**Name:** Jaydeep Solanki

**Roll:** 22BEC059

**Practical 8:** Write a program to cluster data in iris flower dataset using k-means algorithm.

* **Using Inbuilt module**

import numpy as np  
import pandas as pd  
from sklearn.model\_selection import train\_test\_split  
from tabulate import tabulate  
  
  
def makeTable(k, n):  
 data = pd.read\_csv("Iris.csv", header='infer').values  
  
 x = data[:, 1:-1]  
 y = data[:, -1]  
 x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.3)  
 centroids = np.zeros(shape=(k, x\_train.shape[1]))  
 per = np.random.permutation(x\_train.shape[0])  
 for i in range(k):  
 centroids[i, :] = x\_train[per[i], :]  
 for it in range(n):  
 dist = np.zeros(shape=(k, x\_train.shape[0]))  
 for i in range(k):  
 dist[i, :] = np.sqrt(np.sum((x\_train - centroids[i, :]) \*\* 2, axis=1))  
 membership = np.argmin(dist, axis=0)  
 for i in range(k):  
 centroids[i, :] = np.mean(x\_train[membership == i, :], axis=0)  
 dist = np.zeros(shape=(k, x\_test.shape[0]))  
 for i in range(k):  
 dist[i] = np.sqrt(np.sum((x\_test - centroids[i]) \*\* 2, axis=1))  
 membership = np.argmin(dist, axis=0)  
 return [centroids.tolist(), f"Y\_Test= {y\_test.astype(int).tolist()}\nMembership= {membership.tolist()}\n."]  
  
  
k = int(input("Enter the nearest neighbor number(k) : "))  
n = int(input("Enter the number of iterations :"))  
display = {}  
for i in range(1, k + 1):  
 ans = makeTable(i, n)  
 display[i] = ans  
tabulated\_data = tabulate(pd.DataFrame(display).T, tablefmt="pretty", headers=["Centroids", "Comparison"])  
print(tabulated\_data)

* Output: 

