Exercise-5

Python Program to implement Decision Tree.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.tree import plot_tree

# Input: Dataset
iris = sns.load_dataset('iris')

target = iris['species']
iris1 = iris.copy()
iris1 = iris1.drop('species', axis =1)

# Defining the attributes
X = iris1
le = LabelEncoder()                # Label Encoding
target = le.fit_transform(target)
Y = target

# Dividing into test and training sets
X_train, X_test, Y_train, Y_test = train_test_split(X , Y, test_size = 0.3)

# Create Decision Tree Classifier object
dtree = DecisionTreeClassifier(criterion = "entropy", max_depth = 5)

# Train the model using the training sets
dtree.fit(X_train,Y_train)

# Make predictions using the testing set
Y_pred = dtree.predict(X_test)
```

Exercise-5

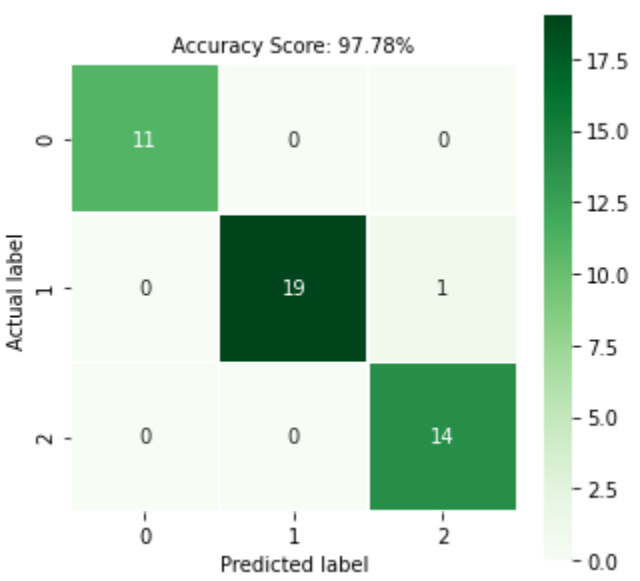
Output for the Decision Tree.

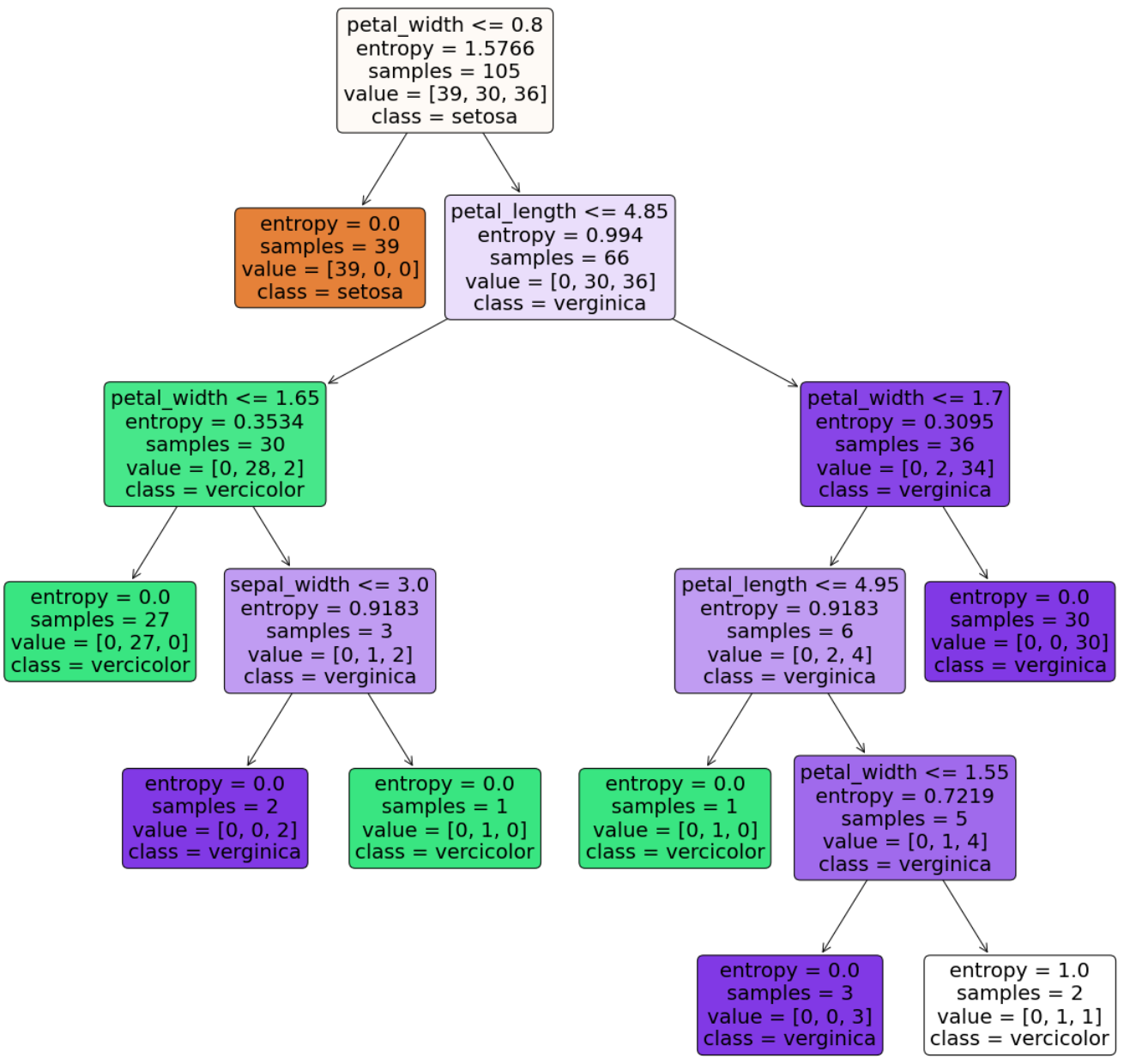
In [3]:
```
# Output: The Classification Report, Confusion Matrix and Decision Tree
print("Classification Report - \n", classification_report(Y_test,Y_pred))
cm = confusion_matrix(Y_test, Y_pred)
plt.figure(figsize=(5,5))
sns.heatmap(data=cm,linewidths=.5, annot=True,square = True,  cmap = 'Greens')
plt.ylabel('Actual label')
plt.xlabel('Predicted label')
all_sample_title = 'Accuracy Score: {0:.2f}%'.format(dtree.score(X_test, Y_test)*100)
plt.title(all_sample_title, size = 10)
plt.figure(figsize = (20,20))
dec_tree = plot_tree(decision_tree=dtree, feature_names = iris.columns,
                     class_names =["setosa", "vercicolor", "verginica"] ,
                     filled = True , precision = 4, rounded = True)
```

```
Classification Report -
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        11
           1       1.00      0.95      0.97        20
           2       0.93      1.00      0.97        14

    accuracy                           0.98        45
   macro avg       0.98      0.98      0.98        45
weighted avg       0.98      0.98      0.98        45
```





On Implementation of Decision Tree Classifier over Iris Dataset with Maximum Depth of 5, we got an accuracy of approximately 98%. The Decision Tree shows the Entropy, Frequency of Classes and the Predicted Class for each node. Also the Information Gain can be computed for a node by subtracting its Entropy from the sum of Entropy of its Child Nodes.