```
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, precision_score, recall_score
```

Add text cell

```
column_names =['sepal.length','sepal.width','petal.length','petal.width','Species']
df = pd.read_csv('/content/Iris.csv', header=None, names=column_names)


# Split data into features (X) and target (y)
X = df.drop(columns=['Species'])  # Independent variables
y = df['Species']  # Target variable


# Split into training and testing sets (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Double-click (or enter) to edit

```
# 2. Train Naïve Bayes model
model = GaussianNB()
model.fit(X_train, y_train)
```

```
▸ GaussianNB
```

```
# Predict on test set
y_pred = model.predict(X_test)


# 3. Compute Confusion Matrix
conf_matrix = confusion_matrix(y_test, y_pred)
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='macro')
recall = recall_score(y_test, y_pred, average='macro')
error_rate = 1 - accuracy

# Print evaluation results
print(f"Confusion Matrix:\n{conf_matrix}")
print(f"Accuracy: {accuracy:.4f}")
print(f"Error Rate: {error_rate:.4f}")
print(f"Precision: {precision:.4f}")
print(f"Recall: {recall:.4f}")
```

```
Confusion Matrix:
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
Accuracy: 1.0000
Error Rate: 0.0000
Precision: 1.0000
Recall: 1.0000
```