importing useful libraries

In [1]:
```
from sklearn.cluster import KMeans
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

importing dataset

In [2]:
```
d=pd.read_csv("Iris.csv")
d
```

Out[2]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

split independent & target variable

In [25]:
```
inputs = d.drop(['Id','Species'],axis='columns')#independent variable
target = d.drop(['Id','SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'],axis='c
olumns')
```

In [26]:
```
inputs
```

Out[26]:

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

150 rows × 4 columns

In [5]:
```
target
```

Out[5]:

	Species
0	Iris-setosa
1	Iris-setosa
2	Iris-setosa
3	Iris-setosa
4	Iris-setosa
...	...
145	Iris-virginica
146	Iris-virginica
147	Iris-virginica
148	Iris-virginica
149	Iris-virginica

150 rows × 1 columns

In [6]:
```
#convert target variable into numeric value
from sklearn.preprocessing import LabelEncoder
```

In [18]:
```
le_Species = LabelEncoder()
```

In [29]:
```
target['Species'] =le_Species.fit_transform(target['Species'])
target.head(70)
```

Out[29]:

	Species
0	0
1	0
2	0
3	0
4	0
...	...
65	1
66	1
67	1
68	1
69	1

70 rows × 1 columns

Decision Tree model implementation

In [36]:
```
from sklearn import tree
```

In [37]:
```
model = tree.DecisionTreeClassifier()
```

In [38]:
```
model.fit(inputs,target)
```

Out[38]:
```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                       max_features=None, max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort=False,
                       random_state=None, splitter='best')
```

value prediction

In [39]:
```
model.predict([[5.1,3.5,1.4,0.2]])#iris setosa
```

Out[39]:
```
array([0])
```

In [40]:
```
model.predict([[6.7,3.0,5.2,2.3]])#iris virginica
```

Out[40]:
```
array([2])
```

In [41]:
```
model.predict([[6.4,2.9,4.3,1.3]])#iris versicolor
```

Out[41]:
```
array([1])
```

In [45]:
```
model.predict([[6.6,3.9,5.3,1.3]])
```

Out[45]:
```
array([2])
```

model accuracy

In [42]:
```
score=model.score(inputs,target)
score
```

Out[42]:
```
1.0
```

Thank you

Submitted By GOutami Dey

The Sparks Foundation

In [ ]:
```
```