In [2]:
```
import pandas as pd
```

In [3]:
```
df=pd.read_csv('../pr3/Admission_Predict.csv')
```

In [4]:
```
df.columns
```

Out[4]:
```
Index(['Serial No.', 'GRE Score', 'TOEFL Score', 'University Rating', 'SOP',
       'LOR ', 'CGPA', 'Research', 'Chance of Admit '],
      dtype='object')
```

removing right space

In [6]:
```
df.columns=df.columns.str.rstrip()
```

In [7]:
```
df.columns
```

Out[7]:
```
Index(['Serial No.', 'GRE Score', 'TOEFL Score', 'University Rating', 'SOP',
       'LOR', 'CGPA', 'Research', 'Chance of Admit'],
      dtype='object')
```

indexing by loc function

In [8]:
```
df.loc[df['Chance of Admit']>=0.80,'Chance of Admit']=1
```

In [9]:
```
df.loc[df['Chance of Admit']<0.80,'Chance of Admit']=0
```

In [10]:
```
df['Chance of Admit']
```

Out[10]:
```
0      1.0
1      0.0
2      0.0
3      1.0
4      0.0
      ...
395    1.0
396    1.0
397    1.0
398    0.0
399    1.0
Name: Chance of Admit, Length: 400, dtype: float64
```

In [13]:
```
df=df.drop('Serial No.',axis=1)#dropping serial no column as it it of no use .
```

In [14]:
```
df
```

Out[14]:

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	337	118	4	4.5	4.5	9.65	1	1.0
1	324	107	4	4.0	4.5	8.87	1	0.0
2	316	104	3	3.0	3.5	8.00	1	0.0
3	322	110	3	3.5	2.5	8.67	1	1.0
4	314	103	2	2.0	3.0	8.21	0	0.0
...	...	...	...	...	...	...	...	...
395	324	110	3	3.5	3.5	9.04	1	1.0
396	325	107	3	3.0	3.5	9.11	1	1.0
397	330	116	4	5.0	4.5	9.45	1	1.0
398	312	103	3	3.5	4.0	8.78	0	0.0
399	333	117	4	5.0	4.0	9.66	1	1.0

400 rows × 8 columns

In [15]:
```
x=df.iloc[:,0:7]  #selecting all rows and 0-6 columns value to x
y=df.iloc[:,7]    #selecting all rows and 7th column value to y(chance of admit value)
```

In [16]:
```
from sklearn.model_selection import train_test_split
```

In [17]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25,random_state=0)
```

In [18]:
```
#from sklearn.model_selection import DecisionTreeClassifier
```

```
---------------------------------------------------------------------------
ImportError                               Traceback (most recent call last)
Input In [18], in <cell line: 1>()
----> 1 from sklearn.model_selection import DecisionTreeClassifier

ImportError: cannot import name 'DecisionTreeClassifier' from 'sklearn.model_selectio
n' (C:\Users\Madhuri Wavhal\AppData\Local\Programs\Python\Python310\lib\site-packages
\sklearn\model_selection\__init__.py)
```

In [19]:
```
from sklearn.tree import DecisionTreeClassifier
```

In [22]:
```
model=DecisionTreeClassifier(criterion="entropy" ,max_depth=4)
```

In [23]:
```
model.fit(x_train,y_train)
y_pred=model.predict(x_test)
```

In [24]:
```
y_pred
```

Out[24]:
```
array([0., 0., 0., 0., 0., 0., 0., 0., 1., 1., 0., 1., 0., 0., 1., 0., 0.,
       1., 0., 0., 1., 1., 0., 0., 1., 0., 0., 0., 1., 0., 0., 0., 0., 0.,
       1., 0., 0., 1., 0., 1., 0., 0., 0., 1., 1., 0., 0., 0., 0., 0., 0.,
       1., 0., 1., 0., 0., 0., 0., 0., 1., 1., 0., 0., 0., 1., 1., 0., 1.,
       1., 0., 0., 1., 1., 0., 1., 0., 0., 0., 1., 0., 0., 0., 0., 1., 0.,
       1., 0., 0., 1., 1., 0., 1., 0., 0., 1., 0., 0., 0., 0., 0.])
```

In [25]:
```
from sklearn.metrics import confusion_matrix
```

In [26]:
```
matrix=confusion_matrix(y_test,y_pred,labels=[0.0,1.0])
```

In [27]:
```
x_train.shape
x_test.shape
```

Out[27]:
```
(100, 7)
```

In [28]:
```
matrix
```

Out[28]:
```
array([[66,  5],
       [ 2, 27]], dtype=int64)
```

In [29]:
```
from sklearn.metrics import accuracy_score
```

In [30]:
```
acc=accuracy_score(y_test,y_pred) #calculating accuracy score
```

In [31]:
```
print(acc)
```

```
0.93
```

In [32]:
```
from sklearn.metrics import classification_report
```

In [33]:
```
cr=classification_report(y_test,y_pred)
```

In [34]:
```
print(cr)
```

```
              precision    recall  f1-score   support

         0.0       0.97      0.93      0.95        71
         1.0       0.84      0.93      0.89        29

    accuracy                           0.93       100
   macro avg       0.91      0.93      0.92       100
weighted avg       0.93      0.93      0.93       100
```

In [ ]:
```

```