Experiment-8

8) Consider the Breast canser predictions dataset and perform the following operations

a) Build a model using Decision Tree algorithm

b) Test the model using Different matrices

c) compare the Decision tree and Random forest algorithm

a) Build a model using Decision Tree algorithm

In [1]:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

In [2]:

```
df=pd.read_csv('C:/Users/Guru Kiran/All CSV files/Breast_Cancer_data.csv')
df.head()
```

Out[2]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoo
0	842302	M	17.99	10.38	122.80	1001.0	
1	842517	M	20.57	17.77	132.90	1326.0	
2	84300903	M	19.69	21.25	130.00	1203.0	
3	84348301	M	11.42	20.38	77.58	386.1	
4	84358402	M	20.29	14.34	135.10	1297.0	

5 rows × 33 columns

In [3]:

```
x=df.drop('diagnosis',axis=1)
y=df['diagnosis']
```

In [4]:

```
## splitting data into train & test
x_test,x_train,y_test,y_train=train_test_split(x,y,test_size=0.2,random_state=10
x_test.shape,x_train.shape,y_test.shape,y_train.shape
```

Out[4]:

```
((455, 32), (114, 32), (455,), (114,))
```

In [5]:

```
## Build a model using Decision Tree


dtc=DecisionTreeClassifier()
model=dtc.fit(x_train,y_train)
dtc_y_pred=model.predict(x_test)
```

b) Test the model using Different matrices

In [6]:

```
print("Accuracy:",metrics.accuracy_score(y_test, dtc_y_pred))
```

```
Accuracy: 0.9362637362637363
```

In [7]:

```
## confusion matrix
confusion_matrix(y_test, dtc_y_pred)
```

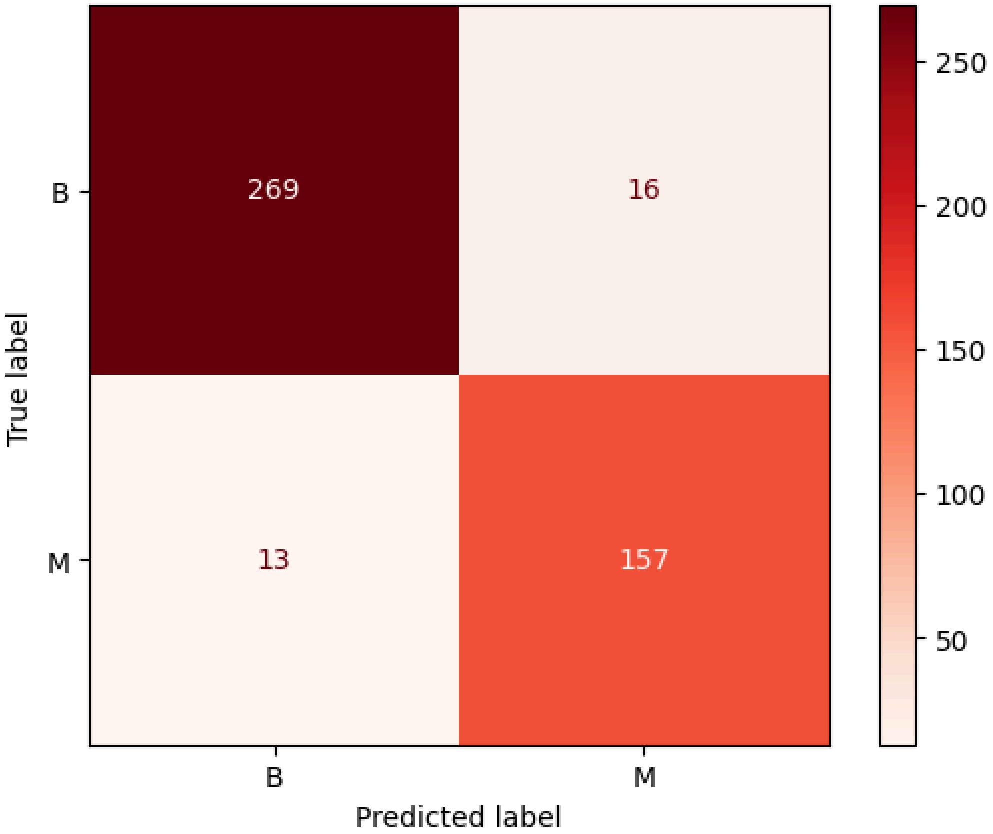
Out[7]:

```
array([[269,  16],
       [ 13, 157]])
```

In [8]:

```
## confusion_matrix show in graph

import matplotlib.pyplot as plt
from sklearn.metrics import ConfusionMatrixDisplay
ConfusionMatrixDisplay.from_estimator(model, x_test, y_test, cmap=plt.cm.Reds)
plt.show()
```



In [9]:

```
## To find the Accuracy,precision,recall,F1 score

from sklearn.metrics import classification_report
print(classification_report(y_test, dtc_y_pred))
```

```
              precision    recall  f1-score   support

           B       0.95      0.94      0.95       285
           M       0.91      0.92      0.92       170

    accuracy                           0.94       455
   macro avg       0.93      0.93      0.93       455
weighted avg       0.94      0.94      0.94       455
```

c) compare the Decision tree and Random forest algorithm

In [10]:

```
rfc=DecisionTreeClassifier()
model1=rfc.fit(x_train,y_train)
y_pred=model1.predict(x_test)
```

In [11]:

```
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.9230769230769231
```

In [12]:

```
from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           B       0.97      0.91      0.94       285
           M       0.86      0.95      0.90       170

    accuracy                           0.92       455
   macro avg       0.91      0.93      0.92       455
weighted avg       0.93      0.92      0.92       455
```

In [13]:

```
print("Accuracy:",metrics.accuracy_score(y_test, dtc_y_pred))
```

```
Accuracy: 0.9362637362637363
```

In [14]:

```
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.9230769230769231
```