

```
In [1]: import pandas as pd
        from sklearn.model_selection import train_test_split
        from sklearn.preprocessing import LabelEncoder
        from sklearn.ensemble import RandomForestClassifier
        from sklearn.metrics import accuracy_score, classification_report, confusion
```

```
In [2]: data = pd.read_csv("ML Datasets/car_evaluation.csv")
```

```
In [3]: data = data.apply(LabelEncoder().fit_transform)
```

```
In [4]: X = data.iloc[:, :-1]
        y = data.iloc[:, -1]
```

```
In [5]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, ran
```

```
In [6]: rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)
```

```
In [7]: rf_classifier.fit(X_train, y_train)
```

```
Out[7]: ▼      RandomForestClassifier ⓘ ?
        RandomForestClassifier(random_state=42)
```

```
In [8]: y_pred = rf_classifier.predict(X_test)
```

```
In [9]: accuracy = accuracy_score(y_test, y_pred)
        confusion = confusion_matrix(y_test, y_pred)
        classification_rep = classification_report(y_test, y_pred)
```

```
In [10]: print(f"Accuracy: {accuracy}")
          print("\nConfusion Matrix:\n", confusion)
          print("\nClassification Report:\n", classification_rep)
```

Accuracy: 0.9624277456647399

Confusion Matrix:

```
[[ 72  1  3  1]
 [  2 10  0  3]
 [  1  0 236  0]
 [  2  0  0 15]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.94	0.94	0.94	77
1	0.91	0.67	0.77	15
2	0.99	1.00	0.99	237
3	0.79	0.88	0.83	17
accuracy			0.96	346
macro avg	0.91	0.87	0.88	346
weighted avg	0.96	0.96	0.96	346