Program:

In this example, Iris dataset, which contains measurements of the sepal length, sepal width, petal length, and petal width for three species of iris flowers (setosa, versicolor, and virginica).

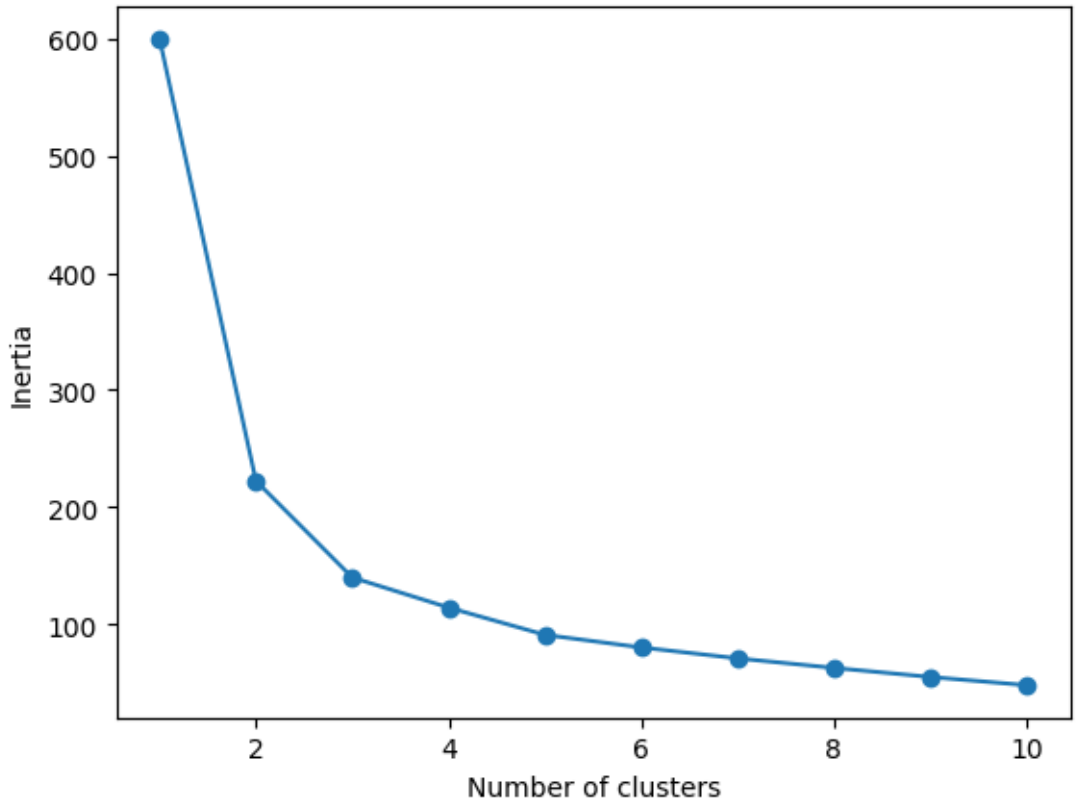
```
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler

# Load iris dataset and scale features
iris = load_iris()
X = iris.data
X_scaled = StandardScaler().fit_transform(X)
```

Implementation of K-means clustering with k=3

```
from sklearn.cluster import KMeans

# Initialize KMeans object and fit to data
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(X_scaled)

# Track convergence using inertia
inertia = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, random_state=42)
    kmeans.fit(X_scaled)
    inertia.append(kmeans.inertia_)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The defa
  warnings.warn(
```

```
import matplotlib.pyplot as plt

# Plot inertia vs. number of clusters
plt.plot(range(1, 11), inertia, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia')
plt.show()
```



Implementation of hierarchical clustering

```
from scipy.cluster.hierarchy import linkage, dendrogram

# Calculate linkage matrix using Ward method
Z = linkage(X_scaled, 'ward')

# Plot dendrogram
plt.figure(figsize=(10, 5))
dendrogram(Z)
plt.show()
```