```
#Name: SHIVANI GADKARI
#Roll no: 13342

import pandas as pd
import numpy as np
import matplotlib as plt

df=pd.read_csv("social_network_ads.csv")

df
```

```
      User ID  Gender  Age  EstimatedSalary  Purchased
0    15624510    Male   19            19000          0
1    15810944    Male   35            20000          0
2    15668575  Female   26            43000          0
3    15603246  Female   27            57000          0
4    15804002    Male   19            76000          0
..        ...     ...  ...              ...        ...
395  15691863  Female   46            41000          1
396  15706071    Male   51            23000          1
397  15654296  Female   50            20000          1
398  15755018    Male   36            33000          0
399  15594041  Female   49            36000          1

[400 rows x 5 columns]
```

```
df.columns
```

```
Index(['User ID', 'Gender', 'Age', 'EstimatedSalary', 'Purchased'],
dtype='object')
```

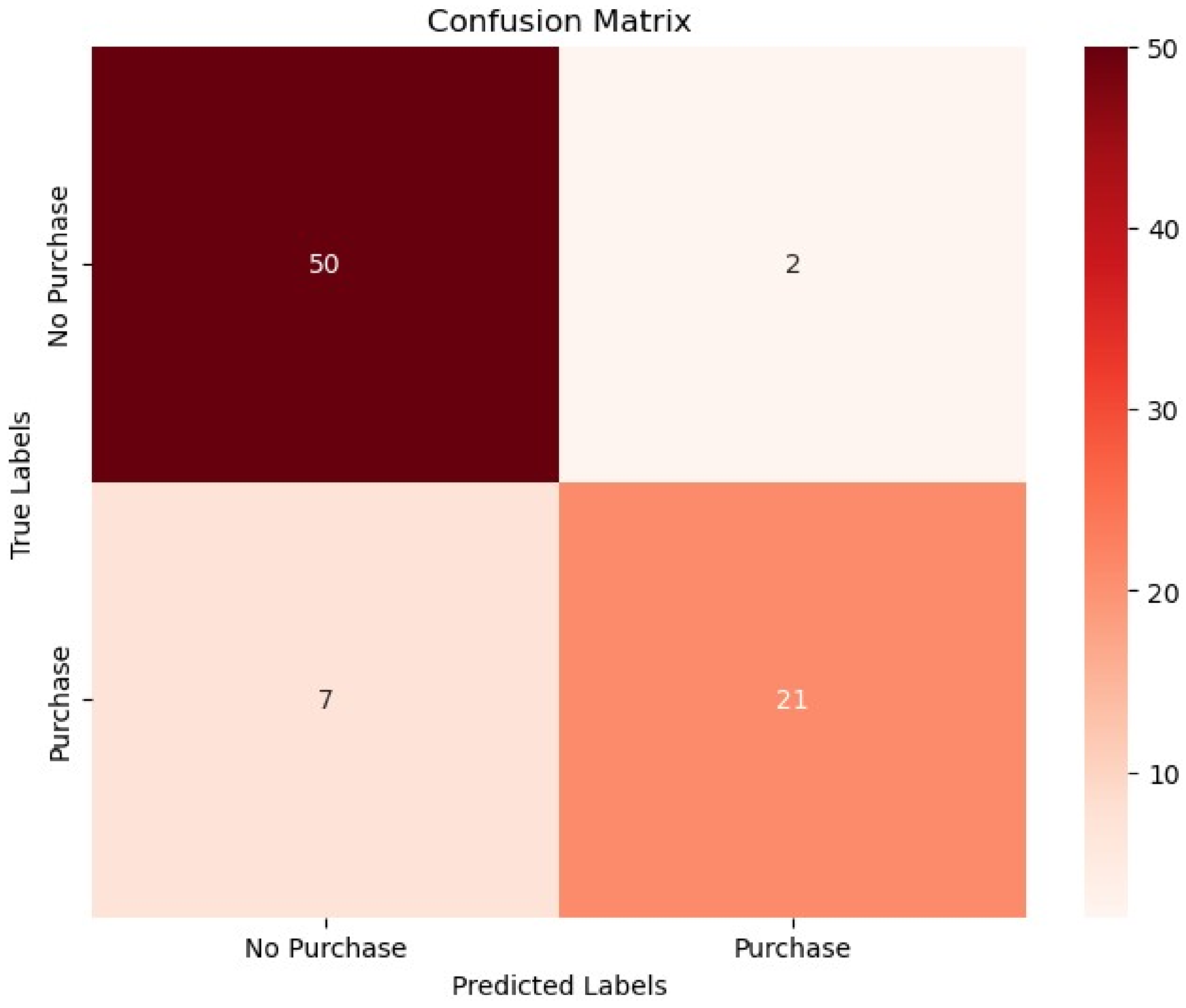
```
df.isnull
```

```
<bound method DataFrame.isnull of       User ID  Gender  Age
EstimatedSalary  Purchased
0    15624510    Male   19            19000          0
1    15810944    Male   35            20000          0
2    15668575  Female   26            43000          0
3    15603246  Female   27            57000          0
4    15804002    Male   19            76000          0
..        ...     ...  ...              ...        ...
395  15691863  Female   46            41000          1
396  15706071    Male   51            23000          1
397  15654296  Female   50            20000          1
398  15755018    Male   36            33000          0
399  15594041  Female   49            36000          1

[400 rows x 5 columns]>
```

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

```
df['Gender'] = le.fit_transform(df['Gender'])
newdf=df

df

      User ID  Gender  Age  EstimatedSalary  Purchased
0    15624510       1   19            19000          0
1    15810944       1   35            20000          0
2    15668575       0   26            43000          0
3    15603246       0   27            57000          0
4    15804002       1   19            76000          0
..        ...     ...  ...              ...        ...
395  15691863       0   46            41000          1
396  15706071       1   51            23000          1
397  15654296       0   50            20000          1
398  15755018       1   36            33000          0
399  15594041       0   49            36000          1

[400 rows x 5 columns]

 x = df.drop(['Purchased'], axis=1)
 y = df['Purchased']

from sklearn.model_selection import train_test_split

 from sklearn.model_selection import train_test_split
 X_train, X_test, Y_train, Y_test = train_test_split(x, y,
test_size=0.4,random_state=10)

from sklearn.linear_model import LogisticRegression

 print(X_train.head())

      User ID  Gender  Age  EstimatedSalary
60   15814004       1   27            20000
21   15736760       0   47            49000
299  15747043       1   46           117000
106  15706185       0   26            35000
139  15741094       1   19            25000

 X_train= pd.get_dummies(X_train,drop_first=True)

from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
X =df.drop('Purchased',axis=1)
y =df['Purchased']
X_train,X_test, Y_train,Y_test=
train_test_split(X,y,test_size=0.2,random_state=42)
X_train= pd.get_dummies(X_train,drop_first=True)
X_test =pd.get_dummies(X_test,drop_first=True)
logreg =LogisticRegression()
logreg.fit(X_train,Y_train)
```

```
LogisticRegression()

Y_pred =logreg.predict(X_test)
print("Predictions:", Y_pred)

Predictions: [0 1 0 1 0 0 1 0 0 0 0 1 0 0 0 0 1 1 0 1 0 0 0 1 0 1 1 0
1 0 0 0 1 0 1 0 0
 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 1 0 0 1 0
0 0
 0 0 1 1 0 0]

 import sklearn
 from sklearn.linear_model import LogisticRegression
 logreg =LogisticRegression()
 model=logreg.fit(X_train,Y_train)

Ytrain_pred= logreg.predict(X_train)
Ytest_pred= logreg.predict(X_test)

 df=pd.DataFrame(Ytrain_pred,Y_train)
 df=pd.DataFrame(Ytest_pred,Y_test)

 from sklearn.metrics import
precision_score,confusion_matrix,accuracy_score,recall_score
 cm =confusion_matrix(Y_test,Y_pred)

 cm =confusion_matrix(Y_test,Y_pred)

 cm =confusion_matrix(Y_test,Y_pred)
 print("ConfusionMatrix:\n",cm)

ConfusionMatrix:
 [[50  2]
 [ 7 21]]

 print("Accuracy:", accuracy_score(Y_test, Y_pred))
 print("Precision:", precision_score(Y_test, Y_pred,
average='weighted'))
 print("Recall:", recall_score(Y_test, Y_pred, average='weighted'))

Accuracy: 0.8875
Precision: 0.8897406559877956
Recall: 0.8875

 import matplotlib.pyplot as plt
 import seaborn as sns
 from sklearn.metrics import confusion_matrix

 # Generate confusion matrix
 cm = confusion_matrix(Y_test, Y_pred)
 # Plot confusion matrix with correct labels
 labels = ['No Purchase', 'Purchase'] # Your class labels
 plt.figure(figsize=(8, 6))
 sns.heatmap(cm, annot=True, fmt='d', cmap='Reds',
xticklabels=labels,yticklabels=labels)
 plt.xlabel('Predicted Labels')
 plt.ylabel('True Labels')
 plt.title('Confusion Matrix')
 plt.show()
```


Confusion Matrix
True Labels
No Purchase
Purchase
Predicted Labels
No Purchase
Purchase