

```
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Step 1: Load and preprocess the data
data = pd.read_csv('insurance_data.csv') # Replace with your insurance claim dataset file

# Perform any necessary data preprocessing, such as handling missing values,
# encoding categorical variables, scaling numerical features, etc.

# Separate the features (X) and the target variable (y)
X = data.drop('Fraudulent', axis=1)
y = data['Fraudulent']

# Step 2: Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Step 3: Train a Random Forest Classifier model
model = RandomForestClassifier()
model.fit(X_train, y_train)

# Step 4: Make predictions on the test set
y_pred = model.predict(X_test)

# Step 5: Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
classification_report = classification_report(y_test, y_pred)

print('Accuracy:', accuracy)
print('Classification Report:\n', classification_report)
```

```
# Step 6: Predict fraud for a new insurance claim

# Prepare the input for a new claim as a pandas DataFrame
new_claim = pd.DataFrame({
    'Feature1': [value1],
    'Feature2': [value2],
    # Add more features based on your dataset
})

# Make a prediction using the trained model
new_claim_fraud = model.predict(new_claim)

print('New Claim Fraud:', new_claim_fraud)
```