TASK DS_03

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

# Create a sample dataset
data = {
    'age': [25, 30, 35, 40, 45, 50, 55, 60],
    'income': [50000, 60000, 70000, 80000, 90000, 100000, 110000, 120000],
    'credit_score': [600, 650, 700, 750, 800, 850, 900, 950],
    'family_size': [2, 3, 4, 5, 6, 7, 8, 9],
    'has_car': [0, 1, 1, 0, 1, 1, 0, 1],
    'has_house': [0, 1, 1, 1, 1, 1, 1, 1],
    'y': [0, 0, 1, 1, 1, 0, 0, 1]
}

# Create a Pandas DataFrame
df = pd.DataFrame(data)

# Split the dataset into features (X) and target (y)
X = df.drop('y', axis=1)
y = df['y']
```

```
# Split the dataset 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)


# Create a Decision Tree Classifier

clf = DecisionTreeClassifier(random_state=42)


# Train the classifier

clf.fit(X_train, y_train)


# Make predictions on the testing set

y_pred = clf.predict(X_test)


# Evaluate the classifier

accuracy = accuracy_score(y_test, y_pred)


# Display the accuracy in a bar chart

plt.bar(['Training', 'Testing'], [accuracy, accuracy])

plt.xlabel('Dataset')

plt.ylabel('Accuracy')

plt.title('Classifier Accuracy')

plt.show()


# Use the classifier to make predictions on new data

new_customer = pd.DataFrame({

    'age': [35],
    'income': [75000],
    'credit_score': [800],
    'family_size': [4],
    'has_car': [1],
    'has_house': [1]
})

prediction = clf.predict(new_customer)

# Calculate the prediction rate
prediction_rate = (prediction[0] == 1) * 100

# Display the prediction rate in a bar chart
plt.bar(['Prediction Rate'], [prediction_rate])
plt.xlabel('Metric')
plt.ylabel('Value')
plt.title('Prediction Rate')
plt.show()
```