importing pandas

```
import pandas as pd
```

```
df=pd.read_csv('Mall_Customers.csv')
```

printing the dataframe

```
df
```

	CustomerID	Genre	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
...	...	...	...	...	...
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

200 rows × 5 columns

```
df.columns
```

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

printing the data types

```
df.dtypes
```

	0
CustomerID	int64
Genre	object
Age	int64
Annual Income (k$)	int64
Spending Score (1-100)	int64

```
import numpy as np
```

```
import matplotlib.pyplot as plt
import seaborn as sns
```

```
print(df.isnull().sum())
```

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

```
df.head()
```

	CustomerID	Genre	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

```
df.drop(['CustomerID'],axis=1,inplace=True)
```

```
df.head()
```

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

preprocessing

```
from sklearn.preprocessing import LabelEncoder
```

```
from sklearn import metrics
```

```
le = LabelEncoder()
```

```
df["Genre"] = le.fit_transform(df["Genre"])
```

```
df.head()
```

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

```
data = df.copy()
x = data.iloc[:,[2,3]]
from sklearn.cluster import KMeans
```

```
data
```

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

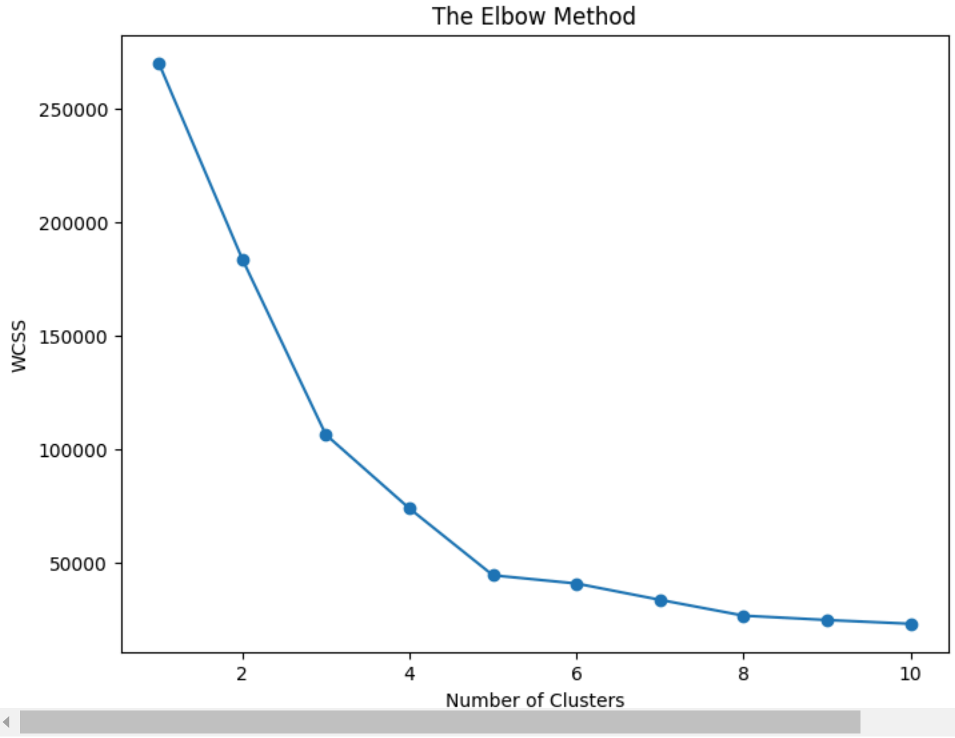
200 rows × 4 columns

```
wcss = []
for i in range(1,11):
    kmeans = KMeans(n_clusters=i,init='k-means++',random_state=42)
    kmeans.fit(x)
    wcss.append(kmeans.inertia_)
    print('k:',i,"->wcss:",kmeans.inertia_)
```

```
k: 1 ->wcss: 269981.28000000014
k: 2 ->wcss: 183653.3289473683
k: 3 ->wcss: 106348.37306211119
k: 4 ->wcss: 73880.64496247198
k: 5 ->wcss: 44448.45544793369
k: 6 ->wcss: 40825.16946386947
k: 7 ->wcss: 33642.57922077922
k: 8 ->wcss: 26686.837785187785
k: 9 ->wcss: 24766.471609793436
k: 10 ->wcss: 23103.122085983905
```

plotting the graph

```
plt.figure(figsize=(8, 6))
plt.plot(range(1,11),wcss,marker='o')
plt.title('The Elbow Method')
plt.xlabel('Number of Clusters')
plt.ylabel('WCSS')
plt.show()
```

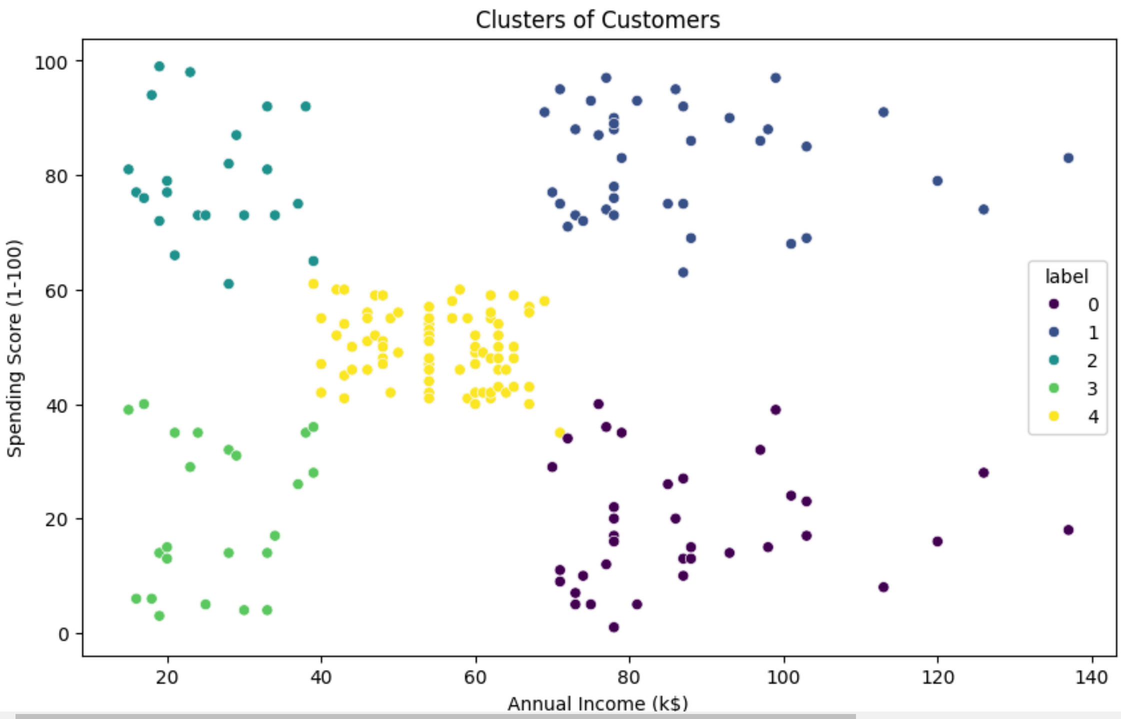


fitting of data

```
kmeans = KMeans(n_clusters=5)
kmeans.fit(data)
y=kmeans.predict(data)
data["label"] = y
```

visualization of clusters

```
plt.close()
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Annual Income (k$)', y='Spending Score (1-100)', hue='label', data=data, palette='viridis')
plt.title('Clusters of Customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.show()
```

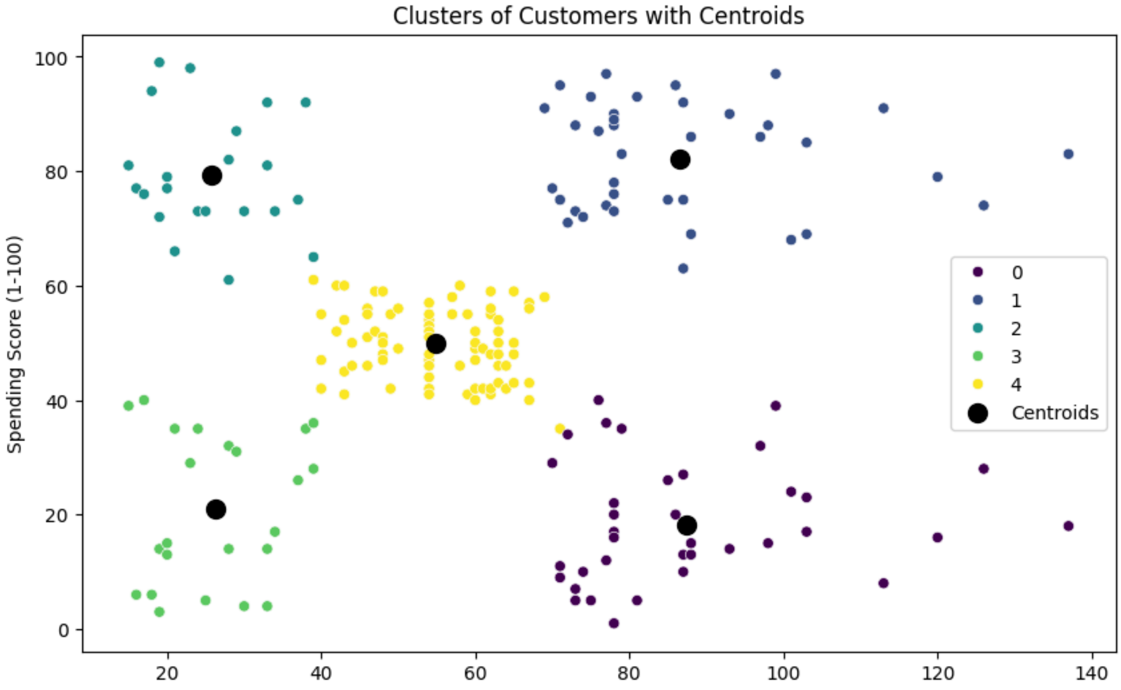


```
centroids = kmeans.cluster_centers_
print(centroids)
```

```
[[ 0.51351351 40.32432432 87.43243243 18.18918919]
 [ 0.46153846 32.69230769 86.53846154 82.12820513]
 [ 0.40909091 25.27272727 25.72727273 79.36363636]
 [ 0.39130435 45.2173913  26.30434783 20.91304348]
 [ 0.41772152 43.12658228 54.82278481 49.83544304]]
```

plotting the centroid

```
plt.close()
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Annual Income (k$)', y='Spending Score (1-100)', hue='label', data=data, palette='viridis')
plt.scatter(centroids[:, 2], centroids[:, 3], s=100, c='black', label='Centroids')
plt.title('Clusters of Customers with Centroids')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```


Clusters of Customers with Centroids
Spending Score (1-100)
0
20
40
60
80
100
20
40
60
80
100
120
140
0
1
2
3
4
Centroids