In [1]:
```
#Import necessary libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
```

In [2]:
```
#Load the dataset
dta=pd.read_csv("C:/Users/hp/Downloads/archive (8).zip")
```

In [3]:
```
#Check the shape of the dataset
dta.shape
```

Out[3]:
```
(150, 6)
```

In [9]:
```
#Drop 'Id' column since it is not needed for classification
dta1=dta.drop(columns=["Id"]);dta1
```

Out[9]:

| | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |
|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |
| **...** | ... | ... | ... | ... | ... |
| **145** | 6.7 | 3.0 | 5.2 | 2.3 | Iris-virginica |
| **146** | 6.3 | 2.5 | 5.0 | 1.9 | Iris-virginica |
| **147** | 6.5 | 3.0 | 5.2 | 2.0 | Iris-virginica |
| **148** | 6.2 | 3.4 | 5.4 | 2.3 | Iris-virginica |
| **149** | 5.9 | 3.0 | 5.1 | 1.8 | Iris-virginica |

150 rows × 5 columns

In [10]:
```
dta1.head()
```

Out[10]:

| | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |
|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |

In [11]:
```
#Replace the species names with numerical labels
dta1["Species"].replace({"Iris-setosa":1,"Iris-versicolor":2,"Iris-virginica":3},inplace=True)
```

In [15]:
```
#Extract the features columns
x=pd.DataFrame(dta1,columns=["SepalLengthCm","SepalWidthCm","PetalLengthCm","PetalWidthCm"]).values
```

In [16]:
```
#Extract column Species
y=dta1.Species.values.reshape(-1,1)
```

In [17]:
```
#Split the dataset into training and testing dataset
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)
```

In [18]:
```
#Define number of neighbors k
k=6
```

In [20]:
```
#Create K Nearest Neighbors classifier with k neighbors
knn=KNeighborsClassifier(n_neighbors=k);knn
```

Out[20]:
```
KNeighborsClassifier(n_neighbors=6)
```

In [21]:
```
#Fit the classifier to training dataset
knn.fit(x_train,y_train)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\_classification.py:198: DataConversionWarning:
A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for
example using ravel().
  return self._fit(X, y)
```

Out[21]:
```
KNeighborsClassifier(n_neighbors=6)
```

In [23]:
```
#Prediction on Test data
y_pred=knn.predict(x_test);y_pred
```

Out[23]:
```
array([2, 1, 3, 2, 2, 1, 2, 3, 2, 2, 3, 1, 1, 1, 1, 2, 3, 2, 2, 3, 1, 3,
       1, 3, 3, 3, 3, 3, 1, 1], dtype=int64)
```

In [24]:
```
#Calculate accuracy
accuracy=metrics.accuracy_score(y_test,y_pred);accuracy
```

Out[24]:
```
1.0
```

In [25]:
```
Accuracy=accuracy*100;Accuracy
```

Out[25]:
```
100.0
```

In [ ]: