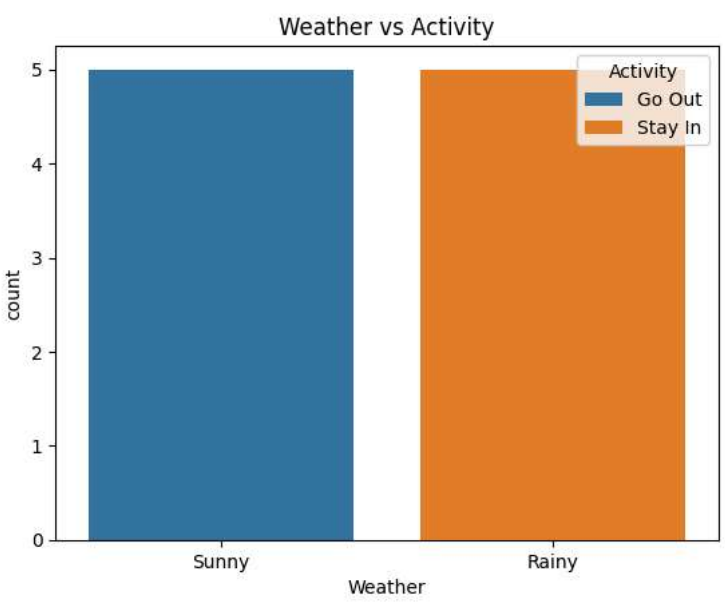
```
# Step 1: Import Libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import classification_report, confusion_matrix
```

```
# Step 2: Create Dataset
data = {
    'Weather': ['Sunny', 'Rainy', 'Sunny', 'Rainy', 'Sunny', 'Rainy', 'Sunny', 'Rainy', 'Sunny', 'Rainy'],
    'Activity': ['Go Out', 'Stay In', 'Go Out', 'Stay In', 'Go Out', 'Stay In', 'Go Out', 'Stay In', 'Go Out', 'Stay In']
}
df = pd.DataFrame(data)
```

```
# Step 3: Encode Categorical Variables
df['Weather_Code'] = df['Weather'].map({'Sunny': 0, 'Rainy': 1})
df['Activity_Label'] = df['Activity'].map({'Stay In': 0, 'Go Out': 1})
```

```
# Step 4: Visualize
sns.countplot(x='Weather', hue='Activity', data=df)
plt.title("Weather vs Activity")
plt.show()
```




```
# Step 5: Model Training
X = df[['Weather_Code']]
y = df['Activity_Label']
```

```
clf = DecisionTreeClassifier(max_depth=1, random_state=42)
clf.fit(X, y)
```

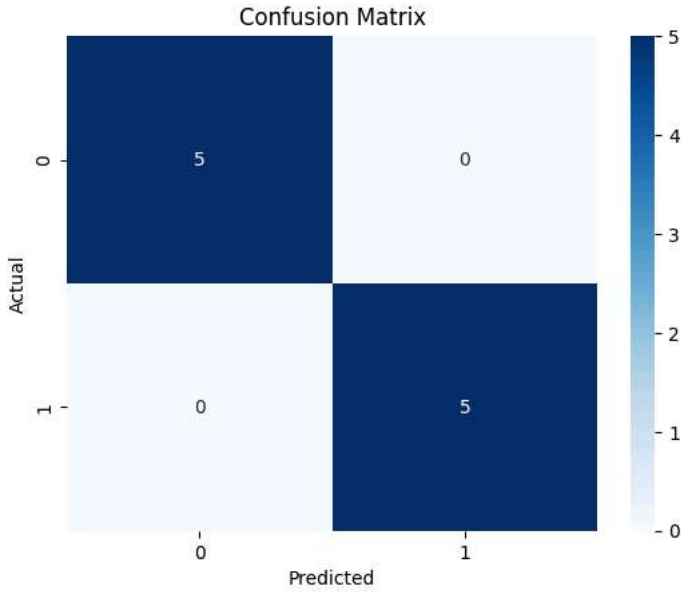
```
DecisionTreeClassifier(max_depth=1, random_state=42)
```

```
# Step 6: Evaluation
y_pred = clf.predict(X)
print("Classification Report:\n", classification_report(y, y_pred))
```

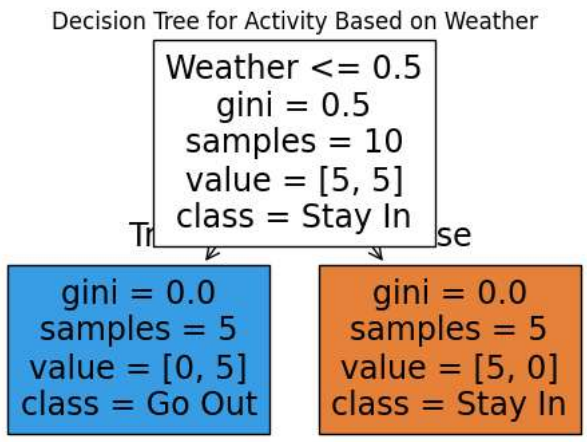
```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         5
           1       1.00      1.00      1.00         5
```

```
          accuracy                           1.00        10
         macro avg       1.00      1.00      1.00        10
      weighted avg       1.00      1.00      1.00        10
```

```
sns.heatmap(confusion_matrix(y, y_pred), annot=True, cmap='Blues')
plt.title("Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```




```
# Step 7: Visualize Tree
plt.figure(figsize=(6,4))
plot_tree(clf, feature_names=["Weather"], class_names=["Stay In", "Go Out"], filled=True)
plt.title("Decision Tree for Activity Based on Weather")
plt.show()
```



Start coding or generate with AI.