

**PROGRAM:**

```
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn import datasets
from sklearn import metrics

# Load the Iris dataset
iris = datasets.load_iris()
X = iris.data # Features
y = iris.target # Target variable

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a Naive Bayes Classifier (Gaussian Naive Bayes in this case)
model = GaussianNB()

# Train the model on the training set
model.fit(X_train, y_train)

# Make predictions on the test set
y_pred = model.predict(X_test)

# Evaluate the performance of the classifier
accuracy = metrics.accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")

# Example: Predict the class of a new sample
new_sample = [[5.1, 3.5, 1.4, 0.2]] # Example input features for a new sample
predicted_class = model.predict(new_sample)
print(f"Predicted class for the new sample: {predicted_class}")
```

**OUTPUT:**

```
Accuracy: 1.0
Predicted class for the new sample: [0]
```