

**EX.NO :13**

**DATE :**

**IMPLEMENTATION OF DECISION TREE CLASSIFICATION  
TECHNIQUES**

**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.

**SOURCE CODE:**

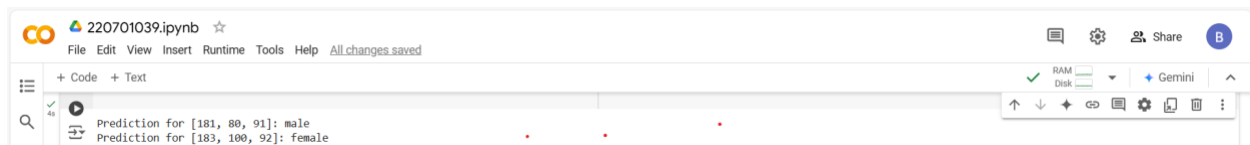
```
from sklearn import tree
#Using DecisionTree classifier for prediction
clf = tree.DecisionTreeClassifier()

#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:**



The image shows a Jupyter Notebook interface with a single code cell. The cell contains two lines of text: "Prediction for [181, 80, 91]: male" and "Prediction for [183, 100, 92]: female". The interface includes a top menu bar with options like File, Edit, View, Insert, Runtime, Tools, and Help. On the right side, there are icons for RAM, Disk, and Gemini, along with a share button. The notebook title is "220701039.ipynb".

```
Prediction for [181, 80, 91]: male  
Prediction for [183, 100, 92]: female
```

**RESULT:**