In [1]:

```
#Name : Ankita Gulde

#Roll no : 44
#Section : 3A
```

In [1]:

```
#Aim : To perform operation on KNN (K Nearest Neighbor)
```

Importing Libraries

In [7]:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.model_selection import train_test_split
import warnings
warnings.filterwarnings('ignore')
```

In [8]:

```
import os
```

In [9]:
```
os.getcwd()
```

Out[9]:
```
'C:\\Users\\HP'
```

In [10]:
```
os.chdir("C:\\Users\\HP\\Desktop")
```

In [11]:
```
df=pd.read_csv("framingham.csv")
```

In [12]:
```
#The "Framingham" heart disease dataset includes over 4,240 records, 15 attrib
#The goal of the dataset is to predict whether the patient has 10-year risk of
```

In [13]:
```
df.head()
```

Out[13]:

| | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentStroke | prevalentHyp diab |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 39 | 4.0 | 0 | 0.0 | 0.0 | 0 | 0 |
| **1** | 0 | 46 | 2.0 | 0 | 0.0 | 0.0 | 0 | 0 |
| **2** | 1 | 48 | 1.0 | 1 | 20.0 | 0.0 | 0 | 0 |
| **3** | 0 | 61 | 3.0 | 1 | 30.0 | 0.0 | 0 | 1 |
| **4** | 0 | 46 | 3.0 | 1 | 23.0 | 0.0 | 0 | 0 |

In [14]:
```
df.describe()
```

Out[14]:

| | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentS |
|---|---|---|---|---|---|---|---|
| **count** | 4238.000000 | 4238.000000 | 4133.000000 | 4238.000000 | 4209.000000 | 4185.000000 | 4238.0 |
| **mean** | 0.429212 | 49.584946 | 1.978950 | 0.494101 | 9.003089 | 0.029630 | 0.0 |
| **std** | 0.495022 | 8.572160 | 1.019791 | 0.500024 | 11.920094 | 0.169584 | 0.0 |
| **min** | 0.000000 | 32.000000 | 1.000000 | 0.000000 | 0.000000 | 0.000000 | 0.0 |
| **25%** | 0.000000 | 42.000000 | 1.000000 | 0.000000 | 0.000000 | 0.000000 | 0.0 |
| **50%** | 0.000000 | 49.000000 | 2.000000 | 0.000000 | 0.000000 | 0.000000 | 0.0 |

```
In     df.info()
```

|  | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentS |
|---|---|---|---|---|---|---|---|
| **max** | 1.000000 | 70.000000 | 4.000000 | 1.000000 | 70.000000 | 1.000000 | 1.0 |
| **75%** | 1.000000 | 56.000000 | 3.000000 | 1.000000 | 20.000000 | 0.000000 | 0.0 |

```
<class
'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to
4237
Data columns (total 16 columns):   Dtype
---   -------        -------        ------
0     male           423  non-      int64
                      8   null
1     age            423  non-      int64
                      8   null
2     education      413  non-      float64
                      3   null
3     currentSmoker  423  non-      int64
                      8   null
4     cigsPerDay     420  non-      float64
                      9   null
5     BPMeds         418  non-      float64
                      5   null
6     prevalentStroke 423 non-      int64
                      8   null
7     prevalentHyp   423  non-      int64
                      8   null
8     diabetes       423  non-      int64
                      8   null
9     totChol        418  non-      float64
                      8   null
10    sysBP          423  non-      float64
                      8   null
11    diaBP          423  non-      float64
                      8   null
12    BMI            421  non-      float64
                      9   null
13    heartRate      423  non-      float64
                      7   null
14    glucose        385  non-      float64
                      0   null
```

```
 15    TenYearCHD        423   non-        int64
                          8    null

 memory usage:         KB
 529.9
```

```
df.isna().sum()
```

```
 male                   0

 age                    0

 education            105

 currentSmoker          0

 cigsPerDay            29

 BPMeds                53

 prevalentStroke        0

 prevalentHyp           0

 diabetes               0

 totChol               50

 sysBP                  0

 diaBP                  0

 BMI                   19

 heartRate              1

 glucose              388

 TenYearCHD             0
 dtype: int64
dtypes: float64(9), int64(7)
```

In [16]:

Out[16]:

In [ ]:

```
#Since, only a few rows have null values in them, we are only removing those
#df = df.dropna(subset=['heartRate','BMI','cigsPerDay','totChol','BPMeds'])
```

In [17]:

```
df
```

Out[17]:

| | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentStroke | prevalentHyp | d |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 39 | 4.0 | 0 | 0.0 | 0.0 | 0 | 0 | |

|  | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentStroke | prevalentHyp | d |
|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 46 | 2.0 | 0 | 0.0 | 0.0 | 0 | 0 |  |
| **2** | 1 | 48 | 1.0 | 1 | 20.0 | 0.0 | 0 | 0 |  |
| **3** | 0 | 61 | 3.0 | 1 | 30.0 | 0.0 | 0 | 1 |  |
| **4** | 0 | 46 | 3.0 | 1 | 23.0 | 0.0 | 0 | 0 |  |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |  |
| **4233** | 1 | 50 | 1.0 | 1 | 1.0 | 0.0 | 0 | 1 |  |
| **4234** | 1 | 51 | 3.0 | 1 | 43.0 | 0.0 | 0 | 0 |  |
| **4235** | 0 | 48 | 2.0 | 1 | 20.0 | NaN | 0 | 0 |  |
| **4236** | 0 | 44 | 1.0 | 1 | 15.0 | 0.0 | 0 | 0 |  |
| **4237** | 0 | 52 | 2.0 | 0 | 0.0 | 0.0 | 0 | 0 |  |

4238 rows × 16 columns

# Missing Value Treatment

Since, 'glucose' and 'education' columns had a significant amount of null values, so we replaced them with the mean of values for their respective columns

In [18]:

```
 df['glucose'].fillna(value = df['glucose'].mean(),inplace=True)
```

In [19]:

```
df['education'].fillna(value = df['education'].mean(),inplace=True)
```

In [20]:

```
df['heartRate'].fillna(value = df['heartRate'].mean(),inplace=True)
```

In [21]:

```
 df['BMI'].fillna(value = df['BMI'].mean(),inplace=True)
```

In [22]:

```
 df['cigsPerDay'].fillna(value = df['cigsPerDay'].mean(),inplace=True)
```

In [23]:

```
df['totChol'].fillna(value = df['totChol'].mean(),inplace=True)
```

In [24]:

```
df['BPMeds'].fillna(value = df['BPMeds'].mean(),inplace=True)
```

In [26]:

```
df.isna().sum()
```

Out[26]:

```
male               0
age                0
education          0
currentSmoker      0
cigsPerDay         0
BPMeds             0
prevalentStroke    0
```

```
prevalent
Hyp
0
diabetes
0
totChol
0
sysBP
0
diaBP
0
BMI
0
```

```
heartRate          0
glucose            0
TenYearCHD         0
dtype: int64
```

Logistic Regression Model

In [27]:
```
#Splitting the dependent and independent variables.
x = df.drop("TenYearCHD",axis=1)
y = df['TenYearCHD']
```

In [28]:
```
x #checking the features
```

Out[28]:

| | male | age | education | currentSmoker | cigsPerDay | BPMeds | prevalentStroke | prevalentHyp | d |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 39 | 4.0 | 0 | 0.0 | 0.00000 | 0 | 0 | |
| **1** | 0 | 46 | 2.0 | 0 | 0.0 | 0.00000 | 0 | 0 | |
| **2** | 1 | 48 | 1.0 | 1 | 20.0 | 0.00000 | 0 | 0 | |
| **3** | 0 | 61 | 3.0 | 1 | 30.0 | 0.00000 | 0 | 1 | |
| **4** | 0 | 46 | 3.0 | 1 | 23.0 | 0.00000 | 0 | 0 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | |
| **4233** | 1 | 50 | 1.0 | 1 | 1.0 | 0.00000 | 0 | 1 | |
| **4234** | 1 | 51 | 3.0 | 1 | 43.0 | 0.00000 | 0 | 0 | |
| **4235** | 0 | 48 | 2.0 | 1 | 20.0 | 0.02963 | 0 | 0 | |
| **4236** | 0 | 44 | 1.0 | 1 | 15.0 | 0.00000 | 0 | 0 | |
| **4237** | 0 | 52 | 2.0 | 0 | 0.0 | 0.00000 | 0 | 0 | |

4238 rows × 15 columns

# Train Test Split

In [29]:
```
x_train,x_test,y_train,y_test  =  train_test_split(x,y,test_size=0.2,random_stat
```

In [30]:
```
y_train
```

Out[30]:
```
3252    0
 3946    0
 1261    0
 2536    0
```

```
4089    0
       ..
3444    0
466     0
3092    0
3772    0
860     0
Name   TenYearCHD, Length: 3390, dtype:
:      int64
```

# KNN Classifier

In [31]:

```
 from sklearn.neighbors import KNeighborsClassifier
 knn = KNeighborsClassifier(n_neighbors=5, p=2, metric='minkowski')
 knn.fit(x_train, y_train)
 acc = knn.score(x_test,y_test)*100
print(acc)
```

```
83.13679245283019
```

In [ ]: