## **AIM**

To implement the K-Means clustering algorithm to group unlabeled data points into clusters based on feature similarity.

## **PROCEDURE**

1. **Import dataset**: Use a dataset with unlabeled or labeled data (e.g., Iris dataset without labels).
2. **Preprocess data**: Extract features for clustering.
3. **Apply K-Means algorithm**:  
   * Choose the number of clusters (k).
   * Initialize cluster centroids randomly.
   * Assign each data point to the nearest centroid.
   * Recalculate centroids as mean of assigned points.
   * Repeat assignment and centroid update until convergence.
4. **Visualize clusters** using 2D plot.
5. **Evaluate results** by comparing clusters to actual labels (optional).

## **CODE**

import matplotlib.pyplot as plt

from sklearn.datasets import load\_iris

from sklearn.cluster import KMeans

from sklearn.decomposition import PCA

# Load dataset

iris = load\_iris()

X = iris.data

# Apply PCA to reduce to 2 dimensions for visualization

pca = PCA(n\_components=2)

X\_reduced = pca.fit\_transform(X)

# Initialize and fit K-Means with 3 clusters

kmeans = KMeans(n\_clusters=3, random\_state=42)

kmeans.fit(X)

# Get cluster labels

labels = kmeans.labels\_

# Plot the clusters

plt.figure(figsize=(8,6))

scatter = plt.scatter(X\_reduced[:, 0], X\_reduced[:, 1], c=labels, cmap='viridis', alpha=0.7)

plt.title('K-Means Clustering on Iris Dataset (PCA-reduced)')

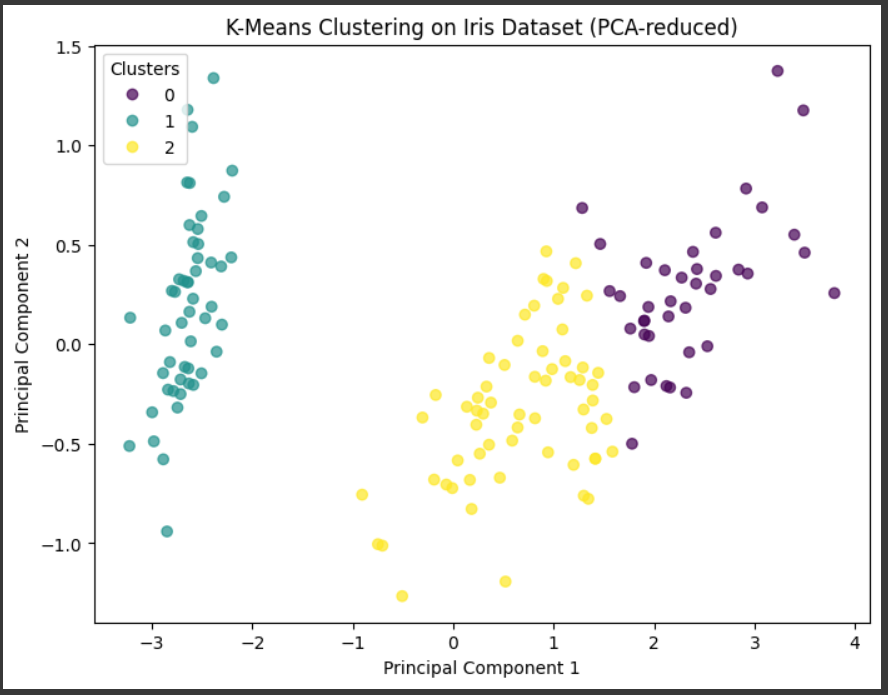
plt.xlabel('Principal Component 1')

plt.ylabel('Principal Component 2')

plt.legend(\*scatter.legend\_elements(), title="Clusters")

plt.show()

## **OUTPUT :**



## **EXPLANATION**

* PCA reduces 4D Iris data to 2D for visualization.
* K-Means partitions data into 3 clusters based on feature similarity.
* Each cluster corresponds roughly to one Iris species.
* Algorithm iteratively minimizes within-cluster variance.

## **CONCLUSION**

* K-Means is an unsupervised algorithm that groups similar data points.
* It requires choosing the number of clusters in advance.
* Visualizing clusters in reduced dimensions aids understanding.
* K-Means works well on well-separated data but may struggle on complex shapes.