In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df=pd.read_csv('Heart.csv')
#print(df.head(3))
df=df.drop("Unnamed: 0", axis=1)
# Task 1
print("==================== Dimentions of the Dataset ====================")
print("Number of rows : ",df.shape[0]) # index zero returns rows
print("Number of columns : ",df.shape[1]) #index on
```

```
==================== Dimentions of the Dataset ====================
Number of rows :  303
Number of columns :  14
```

In [3]:
```
# Task 2
print("==================== Null values in the Dataset ====================")
print(df.isnull().sum()) # print zero if null values not found
```

```
==================== Null values in the Dataset ====================
Age          0
Sex          0
ChestPain    0
RestBP       0
Chol         0
Fbs          0
RestECG      0
MaxHR        0
ExAng        0
Oldpeak      0
Slope        0
Ca           4
Thal         2
AHD          0
dtype: int64
```

In [4]:
```
# Task 3
print("==================== Fetures of the Dataset ======================")
print(df.info()) # retuns count,datatypes,presense of null values,Memory usage
#df.dtypes It also works fine
```

```
==================== Fetures of the Dataset ======================
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Age        303 non-null    int64
 1   Sex        303 non-null    int64
 2   ChestPain  303 non-null    object
 3   RestBP     303 non-null    int64
 4   Chol       303 non-null    int64
 5   Fbs        303 non-null    int64
 6   RestECG    303 non-null    int64
 7   MaxHR      303 non-null    int64
 8   ExAng      303 non-null    int64
 9   Oldpeak    303 non-null    float64
 10  Slope      303 non-null    int64
 11  Ca         299 non-null    float64
 12  Thal       301 non-null    object
 13  AHD        303 non-null    object
dtypes: float64(2), int64(9), object(3)
memory usage: 33.3+ KB
None
```

In [6]:
```
#Task 4
for column_name in df.columns:
    column = df[column_name]
    # Get the count of Zeros in column
    count = (column == 0).sum()
    print('Count of zeros in column ', column_name, ' is : ', count)
```

```
Count of zeros in column  Age  is :  0
Count of zeros in column  Sex  is :  97
Count of zeros in column  ChestPain  is :  0
Count of zeros in column  RestBP  is :  0
Count of zeros in column  Chol  is :  0
Count of zeros in column  Fbs  is :  258
Count of zeros in column  RestECG  is :  151
Count of zeros in column  MaxHR  is :  0
Count of zeros in column  ExAng  is :  204
Count of zeros in column  Oldpeak  is :  99
Count of zeros in column  Slope  is :  0
Count of zeros in column  Ca  is :  176
Count of zeros in column  Thal  is :  0
Count of zeros in column  AHD  is :  0
```

In [8]:
```
# Task 5
print("====================== Mean Age of the patients ====================")
print(df.Age.mean())
```

```
====================== Mean Age of the patients ====================
54.43894389438944
```

In [9]:
```
# Task 6
print("====================== Extracting the columns ============================")
dff = pd.DataFrame(df)
cols = [1,2,3,4,5]
dff = dff[dff.columns[cols]]
print(dff)
```

```
====================== Extracting the columns ============================
     Sex     ChestPain  RestBP  Chol  Fbs
0      1       typical     145   233    1
1      1  asymptomatic     160   286    0
2      1  asymptomatic     120   229    0
3      1    nonanginal     130   250    0
4      0    nontypical     130   204    0
..   ...           ...     ...   ...  ...
298    1       typical     110   264    0
299    1  asymptomatic     144   193    1
300    1  asymptomatic     130   131    0
301    0    nontypical     130   236    0
302    1    nonanginal     138   175    0

[303 rows x 5 columns]
```

In [17]:

```
# Task 7
print("========================= Randomly devide the dataset into tranning dataset ==================")
from sklearn.model_selection import train_test_split
dff = pd.DataFrame(df)
cols = [1,2,3,4,5]
dff = dff[dff.columns[cols]]
#print(dff)
x=df[["Age","Sex","ChestPain","RestBP","Chol"]]
print(x)

y=x.iloc[:,-1]
print(y)

train_test_split(x,y)
y=y.map({'No':0,'Yes':1})#maping the output with 1,0
train_x,test_x,train_y,test_y=train_test_split(x,y,test_size=0.25)
```

```
========================= Randomly devide the dataset into tranning dataset ==================
     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]
0      233
1      286
2      229
3      250
4      204
      ...
298    264
299    193
300    131
301    236
302    175
Name: Chol, Length: 303, dtype: int64
```

In [18]:
```
train_x
```

Out[18]:

	Age	Sex	ChestPain	RestBP	Chol
253	51	0	nonanginal	120	295
232	49	1	nonanginal	118	149
41	40	1	typical	140	199
46	51	1	nonanginal	110	175
107	57	1	nonanginal	128	229
...	...	...	...	...	...
28	43	1	asymptomatic	150	247
272	46	1	asymptomatic	140	311
48	65	0	nonanginal	140	417
262	60	0	typical	150	240
132	29	1	nontypical	130	204

227 rows × 5 columns

In [19]:
```
test_x
```

Out[19]:

	Age	Sex	ChestPain	RestBP	Chol
60	51	0	asymptomatic	130	305
163	58	0	asymptomatic	100	248
56	50	1	nonanginal	140	233
182	42	1	typical	148	244
120	48	1	asymptomatic	130	256
...	...	...	...	...	...
0	63	1	typical	145	233
198	50	0	nontypical	120	244
138	35	1	asymptomatic	120	198
297	57	0	asymptomatic	140	241
94	63	0	nonanginal	135	252

76 rows × 5 columns

In [20]:
```
train_y
```

Out[20]:
```
253   NaN
232   NaN
41    NaN
46    NaN
107   NaN
      ..
28    NaN
272   NaN
48    NaN
262   NaN
132   NaN
Name: Chol, Length: 227, dtype: float64
```

In [21]:
```
test_y
```

Out[21]:
```
60    NaN
163   NaN
56    NaN
182   NaN
120   NaN
      ..
0     NaN
198   NaN
138   NaN
297   NaN
94    NaN
Name: Chol, Length: 76, dtype: float64
```

In [ ]:
```

```