Leetcode Problem 2. (Easy)

Binary Tree Preorder Traversal

Given the root of a binary tree, return *the preorder traversal of its nodes' values*.

**Example 1:**



**Input:** root = [1,null,2,3]

**Output:** [1,2,3]

**Example 2:**

**Input:** root = []

**Output:** []

**Example 3:**

**Input:** root = [1]

**Output:** [1]

**Constraints:**

* The number of nodes in the tree is in the range [0, 100].
* -100 <= Node.val <= 100

Link: <https://leetcode.com/problems/binary-tree-preorder-traversal/>

class Solution {

public List<Integer> preorderTraversal(TreeNode root) {

List<Integer> result = new ArrayList<>();

if (root == null) {

return result;

}

Stack<TreeNode> stack = new Stack<>();

stack.push(root);

while (!stack.isEmpty()) {

TreeNode node = stack.pop();

result.add(node.val);

if (node.right != null) {

stack.push(node.right);

}

if (node.left != null) {

stack.push(node.left);

}

}

return result;

}

}

