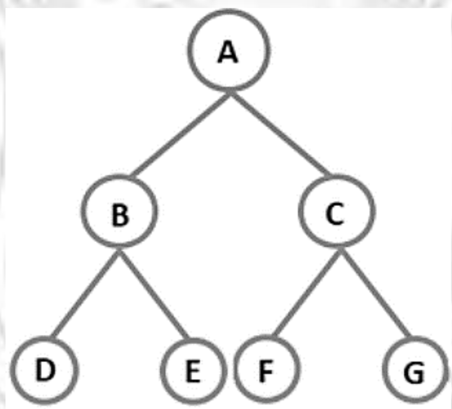
TREES

INTRODUCTION TO TREES

- A tree is non-linear and a hierarchical data structure consisting of a collection of nodes such that each node of the tree stores a value and a list of references to other nodes (the "children"). This data structure is a specialized method to organize and store data in the computer to be used more effectively.

Example of Tree data structure



Here,

- Node A is the root node
- B is the parent of D and E
- D and E are the siblings
- D, E, F and G are the leaf nodes
- A and B are the ancestors of E

Basic Terminologies in Tree Data Structure

- **Parent Node:** The node which is a predecessor of a node is called the parent node of that node.
- **Child Node:** The node which is the immediate successor of a node is called the child node of that node.
- **Root Node:** The topmost node of a tree or the node which does not have any parent node is called the root node. 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.
- **Ancestor of a Node:** Any predecessor nodes on the path of the root to that node are called Ancestors of that node.
- **Descendant:** Any successor node on the path from the leaf node to that node.
- **Sibling:** Children of the same parent node 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.

Properties of a Tree

- **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.

Syntax for creating a node

```
struct Node
{
  int data;
```

```
  struct Node *left_child;
  struct Node *right_child;
};
```