DAY-32

①. Minimum value in Binary Search Tree.

→ Traverse the node from root to left recursively until left is NULL.

* the node whose left is NULL is the node with minimumvalue.

~~def~~ Code part :-

```
def minValue(node):
current =node
~~while current.left is notnone~~
while(current.lef is not None):
        current =current.left

    return current.data.
```

② Height of a Binary Tree

Algorithm :-

⇒ if tree is empty then return 0

⇒ Else

→ Get the max depth of left subtree recursively.

→ Get the max depth of right subtree recursively

→ Get the max of max depths of left and right subtreed and add 1 to it for the current node.

→ Return the max depth.

code part:

```
def  maxDepth(node):
   if node is None:
       return 0.

   else:
      lDepth = maxDepth(node.left)
      rDepth = maxDepth(node.right)

      if (lDepth > rDepth):
          return lDepth + 1
      else :
          return rDepth +1
```