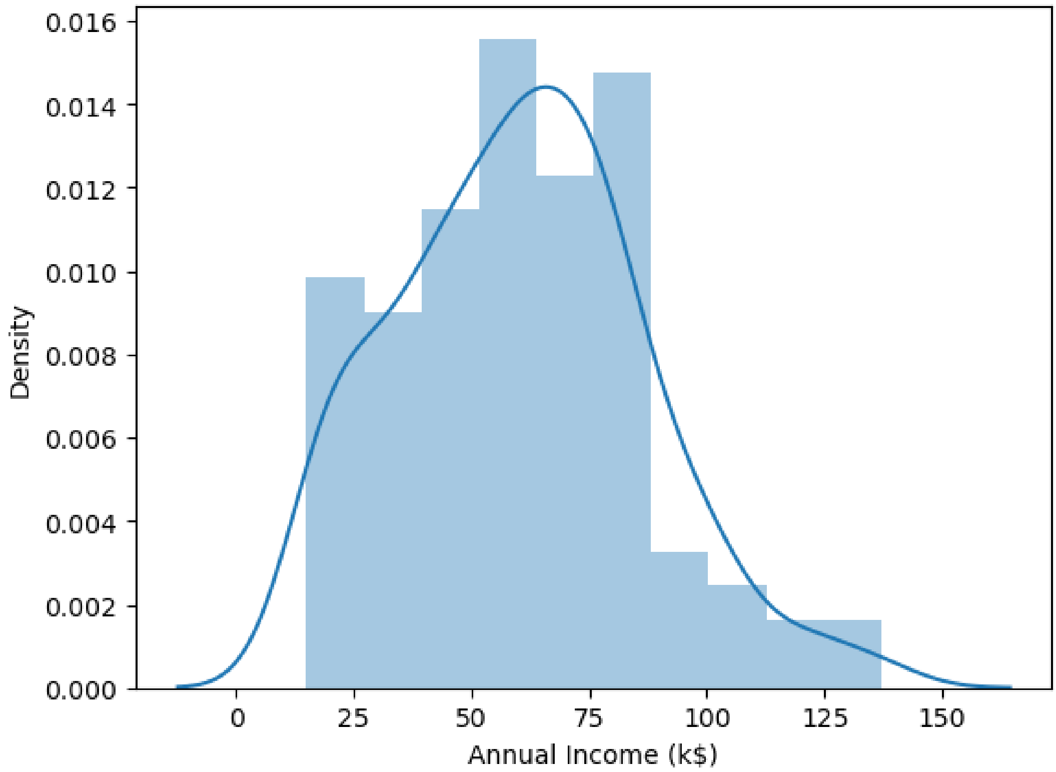
Out[20]:

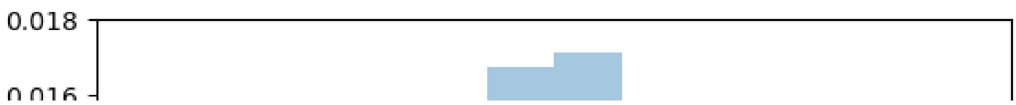
```
Index(['CustomerID', 'Gender', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

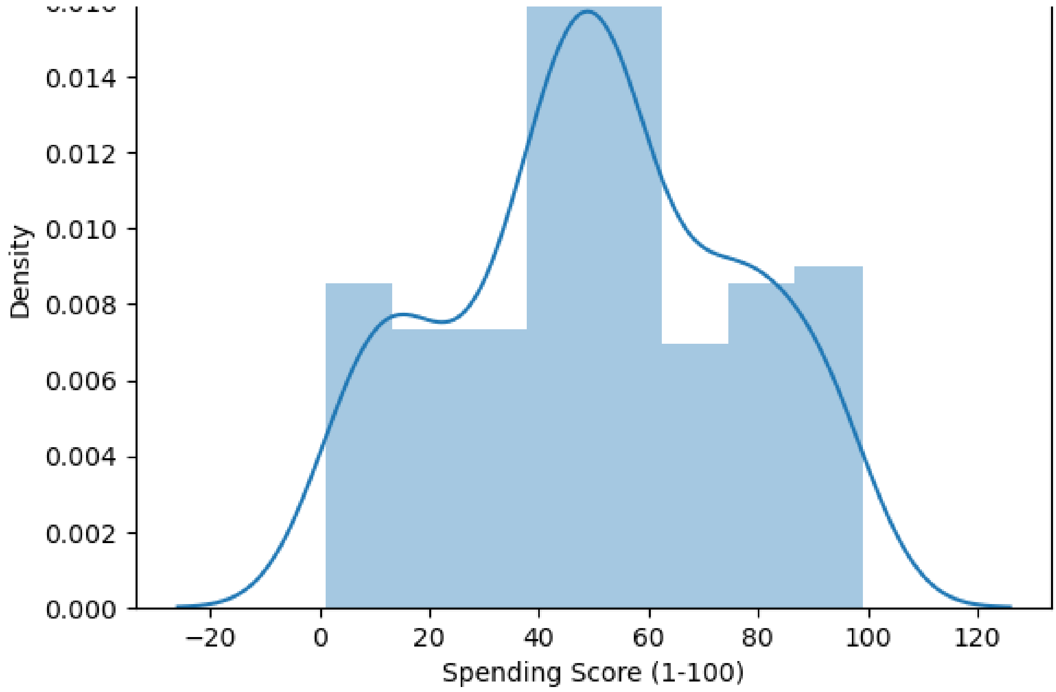
In [24]:

```
#plot the distribution of variables (Age, Annual Income (k$), Spending Score (1-100))
columns=['Age', 'Annual Income (k$)',
       'Spending Score (1-100)']
for i in columns:
    plt.figure()
    sns.distplot(df[i])
```









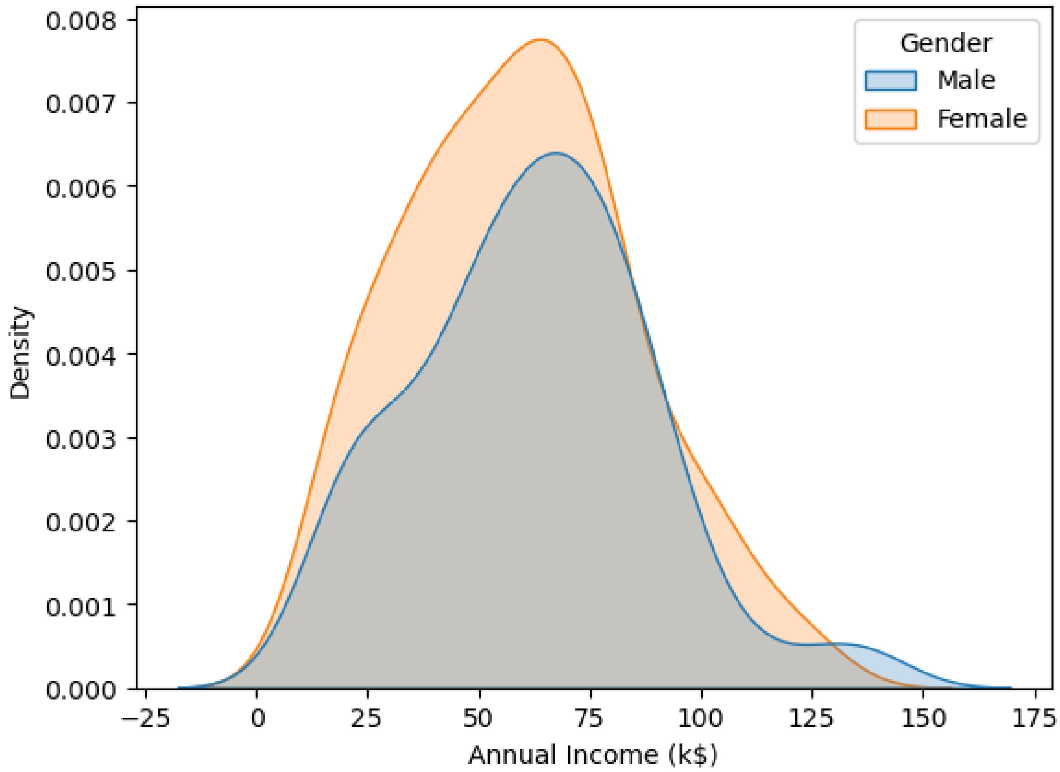
The distribution of age and annual income are slightly right skewed. it suggests that there are more younger customers(less than 50) compared to older custmers and that a single peak of the range 60-70 thousand dollars is most common. Moreover the distribution of spending score apears to be roughly normally distributed with the most common score rughly at 50.

In [26]:

```
sns.kdeplot(df['Annual Income (k$)'],shade=True,hue=df['Gender'])
```

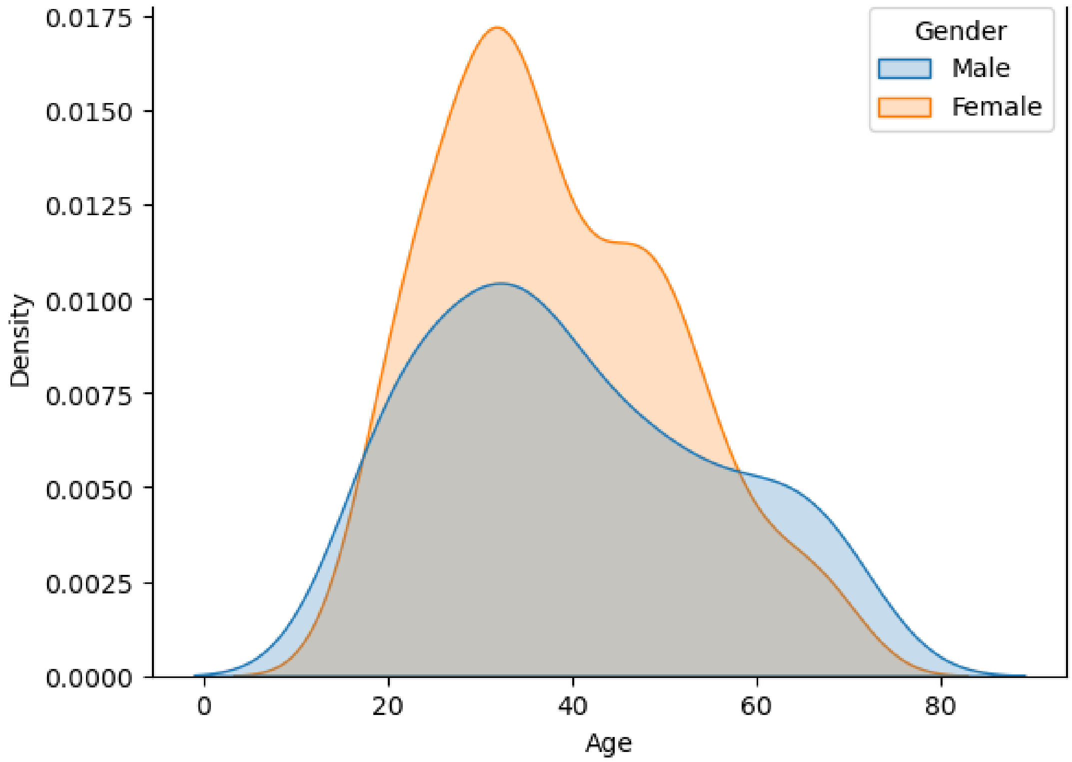
Out[26]:

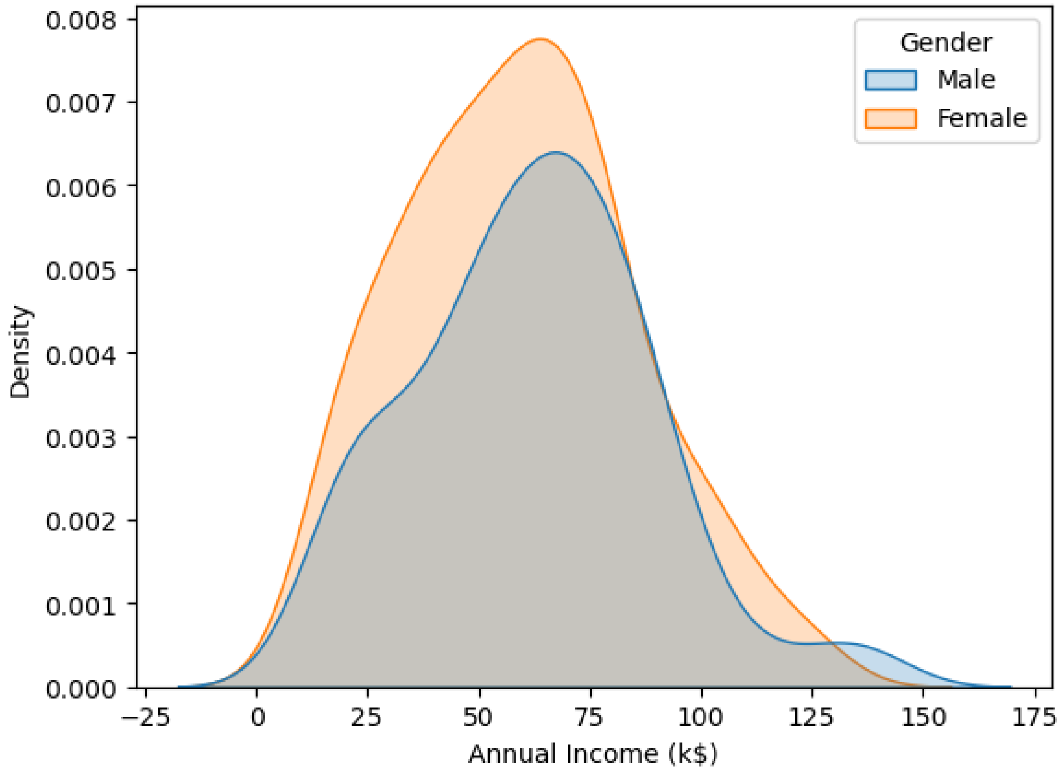
```
<AxesSubplot:xlabel='Annual Income (k$)', ylabel='Density'>
```

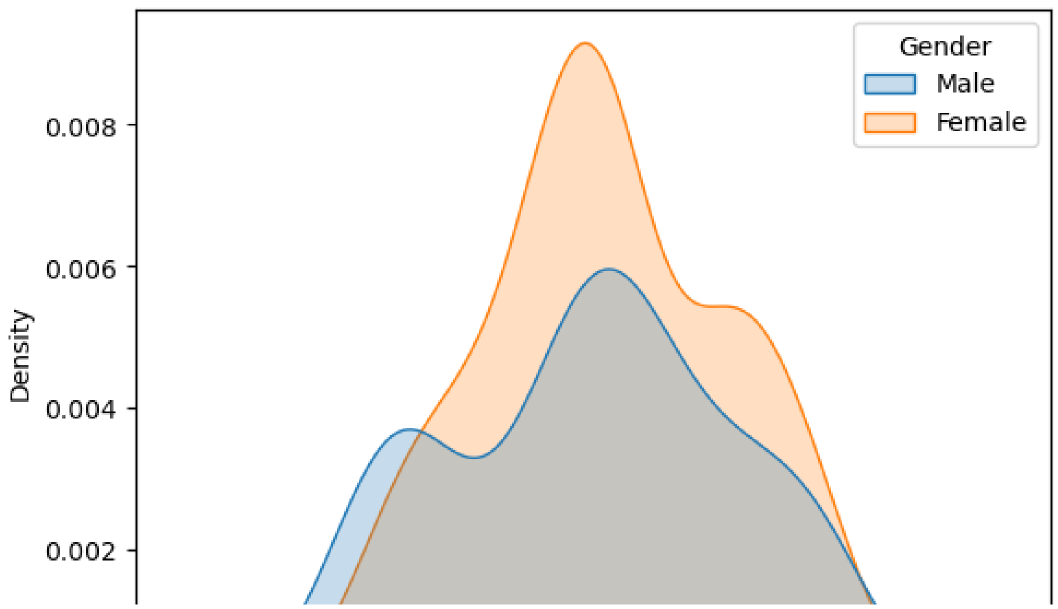


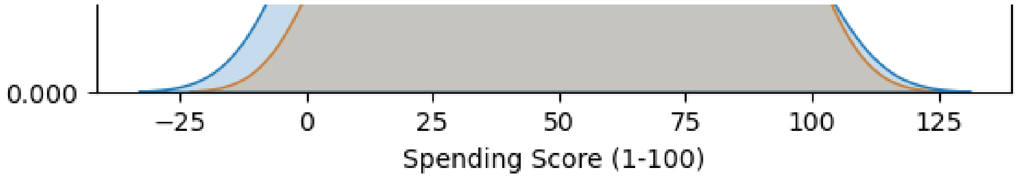
In [27]:

```
columns = ['Age', 'Annual Income (k$)','Spending Score (1-100)']
for i in columns:
    plt.figure()
    sns.kdeplot(df[i],shade=True,hue=df['Gender'])
```


Gender
Male
Female
Density
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
0
20
40
60
80
Age

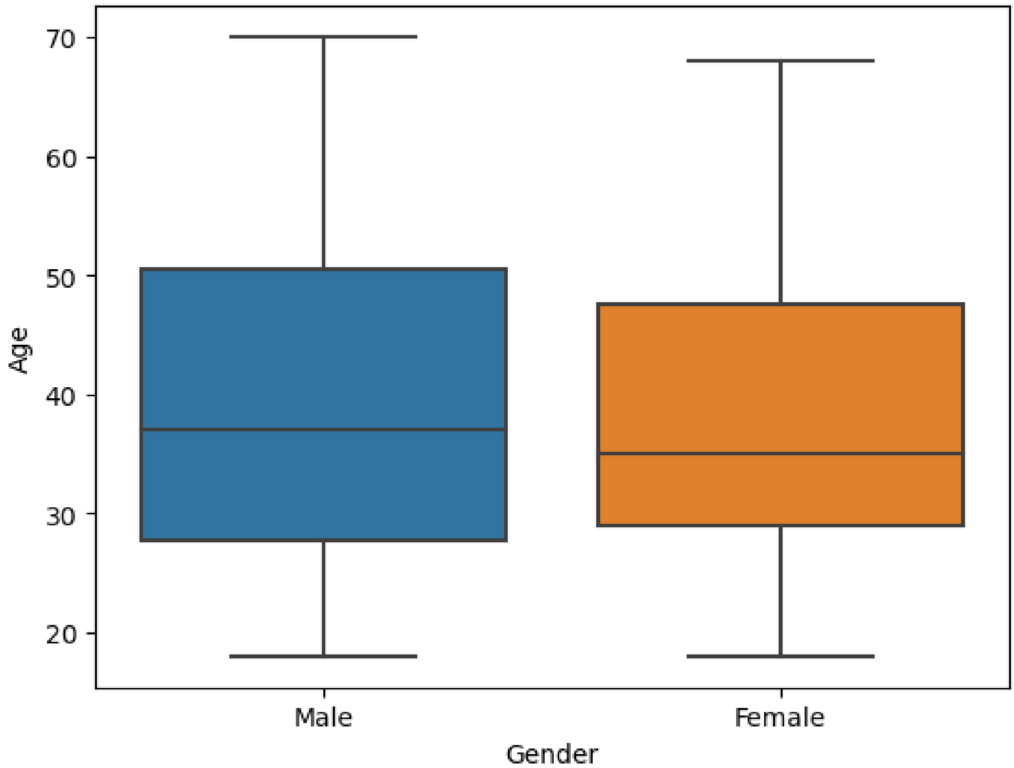

Gender
Male
Female
Density
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
−25
0
25
50
75
100
125
150
175
Annual Income (k$)

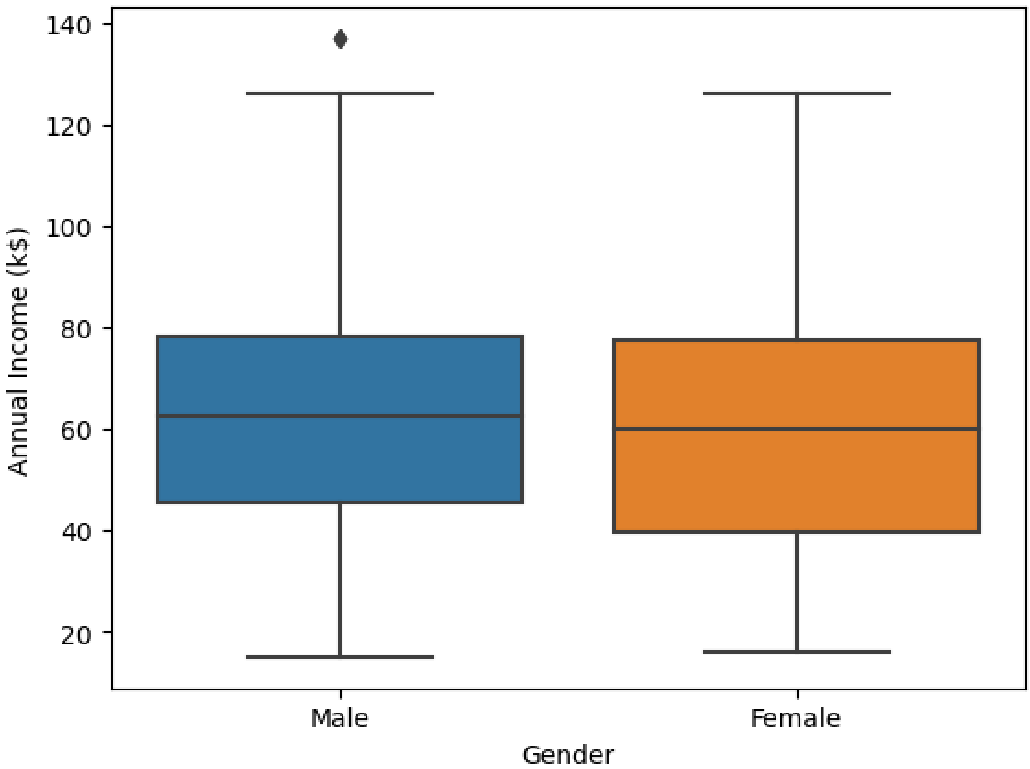

Gender
Male
Female
Density
0.008
0.006
0.004
0.002

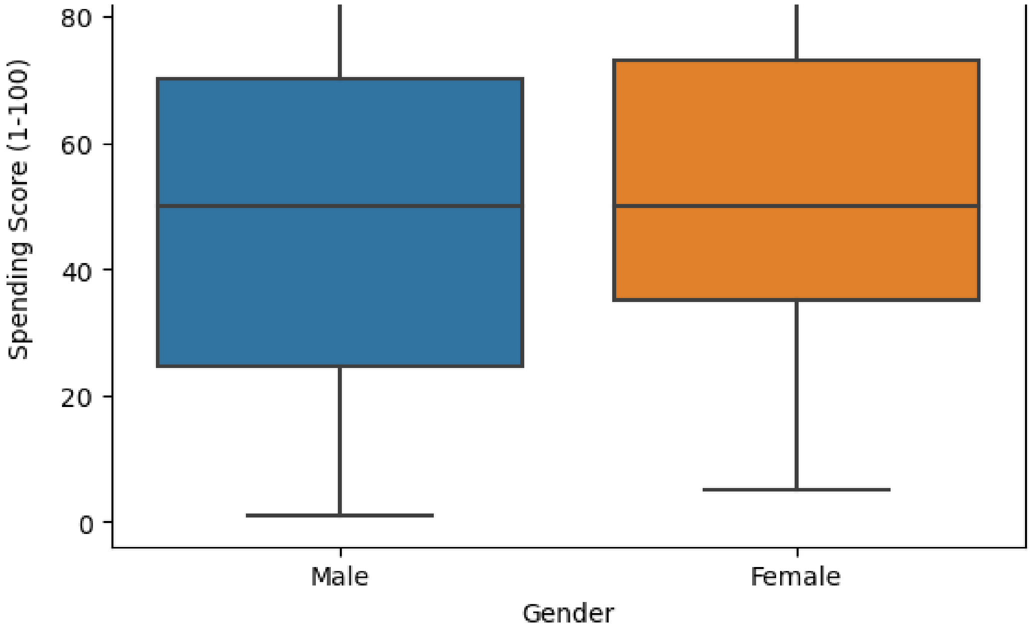


In [29]:

```
columns = ['Age', 'Annual Income (k$)','Spending Score (1-100)']
for i in columns:
    plt.figure()
    sns.boxplot(data=df,x='Gender',y=df[i])
```







In [32]:

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

Out[32]:

```
Female    0.56
Male      0.44
Name: Gender, dtype: float64
```

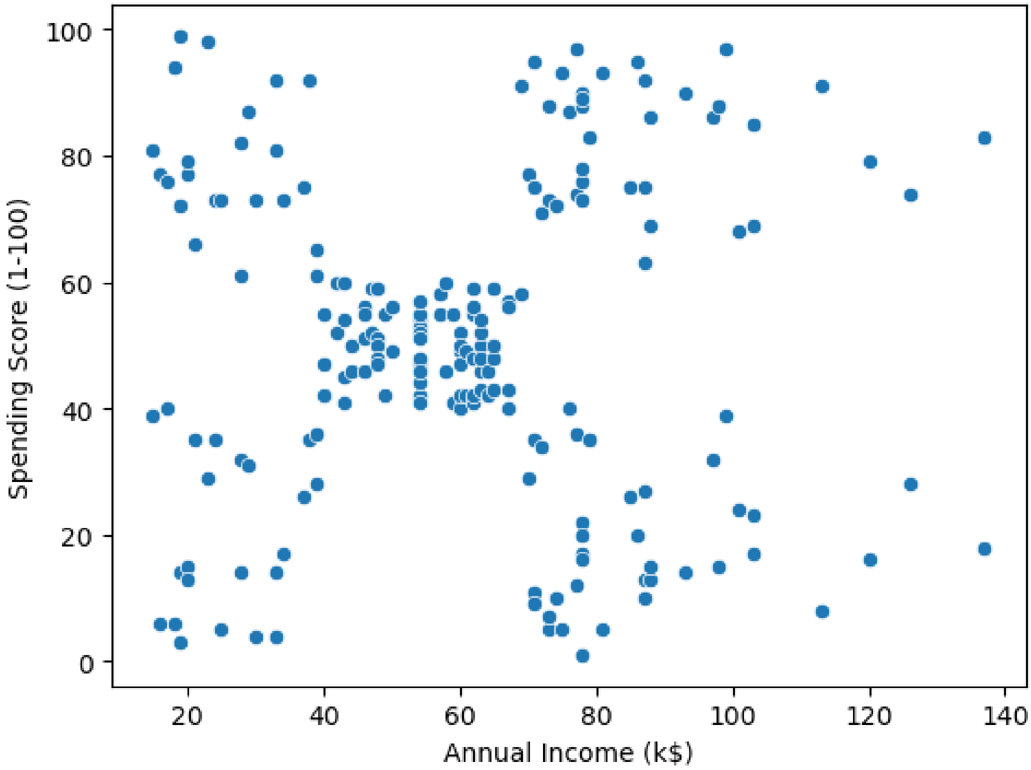
Bivariate analysis

In [36]:

```
sns.scatterplot(data=df,x='Annual Income (k$)',y='Spending Score (1-100)')
```

Out[36]:

```
<AxesSubplot:xlabel='Annual Income (k$)', ylabel='Spending Score (1-100)'>
```



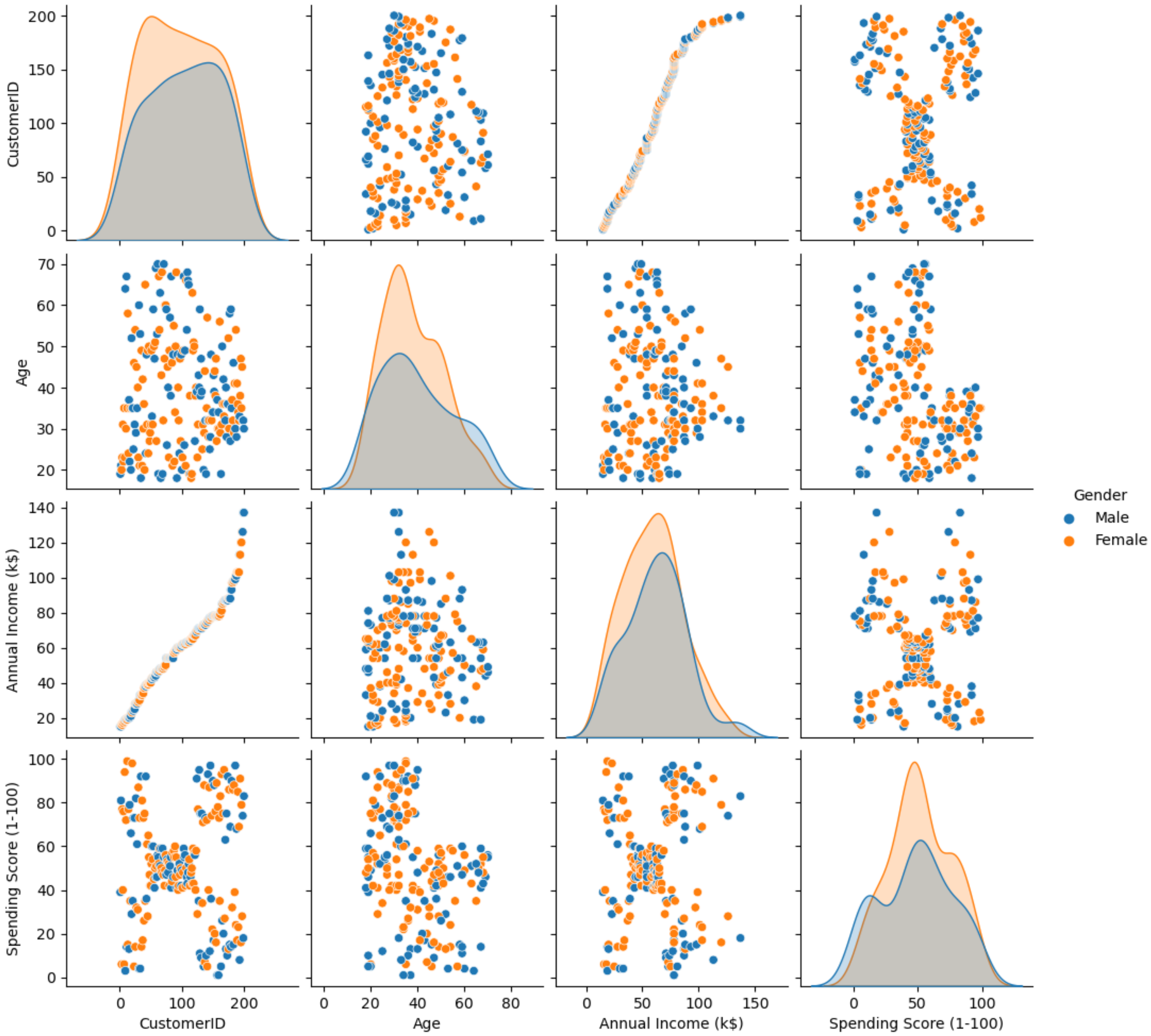
In [37]:

```
sns.pairplot(df,hue='Gender')
```

```
sns.pairplot(df,hue='Gender')
```

Out[37]:

<seaborn.axisgrid.PairGrid at 0x156a7c34400>



In [38]:

```
df.groupby(['Gender'])['Age', 'Annual Income (k$)',
       'Spending Score (1-100)'].mean()
```

Out[38]:

	Age	Annual Income (k$)	Spending Score (1-100)
Gender			
Female	38.098214	59.250000	51.526786
Male	39.806818	62.227273	48.511364

In [39]:

```
df.corr()
```

Out[39]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
CustomerID	1.000000	-0.026763	0.977548	0.013835

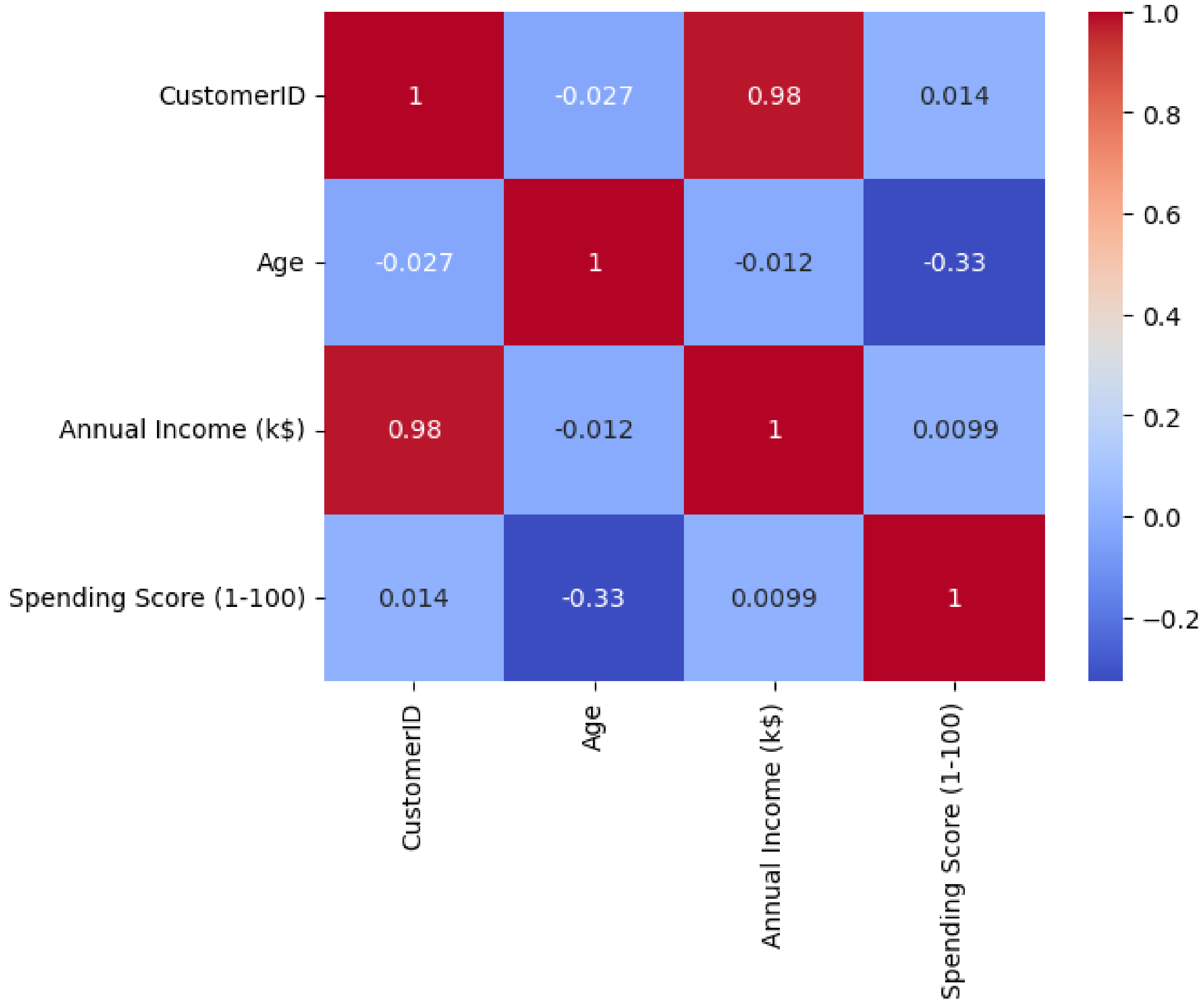
	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
Age	-0.026763	1.000000	-0.012398	-0.327227
Annual Income (k$)	0.977548	-0.012398	1.000000	0.009903
Spending Score (1-100)	0.013835	-0.327227	0.009903	1.000000

In [42]:

```
sns.heatmap(df.corr(),annot=True,cmap='coolwarm')
```

Out[42]:

<AxesSubplot:>



Annual income has a positive correlation with the spending score of customers while age has a negative correlation with annual income and spending score.

Clustering

In [43]:

```
clustering1=KMeans(n_clusters=3)
```

In [44]:

```
clustering1.fit(df[['Annual Income (k$)']])
```

Out[44]:

KMeans(n_clusters=3)

In [45]:

```
clustering1.labels_
```

Out[45]:

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1])
```

In [46]:

```
df['Income Cluster']=clustering1.labels_
df.head()
```

Out[46]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster
0	1	Male	19	15	39	0
1	2	Male	21	15	81	0
2	3	Female	20	16	6	0
3	4	Female	23	16	77	0
4	5	Female	31	17	40	0

In [47]:

```
df['Income Cluster'].value_counts()
```

Out[47]:

```
2    90
0    74
1    36
Name: Income Cluster, dtype: int64
```

In [48]:

```
clustering1.inertia_
```

Out[48]:

```
23517.330930930926
```

In [50]:

```
intertia_scores=[]
for i in range(1,11):
    kmeans=KMeans(n_clusters=i)
    kmeans.fit(df[['Annual Income (k$)']])
    intertia_scores.append(kmeans.inertia_)
```

In [51]:

```
intertia_scores
```

Out[51]:

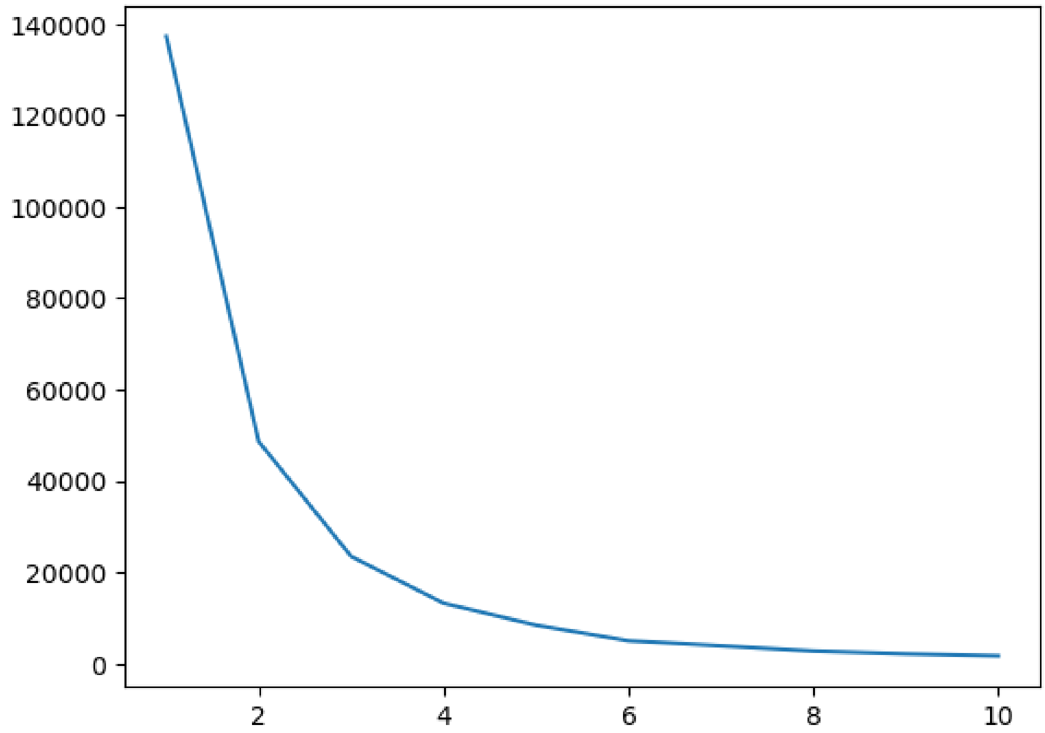
```
[137277.28000000003,
 48660.88888888889,
 23517.330930930926,
 13278.112713472487,
 8481.496190476191,
 5050.904761904763,
 3949.2756132756135,
 2822.4996947496943,
 2222.930303030303,
 1766.6142857142859]
```

In [52]:

```
plt.plot(range(1,11),intertia_scores)
```

Out[52]:

```
[<matplotlib.lines.Line2D at 0x156a85a8be0>]
```



In [53]:

```
df.columns
```

Out[53]:

```
Index(['CustomerID', 'Gender', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)', 'Income Cluster'],
      dtype='object')
```

In [54]:

```
df.groupby('Income Cluster')['Age', 'Annual Income (k$)',
       'Spending Score (1-100)'].mean()
```

Out[54]:

	Age	Annual Income (k$)	Spending Score (1-100)
Income Cluster			
0	39.500000	33.486486	50.229730
1	37.833333	99.888889	50.638889
2	38.722222	67.088889	50.000000

In [55]:

```
clustering2 = KMeans(n_clusters=5)
clustering2.fit(df[['Annual Income (k$)','Spending Score (1-100)']])
df['Spending and Income Cluster'] =clustering2.labels_
df.head()
```

Out[55]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster	Spending and Income Cluster
0	1	Male	19	15	39	0	3

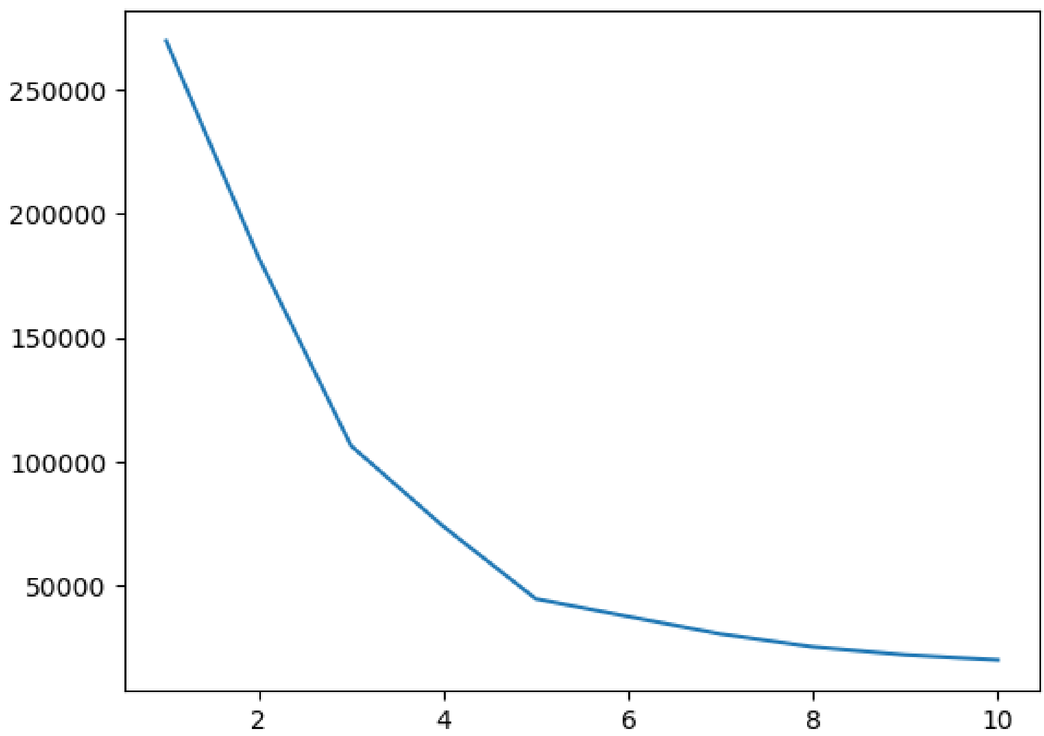
	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster	Spending and Income Cluster
0	1	Male	19	15	39	0	3
1	2	Male	21	15	81	0	1
2	3	Female	20	16	6	0	3
3	4	Female	23	16	77	0	1
4	5	Female	31	17	40	0	3

In [56]:

```
intertia_scores2=[]
for i in range(1,11):
    kmeans2=KMeans(n_clusters=i)
    kmeans2.fit(df[['Annual Income (k$)','Spending Score (1-100)']])
    intertia_scores2.append(kmeans2.inertia_)
plt.plot(range(1,11),intertia_scores2)
```

Out[56]:

```
[<matplotlib.lines.Line2D at 0x156a8618670>]
```

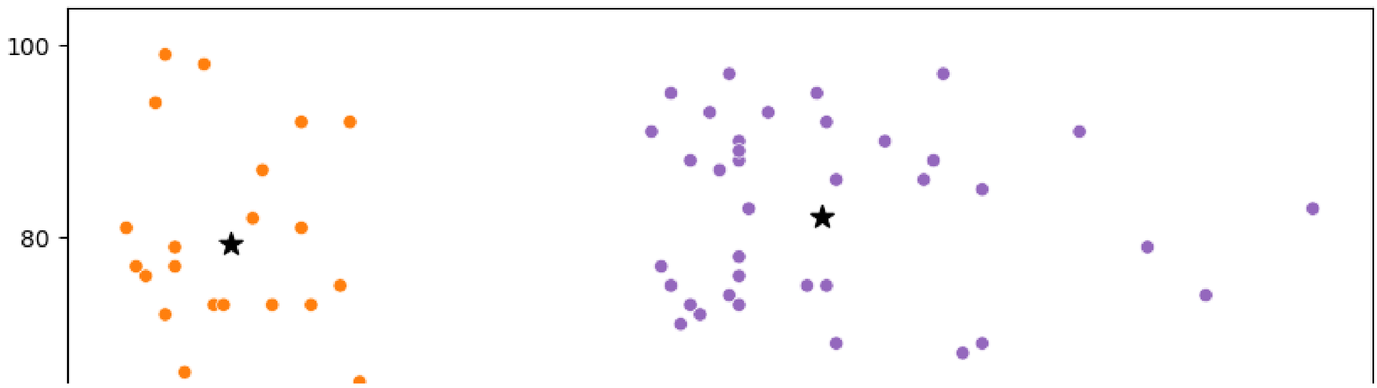


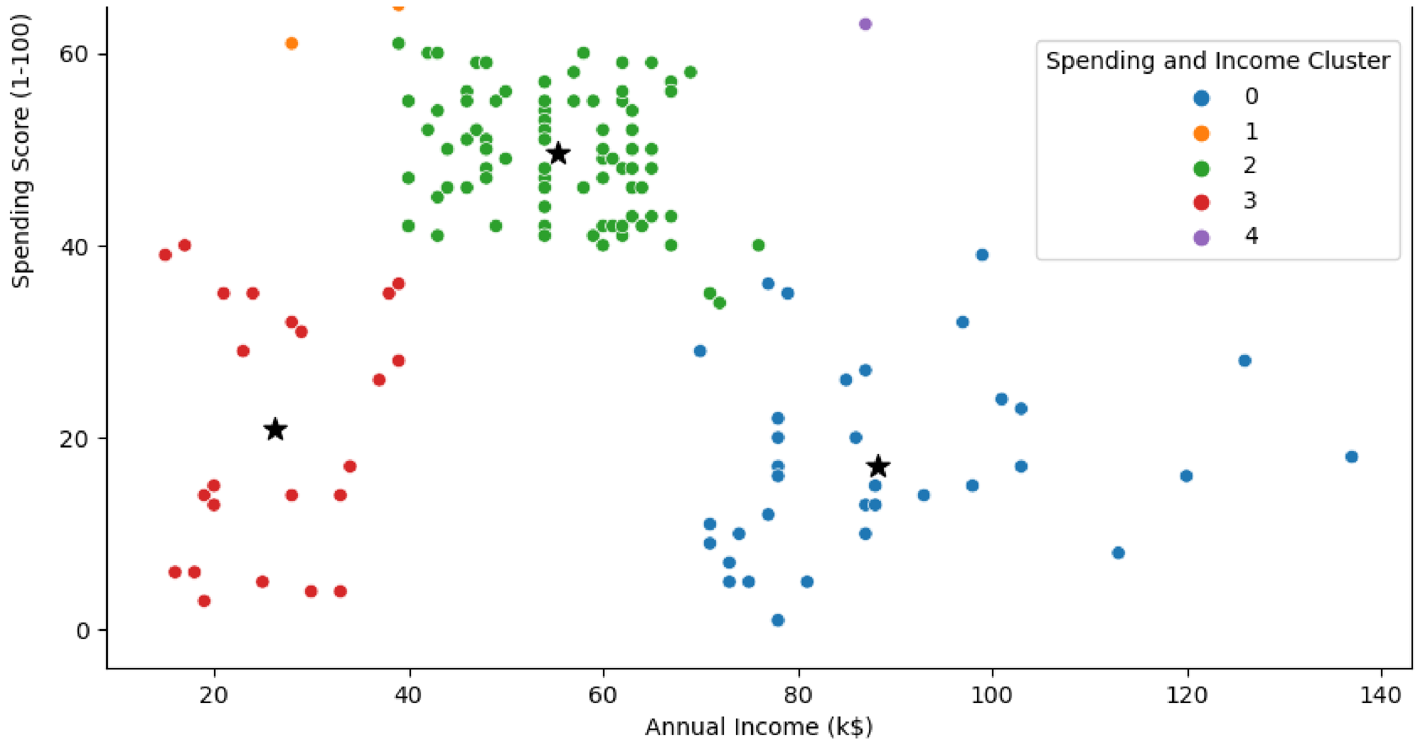
In [57]:

```
centers =pd.DataFrame(clustering2.cluster_centers_)
centers.columns = ['x','y']
```

In [58]:

```
plt.figure(figsize=(10,8))
plt.scatter(x=centers['x'],y=centers['y'],s=100,c='black',marker='*')
sns.scatterplot(data=df, x ='Annual Income (k$)',y='Spending Score (1-100)',hue='Spendin
g and Income Cluster',palette='tab10')
plt.savefig('clustering_bivaraiate.png')
```





cluster 0 (blue) represents high income and low spending customers

cluster 1 (orange) represents low income and high spendng customers

cluster 2 (green) represents midium income and medium spending customers

cluster 3 (red) represents low income and low spending customers

cluster 4 (purple) represents high income and high spending customers

In [59]:

```
pd.crosstab(df['Spending and Income Cluster'],df['Gender'],normalize='index')
```

Out[59]:

Gender	Female	Male
Spending and Income Cluster		
0	0.457143	0.542857
1	0.590909	0.409091
2	0.592593	0.407407
3	0.608696	0.391304
4	0.538462	0.461538

cluster 3 and cluster 1 are female dominated clusters that include both low income, low spending customer (cluster 3,females at 60%) and low income, high spending customer(cluster 1, female at 59 %).

In contrast cluster 0 which represents high income and low spending customers is male dominated with 54.2% males.cluster 2 which indicates loyal customers is famale dominated with females at 59.2%.cluster 4 which represents a high income and high spending customers is gender balanced.

In [60]:

```
df.groupby('Spending and Income Cluster')['Age', 'Annual Income (k$)',
       'Spending Score (1-100)'].mean()
```

Out[60]:

Age	Annual Income (k$)	Spending Score (1-100)

Spending and Income Cluster	Age	Annual Income (k$)	Spending Score (1-100)
Spending and Income Cluster			
0	41.114286	88.200000	17.114286
1	25.272727	25.727273	79.363636
2	42.716049	55.296296	49.518519
3	45.217391	26.304348	20.913043
4	32.692308	86.538462	82.128205

The results above indicates that:

cluster 0 has old customers who are at an average age of 41 years and earning 88.2k$ avarage income.

cluster 1 on the other hand has customers who are young with 25 years in average and earning an average income of 25.7k$.

Cluster 2 has middle-aged customers with an average age of 42.7 years who are earning an income of 55.4k$.

Cluster 3 has oldest customers with an average age of 45 years and earning an average income of 26.3k$.

lastly cluster 4 has young to middle aged customers with an average age of 32 years and earning an average income of 86.5k$

Mulivariate clustering

In [61]:

```
from sklearn.preprocessing import StandardScaler
```

In [62]:

```
scale = StandardScaler()
```

In [63]:

```
df.head()
```

Out[63]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster	Spending and Income Cluster
0	1	Male	19	15	39	0	3
1	2	Male	21	15	81	0	1
2	3	Female	20	16	6	0	3
3	4	Female	23	16	77	0	1
4	5	Female	31	17	40	0	3

In [64]:

```
dff = pd.get_dummies(df,drop_first=True)
dff.head()
```

Out[64]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster	Spending and Income Cluster	Gender_Male
0	1	19	15	39	0	3	1
1	2	21	15	81	0	1	1
2	3	20	16	6	0	3	0
3	4	23	16	77	0	1	0
4	5	31	17	40	0	3	0

In [65]:

```
dff.columns
```

Out[65]:

```
Index(['CustomerID', 'Age', 'Annual Income (k$)', 'Spending Score (1-100)',
       'Income Cluster', 'Spending and Income Cluster', 'Gender_Male'],
      dtype='object')
```

In [66]:

```
dff = dff[['Age', 'Annual Income (k$)', 'Spending Score (1-100)','Gender_Male']]
dff.head()
```

Out[66]:

	Age	Annual Income (k$)	Spending Score (1-100)	Gender_Male
0	19	15	39	1
1	21	15	81	1
2	20	16	6	0
3	23	16	77	0
4	31	17	40	0

In [67]:

```
dff = scale.fit_transform(dff)
```

In [68]:

```
dff = pd.DataFrame(scale.fit_transform(dff))
dff.head()
```

Out[68]:

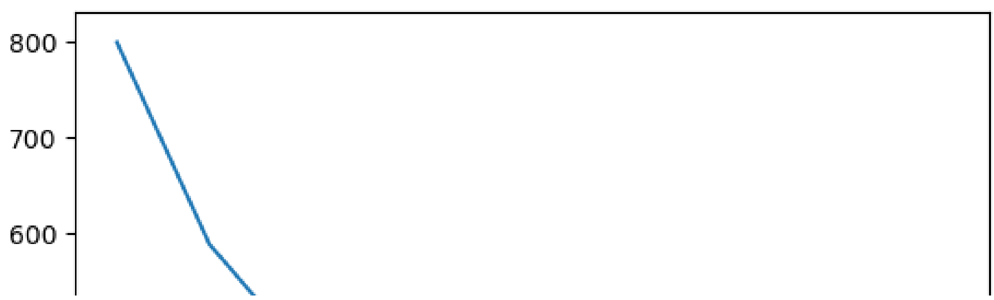
	0	1	2	3
0	-1.424569	-1.738999	-0.434801	1.128152
1	-1.281035	-1.738999	1.195704	1.128152
2	-1.352802	-1.700830	-1.715913	-0.886405
3	-1.137502	-1.700830	1.040418	-0.886405
4	-0.563369	-1.662660	-0.395980	-0.886405

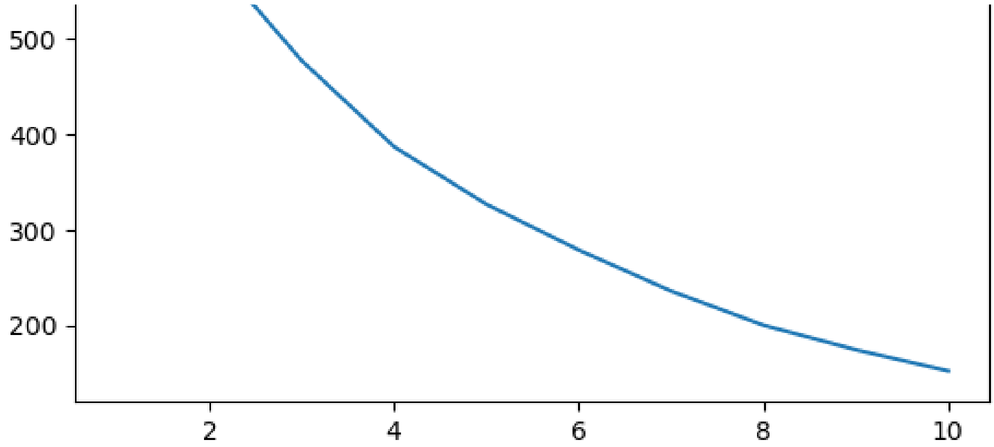
In [69]:

```
intertia_scores3=[]
for i in range(1,11):
    kmeans3=KMeans(n_clusters=i)
    kmeans3.fit(dff)
    intertia_scores3.append(kmeans3.inertia_)
plt.plot(range(1,11),intertia_scores3)
```

Out[69]:

```
[<matplotlib.lines.Line2D at 0x156a88fd580>]
```





In [70]:

```
df
```

Out[70]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Income Cluster	Spending and Income Cluster
0	1	Male	19	15	39	0	3
1	2	Male	21	15	81	0	1
2	3	Female	20	16	6	0	3
3	4	Female	23	16	77	0	1
4	5	Female	31	17	40	0	3
...	...	...	...	...	...	...	...
195	196	Female	35	120	79	1	4
196	197	Female	45	126	28	1	0
197	198	Male	32	126	74	1	4
198	199	Male	32	137	18	1	0
199	200	Male	30	137	83	1	4

200 rows × 7 columns

In [ ]:

```
df.to_csv('Clustering.csv')
```