In [1]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
```

In [2]:
```
df=pd.read_csv("C:/sameer/Heart.csv")
```

In [3]:
```
print(df.ndim)
```

```
2
```

In [4]:
```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 15 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  303 non-null    int64  
 1   Age         303 non-null    int64  
 2   Sex         303 non-null    int64  
 3   ChestPain   303 non-null    object 
 4   RestBP      303 non-null    int64  
 5   Chol        303 non-null    int64  
 6   Fbs         303 non-null    int64  
 7   RestECG     303 non-null    int64  
 8   MaxHR       303 non-null    int64  
 9   ExAng       303 non-null    int64  
 10  Oldpeak     303 non-null    float64
 11  Slope       303 non-null    int64  
 12  Ca          299 non-null    float64
 13  Thal        301 non-null    object 
 14  AHD         303 non-null    object 
dtypes: float64(2), int64(10), object(3)
memory usage: 35.6+ KB
None
```

In [5]:
```
print("The number of instances(row) ",len(df))
print("The number of columns(features) %d"%len(df.columns))
```

```
The number of instances(row)  303
The number of columns(features) 15
```

In [6]:
```
print(df.shape)
print(df.shape[0])
print(df.shape[1])
```

```
(303, 15)
303
15
```

In [7]:
```
print(df.size)
```

```
4545
```

In [8]:
```
print("number of total element in given dataset",df.shape[0]*df.shape[1])
```

```
number of total element in given dataset 4545
```

In [9]:
```
age_series=df['Age']
```

In [10]:
```
print(age_series)
```

```
0      63
1      67
2      67
3      37
4      41
       ..
298    45
299    68
300    57
301    57
302    38
Name: Age, Length: 303, dtype: int64
```

In [11]:
```
print(type(age_series))
```

```
<class 'pandas.core.series.Series'>
```

In [12]:
```
age_series1=df[['Age']]
print(age_series1)
```

```
     Age
0     63
1     67
2     67
3     37
4     41
..   ...
298   45
299   68
300   57
301   57
302   38

[303 rows x 1 columns]
```

In [13]:
```
print(type(age_series1))
```

```
<class 'pandas.core.frame.DataFrame'>
```

In [14]:
```
print(age_series1.shape)
```

```
(303, 1)
```

In [15]:
```
print(list(df.columns))
```

```
['Unnamed: 0', 'Age', 'Sex', 'ChestPain', 'RestBP', 'Chol', 'Fbs', 'RestECG', 'MaxHR', 'ExAng', 'Oldpeak', 'Slope', 'Ca', 'Thal', 'AHD']
```

In [16]:
```
print(list(df.columns.values.tolist()))
```

```
['Unnamed: 0', 'Age', 'Sex', 'ChestPain', 'RestBP', 'Chol', 'Fbs', 'RestECG', 'MaxHR', 'ExAng', 'Oldpeak', 'Slope', 'Ca', 'Thal', 'AHD']
```

In [17]:
```
print(df.dtypes)
```

```
Unnamed: 0      int64
Age             int64
Sex             int64
ChestPain      object
RestBP          int64
Chol            int64
Fbs             int64
RestECG         int64
MaxHR           int64
ExAng           int64
Oldpeak       float64
Slope           int64
Ca            float64
Thal           object
AHD            object
dtype: object
```

In [19]:
```
missing_values = ["n/a", "na", "--"]
df=pd.read_csv("C:/sameer/Heart.csv",na_values=missing_values)
```

In [20]:
```
na_values=missing_values
```

In [21]:
```
print(df.describe())
print(df.shape)
print(df.dtypes.value_counts())
```

```
       Unnamed: 0         Age         Sex      RestBP        Chol         Fbs  \
count  303.000000  303.000000  303.000000  303.000000  303.000000  303.000000
mean   152.000000   54.438944    0.679868  131.689769  246.693069    0.148515
std     87.612784    9.038662    0.467299   17.599748   51.776918    0.356198
min      1.000000   29.000000    0.000000   94.000000  126.000000    0.000000
25%     76.500000   48.000000    0.000000  120.000000  211.000000    0.000000
50%    152.000000   56.000000    1.000000  130.000000  241.000000    0.000000
75%    227.500000   61.000000    1.000000  140.000000  275.000000    0.000000
max    303.000000   77.000000    1.000000  200.000000  564.000000    1.000000

          RestECG       MaxHR       ExAng     Oldpeak       Slope          Ca
count  303.000000  303.000000  303.000000  303.000000  303.000000  299.000000
mean     0.990099  149.607261    0.326733    1.039604    1.600660    0.672241
std      0.994971   22.875003    0.469794    1.161075    0.616226    0.937438
min      0.000000   71.000000    0.000000    0.000000    1.000000    0.000000
25%      0.000000  133.500000    0.000000    0.000000    1.000000    0.000000
50%      1.000000  153.000000    0.000000    0.800000    2.000000    0.000000
75%      2.000000  166.000000    1.000000    1.600000    2.000000    1.000000
max      2.000000  202.000000    1.000000    6.200000    3.000000    3.000000
(303, 15)
int64      10
object      3
float64     2
dtype: int64
```

In [22]:
```
numerical_var=df.columns[df.dtypes!=object]
```

In [23]:
```
numerical_var=df.columns[df.dtypes!=object]
```

In [24]:
```
print(numerical_var)
```

```
Index(['Unnamed: 0', 'Age', 'Sex', 'RestBP', 'Chol', 'Fbs', 'RestECG', 'MaxHR',
       'ExAng', 'Oldpeak', 'Slope', 'Ca'],
      dtype='object')
```

In [26]:
```
print(df["Ca"].isnull())
```

```
0      False
1      False
2      False
3      False
4      False
       ...
298    False
299    False
300    False
301    False
302     True
Name: Ca, Length: 303, dtype: bool
```

In [27]:
```
print(df["Ca"].notnull())
```

```
0       True
1       True
2       True
3       True
4       True
       ...
298     True
299     True
300     True
301     True
302    False
Name: Ca, Length: 303, dtype: bool
```

In [28]:
```
print(df[numerical_var].notnull())
```

```
     Unnamed: 0   Age   Sex  RestBP  Chol   Fbs  RestECG  MaxHR  ExAng  \
0          True  True  True    True  True  True     True   True   True
1          True  True  True    True  True  True     True   True   True
2          True  True  True    True  True  True     True   True   True
3          True  True  True    True  True  True     True   True   True
4          True  True  True    True  True  True     True   True   True
..          ...   ...   ...     ...   ...   ...      ...    ...    ...
298        True  True  True    True  True  True     True   True   True
299        True  True  True    True  True  True     True   True   True
300        True  True  True    True  True  True     True   True   True
301        True  True  True    True  True  True     True   True   True
302        True  True  True    True  True  True     True   True   True

     Oldpeak  Slope     Ca
0       True   True   True
1       True   True   True
2       True   True   True
3       True   True   True
4       True   True   True
..       ...    ...    ...
298     True   True   True
299     True   True   True
300     True   True   True
301     True   True   True
302     True   True  False

[303 rows x 12 columns]
```

In [30]:
```
print(df.dtypes)
```

```
Unnamed: 0      int64
Age             int64
Sex             int64
ChestPain      object
RestBP          int64
Chol            int64
Fbs             int64
RestECG         int64
MaxHR           int64
ExAng           int64
Oldpeak       float64
Slope           int64
Ca            float64
Thal           object
AHD            object
dtype: object
```

In [31]:
```
print((df==0).sum(axis=0))
```

```
Unnamed: 0      0
Age             0
Sex            97
ChestPain       0
RestBP          0
Chol            0
Fbs           258
RestECG       151
MaxHR           0
ExAng         204
Oldpeak        99
Slope           0
Ca            176
Thal            0
AHD             0
dtype: int64
```

In [32]:
```
pandas_dataframe = pd.DataFrame({'a':[1,0,0,1,3], 'b':[0,0,1,0,1], 'c':[0,0,0,0,0]})
print((pandas_dataframe==0).sum(axis=0))
age_mean= np.mean(df['Age'])
print("The mean age of patient = ",age_mean)
age_mean= np.median(df['Age'])
print("The median age of patient = ",age_mean)
```

```
a    2
b    3
c    5
dtype: int64
The mean age of patient =  54.43894389438944
The median age of patient =  56.0
```

In [33]:
```
temp1=df.iloc[:,[1,2,3,4,5]]
print(temp1)
print(type(temp1))
tra_set=temp1.sample(frac=0.75)
tes_set=temp1.drop(tra_set.index)
print(tra_set)
print(tes_set)
```

```
     Age  Sex     ChestPain  RestBP  Chol
0     63    1       typical     145   233
1     67    1  asymptomatic     160   286
2     67    1  asymptomatic     120   229
3     37    1    nonanginal     130   250
4     41    0    nontypical     130   204
..   ...  ...           ...     ...   ...
298   45    1       typical     110   264
299   68    1  asymptomatic     144   193
300   57    1  asymptomatic     130   131
301   57    0    nontypical     130   236
302   38    1    nonanginal     138   175

[303 rows x 5 columns]
<class 'pandas.core.frame.DataFrame'>
     Age  Sex     ChestPain  RestBP  Chol
96    59    1  asymptomatic     110   239
163   58    0  asymptomatic     100   248
173   62    0  asymptomatic     140   394
227   67    0    nonanginal     152   277
295   41    1    nontypical     120   157
..   ...  ...           ...     ...   ...
291   55    0    nontypical     132   342
206   58    1  asymptomatic     128   259
201   64    0  asymptomatic     180   325
280   57    1  asymptomatic     110   335
165   57    1  asymptomatic     132   207

[227 rows x 5 columns]
     Age  Sex     ChestPain  RestBP  Chol
0     63    1       typical     145   233
2     67    1  asymptomatic     120   229
4     41    0    nontypical     130   204
13    44    1    nontypical     120   263
18    48    0    nonanginal     130   275
..   ...  ...           ...     ...   ...
272   46    1  asymptomatic     140   311
276   66    0    nonanginal     146   278
281   47    1    nonanginal     130   253
290   67    1    nonanginal     152   212
300   57    1  asymptomatic     130   131

[76 rows x 5 columns]
```

In [34]:
```
print(df)
```

```
     Unnamed: 0  Age  Sex     ChestPain  RestBP  Chol  Fbs  RestECG  MaxHR  \
0             1   63    1       typical     145   233    1        2    150
1             2   67    1  asymptomatic     160   286    0        2    108
2             3   67    1  asymptomatic     120   229    0        2    129
3             4   37    1    nonanginal     130   250    0        0    187
4             5   41    0    nontypical     130   204    0        2    172
..          ...  ...  ...           ...     ...   ...  ...      ...    ...
298         299   45    1       typical     110   264    0        0    132
299         300   68    1  asymptomatic     144   193    1        0    141
300         301   57    1  asymptomatic     130   131    0        0    115
301         302   57    0    nontypical     130   236    0        2    174
302         303   38    1    nonanginal     138   175    0        0    173

     ExAng  Oldpeak  Slope   Ca        Thal  AHD
0        0      2.3      3  0.0       fixed   No
1        1      1.5      2  3.0      normal  Yes
2        1      2.6      2  2.0  reversable  Yes
3        0      3.5      3  0.0      normal   No
4        0      1.4      1  0.0      normal   No
..     ...      ...    ...  ...         ...  ...
298      0      1.2      2  0.0  reversable  Yes
299      0      3.4      2  2.0  reversable  Yes
300      1      1.2      2  1.0  reversable  Yes
301      0      0.0      2  1.0      normal  Yes
302      0      0.0      1  NaN      normal   No

[303 rows x 15 columns]
```

In [35]:
```
y=df.AHD
print(y)
```

```
0       No
1      Yes
2      Yes
3       No
4       No
      ...
298    Yes
299    Yes
300    Yes
301    Yes
302     No
Name: AHD, Length: 303, dtype: object
```

In [36]:
```
x=df.drop('AHD',axis=1)
print(x)
```

```
     Unnamed: 0  Age  Sex     ChestPain  RestBP  Chol  Fbs  RestECG  MaxHR  \
0             1   63    1       typical     145   233    1        2    150
1             2   67    1  asymptomatic     160   286    0        2    108
2             3   67    1  asymptomatic     120   229    0        2    129
3             4   37    1    nonanginal     130   250    0        0    187
4             5   41    0    nontypical     130   204    0        2    172
..          ...  ...  ...           ...     ...   ...  ...      ...    ...
298         299   45    1       typical     110   264    0        0    132
299         300   68    1  asymptomatic     144   193    1        0    141
300         301   57    1  asymptomatic     130   131    0        0    115
301         302   57    0    nontypical     130   236    0        2    174
302         303   38    1    nonanginal     138   175    0        0    173

     ExAng  Oldpeak  Slope   Ca        Thal
0        0      2.3      3  0.0       fixed
1        1      1.5      2  3.0      normal
2        1      2.6      2  2.0  reversable
3        0      3.5      3  0.0      normal
4        0      1.4      1  0.0      normal
..     ...      ...    ...  ...         ...
298      0      1.2      2  0.0  reversable
299      0      3.4      2  2.0  reversable
300      1      1.2      2  1.0  reversable
301      0      0.0      2  1.0      normal
302      0      0.0      1  NaN      normal

[303 rows x 14 columns]
```

In [37]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25)
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(227, 14)
(76, 14)
(227,)
(76,)
```

In [38]:
```
from sklearn import metrics
import matplotlib.pyplot as plt
```

In [39]:
```
actual=[1,0,1,1,1,0,0,1,1,0,1,0]
```

In [40]:
```
predicted=[1,1,0,1,1,0,0,1,1,0,1,1]
```

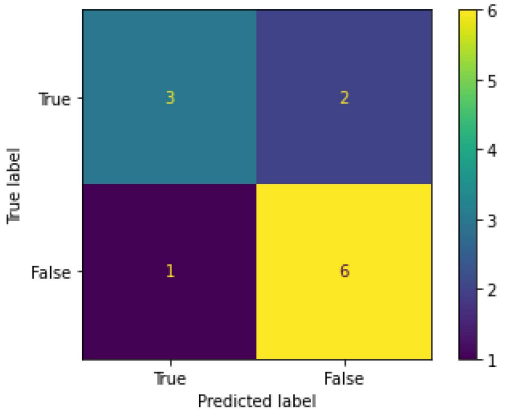
In [41]:
```
print(metrics.confusion_matrix(actual, predicted))
```

```
[[3 2]
 [1 6]]
```

In [42]:
```
confusion_matrix = metrics.confusion_matrix(actual, predicted)

cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [True, False])

cm_display.plot()
plt.show()
```



In [43]:
```
from sklearn.metrics import accuracy_score
from sklearn.metrics import f1_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
```

In [44]:
```
print("Accuracy: %.3f" % accuracy_score(actual, predicted))
print("F1-Score: %.3f" % f1_score(actual, predicted))
print("Precision: %.3f" % precision_score(actual, predicted))
print("Recall: %.3f" % recall_score(actual, predicted))
```

```
Accuracy: 0.750
F1-Score: 0.800
Precision: 0.750
Recall: 0.857
```

In [ ]: