import numpy as np  
import pandas as pd  
from sklearn.model\_selection import train\_test\_split  
from sklearn.linear\_model import LogisticRegression  
from sklearn.metrics import accuracy\_score  
datas = pd.read\_csv ('datas.csv')  
datas.head()  
datas.shape  
datas.info()  
datas['target'].value\_counts()  
X = datas.drop (columns='target', axis=1)  
Y = datas ['target']  
  
print(X)  
print(Y)  
X\_train, X\_test, Y\_train, Y\_test = train\_test\_split (X, Y, test\_size=0.2, stratify=Y, random\_state=2)  
print(X.shape, X\_train.shape, X\_test.shape)  
model = LogisticRegression()  
model.fit (X\_train.values, Y\_train)  
y\_pred = model.predict(X\_test.values)  
print(y\_pred)  
print(accuracy\_score(Y\_test,y\_pred))  
input\_data = (62, 0, 0, 140, 268, 0, 0, 160, 0, 3.6, 0, 2, 2)  
# Change the input data to a numpy array  
numpy\_data= np.asarray (input\_data)  
# reshape the numpy array as we are predicting for only on instance  
input\_reshaped = numpy\_data.reshape (1,-1)  
prediction = model.predict (input\_reshaped)  
if(prediction[0]== 0):  
 print("The Person does not have a Heart Disease")  
else:  
 print("The Person has Heart Disease")  
 #Saving the trained model  
import pickle  
filename = 'trained\_model.sav'  
#dump=save your trained model  
pickle.dump (model,open (filename,'wb'))  
#loading the saved model  
loaded\_model = pickle.load (open ('trained\_model.sav','rb'))