import Libraries

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

import Linear Regression

In [2]:

```
from sklearn.linear_model import LogisticRegression
```

In [3]:

```
lgr=LogisticRegression()
```

Select Required data from certain columns

In [4]:

```
a=pd.read_csv("fram.csv")
a
```

Out[4]:

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

4238 rows × 16 columns

In [5]:

```
c=a.dropna()
c
```

Out[5]:

e	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes
1	39	4.0	0	0.0	0.0	0	0	(
)	46	2.0	0	0.0	0.0	0	0	(
1	48	1.0	1	20.0	0.0	0	0	(
)	61	3.0	1	30.0	0.0	0	1	(
)	46	3.0	1	23.0	0.0	0	0	(
.	...	...	...	...	...	...	...	..
1	58	3.0	0	0.0	0.0	0	1	(
1	68	1.0	0	0.0	0.0	0	1	(
1	50	1.0	1	1.0	0.0	0	1	(
1	51	3.0	1	43.0	0.0	0	0	(
)	52	2.0	0	0.0	0.0	0	0	(

× 16 columns

In [6]:

```
c.columns
```

Out[6]:

```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMed
s',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

In [7]:

```
fm=c.iloc[:,0:16]
tv=c.iloc[:,-1]
```

Shape

In [8]:

```
fm.shape
```

Out[8]:

```
(3656, 16)
```

In [9]:

```
tv.shape
```

Out[9]:

```
(3656,)
```

To make the data in order (feature matrix)

In [10]:

```
from sklearn.preprocessing import StandardScaler
```

In [11]:

```
fs=StandardScaler().fit_transform(fm)
```

Imply Logistic Regression

In [12]:

```
lgr.fit(fs,tv)
```

Out[12]:

```
LogisticRegression()
```

Prediction

In [16]:

```
ab=[[3,90,543,34,5,6,67,4,3,21,54,34,56,4,543,45]]
```

In [17]:

```
pre=lgr.predict(ab)
```

In [18]:

```
print(pre)
```

```
[1]
```

To check the output var we have got

In [19]:

```
lgr.classes_
```

Out[19]:

```
array([0, 1], dtype=int64)
```

Prediction in Probablity value

In [20]:

```
lgr.predict_proba(ab)[0][1]
```

Out[20]:

```
1.0
```