In [1]:

```
import pandas as pd
import numpy as np
from sklearn import datasets
```

```
C:\Users\Navaneetha\anaconda3\lib\site-packages\scipy\__init__.py:138: Use
rWarning: A NumPy version >=1.16.5 and <1.23.0 is required for this versio
n of SciPy (detected version 1.23.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} i
s required for this version of "
```

In [2]:

```
iris=datasets.load_iris()
```

In [3]:

```
x=iris.data
y=iris.target
```

In [4]:

```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.6,random_state=1)
```

Desicion Tree

In [5]:

```
from sklearn.tree import DecisionTreeClassifier

dt=DecisionTreeClassifier()
dt=dt.fit(x_train,y_train)
```

In [6]:

```
y_pred=dt.predict(x_test)
y_pred
```

Out[6]:

```
array([0, 1, 1, 0, 2, 1, 2, 0, 0, 2, 1, 0, 2, 1, 1, 0, 1, 1, 0, 0, 1, 1,
       2, 0, 2, 1, 0, 0, 1, 2, 1, 2, 1, 2, 2, 0, 1, 0, 1, 2, 2, 0, 1, 2,
       1, 2, 0, 0, 0, 1, 0, 0, 2, 2, 2, 2, 2, 1, 2, 1])
```

In [7]:

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

In [8]:

```
print(confusion_matrix(y_test,y_pred))
```

```
[[19  0  0]
 [ 0 20  1]
 [ 0  1 19]]
```

In [9]:

```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        19
           1       0.95      0.95      0.95        21
           2       0.95      0.95      0.95        20

    accuracy                           0.97        60
   macro avg       0.97      0.97      0.97        60
weighted avg       0.97      0.97      0.97        60
```

Random Forest

In [10]:

```
from sklearn.ensemble import RandomForestClassifier

rf=RandomForestClassifier()
rf=dt.fit(x_train,y_train)
```

In [11]:

```
y_pred=rf.predict(x_test)
y_pred
```

Out[11]:

```
array([0, 1, 1, 0, 2, 1, 2, 0, 0, 2, 1, 0, 2, 1, 1, 0, 1, 1, 0, 0, 1, 1,
       2, 0, 2, 1, 0, 0, 1, 2, 1, 2, 1, 2, 2, 0, 1, 0, 1, 2, 2, 0, 1, 2,
       1, 2, 0, 0, 0, 1, 0, 0, 2, 2, 2, 2, 2, 1, 2, 1])
```

In [12]:

```
print(confusion_matrix(y_test,y_pred))
```

```
[[19  0  0]
 [ 0 20  1]
 [ 0  1 19]]
```

In [13]:

```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        19
           1       0.95      0.95      0.95        21
           2       0.95      0.95      0.95        20

    accuracy                           0.97        60
   macro avg       0.97      0.97      0.97        60
weighted avg       0.97      0.97      0.97        60
```

In [ ]: