Write a programme for decision tree classifier

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
import matplotlib.pyplot as plt
import numpy as np

# Generating random data for demonstration
np.random.seed(0)
X = 2 * np.random.rand(100, 1)
y = np.where(X > 1, 1, 0)  # Generating binary classification labels
based on X values

# Creating a Decision Tree Classifier
model = DecisionTreeClassifier(random_state=42)
model.fit(X, y)
```

```
DecisionTreeClassifier(random_state=42)
```

```
# Generating new data for prediction
X_new = np.linspace(0, 2, 100).reshape(-1, 1)  # Generating 100 evenly
spaced values between 0 and 2

# Making predictions using the model
y_pred = model.predict(X_new)

# Evaluating the model
accuracy = accuracy_score(y, model.predict(X))
cm = confusion_matrix(y, model.predict(X))
class_report = classification_report(y, model.predict(X))

print("Accuracy:", accuracy)
print("Confusion Matrix:")
print(cm)
print("Classification Report:")
print(class_report)
```

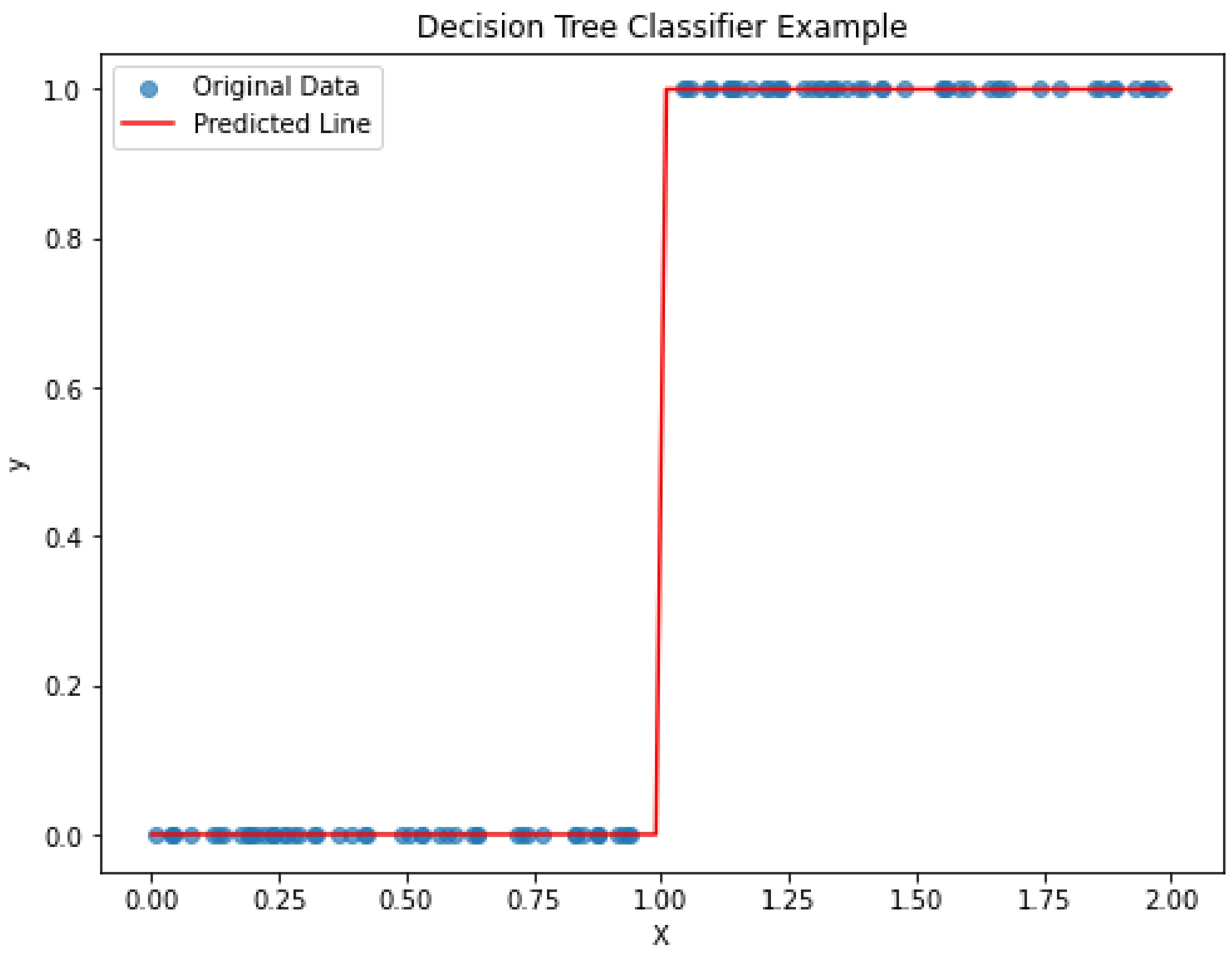
```
Accuracy: 1.0
Confusion Matrix:
[[51  0]
 [ 0 49]]
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        51
           1       1.00      1.00      1.00        49

    accuracy                           1.00       100
```

```
   macro avg       1.00      1.00      1.00       100
weighted avg       1.00      1.00      1.00       100


# Visualizing decision boundary
plt.figure(figsize=(8, 6))
plt.scatter(X, y, alpha=0.7, label='Original Data')
plt.plot(X_new, y_pred, color='red', label='Predicted Line')
plt.xlabel('X')
plt.ylabel('y')
plt.title('Decision Tree Classifier Example')
plt.legend()
plt.show()
```


Decision Tree Classifier Example
Original Data
Predicted Line
y
X