Problem # 104. Maximum Depth of Binary Tree (Easy)

<https://leetcode.com/problems/maximum-depth-of-binary-tree/>

My Solution:

This Solution uses recursion.

1. Terminal condition (for exiting) is the root is null. In this case return 0.
2. Otherwise, get the heights of the left subtree and the right subtree and find the maximum of these two heights. Add 1 to the maximum value for the root.

# Definition for a binary tree node.

# class TreeNode:

# def \_\_init\_\_(self, val=0, left=None, right=None):

# self.val = val

# self.left = left

# self.right = right

class Solution:

def maxDepth(self, root: TreeNode) -> int:

if not root:

return 0

return 1 + max(self.maxDepth(root.left), self.maxDepth(root.right))