Program:

To implement the Naive Bayes Classifier on the Iris dataset, we first need to import the necessary libraries and load the dataset. Here's the code for it:

```
# Importing the necessary libraries
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, precision_score

# Loading the Iris dataset
iris = load_iris()
X = iris.data
y = iris.target

# Splitting 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)
```

Next, we'll create an instance of the Gaussian Naive Bayes classifier and fit it on the training data.

```
# Creating an instance of the Gaussian Naive Bayes classifier
gnb = GaussianNB()

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

```
GaussianNB()
```

Now, we can use the trained model to make predictions on the test data and evaluate its accuracy and precision.

```
# Making predictions on the test data
y_pred = gnb.predict(X_test)

# Calculating the accuracy and precision of the classifier
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')

print("Accuracy:", accuracy)
print("Precision:", precision)
```

```
Accuracy: 1.0
Precision: 1.0
```