Exp No : 09
Date : 02/11/24

IMPLEMENTATION OF CLUSTERING TECHNIQUES - K MEANS

AIM:

To implement a k-Means clustering techniques using Python.

EXPLANATION:

* Import k means from sk learn_cluste
* Assign X & Y
* Call the function k means()
* Perform scatter operation and display output.

ALGORITHM:

* Initialize
  → Choose the no of clusters k
  → Randomly initialize k centroids
* Assign Data Points to clusters
  → for each dataset
  → Calculate the distance between the data point and each centroid.
  → Assign the data point to the cluster where centroid is the closest.

* Recalculate centroids

→ For each cluster, compute the new centroid by calculating the mean of all data points assigned to that cluster.

* Repeat

→ Repeat ~~step~~ 2 and 3 until the cluster assignment do not change. This is called convergence.

* stopping Criteria :

→ Algorithm stops when one of the following occurs.

→ The centroids do not change between iterations

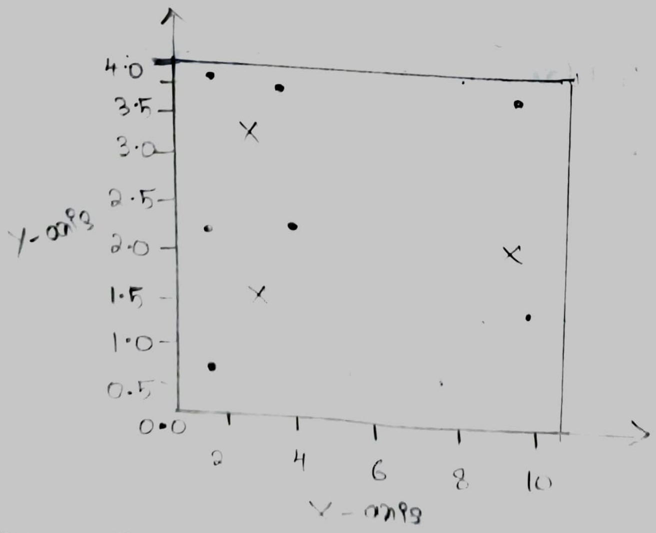
→ A Maximum no of iterations is reached.

CODE:

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import nump as np

X = np.array([1,2], [1,4], [1,8], [4,2],[4,4]
       [4,0] , [10,2], [10,4], [10,0]])

kmeans = kmeans (n.clusters=3, random
                               state=0)
kmeans.fit(X)
```

```
y_kmeans = kmeans.predict(x)

plt.scatter(X[:,0], X[:,1], c= y_kMeans,
                        s=50, cmap='viridis')
centroids = kmeans.cluster_centers
plt.scatter(centroids[:,0], centroids[:,1],
                 c='red', s=200, alpha=0.75,
                        marker='x')
plt.xlabel("x-axis")
plt.ylabel("Y-axis")
plt.title("k-means Clustering")
plt.show()
```

OUTPUT:



RESULT:

Thus, the K-Means clustering Program is executed successfully and output is verified.