**PROGRAM NO:4 DATE:06-10-2022**

**AIM: Predict the class label of an unseen observation using Naïve-Bayes.**

**Source\_code:**

weather=['sunny','sunny','overcast','rainy','rainy','rainy','overcast','sunny','sunny','rainy','sunny','overcast','overcast','rainy']

temp=['hot','hot','hot','mild','cool','cool','cool','mild','cool','mild','mild','mild','hot','mild']

play=['no','no','yes','yes','yes','no','yes','no','yes','yes','yes','yes','yes','no']

from sklearn import preprocessing

le=preprocessing.LabelEncoder()

weather\_encoded=le.fit\_transform(weather)

print(weather\_encoded)

temp\_encoded=le.fit\_transform(temp)

label=le.fit\_transform(play)

print("temp:",temp\_encoded)

print("play:",label)

features=zip(weather\_encoded,temp\_encoded)

features=list(features)

print("features",features)

features

from sklearn.naive\_bayes import GaussianNB

model=GaussianNB()

model.fit(features,label)

predicted=model.predict([[0,2]])

print("predicted value:",predicted)

**Output**

[2 2 0 1 1 1 0 2 2 1 2 0 0 1]

temp: [1 1 1 2 0 0 0 2 0 2 2 2 1 2]

play: [0 0 1 1 1 0 1 0 1 1 1 1 1 0]

features [(2, 1), (2, 1), (0, 1), (1, 2), (1, 0), (1, 0), (0, 0), (2, 2), (2, 0), (1, 2), (2, 2), (0, 2), (0, 1), (1, 2)]

predicted value: [1]