```python
# Online Payment Fraud Detection using Python

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 classification_report, confusion_matrix

# Sample DataFrame construction based on the screenshot
data = {
    'step': [1, 1, 1, 1, 1],
    'type': ['PAYMENT', 'PAYMENT', 'TRANSFER', 'CASH_OUT', 'PAYMENT'],
    'amount': [9839.64, 1864.28, 181.00, 181.00, 11668.14],
        'nameOrig': ['C1231006815', 'C1666544295', 'C1305486145', 'C840083671',
'C2048537720'],
    'oldbalanceOrg': [170136.0, 21249.0, 181.0, 181.0, 41554.0],
    'newbalanceOrig': [160296.36, 19384.72, 0.0, 0.0, 29885.86],
        'nameDest': ['M1979787155', 'M2044282225', 'C553264065', 'C38997010',
'M1230701703'],
    'oldbalanceDest': [0.0, 0.0, 0.0, 21182.0, 0.0],
    'newbalanceDest': [0.0, 0.0, 0.0, 0.0, 0.0],
    'isFraud': [0, 0, 0, 1, 0],
    'isFlaggedFraud': [0, 0, 0, 0, 0]
}

df = pd.DataFrame(data)

# Encode categorical features
le_type = LabelEncoder()
df['type'] = le_type.fit_transform(df['type'])

# Drop non-informative columns
df.drop(['nameOrig', 'nameDest'], axis=1, inplace=True)

# Split dataset
X = df.drop(['isFraud'], axis=1)
y = df['isFraud']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

# Train model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict and evaluate
y_pred = model.predict(X_test)

print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```