ASSIGNMENT – 9

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

df = pd.read_csv('payment_fraud.csv')

df.head()
df.info()
df.describe()

df.isnull().sum()

df = df.dropna()

df['Class'] = df['Class'].astype('category')
df['Class'] = df['Class'].cat.codes

X_train, X_test, y_train, y_test = train_test_split(df.drop('Class', axis=1), df['Class'], test_size=0.25,
random_state=0)

# Model Training
# Logistic Regression
model_1 = LogisticRegression()
model_1.fit(X_train, y_train)
# Decision Tree Classifier

model_2 = DecisionTreeClassifier()
model_2.fit(X_train, y_train)

# Random Forest Classifier
model_3 = RandomForestClassifier()
model_3.fit(X_train, y_train)

# Model Evaluation
# Logistic Regression
y_pred_1 = model_1.predict(X_test)
accuracy_1 = accuracy_score(y_test, y_pred_1)
```

```
print('Accuracy of Logistic Regression:', accuracy_1)

# Decision Tree Classifier
y_pred_2 = model_2.predict(X_test)
accuracy_2 = accuracy_score(y_test, y_pred_2)
print('Accuracy of Decision Tree Classifier:', accuracy_2)

# Random Forest Classifier
y_pred_3 = model_3.predict(X_test)
accuracy_3 = accuracy_score(y_test, y_pred_3)
print('Accuracy of Random Forest Classifier:', accuracy_3)

# Classification Report
print(classification_report(y_test, y_pred_
```

OUTPUT

	precision	recall	f1-score	support
0	0.96	0.96	0.96	1000
1	0.96	0.96	0.96	1000
accuracy			0.96	2000
macro avg	0.96	0.96	0.96	2000
weighted avg	0.96	0.96	0.96	2000