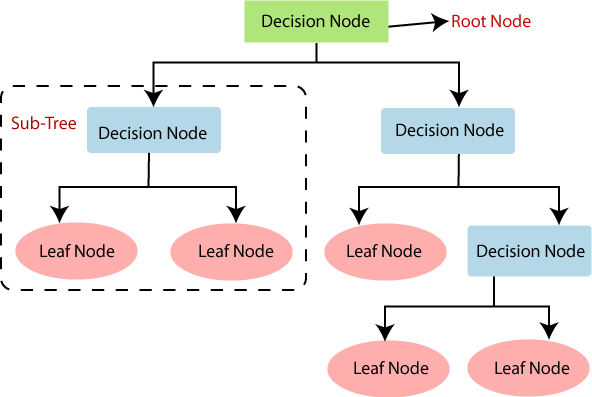
# Decision Tree

* Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems
* In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
* The decisions or the test are performed on the basis of features of the given dataset.
* It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.
* In order to build a tree, we use the **CART algorithm,** which stands for **Classification and Regression Tree algorithm.**
* A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.
* Below diagram explains the general structure of a decision tree:



* **Root Node:** Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
* **Leaf Node:** Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
* **Splitting:** Splitting is the process of dividing the decision node/root node into sub- nodes according to the given conditions.
* **Branch/Sub Tree:** A tree formed by splitting the tree.
* **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
* **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.

**How does the Decision Tree algorithm Work?**

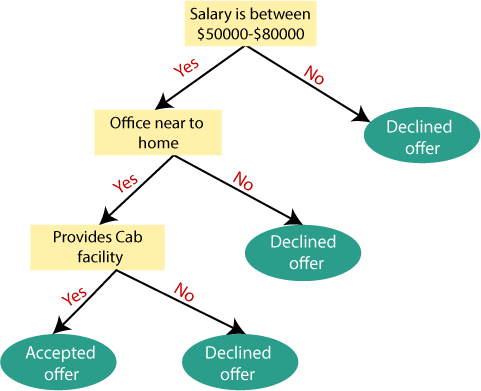
**Step-1:** Begin the tree with the root node, says S, which contains the complete dataset. **Step-2:** Find the best attribute in the dataset using **Attribute Selection Measure (ASM). Step-3:** Divide the S into subsets that contains possible values for the best attributes.

**Step-4:** Generate the decision tree node, which contains the best attribute.

**Step-5:** Recursively make new decision trees using the subsets of the dataset created in step

-3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

**Example:** Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not. So, to solve this problem, the decision tree starts with the root node (Salary attribute by ASM). The root node splits further into the next decision node (distance from the office) and one leaf node based on the corresponding labels. The next decision node further gets split into one decision node (Cab facility) and one leaf node. Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer). Consider the below diagram:



# Attribute Selection Measures

While implementing a Decision tree, the main issue arises that how to select the best attribute for the root node and for sub-nodes. So, to solve such problems there is a technique which is called as **Attribute selection measure or ASM.** By this measurement, we can easily select the best attribute for the nodes of the tree. There are two popular techniques for ASM, which are:

# Information Gain:

* + It calculates how much information a feature provides us about a class.
  + According to the value of information gain, we split the node and build the decision tree.
  + A decision tree algorithm always tries to maximize the value of information gain, and a node/attribute having the highest information gain is split first
  + It can be calculated using the below formula

Information Gain= Entropy(S)- [(Weighted Avg) \*Entropy (each feature)

Were,

**Entropy** is an information theory metric that measures the impurity or uncertainty in a group of observations

# Gini Index:

* + Gini index is a measure of impurity or purity used while creating a decision tree in the CART (Classification and Regression Tree) algorithm.
  + An attribute with the low Gini index should be preferred as compared to the high Gini index.
  + It only creates binary splits, and the CART algorithm uses the Gini index to create binary splits.
  + Gini index can be calculated using the below formula:

Gini Index= 1- ∑ P 2

j j

# Pruning:

Pruning is a process of deleting the unnecessary nodes from a tree in order to get the optimal decision tree.

A too-large tree increases the risk of overfitting, and a small tree may not capture all the important features of the dataset. Therefore, a technique that decreases the size of the learning tree without reducing accuracy is known as Pruning.

**Hyperparameter Tuning:**

Hyperparameters in decision tree are:

1. Max\_depth  Indicates how deep decision tree can be, deeper the tree more splits it has and captures more information about data, A tree overfits for larger depth values
2. Min\_samples\_split  minimum number of samples required to split an internal node
3. Min\_samples\_leaf  minimum number of samples required to be at a leaf node
4. Max\_features  No of features to consider when looking for the best split
5. Criterion  supported criterion are ‘gini’ and ‘entropy’

**Random forest**

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

