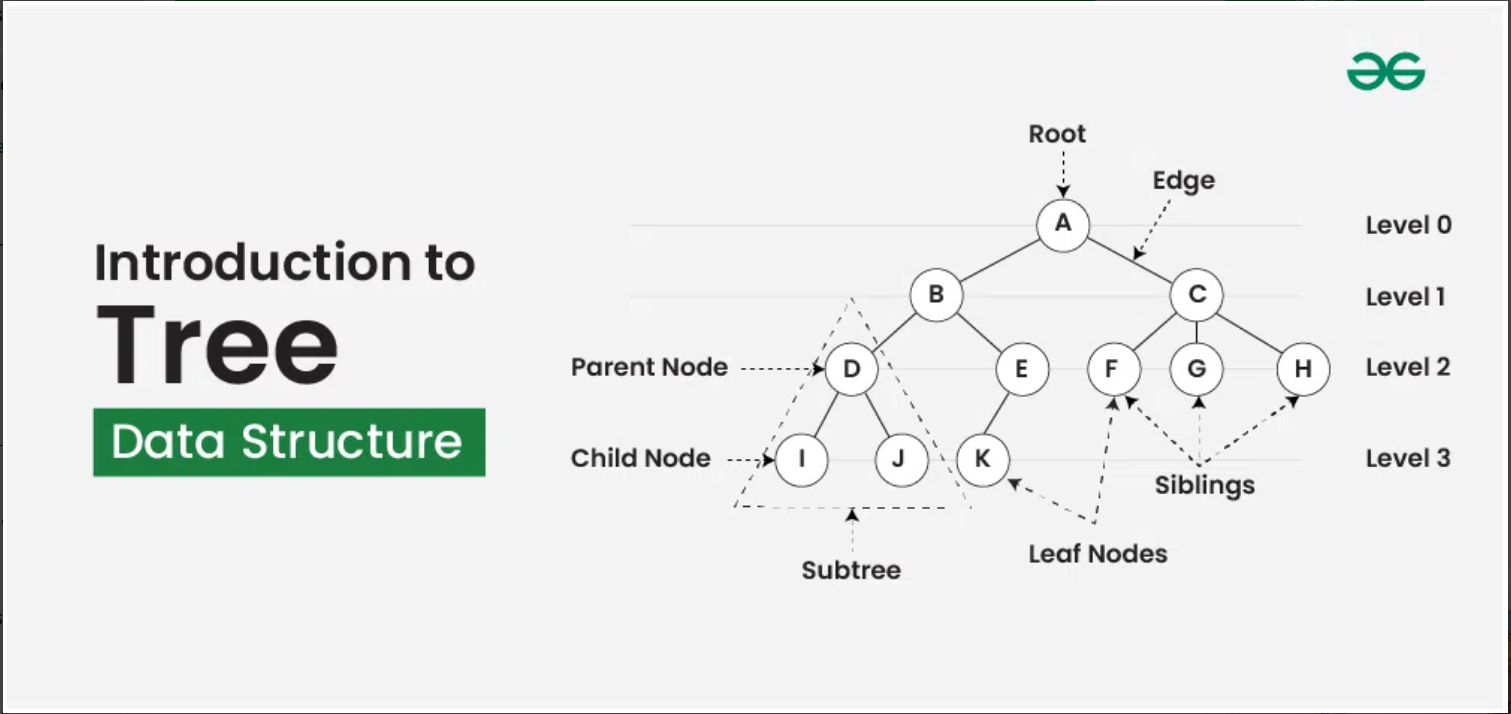
**Introduction to Tree Data Structure**

**Tree data structure** is a hierarchical structure that is used to represent and organize data in the form of parent child relationship. The following are some real world situations which are naturally a tree.

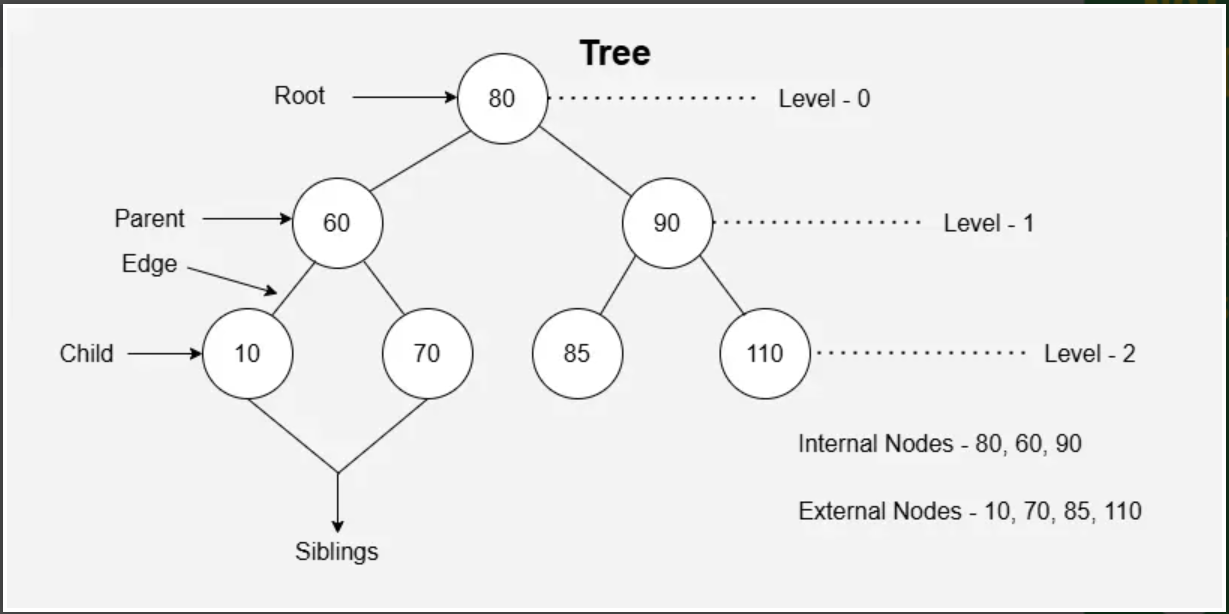
* Folder structure in an operating system.
* Tag structure in an HTML (root tag the as html tag) or XML document.

The topmost node of the tree is called the **root**, and the nodes below it are called the child nodes. Each node can have multiple child nodes, and these child nodes can also have their own child nodes, forming a recursive structure



### Basic Terminologies In Tree Data Structure:

* **Parent Node:** The node which is an immediate predecessor of a node is called the parent node of that node.**{B}** is the parent node of **{D, E}**.
* **Child Node:** The node which is the immediate successor of a node is called the child node of that node. Examples: **{D, E}** are the child nodes of **{B}.**
* **Root Node:** The topmost node of a tree or the node which does not have any parent node is called the root node. {A**}** 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.
* **Leaf Node or External Node:** The nodes which do not have any child nodes are called leaf nodes. **{I, J, K, F, G, H}** 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.**{A,B}** are the ancestor nodes of the node**{E}**
* **Descendant:** A node x is a descendant of another node y if and only if y is an ancestor of x.
* **Sibling:** Children of the same parent node are called siblings.**{D,E}** are called siblings.
* **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.

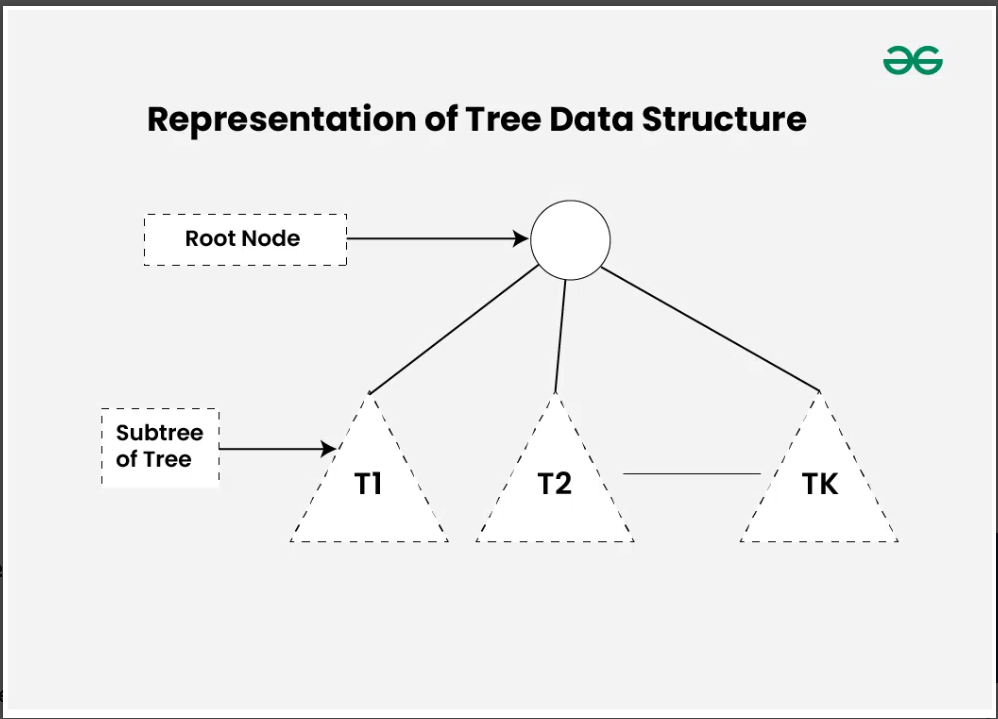


### Why Tree is considered a non-linear data structure?

The data in a tree are not stored in a sequential manner i.e., they are not stored linearly. Instead, they are arranged on multiple levels or we can say it is a hierarchical structure. For this reason, the tree is considered to be a [non-linear data structure](https://www.geeksforgeeks.org/difference-between-linear-and-non-linear-data-structures).

## ****Representation of Tree Data Structure:****

A tree consists of a root node, and zero or more subtrees T1, T2, ... , Tk such that there is an edge from the root node of the tree to the root node of each subtree. Subtree of a node X consists of all the nodes which have node X as the ancestor node.



### Representation of a Node in Tree Data Structure:

A tree can be represented using a collection of nodes. Each of the nodes can be represented with the help of class or structs. Below is the representation of Node in different languages:

### ****Importance for Tree Data Structure:****

One reason to use trees might be because you want to store information that naturally forms a hierarchy. For example, the file system on a computer: The [DOM model](https://www.geeksforgeeks.org/dom-document-object-model/) of an HTML page is also tree where we have html tag as root, head and body its children and these tags, then have their own children.Please refer [Applications, Advantages and Disadvantages of Tree](https://www.geeksforgeeks.org/applications-advantages-and-disadvantages-of-tree/) for details.

### Types of Tree data structures:

Tree data structure can be classified into three types based upon the number of children each node of the tree can have. The types are:

* [**Binary tree**](https://www.geeksforgeeks.org/types-of-trees-in-data-structures)**:** In a binary tree, each node can have a maximum of two children linked to it. Some common types of binary trees include full binary trees, complete binary trees, balanced binary trees, and degenerate or pathological binary trees. Examples of Binary Tree are Binary Search Tree and Binary Heap.
* [**Ternary Tree**](https://www.geeksforgeeks.org/ternary-tree)**:** A Ternary Tree is a tree data structure in which each node has at most three child nodes, usually distinguished as “left”, “mid” and “right”.
* [**N-ary Tree or Generic Tree**](https://www.geeksforgeeks.org/generic-treesn-array-trees)**:** Generic trees are a collection of nodes where each node is a data structure that consists of records and a list of references to its children(duplicate references are not allowed). Unlike the linked list, each node stores the address of multiple nodes.

### Basic Operations Of Tree Data Structure:

* **Create** – create a tree in the data structure.
* **Insert** − Inserts data in a tree.
* **Search** − Searches specific data in a tree to check whether it is present or not.
* **Traversal**:
  + [Depth-First-Search Traversal](https://www.geeksforgeeks.org/depth-first-search-or-dfs-for-a-graph)
  + [Breadth-First-Search Traversal](https://www.geeksforgeeks.org/breadth-first-search-or-bfs-for-a-graph)

### Properties of Tree Data Structure:

**Number of edges:**An edge can be defined as the connection between two nodes. If a tree has N nodes then it will have (N-1) edges. There is only one path from each node to any other node of the tree.

**Depth of a node:** The depth of a node is defined as the length of the path from the root to that node. Each edge adds 1 unit of length to the path. So, it can also be defined as the number of edges in the path from the root of the tree to the node.

* **Height of a node:** The height of a node can be defined as the length of the longest path from the node to a leaf node of the tree.
* **Height of the Tree:** The height of a tree is the length of the longest path from the root of the tree to a leaf node 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.