```
import seaborn as sns
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
Insert code cell below (Ctrl+M B) import accuracy_score,classification_report
```

```
data=pd.read_csv('heart_disease_dataset.csv')
```

```
data
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Alcohol Intake	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina
0	75	Female	228	119	66	Current	Heavy	1	No	No	Yes	8	119	Yes
1	48	Male	204	165	62	Current	NaN	5	No	No	No	9	70	Yes
2	53	Male	234	91	67	Never	Heavy	3	Yes	No	Yes	5	196	Yes
3	69	Female	192	90	72	Current	NaN	4	No	Yes	No	7	107	Yes
4	62	Female	172	163	93	Never	NaN	6	No	Yes	No	2	183	Yes
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	56	Female	269	111	86	Never	Heavy	5	No	Yes	Yes	10	120	No
996	78	Female	334	145	76	Never	NaN	6	No	No	No	10	196	Yes
997	79	Male	151	179	81	Never	Moderate	4	Yes	No	Yes	8	189	Yes

```
data.head()
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Alcohol Intake	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina
0	75	Female	228	119	66	Current	Heavy	1	No	No	Yes	8	119	Yes
1	48	Male	204	165	62	Current	NaN	5	No	No	No	9	70	Yes
2	53	Male	234	91	67	Never	Heavy	3	Yes	No	Yes	5	196	Yes

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 16 columns):
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   Age                      1000 non-null   int64 
 1   Gender                   1000 non-null   object
 2   Cholesterol              1000 non-null   int64 
 3   Blood Pressure           1000 non-null   int64 
 4   Heart Rate               1000 non-null   int64 
 5   Smoking                  1000 non-null   object
 6   Alcohol Intake           660 non-null    object
 7   Exercise Hours           1000 non-null   int64 
 8   Family History           1000 non-null   object
 9   Diabetes                 1000 non-null   object
 10  Obesity                  1000 non-null   object
 11  Stress Level             1000 non-null   int64 
 12  Blood Sugar              1000 non-null   int64 
 13  Exercise Induced Angina  1000 non-null   object
 14  Chest Pain Type          1000 non-null   object
 15  Heart Disease            1000 non-null   int64 
dtypes: int64(8), object(8)
memory usage: 125.1+ KB
```

```
data.describe()
```

	Age	Cholesterol	Blood Pressure	Heart Rate	Exercise Hours	Stress Level	Blood Sugar	Heart Disease
count	1000.000000	1000.000000	1000.0000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000
mean	52.293000	249.939000	135.2810	79.204000	4.529000	5.646000	134.941000	0.392000
Insert code cell below (Ctrl+M B)		57.914673	26.3883	11.486092	2.934241	2.831024	36.699624	0.488441
min	25.000000	150.000000	90.0000	60.000000	0.000000	1.000000	70.000000	0.000000
25%	39.000000	200.000000	112.7500	70.000000	2.000000	3.000000	104.000000	0.000000
50%	52.000000	248.000000	136.0000	79.000000	4.500000	6.000000	135.000000	0.000000
75%	66.000000	299.000000	159.0000	89.000000	7.000000	8.000000	167.000000	1.000000
max	79.000000	349.000000	179.0000	99.000000	9.000000	10.000000	199.000000	1.000000

```
data.isnull()
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Alcohol Intake	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	True	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	True	False	False	False	False	False	False	False
4	False	False	False	False	False	False	True	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	False	False	False	False	False	False	False	False	False	False	False	False	False	False
996	False	False	False	False	False	False	True	False	False	False	False	False	False	False
997	False	False	False	False	False	False	False	False	False	False	False	False	False	False
998	False	False	False	False	False	False	True	False	False	False	False	False	False	False
999	False	False	False	False	False	False	True	False	False	False	False	False	False	False

```
data.isnull().sum()*100
```

	0
Age	0
Gender	0
Cholesterol	0
Blood Pressure	0
Heart Rate	0
Smoking	0
Alcohol Intake	34000
Exercise Hours	0
Family History	0
Diabetes	0
Obesity	0
Stress Level	0
Blood Sugar	0
Exercise Induced Angina	0
Chest Pain Type	0
Heart Disease	0

dtype: int64

```
data=data.drop('Alcohol Intake',axis=1)
```

```
data
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest P T
0	75	Female	228	119	66	Current	1	No	No	Yes	8	119	Yes	Aty An
1	48	Male	204	165	62	Current	5	No	No	No	9	70	Yes	Ty An
2	53	Male	234	91	67	Never	3	Yes	No	Yes	5	196	Yes	Aty An
3	69	Female	192	90	72	Current	4	No	Yes	No	7	107	Yes	Non-ang
4	62	Female	172	163	93	Never	6	No	Yes	No	2	183	Yes	Asympton
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
995	56	Female	269	111	86	Never	5	No	Yes	Yes	10	120	No	Non-ang
996	78	Female	334	145	76	Never	6	No	No	No	10	196	Yes	Ty An
997	79	Male	151	179	81	Never	4	Yes	No	Yes	8	189	Yes	Asympton

Insert code cell below (Ctrl+M B)

```
data.nunique()
```

	0
Age	55
Gender	2
Cholesterol	200
Blood Pressure	90
Heart Rate	40
Smoking	3
Exercise Hours	10
Family History	2
Diabetes	2
Obesity	2
Stress Level	10
Blood Sugar	130
Exercise Induced Angina	2
Chest Pain Type	4
Heart Disease	2

dtype: int64

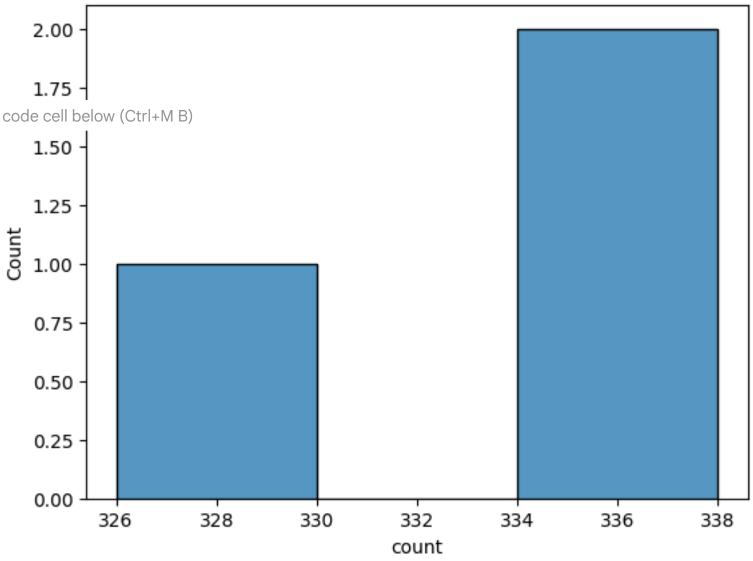
```
data['Smoking'].value_counts()
```

	count
Smoking	
Never	338
Current	336
Former	326

dtype: int64

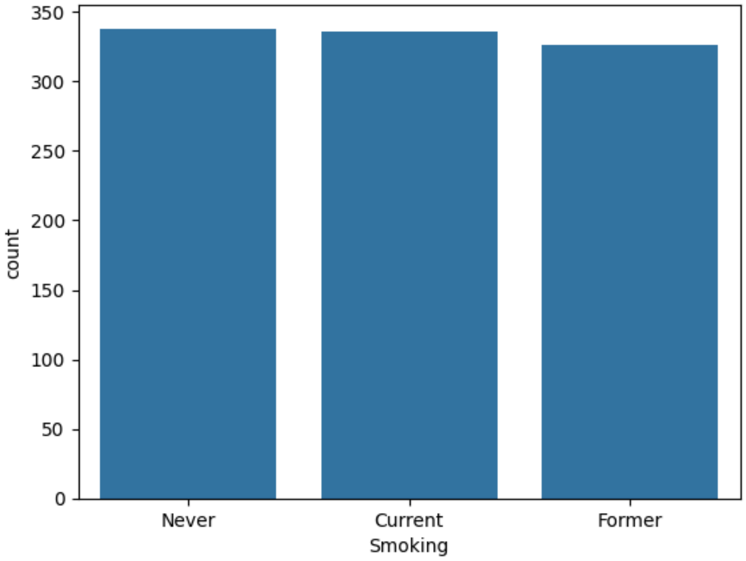
```
sns.histplot(data['Smoking'].value_counts())
```

```
<Axes: xlabel='count', ylabel='Count'>
```



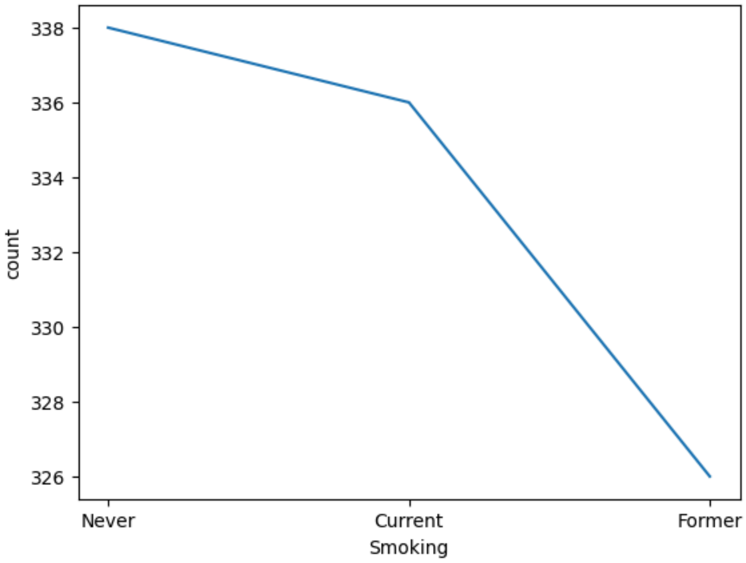
```
sns.barplot(data['Smoking'].value_counts())
```

```
<Axes: xlabel='Smoking', ylabel='count'>
```



```
sns.lineplot(data['Smoking'].value_counts())
```

```
<Axes: xlabel='Smoking', ylabel='count'>
```



```
data.columns
```

```
Index(['Age', 'Gender', 'Cholesterol', 'Blood Pressure', 'Heart Rate',
       'Smoking', 'Exercise Hours', 'Family History', 'Diabetes', 'Obesity',
       'Stress Level', 'Blood Sugar', 'Exercise Induced Angina',
       'Chest Pain Type', 'Heart Disease'],
      dtype='object')
```

Insert code cell below (Ctrl+M B)

```
x=[['Age'],'BloodPressure','SkinThickness','Insulin','BMI','DiabetesPedigreeFunction','Age']
y=['Outcome']
```

```
data.notnull()
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest Pain Type
0	True	True	True	True	True	True	True	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True	True	True	True	True	True	True	True
2	True	True	True	True	True	True	True	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True	True	True	True	True	True	True	True
4	True	True	True	True	True	True	True	True	True	True	True	True	True	True
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	True	True	True	True	True	True	True	True	True	True	True	True	True	True
996	True	True	True	True	True	True	True	True	True	True	True	True	True	True
997	True	True	True	True	True	True	True	True	True	True	True	True	True	True
998	True	True	True	True	True	True	True	True	True	True	True	True	True	True
999	True	True	True	True	True	True	True	True	True	True	True	True	True	True

```
data.fillna(2)
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest P T
0	75	Female	228	119	66	Current	1	No	No	Yes	8	119	Yes	Aty An
1	48	Male	204	165	62	Current	5	No	No	No	9	70	Yes	Ty An
2	53	Male	234	91	67	Never	3	Yes	No	Yes	5	196	Yes	Aty An
3	69	Female	192	90	72	Current	4	No	Yes	No	7	107	Yes	Non-ang
4	62	Female	172	163	93	Never	6	No	Yes	No	2	183	Yes	Asympton
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
995	56	Female	269	111	86	Never	5	No	Yes	Yes	10	120	No	Non-ang
996	78	Female	334	145	76	Never	6	No	No	No	10	196	Yes	Ty An
997	79	Male	151	179	81	Never	4	Yes	No	Yes	8	189	Yes	Asympton

```
data.replace()
```

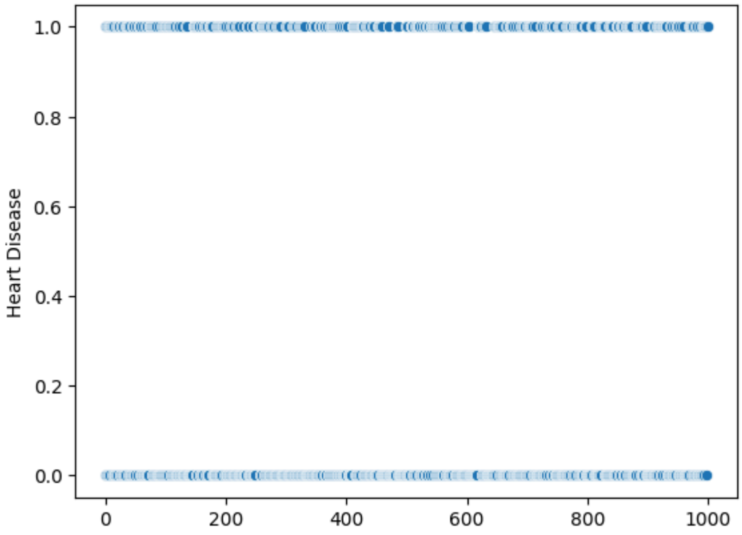
```
/tmp/ipython-input-295544180.py:1: FutureWarning: DataFrame.replace without 'value' and with non-dict-like 'to_replace' is depr
  data.replace()
```

Insert code cell below (Ctrl+M B)

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest I T
0	75	Female	228	119	66	Current	1	No	No	Yes	8	119	Yes	Aty An
1	48	Male	204	165	62	Current	5	No	No	No	9	70	Yes	Ty An
2	53	Male	234	91	67	Never	3	Yes	No	Yes	5	196	Yes	Aty An
3	69	Female	192	90	72	Current	4	No	Yes	No	7	107	Yes	Non-an
4	62	Female	172	163	93	Never	6	No	Yes	No	2	183	Yes	Asympton
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
995	56	Female	269	111	86	Never	5	No	Yes	Yes	10	120	No	Non-an
996	78	Female	334	145	76	Never	6	No	No	No	10	196	Yes	Ty An
997	79	Male	151	179	81	Never	4	Yes	No	Yes	8	189	Yes	Asympton
998	60	Female	326	151	68	Former	8	Yes	Yes	No	5	174	Yes	Aty

```
sns.scatterplot(data['Heart Disease'])
```

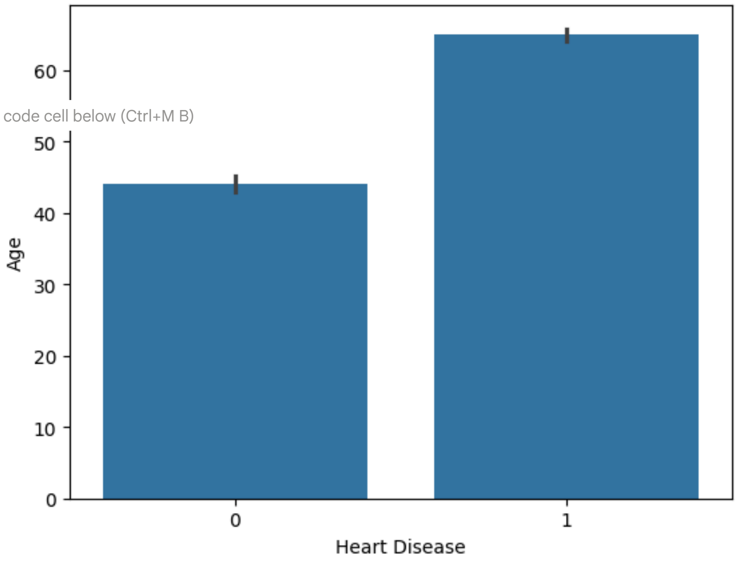
```
<Axes: ylabel='Heart Disease'>
```



```
sns.barplot(x='Heart Disease',y='Age',data=data)
```

```
<Axes: xlabel='Heart Disease', ylabel='Age'>
```

Insert code cell below (Ctrl+M B)



```
#encoding
```

```
data.columns
```

```
Index(['Age', 'Gender', 'Cholesterol', 'Blood Pressure', 'Heart Rate',
       'Smoking', 'Exercise Hours', 'Family History', 'Diabetes', 'Obesity',
       'Stress Level', 'Blood Sugar', 'Exercise Induced Angina',
       'Chest Pain Type', 'Heart Disease'],
      dtype='object')
```

```
data.head(1)
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest Pain Type	Di
0	75	Female	228	119	66	Current	1	No	No	Yes	8	119	Yes	Atypical	

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

```
a=[ 'Gender', 'Smoking', 'Family History', 'Diabetes', 'Obesity', 'Exercise Induced Angina','Chest Pain Type']
for i in a:
  data[i]=le.fit_transform(data[i])
```

```
data
```

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest Pain Type	D
0	75	0	228	119	66	0	1	0	0	1	8	119	1	1	
1	48	1	204	165	62	0	5	0	0	0	9	70	1	3	
2	53	1	234	91	67	2	3	1	0	1	5	196	1	1	
3	69	0	192	90	72	0	4	0	1	0	7	107	1	2	
4	62	0	172	163	93	2	6	0	1	0	2	183	1	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
995	56	0	269	111	86	2	5	0	1	1	10	120	0	2	
996	78	0	334	145	76	2	6	0	0	0	10	196	1	3	
997	79	1	151	179	81	2	4	1	0	1	8	189	1	0	
998	60	0	326	151	68	1	8	1	1	0	5	174	1	1	
999	53	1	226	116	82	0	6	0	0	1	5	161	1	0	

```
x=data.drop('Heart Disease',axis=1)
y=data['Heart Disease']
```

```
x
```

Insert code cell below (Ctrl+M B)

	Age	Gender	Cholesterol	Blood Pressure	Heart Rate	Smoking	Exercise Hours	Family History	Diabetes	Obesity	Stress Level	Blood Sugar	Exercise Induced Angina	Chest Pain Type
0	75	0	228	119	66	0	1	0	0	1	8	119	1	1
1	48	1	204	165	62	0	5	0	0	0	9	70	1	3
2	53	1	234	91	67	2	3	1	0	1	5	196	1	1
3	69	0	192	90	72	0	4	0	1	0	7	107	1	2
4	62	0	172	163	93	2	6	0	1	0	2	183	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	56	0	269	111	86	2	5	0	1	1	10	120	0	2
996	78	0	334	145	76	2	6	0	0	0	10	196	1	3
997	79	1	151	179	81	2	4	1	0	1	8	189	1	0
998	60	0	326	151	68	1	8	1	1	0	5	174	1	1
999	53	1	226	116	82	0	6	0	0	1	5	161	1	0

Train Test Split

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=2)
```

Scaling

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
x_train=sc.fit_transform(x_train)
x_test=sc.transform(x_test)
```

Implementation of Model