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

**PROGRAM**

# Importing the libraries

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd # Importing the dataset

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=0)

kmeans.fit(X)

wcss\_list.append(kmeans.inertia\_)

plt.plot(range(1, 11), wcss\_list)

plt.title('The Elbow Method Graph')

plt.xlabel('Number of clusters')

plt.ylabel('WCSS')

pltkmeans = KMeans(n\_clusters=5, init="k-means++", random\_state=42)

y\_predict = kmeans.fit\_predict(X)

print(y\_predict)

plt.scatter(X[y\_predict == 0, 0], X[y\_predict == 0, 1], s=60, c='red', label='Cluster1')

plt.scatter(X[y\_predict == 1, 0], X[y\_predict == 1, 1], s=60, c='blue', label='Cluster2')

plt.scatter(X[y\_predict == 2, 0], X[y\_predict == 2, 1], s=60, c='green', label='Cluster3')

plt.scatter(X[y\_predict == 3, 0], X[y\_predict == 3, 1], s=60, c='violet', label='Cluster4')

plt.scatter(X[y\_predict == 4, 0], X[y\_predict == 4, 1], s=60, c='yellow', label='Cluster5')

plt.scatter(kmeans.cluster\_centers\_[:, 0], kmeans.cluster\_centers\_[:, 1], s=100, c='black', label='Centroids')

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

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

plt.legend()

plt.show()

**OUTPUT**

[[ 15 39]

[ 15 81]

[ 16 6]

[ 16 77]

[ 17 40]

[ 17 76]

[ 18 6]

[ 18 94]

[126 28]

[126 74]

[137 18]

[137 83]]

[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 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 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 1 4]



