Program No:- 9

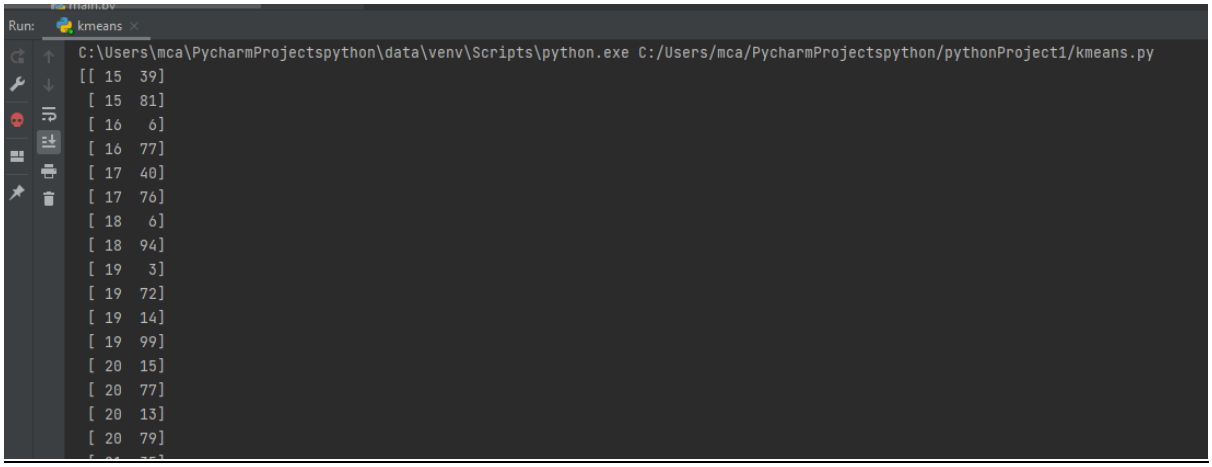
Aim:- Program to implement k-means clustering technique using any standard dataset available in the public domain

Program Code

```
import numpy as nm
import matplotlib.pyplot as mtp
import pandas as pd
dataset = pd.read_csv('Mall_Customers.csv')
x = dataset.iloc[:, [3, 4]].values
print(x)
from sklearn.cluster import KMeans
wcss_list = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state=42)
    kmeans.fit(x)
    wcss_list.append(kmeans.inertia_)
mtp.plot(range(1, 11), wcss_list)
mtp.title('The Elbow Method Graph')
mtp.xlabel('Number of clusters(k)')
mtp.ylabel('wcss_list')
mtp.show()
kmeans = KMeans(n_clusters=5, init='k-means++', random_state=42)
```

```
y_predict = kmeans.fit_predict(x)

print(y_predict)

mtp.scatter(x[y_predict == 0, 0], x[y_predict == 0, 1], s =
100, c = 'blue', label = 'Cluster0')

mtp.scatter(x[y_predict == 1, 0], x[y_predict == 1, 1], s =
100, c = 'green', label = 'Cluster1')

mtp.scatter(x[y_predict == 2, 0], x[y_predict == 2, 1], s =
100, c = 'red', label = 'Cluster2')

mtp.scatter(x[y_predict == 3, 0], x[y_predict == 3, 1], s =
100, c = 'cyan', label = 'Cluster3')

mtp.scatter(x[y_predict == 4, 0], x[y_predict == 4, 1], s =
100, c = 'magenta', label = 'Cluster4')

mtp.scatter(kmeans.cluster_centers_[:, 0],
kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow')

mtp.title('Clusters of customers')

mtp.xlabel('Annual Income (k$)')

mtp.ylabel('Spending Score (1-100)')

mtp.legend()

mtp.show()
```

Output


Run:
kmeans
C:\Users\mca\PycharmProjectspython\data\venv\Scripts\python.exe C:/Users/mca/PycharmProjectspython/pythonProject1/kmeans.py
[[15 39]
 [15 81]
 [16   6]
 [16 77]
 [17 40]
 [17 76]
 [18   6]
 [18 94]
 [19   3]
 [19 72]
 [19 14]
 [19 99]
 [20 15]
 [20 77]
 [20 13]
 [20 79]

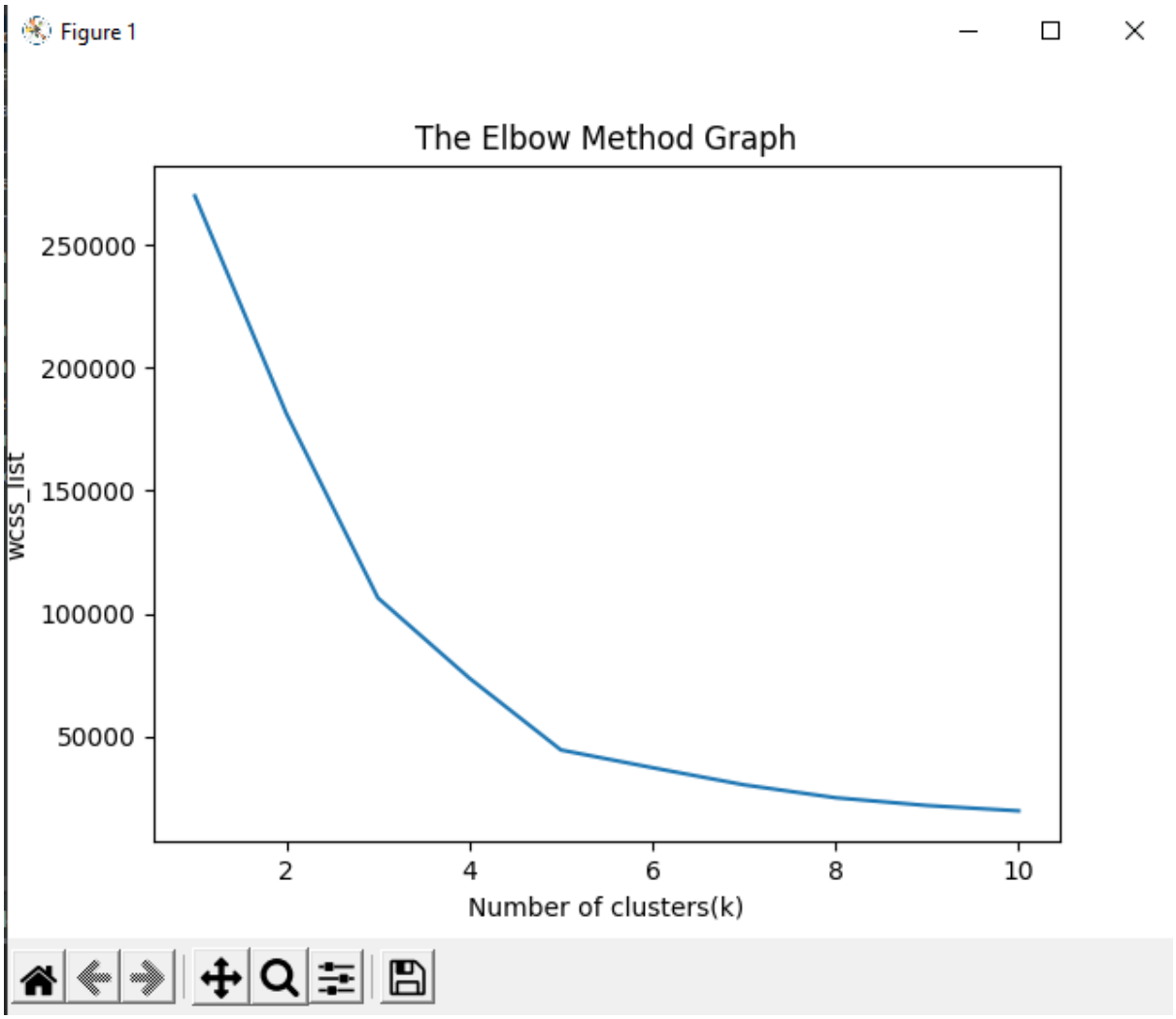


Figure 1
The Elbow Method Graph
250000
200000
150000
100000
50000
WCSS_list
2
4
6
8
10
Number of clusters(k)

```
[2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2
 3 2 3 2 3 2 0 2 3 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 4 1 4 0 4 1 4 1 4 0 4 1 4 1 4 1 4 1 4 0 4 1 4 1 4
 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1
 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4]
```