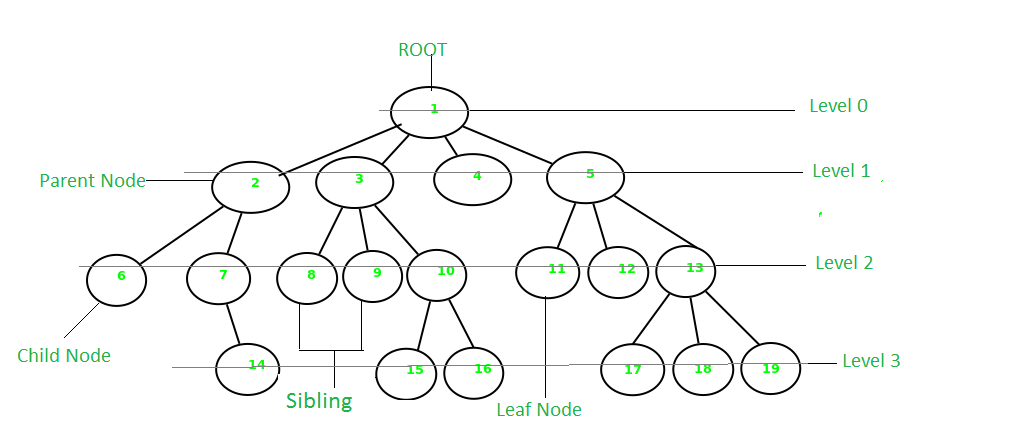
**Tree Data Structure**

A tree is *non-linear and hierarchical tree data structure* consisting of a collection of nodes such that each node of the tree stores a value, a list of references to nodes (the “children”).

Link: <https://www.geeksforgeeks.org/introduction-to-tree-data-structure/>,

<https://www.andrew.cmu.edu/course/15-121/lectures/Trees/trees.html#:~:text=Every%20node%20(excluding%20a%20root,called%20leaves%2C%20or%20external%20nodes>



**Basic Terminology In Tree Data Structure:**

* **Parent Node:** The node which is a predecessor of a node is called the parent node of that node. {2} is the parent node of {6, 7}.
* **Child Node:** The node which is the immediate successor of a node is called the child node of that node. Examples: {6, 7} are the child nodes of {2}.
* **Root Node:** The topmost node of a tree or the node which does not have any parent node is called the root node. {1} is the root node of the tree. A non-empty tree must contain exactly one root node and exactly one path from the root to all other nodes of the tree.
* **Degree of a Node:** The total count of subtrees attached to that node is called the degree of the node. The degree of a leaf node must be 0. The degree of a tree is the maximum degree of a node among all the nodes in the tree. The degree of the node {3} is 3.
* **Leaf Node or External Node:** The nodes which do not have any child nodes are called leaf nodes. {6, 14, 8, 9, 15, 16, 4, 11, 12, 17, 18, 19} are the leaf nodes of the tree.
* **Ancestor of a Node:** Any predecessor nodes on the path of the root to that node are called Ancestors of that node. {1, 2} are the parent nodes of the node {7}
* **Descendant:** Any successor node on the path from the leaf node to that node. {7, 14} are the descendants of the node. {2}.
* **Sibling:** Children of the same parent node are called siblings. {8, 9, 10} are called siblings.
* **Depth of a node:** The count of edges from the root to the node. Depth of node {14} is 3.
* **Height of a node:** The number of edges on the longest path from that node to a leaf. Height of node {3} is 2.
* **Height of a tree:** The height of a tree is the height of the root node i.e the count of edges from the root to the deepest node. The height of the above tree is 3.
* **Level of a node:** The count of edges on the path from the root node to that node. The root node has level 0.
* **Internal node:** A node with at least one child is called Internal Node.
* **Neighbour of a Node:** Parent or child nodes of that node are called neighbors of that node.
* **Subtree:** Any node of the tree along with its descendant