```python
#Program-7
"""
Write a program to implement the naïve Bayesian classifier for a
sample training data set stored
as a .CSV file.Compute the accuracy of the classifier, considering few
test data sets. (Iris Dataset)
"""

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import LabelEncoder

# Load the Iris dataset
data = pd.read_csv("Iris.csv")

# Select features and target
X = data.drop("Species", axis=1)
y = data['Species']

X
```

```
      Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
0      1            5.1           3.5            1.4           0.2
1      2            4.9           3.0            1.4           0.2
2      3            4.7           3.2            1.3           0.2
3      4            4.6           3.1            1.5           0.2
4      5            5.0           3.6            1.4           0.2
..   ...            ...           ...            ...           ...
145  146            6.7           3.0            5.2           2.3
146  147            6.3           2.5            5.0           1.9
147  148            6.5           3.0            5.2           2.0
148  149            6.2           3.4            5.4           2.3
149  150            5.9           3.0            5.1           1.8

[150 rows x 5 columns]
```

```python
y
```

```
0         Iris-setosa
1         Iris-setosa
2         Iris-setosa
3         Iris-setosa
4         Iris-setosa
             ...
145    Iris-virginica
146    Iris-virginica
147    Iris-virginica
148    Iris-virginica
```

```
149    Iris-virginica
Name: Species, Length: 150, dtype: object
```

```
# Encoding the Species column to get numerical class
le = LabelEncoder()
y = le.fit_transform(y)
```

```
y
```

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,
       0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2,
2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)
```

```
# Gaussian Naive Bayes classifier
gnb = GaussianNB()
```

```
# Train the classifier on the training data
gnb.fit(X_train, y_train)
```

```
GaussianNB()
```

```
# Make predictions on the testing data
y_pred = gnb.predict(X_test)
```

```
# Calculate the accuracy of the model
accuracy = accuracy_score(y_test, y_pred)
print(f"The Accuracy of Prediction on Iris Flower is: {accuracy}")
```

```
The Accuracy of Prediction on Iris Flower is: 1.0
```

```
# Create a DataFrame to display actual and predicted values
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
```

```
# Print the table
print(df)
```

```
    Actual  Predicted
0        1          1
```

1	0	0
2	2	2
3	1	1
4	1	1
5	0	0
6	1	1
7	2	2
8	1	1
9	1	1
10	2	2
11	0	0
12	0	0
13	0	0
14	0	0
15	1	1
16	2	2
17	1	1
18	1	1
19	2	2
20	0	0
21	2	2
22	0	0
23	2	2
24	2	2
25	2	2
26	2	2
27	2	2
28	0	0
29	0	0
30	0	0
31	0	0
32	1	1
33	0	0
34	0	0
35	2	2
36	1	1
37	0	0
38	0	0
39	0	0
40	2	2
41	1	1
42	1	1
43	0	0
44	0	0