# Import necessary libraries import pandas as pd

from sklearn.model\_selection import train\_test\_split from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import confusion\_matrix, classification\_report, accuracy\_score import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset (replace 'credit\_card\_data.csv' with your actual dataset) data = pd.read\_csv('credit\_card\_data.csv')

# Explore the dataset print(data.head()) print(data.info())

# Data preprocessing

# Add any necessary steps, such as handling missing values, scaling, etc.

# Split the data into features (X) and target variable (y) X = data.drop('Class', axis=1)

y = data['Class']

# 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)

# Train a Random Forest classifier

clf = RandomForestClassifier(n\_estimators=100, random\_state=42) clf.fit(X\_train, y\_train)

# Make predictions on the test set y\_pred = clf.predict(X\_test)

# Evaluate the model

print("Accuracy:", accuracy\_score(y\_test, y\_pred)) print("\nConfusion Matrix:\n", confusion\_matrix(y\_test, y\_pred))

print("\nClassification Report:\n", classification\_report(y\_test, y\_pred))

# Visualize the confusion matrix plt.figure(figsize=(8,6))

sns.heatmap(confusion\_matrix(y\_test, y\_pred), annot=True, cmap='Blues', fmt='g') plt.title('Confusion Matrix')

plt.xlabel('Predicted) plt.ylabel('Actual') plt.show()