Task 4-Exploring Decision Tree Algorithm on Iris Dataset

NAME-SARYU

IMPORTING LIBRARIES

In [2]:

```
import  numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

LOADING DATA

In [3]:

```
os.getcwd()
os.chdir('C:\\Users\\Serving Minds\\Desktop')
df=pd.read_csv('Iris - Iris.csv')
df.head(3)
```

Out[3]:

	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

DATA PREPROCESSING

In [4]:

```
df.isnull().any()
```

Out[4]:

```
Id               False
SepalLengthCm    False
SepalWidthCm     False
PetalLengthCm    False
PetalWidthCm     False
Species          False
dtype: bool
```

In [5]:

```
df.dtypes
```

Out[5]:

```
Id                 int64
SepalLengthCm    float64
SepalWidthCm     float64
PetalLengthCm    float64
PetalWidthCm     float64
Species           object
dtype: object
```

dtype: object

In [6]:

```
df.describe()
```

Out[6]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [4]:

```
df=df.drop(['Id'], axis=1)
df['Species'] = df['Species'].str.replace(r'Iris-', '')
df.head()
```

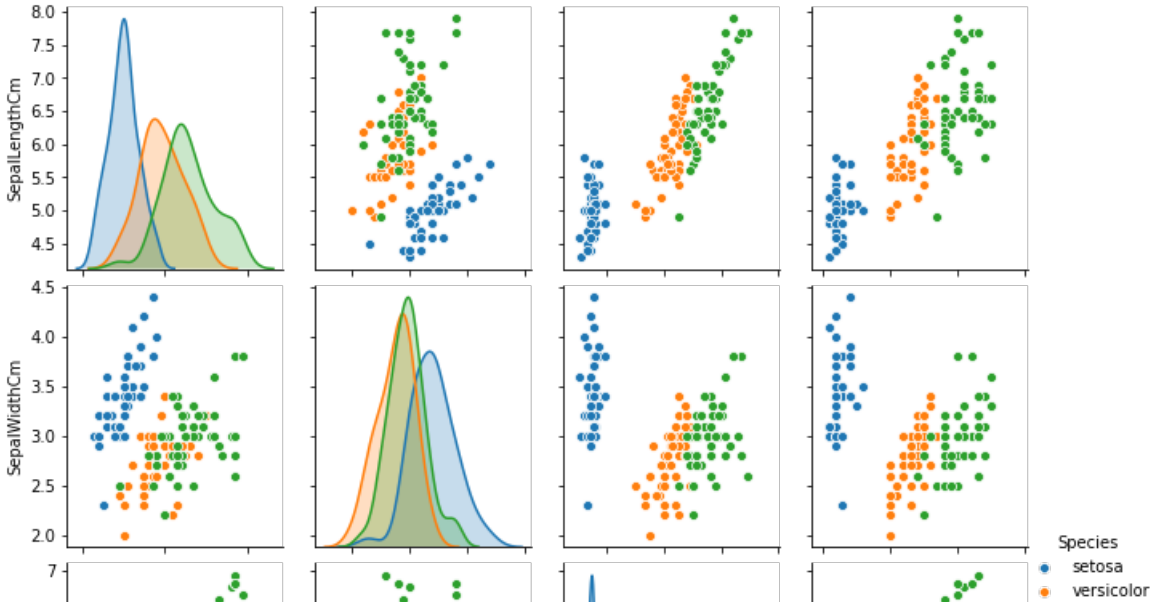
Out[4]:

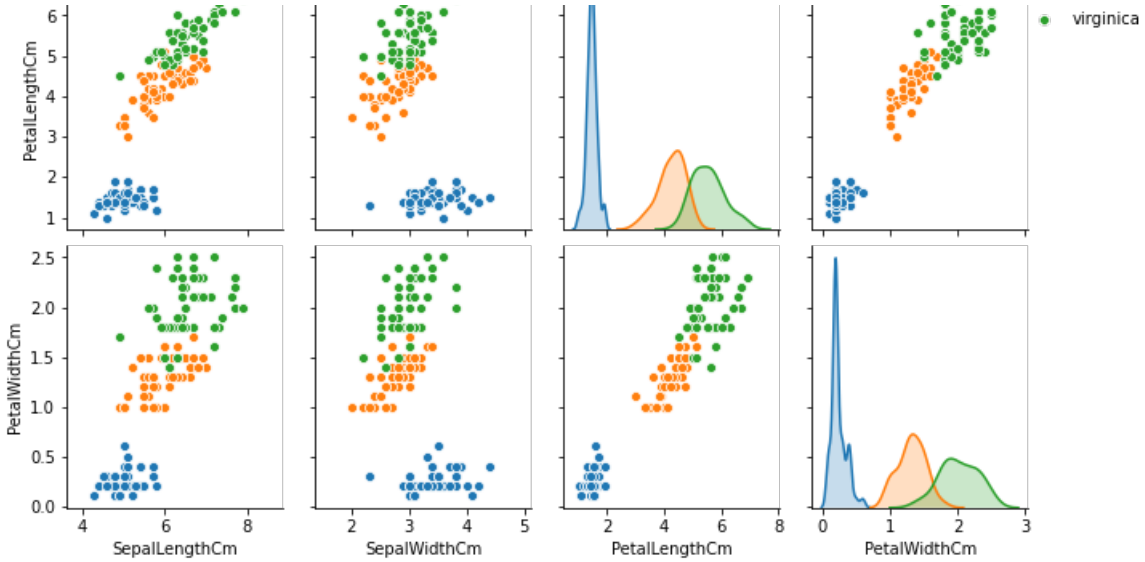
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

VISUALISING PAIR PLOTS

In [7]:

```
g=sns.pairplot(df, hue='Species')
g.fig.set_size_inches(10,10)
```





ENCODING COLUMN

In [8]:

```
# Import label encoder
from sklearn import preprocessing

# label_encoder object knows how to understand word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in column 'species'.
df['Species']= label_encoder.fit_transform(df['Species'])
df.head()
```

Out[8]:

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

SPLITTING DATA INTO TEST AND TRAINING SET

In [9]:

```
x=df.iloc[:,:-1].values
y=df.iloc[:,-1].values

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=0)
```

DECISION TREE ALGORITHM

In [12]:

```
from sklearn.tree import DecisionTreeClassifier
classifier=DecisionTreeClassifier(criterion='entropy',random_state=0)
classifier.fit(X_train,Y_train)
from sklearn import tree
y_pred=classifier.predict(X_test)
df2=pd.DataFrame({'ACTUAL':Y_test,'PREDICTED':y_pred})
```

```
df2.head()
```

Out[12]:

	ACTUAL	PREDICTED
0	2	2
1	1	1
2	0	0
3	2	2
4	0	0

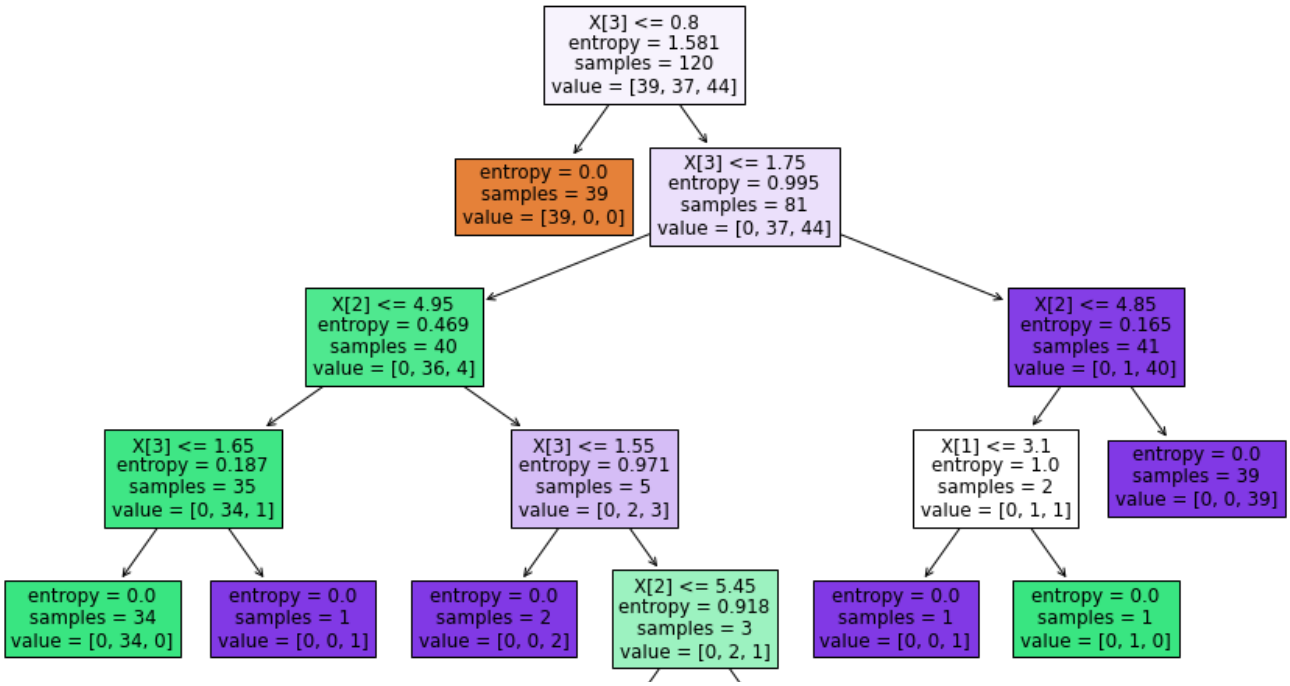
VISUALISING DECISION TREE

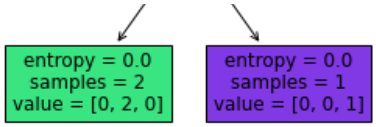
In [13]:

```
fig=plt.figure(figsize=(15,10))
tree.plot_tree(classifier,filled=True)
```

Out[13]:

```
[Text(418.5, 498.3, 'X[3] <= 0.8\nentropy = 1.581\nsamples = 120\nvalue = [39, 37, 44]'),
 Text(354.11538461538464, 407.70000000000005, 'entropy = 0.0\nsamples = 39\nvalue = [39, 0, 0]'),
 Text(482.8846153846154, 407.70000000000005, 'X[3] <= 1.75\nentropy = 0.995\nsamples = 81\nvalue =
[0, 37, 44]'),
 Text(257.53846153846155, 317.1, 'X[2] <= 4.95\nentropy = 0.469\nsamples = 40\nvalue = [0, 36, 4]'
),
 Text(128.76923076923077, 226.5, 'X[3] <= 1.65\nentropy = 0.187\nsamples = 35\nvalue = [0, 34, 1]'
),
 Text(64.38461538461539, 135.89999999999998, 'entropy = 0.0\nsamples = 34\nvalue = [0, 34, 0]'),
 Text(193.15384615384616, 135.89999999999998, 'entropy = 0.0\nsamples = 1\nvalue = [0, 0, 1]'),
 Text(386.3076923076923, 226.5, 'X[3] <= 1.55\nentropy = 0.971\nsamples = 5\nvalue = [0, 2, 3]'),
 Text(321.9230769230769, 135.89999999999998, 'entropy = 0.0\nsamples = 2\nvalue = [0, 0, 2]'),
 Text(450.69230769230774, 135.89999999999998, 'X[2] <= 5.45\nentropy = 0.918\nsamples = 3\nvalue =
[0, 2, 1]'),
 Text(386.3076923076923, 45.299999999999955, 'entropy = 0.0\nsamples = 2\nvalue = [0, 2, 0]'),
 Text(515.0769230769231, 45.299999999999955, 'entropy = 0.0\nsamples = 1\nvalue = [0, 0, 1]'),
 Text(708.2307692307693, 317.1, 'X[2] <= 4.85\nentropy = 0.165\nsamples = 41\nvalue = [0, 1,
40]'),
 Text(643.8461538461538, 226.5, 'X[1] <= 3.1\nentropy = 1.0\nsamples = 2\nvalue = [0, 1, 1]'),
 Text(579.4615384615385, 135.89999999999998, 'entropy = 0.0\nsamples = 1\nvalue = [0, 0, 1]'),
 Text(708.2307692307693, 135.89999999999998, 'entropy = 0.0\nsamples = 1\nvalue = [0, 1, 0]'),
 Text(772.6153846153846, 226.5, 'entropy = 0.0\nsamples = 39\nvalue = [0, 0, 39]')]
```




entropy = 0.0
samples = 2
value = [0, 2, 0]
entropy = 0.0
samples = 1
value = [0, 0, 1]