**Aim**

To implement the K-Means clustering algorithm in Python and visualize the clustering process.

**Procedure**

1. **Import Libraries**: Load necessary Python libraries (sklearn, numpy, matplotlib).
2. **Generate or Load Data**: Use synthetic or real-world data for clustering.
3. **Preprocess Data**: Normalize or scale data if necessary.
4. **Apply K-Means Algorithm**: Initialize and fit the model to the data.
5. **Determine Optimal K**: Use the elbow method to find the best number of clusters.
6. **Visualize Results**: Plot the clustered data points.

**Code**

import numpy as np

import matplotlib.pyplot as plt

from sklearn.cluster import KMeans

from sklearn.datasets import make\_blobs

# Generate synthetic data

X, \_ = make\_blobs(n\_samples=300, centers=4, cluster\_std=0.60, random\_state=42)

# Apply K-Means clustering

kmeans = KMeans(n\_clusters=4, random\_state=42)

kmeans.fit(X)

y\_kmeans = kmeans.predict(X)

# Plot results

plt.scatter(X[:, 0], X[:, 1], c=y\_kmeans, cmap='viridis')

plt.scatter(kmeans.cluster\_centers\_[:, 0], kmeans.cluster\_centers\_[:, 1], s=300, c='red', marker='X', label='Centroids')

plt.title("K-Means Clustering")

plt.legend()

plt.show()

**Output**

Thus the code has been successfully compiled.