

```
In [1]: # in this lab, we will use DT to predict the output class for the Iris flower.

# import the required libraries.
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_text # this is used to show the DT Tree
import pandas as pd # to read the dataset
from sklearn.model_selection import train_test_split
from sklearn import metrics # this is to find the accuracy score
```

```
In [2]: # Task 1: Read the Iris dataset with the following columns and show the first 5 rows
# 'sepal length', 'sepal width', 'petal length', 'petal width', 'target'
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
```

```
In [3]: # Task 2: extract the features which are the first 4 columns and save in X
# the y variable should be the last column
```

```
In [4]: # Task 3: Split the dataset (both X and y) into training set and test set
# using the train_test_split() function
```

```
In [5]: #Task 4: create and train the DT Model with maximum tree depth = 3.
# more information about the sklearn DecisionTreeClassifier can be found here
# https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

# print the tree structure using the export_text() function, more about this function is found here
# https://scikit-learn.org/stable/modules/generated/sklearn.tree.export_text.html
```

```
In [6]: #Task 5: use the DT model to predict the output for the test dataset
# and print the accuracy of the predicted values

# print the accuracy
```

```
In [9]: # Find and print the features importance using
# the feature_importances_ attribute
feature_imp = pd.Series(Model_DT.feature_importances_,
                        index = features).sort_values(ascending=False)
print(Model_DT.feature_importances_)
print (feature_imp)
```

```
[0.          0.          0.96445432 0.03554568]
petal length    0.964454
petal width     0.035546
sepal width     0.000000
sepal length    0.000000
dtype: float64
```

```
In [7]: #Task 6: ignore the least important feature from your dataset
# and use the DT model on the new dataset.
```