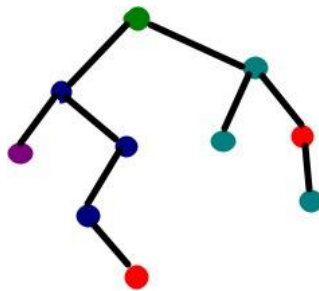


## **EX.NO : 12**

### **IMPLEMENTATION OF DECISION TREE CLASSIFICATION TECHNIQUES**

[Decision Tree](#) is one of the most powerful and popular algorithm. Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.



#### **AIM:**

To implement a decision tree classification technique for gender classification using python.

#### **EXPLANATION:**

- Import tree from sklearn.
- Call the function DecisionTreeClassifier() from tree
- Assign values for X and Y.
- Call the function predict for Predicting on the basis of given random values for each given feature.
- Display the output.

### **CODE :**

```
[ ] from sklearn import tree
    #Using DecisionTree classifier for prediction
    clf = tree.DecisionTreeClassifier()

    #Here the array contains three values which are height,weight and shoe size
    X = [[181, 80, 91], [182, 90, 92], [183, 100, 92], [184, 200, 93], [185, 300, 94], [186, 400, 95],
    [187, 500, 96], [189, 600, 97], [190, 700, 98], [191, 800, 99], [192, 900, 100], [193, 1000, 101]]
    Y = ['male', 'male', 'female', 'male', 'female', 'male', 'female', 'male', 'female', 'male', 'female',
    'male' ]
    clf = clf.fit(X, Y)

    #Predicting on basis of given random values for each given feature
    predictionf = clf.predict([[181, 80, 91]])
    predictionm = clf.predict([[183, 100, 92]])

    #Printing final prediction
    print(predictionf)
    print(predictionm)
```

### **OUTPUT :**

```
⇒ ['male']
   ['female']
```

### **RESULT :**

Thus, the implementation of Decision Tree Classification Techniques is successfully executed and the output is verified.