In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from mpl_toolkits.mplot3d import Axes3D

%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
data=pd.read_csv("C:/Users/zoaah/OneDrive/Documents/Mall_Customers.csv")
```

In [3]:

```
data.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]:

```
data.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

In [5]:

```
data1=data.sample(n=150)
```

In [6]:

```
data1.shape
```

Out[6]:

```
(150, 5)
```

In [7]:

```
data1.head()
```

Out[7]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
149	150	Male	34	78	90
92	93	Male	48	60	49
160	161	Female	56	79	35
118	119	Female	51	67	43
157	158	Female	30	78	78

In [8]:

```
data1.describe()
```

Out[8]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	150.000000	150.000000	150.000000	150.000000
mean	101.406667	38.866667	60.806667	51.120000
std	56.280112	13.825697	25.164078	26.281624
min	1.000000	18.000000	15.000000	1.000000
25%	54.250000	29.000000	43.000000	35.000000
50%	100.500000	36.500000	61.500000	51.000000
75%	150.750000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

In [9]:

```
#null vaues
null_values=data1.isnull().sum()
```

In [10]:

```
null_values
```

Out[10]:

```
CustomerID              0
Gender                  0
Age                     0
Annual Income (k$)      0
Spending Score (1-100)  0
dtype: int64
```

In [11]:

```
#correlation
corr=data1.corr()
corr
```

Out[11]:

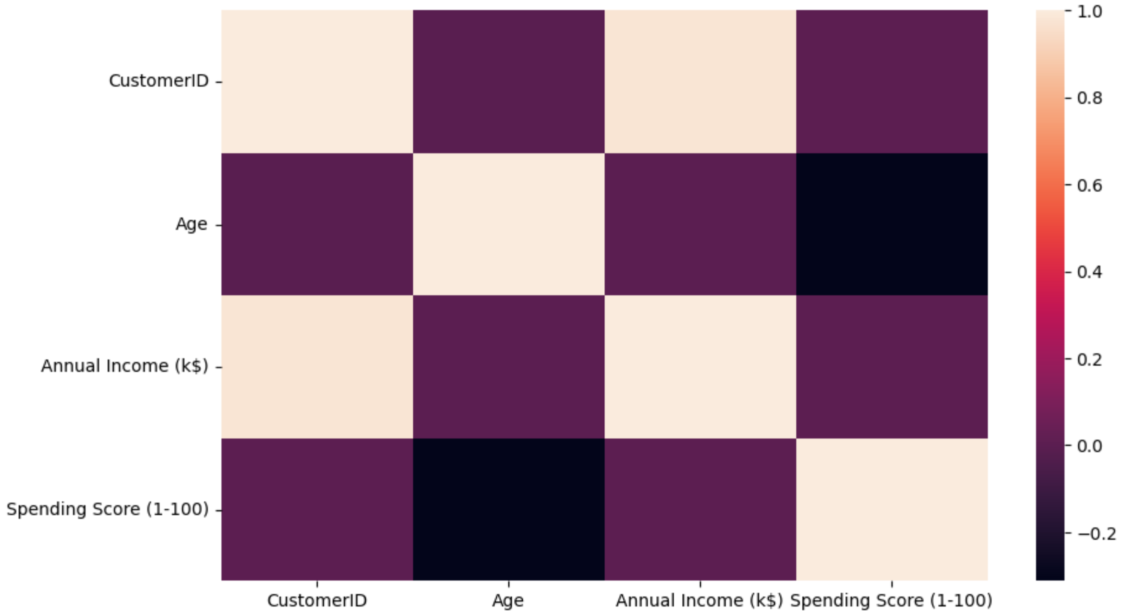
	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
CustomerID	1.000000	-0.007028	0.977109	0.005530
Age	-0.007028	1.000000	0.000736	-0.311827
Annual Income (k$)	0.977109	0.000736	1.000000	0.002633
Spending Score (1-100)	0.005530	-0.311827	0.002633	1.000000

In [16]:

```
#heatmap & exploratory data analysis
plt.figure(figsize=(10,6))
heatmap=sns.heatmap(corr)
heatmap
```

Out[16]:

```
<Axes: >
```

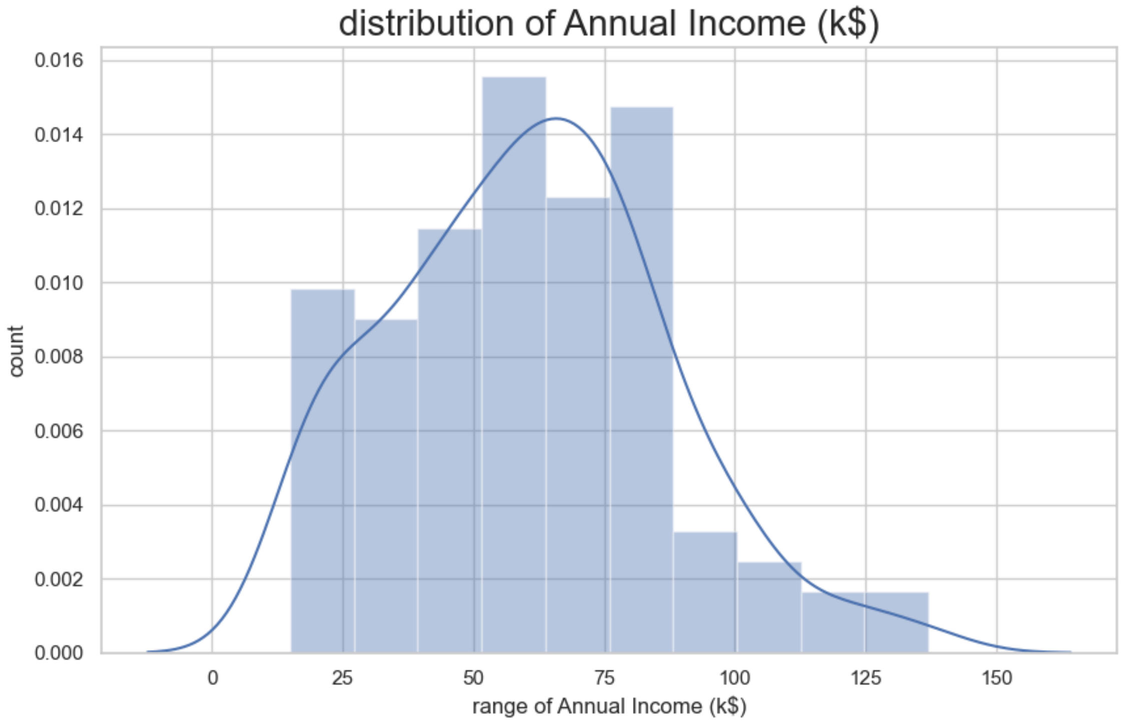


In [17]:

```
#distribution of annual income
plt.figure(figsize=(10,6))
sns.set(style='whitegrid')
sns.distplot(data['Annual Income (k$)'])
plt.title('distribution of Annual Income (k$)', fontsize=20)
plt.xlabel('range of Annual Income (k$)')
plt.ylabel('count')
```

Out[17]:

```
Text(0, 0.5, 'count')
```



In [18]:

```
#distribution of spending score
plt.figure(figsize=(10,6))
sns.set(style='whitegrid')
sns.distplot(data['Spending Score (1-100)'])
plt.title('distribution of Spending Score (1-100)',fontsize=20)
plt.xlabel('range of Spending Score (1-100)')
plt.ylabel('count')
```

Out[18]:

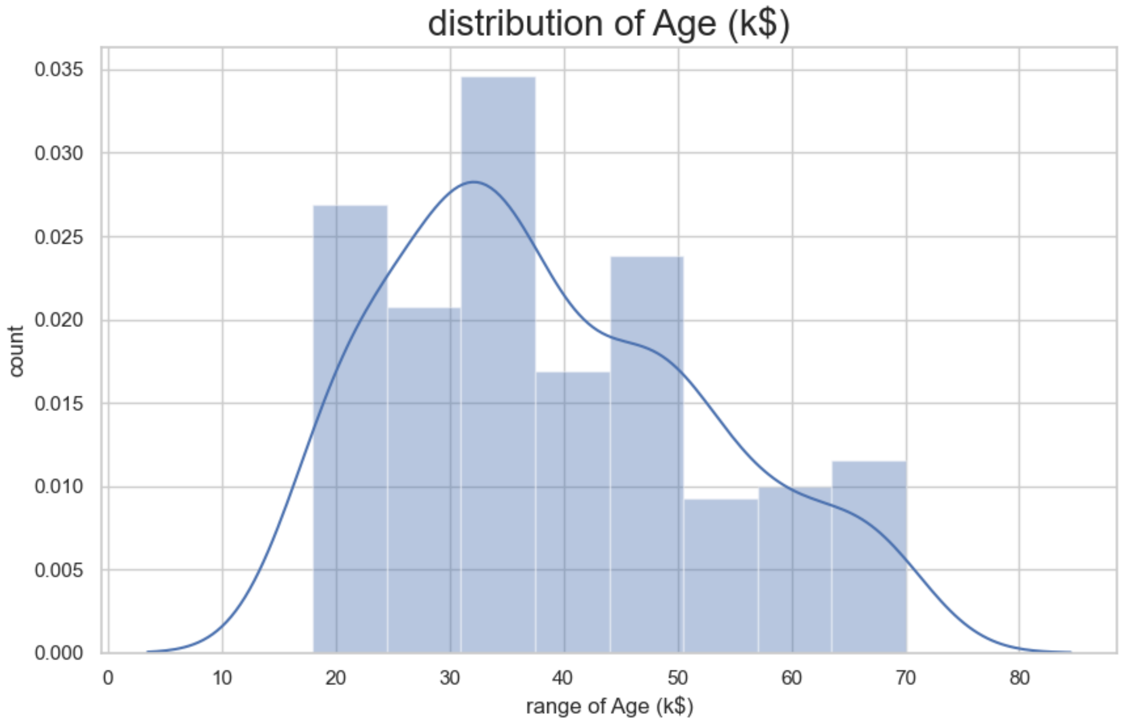
```
Text(0, 0.5, 'count')
```

In [19]:

```
#distribution of age
plt.figure(figsize=(10,6))
sns.set(style='whitegrid')
sns.distplot(data['Age'])
plt.title('distribution of Age (k$)', fontsize=20)
plt.xlabel('range of Age (k$)')
plt.ylabel('count')
```

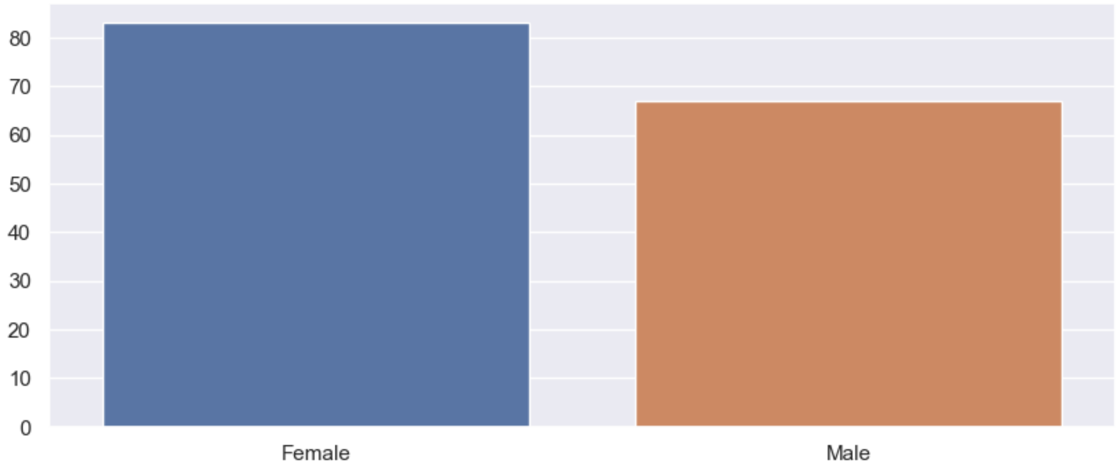
Out[19]:

```
Text(0, 0.5, 'count')
```



In [20]:

```
genders = data1.Gender.value_counts()
sns.set_style("darkgrid")
plt.figure(figsize=(10,4))
sns.barplot(x=genders.index , y=genders.values)
plt.show()
```



In [21]:

```
x=data1[["Annual Income (k$)","Spending Score (1-100)"]]
x.head()
```

Out[21]:

	Annual Income (k$)	Spending Score (1-100)
149	78	90
92	60	49
160	79	35
118	67	43
157	78	78

In [22]:

```
#scattered plot of input data
plt.figure(figsize=(10,6))
sns.scatterplot(x='Annual Income (k$)', y='Spending Score (1-100)', data=x,s=60)
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100) vs Annual Income (k$)')
plt.show()
```

In [23]:

```
#import kmeans from sklearn
from sklearn.cluster import KMeans
```

In [24]:

```
kmeans= KMeans(n_clusters=2, random_state=0).fit(x)
y=kmeans.labels_
```

In [25]:

```
y
```

Out[25]:

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

In [27]:

```
data1["label"]=y
```

In [28]:

```
data1.head()
```

Out[28]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	label
149	150	Male	34	78	90	0
92	93	Male	48	60	49	0
160	161	Female	56	79	35	1
118	119	Female	51	67	43	1
157	158	Female	30	78	78	0

In [52]:

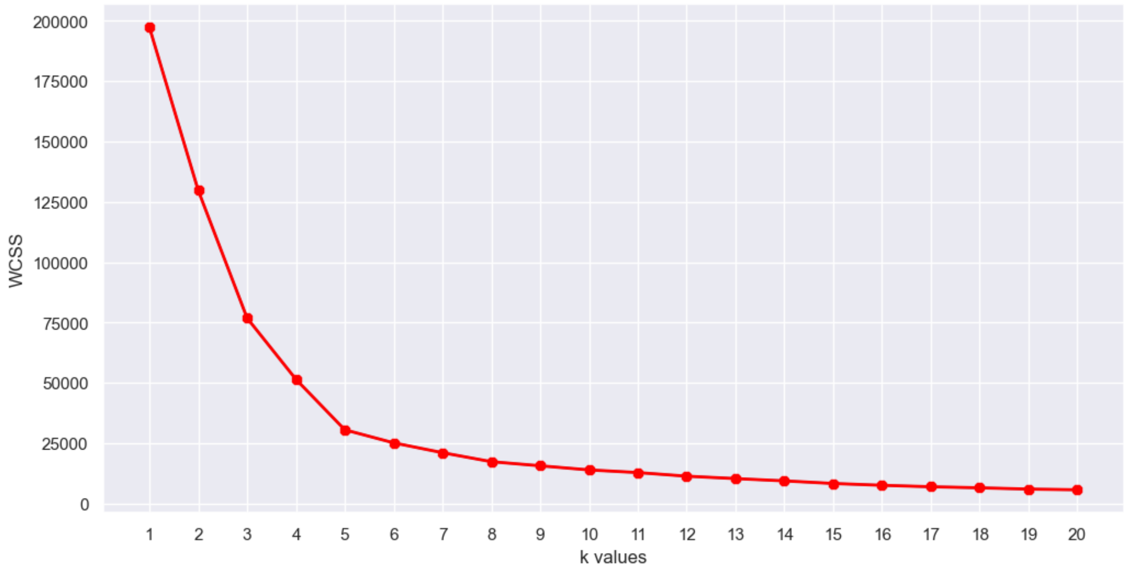
```
#scatter plot with two clusters
plt.figure(figsize=(10,6))
sns.scatterplot(x='Annual Income (k$)',y='Spending Score (1-100)',hue="label", palette=[
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100)vs Annual Income (k$)')
plt.show()
```

In [42]:

```
WCSS=[]
for i in range(1,21):
    km=KMeans(n_clusters=i)
    km.fit(x)
    WCSS.append(km.inertia_)
```

In [43]:

```
#the elbow curve
plt.figure(figsize=(12,6))
plt.plot(range(1,21),WCSS)
plt.plot(range(1,21),WCSS, linewidth=2, color="red", marker="8")
plt.xlabel("k values")
plt.xticks(np.arange(1,21,1))
plt.ylabel("WCSS")
plt.show()
```



In [44]:

```
#taking 5 clusters(kmeans model training)
kmeans_WCSS=KMeans(n_clusters=5)
kmeans_WCSS.fit(x)
y=kmeans_WCSS.predict(x)
data1["label"]=y
data1.head()
```

Out[44]:

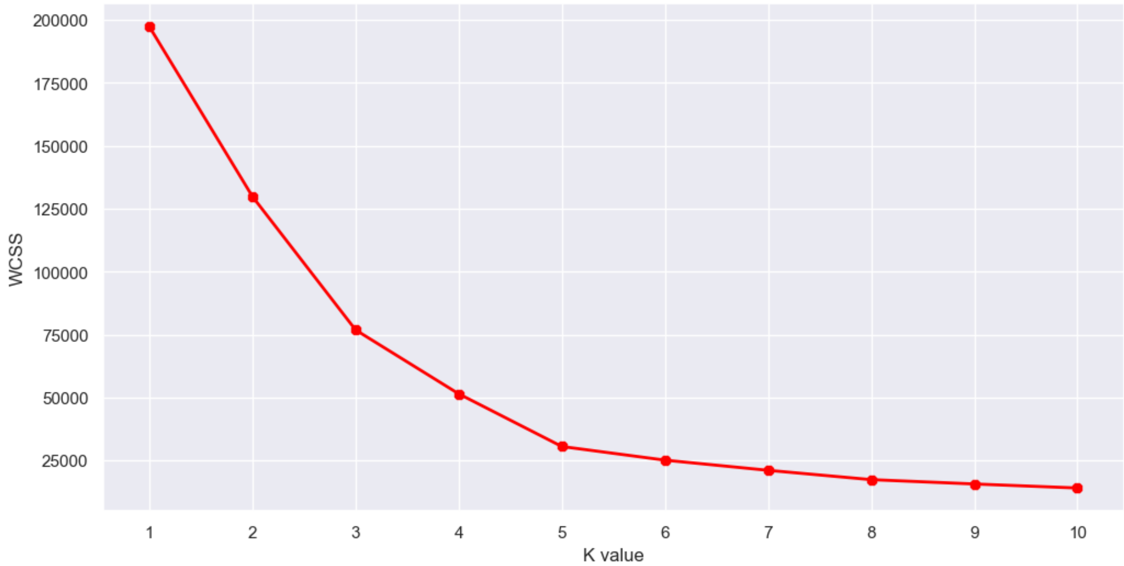
	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	label
149	150	Male	34	78	90	3
92	93	Male	48	60	49	2
160	161	Female	56	79	35	0
118	119	Female	51	67	43	2
157	158	Female	30	78	78	3

In [50]:

```
#scatter plot with 5 clusters
plt.figure(figsize=(10,6))
sns.scatterplot(x='Annual Income (k$)',y='Spending Score (1-100)',hue="label",palette=['
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100)vs Annual Income (k$)')
plt.show()
```

In [56]:

```
#using kmeans++ for elbow curve
#taking the features
x1=data1[["Age","Annual Income (k$)","Spending Score (1-100)"]]
WCSS=[]
for k in range(1,11):
    kmeans=KMeans(n_clusters=k,init="k-means++")
    kmeans.fit(x)
    WCSS.append(kmeans.inertia_)
plt.figure(figsize=(12,6))
plt.plot(range(1,11),WCSS,linewidth=2,color="red",marker="8")
plt.xlabel("K value")
plt.xticks(np.arange(1,11,1))
plt.ylabel("WCSS")
plt.show()
```



In [57]:

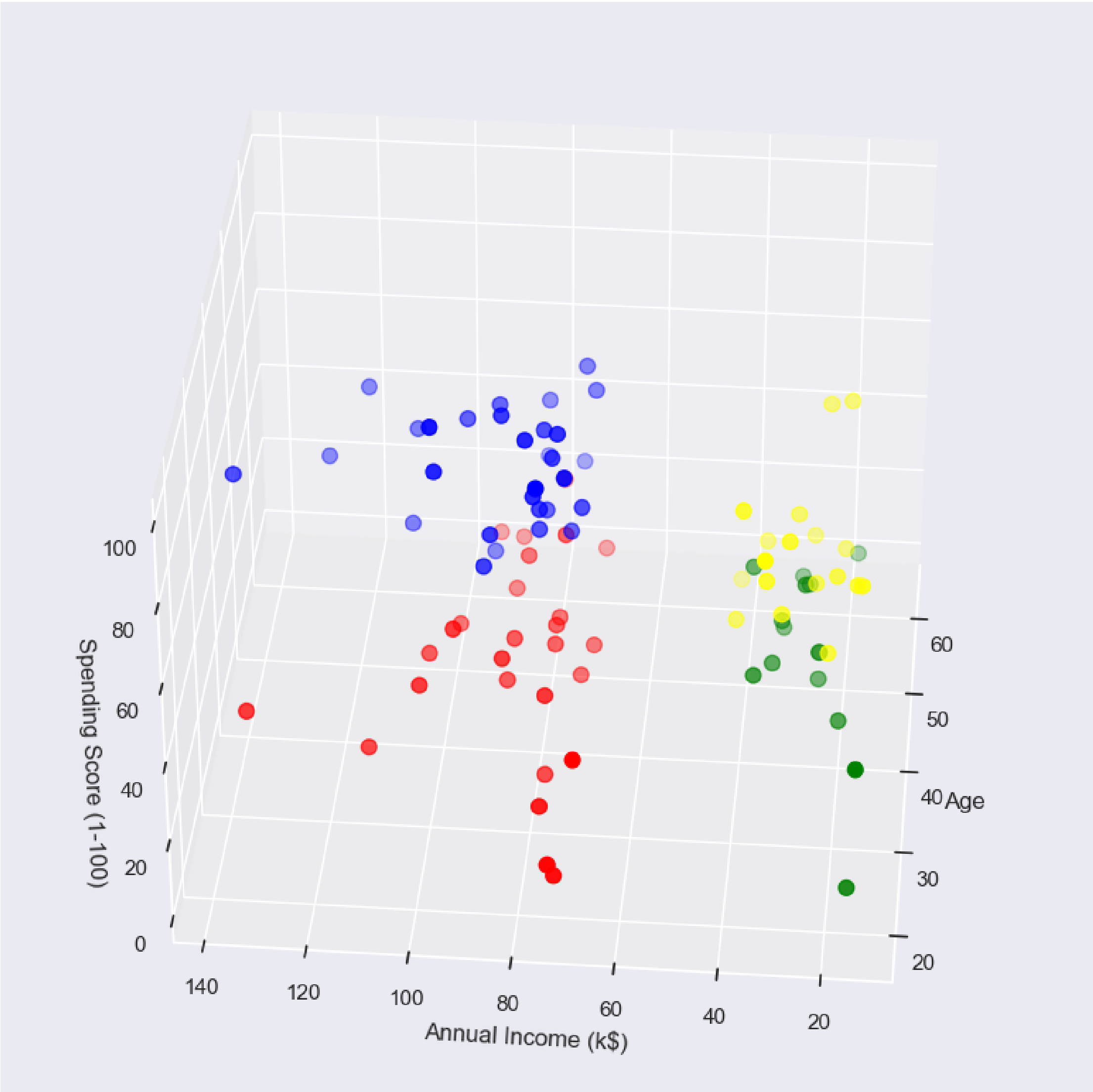
```
kmeans2=KMeans(n_clusters=5)
y2=kmeans2.fit_predict(x1)
data1["label"]=y2
data1.head()
```

Out[57]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	label
149	150	Male	34	78	90	2
92	93	Male	48	60	49	0
160	161	Female	56	79	35	1
118	119	Female	51	67	43	0
157	158	Female	30	78	78	2

In [59]:

```
#3d plot as we  did the clustering on the basis of 3 input features
fig=plt.figure(figsize=(10,15))
ax=fig.add_subplot(111,projection='3d')
ax.scatter(data1.Age[data1.label==1],data1["Annual Income (k$)"][data1.label==1],data1['
ax.scatter(data1.Age[data1.label==2],data1["Annual Income (k$)"][data1.label==2],data1['
ax.scatter(data1.Age[data1.label==3],data1["Annual Income (k$)"][data1.label==3],data1['
ax.scatter(data1.Age[data1.label==4],data1["Annual Income (k$)"][data1.label==4],data1['
ax.view_init(35,185)
plt.xlabel("Age")
plt.ylabel("Annual Income (k$)")
ax.set_zlabel('Spending Score (1-100)')
plt.show()
```



In [60]:

```
df=data1.groupby(['label'])['Age','Annual Income (k$)','Spending Score (1-100)'].mean()
df['N obs']=data1[['label','Gender']].groupby(['label']).count()
```

In [61]:

```
df
```

Out[61]:

label	Age	Annual Income (k$)	Spending Score (1-100)	N obs
0	43.770492	54.950820	49.803279	61
1	40.851852	85.925926	17.444444	27
2	32.766667	86.100000	83.233333	30
3	43.500000	27.214286	17.142857	14
4	25.833333	26.944444	79.000000	18

`conclusion for the overall project`

In this customer behavior analysis project, we employed K-means clustering, a popular algorithm for customer segmentation. The main goal was to categorize customers into distinct groups based on their similarities. To start, we collected relevant data encompassing demographics, purchase history, website interactions, and more. After preprocessing the data to ensure its quality, we selected the most informative features for the segmentation process. Next, we faced the task of determining the ideal number of clusters (K) to create meaningful segments. We used techniques like the Elbow Method or Silhouette Score to find the optimal value for K. Throughout the project, we continuously monitored customer behavior and assessed the success of our segmentation and marketing efforts. This ongoing evaluation ensured that our approach remained effective in accommodating changing customer preferences. Overall, the project's implementation of K-means clustering proved to be a valuable tool for understanding customer behavior and driving targeted business strategies.

`REMARKS`

1. Label 0:
   - Age: The average age is approximately 44 years.
   - Annual Income: The average annual income is around $55,000(k$ = 1,000 dollars).
   - Spending Score: The average spending score is approximately 50 (out of 100).
   - Observations: There are 61 data points in this group.
2. Label 1:
   - Age: The average age is approximately 41 years.
   - Annual Income: The average annual income is relatively high at around $86,000.
   - Spending Score: The average spending score is quite low at approximately 17.
   - Observations: There are 27 data points in this group.
3. Label 2:
   - Age: The average age is around 33 years.
   - Annual Income: The average annual income is high, similar to Label 1, at approximately $86,000.
   - Spending Score: The average spending score is high, approximately 83 out of 100.
   - Observations: There are 30 data points in this group.
4. Label 3:

- Age: The average age is approximately 44 years.
- Annual Income: The average annual income is relatively low, around $27,000.
- Spending Score: The average spending score is quite low, approximately 17 out of 100.
- Observations: There are 14 data points in this group.

5. **Label 4**:

   - Age: The average age is around 26 years.
   - Annual Income: The average annual income is relatively low, similar to Label 3, at approximately $27,000.
   - Spending Score: The average spending score is high, around 79 out of 100.
   - Observations: There are 18 data points in this group.

From the descriptions above, we can see that:

- Labels 1 and 2 represent customers with high annual incomes, but their spending behavior differs significantly. Label 1 has low spending scores, while Label 2 has high spending scores.
- Labels 3 and 4 represent customers with lower annual incomes. Label 4, however, has a higher spending score compared to Label 3.

This information is valuable for market segmentation and understanding customer behavior to devise appropriate marketing strategies for different customer groups.