1. Take fruits.csv and perform classification based on Decision tree. What is the success rate?

In [1]:

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

In [2]:

```
dataset = pd.read_csv("fruits.csv")
dataset.head()
```

Out[2]:

	Index	Weight	size	Class
0	0	69	4.39	orange
1	1	68	4.21	orange
2	2	65	4.28	orange
3	3	72	5.85	apple
4	4	67	4.74	orange

In [3]:

```
X = dataset.iloc[:, 1:-1].values
Y = dataset.iloc[:, -1].values
```

In [17]:

```
y = []
for i in Y:
    if i == 'orange':
        y.append(0)
    else:
        y.append(1)
print(y)
```

```
[0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0]
```

In [5]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size = 0.25, random_state = 0)
```

In [6]:

```
print(X_train)
print(y_train)
print(X_test)
print(y_test)
```

```
[[67.    4.8 ]
 [70.    5.65]
 [68.    4.8 ]
 [67.    4.74]
 [65.    4.28]
 [73.    5.5 ]
 [74.    5.75]
 [65.    4.8 ]
 [75.    5.85]
 [64.    4.9 ]
 [70.    5.3 ]
 [72.    5.85]
 [69.    4.39]
 [75.    5.85]
 [74.    5.5 ]]
[0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1]
[[65.    4.75]
 [68.    4.21]
 [66.    4.8 ]
 [74.    5.9 ]
 [73.    5.5 ]]
[0, 0, 0, 1, 1]
```

In [7]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

In [8]:

```
print(X_train)
```

```
[[-0.7778427  -0.7467675 ]
 [ 0.03617873  0.85076364]
 [-0.50650222 -0.7467675 ]
 [-0.7778427  -0.8595344 ]
 [-1.32052365 -1.72408066]
 [ 0.85020016  0.56884638]
 [ 1.12154064  1.03870848]
 [-1.32052365 -0.7467675 ]
 [ 1.39288112  1.22665332]
 [-1.59186413 -0.55882266]
 [ 0.03617873  0.1929567 ]
 [ 0.57885968  1.22665332]
 [-0.23516175 -1.51734134]
 [ 1.39288112  1.22665332]
 [ 1.12154064  0.56884638]]
```

In [9]:

```
print(X_test)
```

```
[[-1.32052365 -0.84073992]
 [-0.50650222 -1.85564205]
 [-1.04918318 -0.7467675 ]
 [ 1.12154064  1.32062574]
 [ 0.85020016  0.56884638]]
```

In [10]:

```
from sklearn.tree import DecisionTreeClassifier
classifier = DecisionTreeClassifier(criterion = 'entropy', random_state = 0)
classifier.fit(X_train, y_train)
```

Out[10]:

```
DecisionTreeClassifier(criterion='entropy', random_state=0)
```

In [11]:

```
print(classifier.predict(sc.transform([[72,5.85]])))
```

```
[1]
```

In [16]:

```
y_pred = classifier.predict(X_test)
print(y_test)
print()
print(y_pred)
```

```
[0, 0, 0, 1, 1]

[0 0 0 1 1]
```

In [18]:

```
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
print(cm)
print("Success rate: ",100*accuracy_score(y_test, y_pred))
```

```
[[3 0]
 [0 2]]
Success rate:  100.0
```

In [ ]:

```

```