Problem- Create the Decision Tree classifier and visualize it graphically. The purpose is if we feed any new data to this classifier, it would be able to predict the right class accordingly.

Importing Libraries and loading iris dataset

```
import sklearn.datasets as datasets
import pandas as pd

# Loading the iris dataset
iris=datasets.load_iris()

# Forming the iris dataframe
X = pd.DataFrame(iris.data, columns=iris.feature_names)
X.head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
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

```
y=iris.target
y
```

```
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, 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, 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])
```

Describe the dataset

```
X.describe()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
25%	5.100000	2.800000	1.600000	0.300000
75%	6.400000	3.300000	5.100000	1.800000

Training and split the dataset

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)
```

Import the DecisionTreeClassifier and fit the model

```
from sklearn.tree import DecisionTreeClassifier
dtree=DecisionTreeClassifier()
dtree.fit(X_train,y_train)
print('Decision Tree Classifer Created Successfully')
```

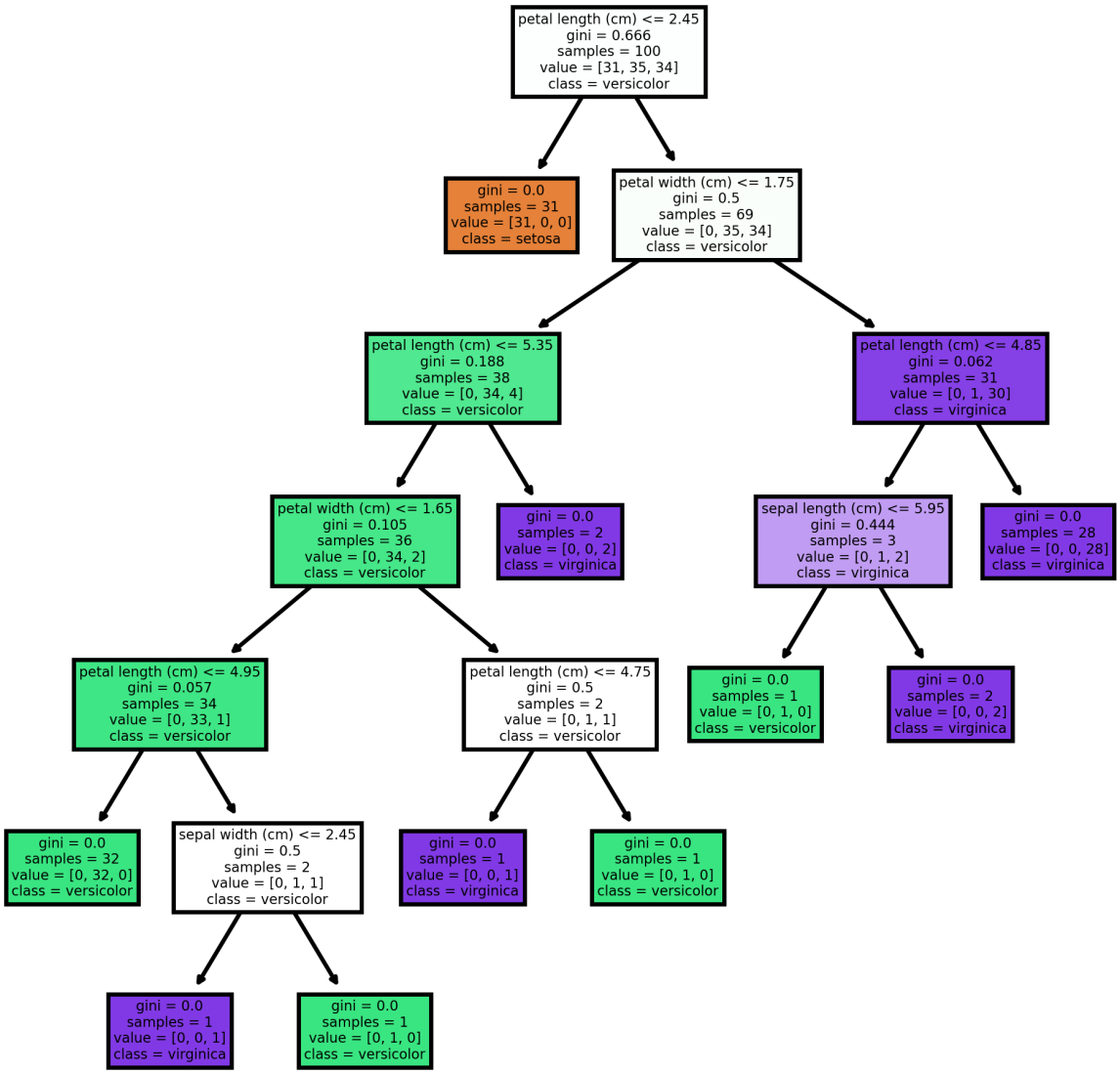
```
Decision Tree Classifer Created Successfully
```

```
y_predict = dtree.predict(X_test)
y_predict
```

```
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, 0, 0, 1, 0, 0, 2, 1, 0, 0, 0, 2, 1, 1, 0,
       0, 1, 1, 2, 1, 2])
```

Constructing Decision Tree using matplotlib.pyplot

```
import matplotlib.pyplot as plt
from sklearn import tree
a=['sepal length (cm)','sepal width (cm)','petal length (cm)','petal width (cm)']
b=['setosa','versicolor','virginica']
fig, axes = plt.subplots(nrows = 1, ncols = 1, figsize = (5,5), dpi = 500)
tree.plot_tree(dtree, feature_names = a, class_names = b, filled = True);
```


petal length (cm) <= 2.45
gini = 0.666
samples = 100
value = [31, 35, 34]
class = versicolor
gini = 0.0
samples = 31
value = [31, 0, 0]
class = setosa
petal width (cm) <= 1.75
gini = 0.5
samples = 69
value = [0, 35, 34]
class = versicolor
petal length (cm) <= 5.35
gini = 0.188
samples = 38
value = [0, 34, 4]
class = versicolor
petal length (cm) <= 4.85
gini = 0.062
samples = 31
value = [0, 1, 30]
class = virginica
petal width (cm) <= 1.65
gini = 0.105
samples = 36
value = [0, 34, 2]
class = versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = virginica
sepal length (cm) <= 5.95
gini = 0.444
samples = 3
value = [0, 1, 2]
class = virginica
gini = 0.0
samples = 28
value = [0, 0, 28]
class = virginica
petal length (cm) <= 4.95
gini = 0.057
samples = 34
value = [0, 33, 1]
class = versicolor
petal length (cm) <= 4.75
gini = 0.5
samples = 2
value = [0, 1, 1]
class = versicolor
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = virginica
gini = 0.0
samples = 32
value = [0, 32, 0]
class = versicolor
sepal width (cm) <= 2.45
gini = 0.5
samples = 2
value = [0, 1, 1]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor