11. Assignment on Classification using KNN: Buid an application to classify a given flower into its specie using KNN (Use Iris dataset from sklearn library)

In [1]:
```
# Import libraries
import pandas as pd
from sklearn.datasets import load_iris
import numpy as np
```

In [2]:
```
# import iris dataset
irisData = load_iris()
```

In [3]:
```
#assigning iris data to the variable x
X = irisData.data

#assigning iris specieses to y
y = irisData.target

#Integer represent the species : 0-setosa, 1-versicolor, 2-virginica
y
```

Out[3]:
```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

In [4]:
```
# split the data into train and test sets
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
            X, y, test_size = 0.2, random_state=42)
```

In [6]:
```
#Importing KNeighborsClassifier from Scikit-learn.
from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier(n_neighbors=7)

## Train the classifier on the training set
model.fit(X_train, y_train)

#Predicting the testing data using predict() function.
pred=model.predict(X_test)
```

In [7]:
```
#Checking the predicting output with real output which is stored in y_test.
y_test
```

Out[7]:
```
array([1, 0, 2, 1, 1, 0, 1, 2, 1, 1, 2, 0, 0, 0, 0, 1, 2, 1, 1, 2, 0, 2,
       0, 2, 2, 2, 2, 2, 0, 0])
```

In [8]:
```
# Calculate the accuracy of the classifier
from sklearn.metrics import accuracy_score
print('Accuracy Score : ',accuracy_score(y_test, pred))
```

```
Accuracy Score :  0.9666666666666667
```

In [9]:
```
#Classification Report: It consists of precision, recall and F1 score.
from sklearn.metrics import classification_report
print(classification_report(y_test, pred))
```

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

```
              0       1.00      1.00      1.00        10
              1       1.00      0.89      0.94         9
              2       0.92      1.00      0.96        11

    accuracy                           0.97        30
   macro avg       0.97      0.96      0.97        30
weighted avg       0.97      0.97      0.97        30
```

In [10]:
```
from sklearn.metrics import confusion_matrix
print(confusion_matrix(y_test, pred))
```

```
[[10  0  0]
 [ 0  8  1]
 [ 0  0 11]]
```