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

```
import warnings
warnings.filterwarnings('ignore')
```

```
data=pd.read_csv("data 2.csv")
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   id                       569 non-null    int64
 1   diagnosis                569 non-null    object
 2   radius_mean              569 non-null    float64
 3   texture_mean             569 non-null    float64
 4   perimeter_mean           569 non-null    float64
 5   area_mean                569 non-null    float64
 6   smoothness_mean          569 non-null    float64
 7   compactness_mean         569 non-null    float64
 8   concavity_mean           569 non-null    float64
 9   concave points_mean      569 non-null    float64
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
```

```
 19  concave points_se        569 non-null    float64
 20  symmetry_se              569 non-null    float64
 21  fractal_dimension_se     569 non-null    float64
 22  radius_worst             569 non-null    float64
 23  texture_worst            569 non-null    float64
 24  perimeter_worst          569 non-null    float64
 25  area_worst               569 non-null    float64
 26  smoothness_worst         569 non-null    float64
 27  compactness_worst        569 non-null    float64
 28  concavity_worst          569 non-null    float64
 29  concave points_worst     569 non-null    float64
 30  symmetry_worst           569 non-null    float64
 31  fractal_dimension_worst  569 non-null    float64
 32  Unnamed: 32              0 non-null      float64
dtypes: float64(31), int64(1), object(1)
memory usage: 146.8+ KB
```

```
data.shape
```

```
(569, 33)
```

```
data.head()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980

```
#data cleaning
data=pd.DataFrame(data)
```

```
data2=data.drop(columns= ['id','diagnosis','Unnamed: 32'])
```

```
data2
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavit
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	C
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	C
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	C
3	11.42	20.38	77.58	386.1	0.14250	0.28390	C
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	C
...	...	...	...	...	...	...	
564	21.56	22.39	142.00	1479.0	0.11100	0.11590	C
565	20.13	28.25	131.20	1261.0	0.09780	0.10340	C
566	16.60	28.08	108.30	858.1	0.08455	0.10230	C
567	20.60	29.33	140.10	1265.0	0.11780	0.27700	C
568	7.76	24.54	47.92	181.0	0.05263	0.04362	C

569 rows × 30 columns

```
disease_type=data['diagnosis'].astype('category')
```

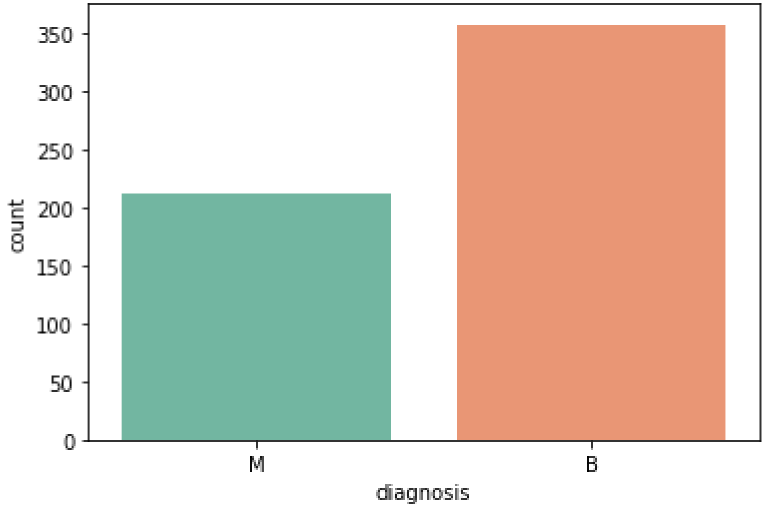
```
disease_type
```

```
0      M
1      M
2      M
3      M
```

```
4      M
      ..
564    M
565    M
566    M
567    M
568    B
Name: diagnosis, Length: 569, dtype: category
Categories (2, object): ['B', 'M']
```

```
sns.countplot(x='diagnosis',data=data,palette='Set2')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f49cc808810>
```




```
target=disease_type.replace(('M','B'),(1,0))
target
```

```
0      1
1      1
2      1
3      1
4      1
      ..
564    1
```

```
565    1
566    1
567    1
568    0
Name: diagnosis, Length: 569, dtype: int64
```

```
from sklearn.model_selection import train_test_split
```

```
x_train,x_test,y_train,y_test = train_test_split(data2,target,test_size =0.3)
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(398, 30)
(171, 30)
(398,)
(171,)
```

```
#applying the Logistic Regression
from sklearn.linear_model import LogisticRegression
classifier=LogisticRegression()
classifier.fit(x_train,y_train)
lr_pridict = classifier.predict(x_test)
```

```
import sklearn.metrics
from sklearn.metrics import classification_report,accuracy_score
```

```
prediction=classifier.predict(x_test)
prediction
print(classification_report(y_test, prediction))
```

```
              precision    recall  f1-score   support
```

```
              0       0.95      0.95      0.95       109
              1       0.92      0.90      0.91        62

       accuracy                           0.94       171
      macro avg       0.93      0.93      0.93       171
   weighted avg       0.94      0.94      0.94       171
```

```
#Decision Tree
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree
dtree = DecisionTreeClassifier()
dtree.fit(x_train,y_train)
y_pred_train = dtree.predict(x_train)
y_pred_test = dtree.predict(x_test)


print("Bias = Accuracy", accuracy_score(y_train,y_pred_train))
print("Variance = Accuracy", accuracy_score(y_test,y_pred_test))
```

```
Bias = Accuracy 1.0
Variance = Accuracy 0.9005847953216374
```

```
tree.plot_tree(dtree.fit(data2,target))
```

```
[Text(209.25, 203.85, 'X[20] <= 16.795\ngini = 0.468\nsamples = 569\nvalue = [357, 212]'),
 Text(136.01250000000002, 176.67000000000002, 'X[27] <= 0.136\ngini = 0.159\nsamples = 379\nvalue = [34
 Text(78.46875, 149.49, 'X[10] <= 1.048\ngini = 0.03\nsamples = 333\nvalue = [328, 5]'),
 Text(68.00625000000001, 122.31, 'X[13] <= 38.605\ngini = 0.024\nsamples = 332\nvalue = [328, 4]'),
 Text(41.85, 95.13, 'X[14] <= 0.003\ngini = 0.012\nsamples = 319\nvalue = [317, 2]'),
 Text(20.925, 67.94999999999999, 'X[19] <= 0.001\ngini = 0.245\nsamples = 7\nvalue = [6, 1]'),
 Text(10.4625, 40.77000000000001, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(31.387500000000003, 40.77000000000001, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
 Text(62.775000000000006, 67.94999999999999, 'X[21] <= 33.27\ngini = 0.006\nsamples = 312\nvalue = [311
 Text(52.3125, 40.77000000000001, 'gini = 0.0\nsamples = 292\nvalue = [292, 0]'),
 Text(73.2375, 40.77000000000001, 'X[21] <= 33.56\ngini = 0.095\nsamples = 20\nvalue = [19, 1]'),
 Text(62.775000000000006, 13.590000000000003, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(83.7, 13.590000000000003, 'gini = 0.0\nsamples = 19\nvalue = [19, 0]'),
 Text(94.16250000000001, 95.13, 'X[25] <= 0.082\ngini = 0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(83.7, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(104.625, 67.94999999999999, 'X[27] <= 0.117\ngini = 0.153\nsamples = 12\nvalue = [11, 1]'),
 Text(94.16250000000001, 40.77000000000001, 'gini = 0.0\nsamples = 11\nvalue = [11, 0]'),
 Text(115.0875, 40.77000000000001, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(88.93125, 122.31, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(193.55625, 149.49, 'X[21] <= 25.67\ngini = 0.476\nsamples = 46\nvalue = [18, 28]'),
 Text(156.9375, 122.31, 'X[23] <= 810.3\ngini = 0.332\nsamples = 19\nvalue = [15, 4]'),
 Text(136.01250000000002, 95.13, 'X[24] <= 0.179\ngini = 0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(125.55000000000001, 67.94999999999999, 'gini = 0.0\nsamples = 14\nvalue = [14, 0]'),
 Text(146.475, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(177.8625, 95.13, 'X[2] <= 92.79\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(167.4, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(188.32500000000002, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(230.175, 122.31, 'X[6] <= 0.097\ngini = 0.198\nsamples = 27\nvalue = [3, 24]'),
 Text(219.7125, 95.13, 'X[1] <= 19.435\ngini = 0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(209.25, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(230.175, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(240.63750000000002, 95.13, 'gini = 0.0\nsamples = 21\nvalue = [0, 21]'),
 Text(282.4875, 176.67000000000002, 'X[1] <= 16.11\ngini = 0.109\nsamples = 190\nvalue = [11, 179]'),
 Text(261.5625, 149.49, 'X[16] <= 0.034\ngini = 0.498\nsamples = 17\nvalue = [9, 8]'),
 Text(251.10000000000002, 122.31, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
 Text(272.02500000000003, 122.31, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]'),
 Text(303.4125, 149.49, 'X[24] <= 0.088\ngini = 0.023\nsamples = 173\nvalue = [2, 171]'),
 Text(292.95, 122.31, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(313.875, 122.31, 'X[26] <= 0.18\ngini = 0.012\nsamples = 172\nvalue = [1, 171]'),
 Text(303.4125, 95.13, 'X[17] <= 0.01\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(292.95, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(313.875, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
```

```
        Text(324.33750000000003, 95.13, 'gini = 0.0\nsamples = 168\nvalue = [0, 168]')]
```

```
tree.plot_tree(dtree.fit(data2,target), fontsize=8)
```

```
[Text(209.25, 203.85, 'X[20] <= 16.795\ngini = 0.468\nsamples = 569\nvalue = [357, 212]'),
 Text(136.01250000000002, 176.67000000000002, 'X[27] <= 0.136\ngini = 0.159\nsamples = 379\nvalue = [34
 Text(78.46875, 149.49, 'X[12] <= 6.597\ngini = 0.03\nsamples = 333\nvalue = [328, 5]'),
 Text(68.00625000000001, 122.31, 'X[13] <= 38.605\ngini = 0.024\nsamples = 332\nvalue = [328, 4]'),
 Text(41.85, 95.13, 'X[14] <= 0.003\ngini = 0.012\nsamples = 319\nvalue = [317, 2]'),
 Text(20.925, 67.94999999999999, 'X[1] <= 19.9\ngini = 0.245\nsamples = 7\nvalue = [6, 1]'),
 Text(10.4625, 40.77000000000001, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
 Text(31.387500000000003, 40.77000000000001, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(62.775000000000006, 67.94999999999999, 'X[21] <= 33.27\ngini = 0.006\nsamples = 312\nvalue = [311
 Text(52.3125, 40.77000000000001, 'gini = 0.0\nsamples = 292\nvalue = [292, 0]'),
 Text(73.2375, 40.77000000000001, 'X[21] <= 33.56\ngini = 0.095\nsamples = 20\nvalue = [19, 1]'),
 Text(62.775000000000006, 13.590000000000003, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(83.7, 13.590000000000003, 'gini = 0.0\nsamples = 19\nvalue = [19, 0]'),
 Text(94.16250000000001, 95.13, 'X[28] <= 0.207\ngini = 0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(83.7, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(104.625, 67.94999999999999, 'X[13] <= 39.15\ngini = 0.153\nsamples = 12\nvalue = [11, 1]'),
 Text(94.16250000000001, 40.77000000000001, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(115.0875, 40.77000000000001, 'gini = 0.0\nsamples = 11\nvalue = [11, 0]'),
 Text(88.93125, 122.31, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
```

```
#Conclusion-
#When we performed logistics regression the precision,recall and F1 scores are more than 50% which says that it can be considered.
#And when we applied Decision tree model the bias score is more than variance which means that it is a case of underfitting so we sho
```

```
 Text(177.8625, 95.13, 'X[23] <= 844.65\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(167.4, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(188.32500000000002, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(230.175, 122.31, 'X[7] <= 0.054\ngini = 0.198\nsamples = 27\nvalue = [3, 24]'),
 Text(219.7125, 95.13, 'X[21] <= 28.545\ngini = 0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(209.25, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(230.175, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(240.63750000000002, 95.13, 'gini = 0.0\nsamples = 21\nvalue = [0, 21]'),
 Text(282.4875, 176.67000000000002, 'X[1] <= 16.11\ngini = 0.109\nsamples = 190\nvalue = [11, 179]'),
 Text(261.5625, 149.49, 'X[7] <= 0.066\ngini = 0.498\nsamples = 17\nvalue = [9, 8]'),
 Text(251.10000000000002, 122.31, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
 Text(272.02500000000003, 122.31, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]'),
 Text(303.4125, 149.49, 'X[24] <= 0.088\ngini = 0.023\nsamples = 173\nvalue = [2, 171]'),
 Text(292.95, 122.31, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(313.875, 122.31, 'X[26] <= 0.18\ngini = 0.012\nsamples = 172\nvalue = [1, 171]'),
 Text(303.4125, 95.13, 'X[6] <= 0.053\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(292.95, 67.94999999999999, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(313.875, 67.94999999999999, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
```

```
 Text(324.33750000000003, 95.13, 'gini = 0.0\nsamples = 168\nvalue = [0, 168]')]
```