**implement K-Nearest Neighbour(KNN) algorithm.**

import pandas as pd

data = pd.read\_csv("/content/Iris.csv")

print(data.head())

x = data.iloc[:,1:5].values

y = data.iloc[:,5].values

data['Species'].value\_counts()

## Split the data into training and testing

from sklearn.model\_selection import train\_test\_split

xtrain,xtest,ytrain,ytest = train\_test\_split(x,y,train\_size=0.80,random\_state=3)

import numpy as np

np.unique(ytrain,return\_counts=True)

## Build the model

from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n\_neighbors=3)

#### Train the model

model.fit(xtrain,ytrain)

ypred = model.predict(xtest)

print(ypred)

print(ytest)

from sklearn.metrics import confusion\_matrix

cm = confusion\_matrix(ytest,ypred)

print(cm);

### Accuracy Score

from sklearn.metrics import accuracy\_score

acc = accuracy\_score(ytest,ypred)

print(acc)

accuracy = []

for i in range(1,100):

  model\_1 = KNeighborsClassifier(n\_neighbors=i)

  model\_1.fit(xtrain,ytrain)

  ypred\_1 = model\_1.predict(xtest)

  acc\_1 = accuracy\_score(ytest,ypred\_1)

  accuracy.append(acc\_1)

print(accuracy)

model.predict([[2.3,4.5,2.3,5.6]])

def flower():

  sl = float(input("Enter sepel length:"))

  sw = float(input("Enter sepel width:"))

  pl = float(input("Enter petel length:"))

  pw = float(input("Enter petel width:"))

  fl = model.predict([[sl,sw,pl,pw]])

  print("The expected type of flowers "+str((fl[0])))

flower()