Aim:- Write a Program to implement Confusion matrix and Roc-Curve on Dataset.

Objective:- Applying Confusion matrix and Roc couve to check accuracy and performance of Dataset.

```
# Import necessary libraries
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, roc_curve, auc
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression

import numpy as np

np.random.seed(0)
X = np.random.rand(100, 2)
y = np.random.randint(2, size=100)

# Split 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 logistic regression model
model = LogisticRegression()
model.fit(X_train, y_train)
```

```
LogisticRegression()
```

```
# Predict probabilities
y_pred_prob = model.predict_proba(X_test)[:, 1]

# Predict classes
y_pred = (y_pred_prob > 0.5).astype(int)

from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

cm_matrix = confusion_matrix(y_test, y_pred)
print(cm_matrix)
```

```
[[ 6  0]
 [12  2]]
```

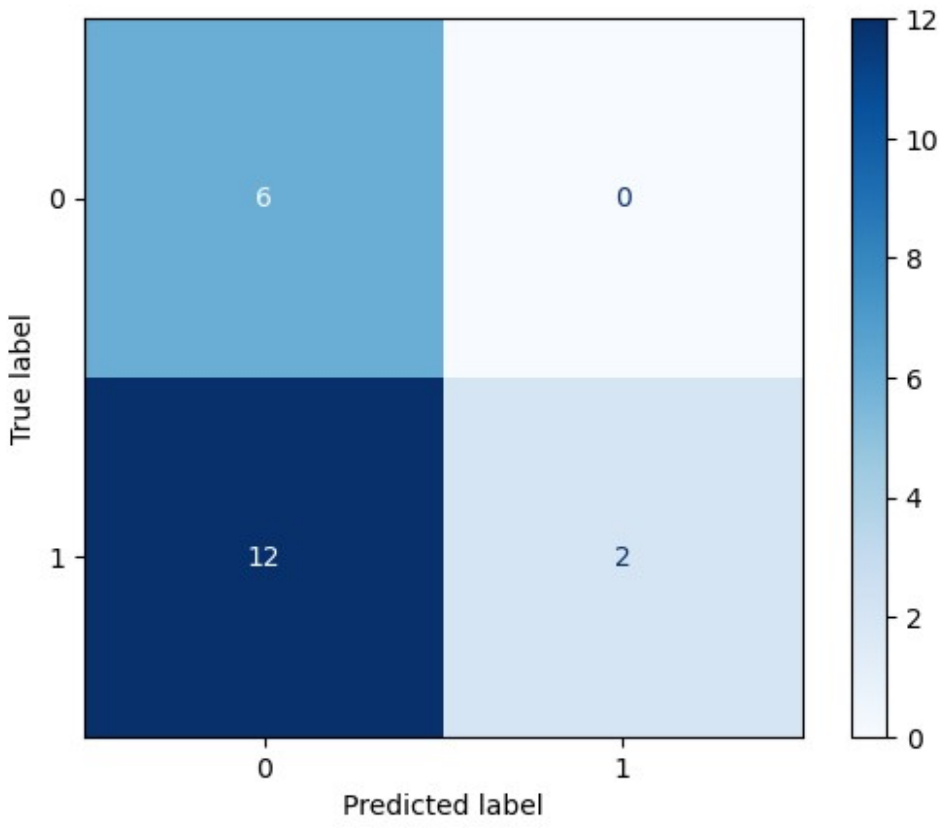
```
from sklearn.metrics import ConfusionMatrixDisplay

# Create a ConfusionMatrixDisplay object
cm = ConfusionMatrixDisplay(cm_matrix)

print("Confusion matrix")
cm.plot(cmap='Blues')
```

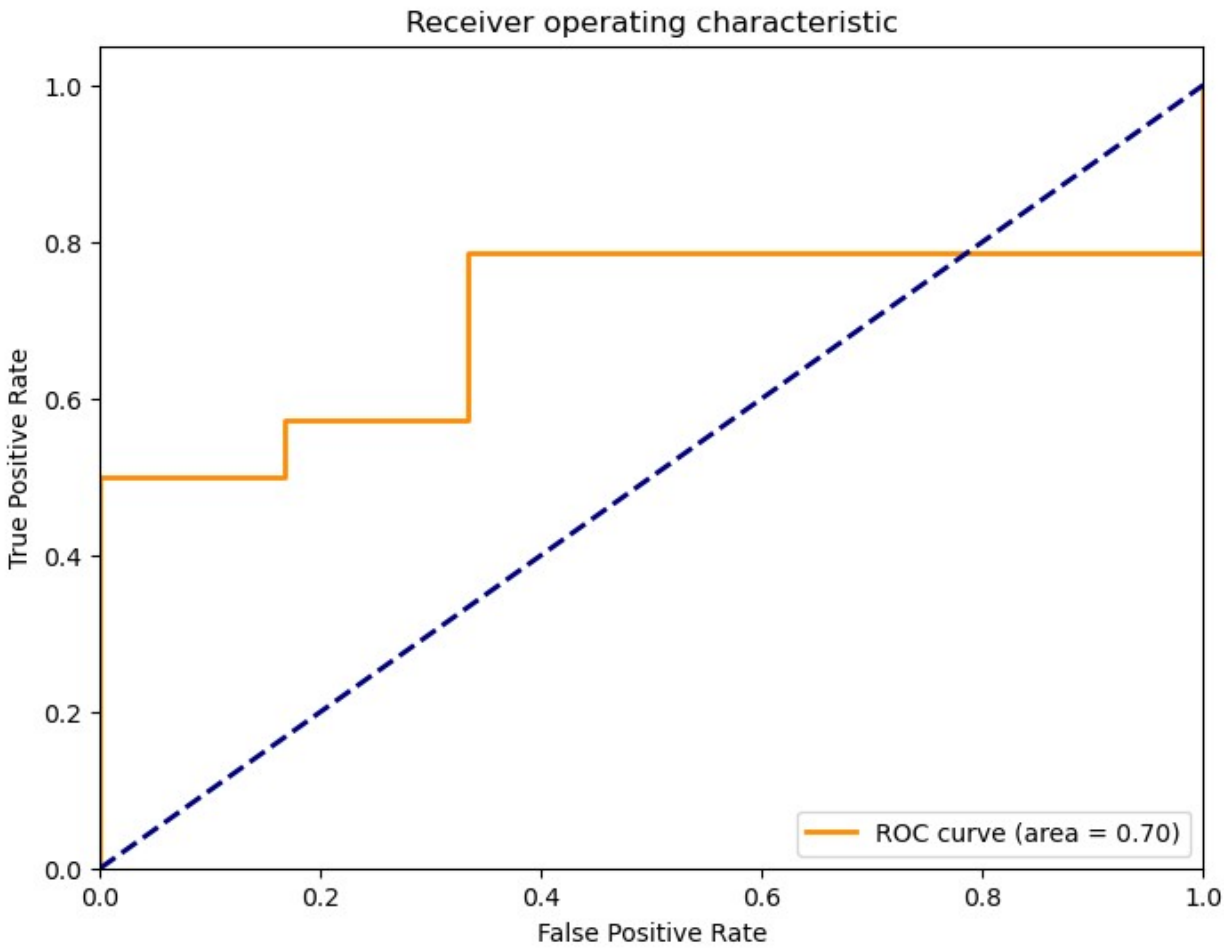
```
Confusion matrix
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at
0x2008d8c4b00>
```




```
# ROC Curve
fpr, tpr, thresholds = roc_curve(y_test, y_pred_prob)
roc_auc = auc(fpr, tpr)

plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area =
%0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic')
plt.legend(loc="lower right")
plt.show()
```




```
print("The End")
```

```
The End
```