```python
import numpy as np

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.naive_bayes import GaussianNB

from sklearn.metrics import confusion_matrix, accuracy_score, classification_report


# Step 1: Create a simple dataset

data = {

    'HoursStudied': [1, 2, 2, 3, 3, 4, 4, 5, 6, 6, 7, 8],

    'Attendance': [30, 40, 50, 45, 60, 70, 75, 80, 85, 90, 95, 100],

    'Pass': [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1]

}

df = pd.DataFrame(data)


# Step 2: Separate features and label

X = df[['HoursStudied', 'Attendance']]

y = df['Pass']


# Step 3: Train-test split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)


# Step 4: Train Naive Bayes Classifier

model = GaussianNB()

model.fit(X_train, y_train)


# Step 5: Predict on test data

y_pred = model.predict(X_test)


# Step 6: Evaluation

conf_matrix = confusion_matrix(y_test, y_pred)

acc = accuracy_score(y_test, y_pred)
```

```python
print("Confusion Matrix:\n", conf_matrix)

print("Accuracy:", acc)

print("\nClassification Report:\n", classification_report(y_test, y_pred))


# Step 7: Predict new test cases

new_data = np.array([[3, 60], [6, 95], [2, 40]])  # new students

new_predictions = model.predict(new_data)


print("\nNew Predictions:")

for i in range(len(new_data)):

    print("Hours Studied:", new_data[i][0],

        "| Attendance:", new_data[i][1],

        "=>", "Pass" if new_predictions[i] == 1 else "Fail")
```