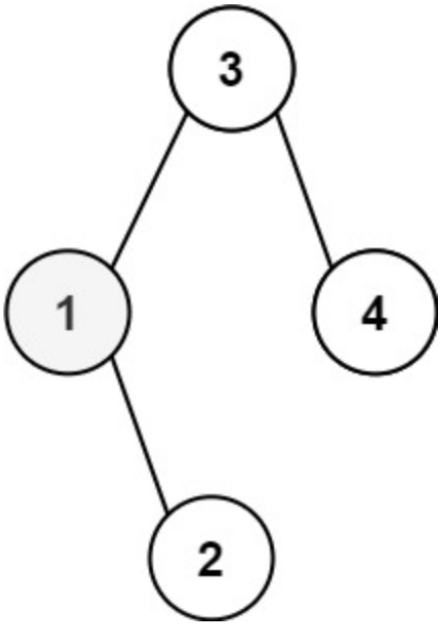
Kth Smallest Element in a BST

Difficulty	Medium
Category	Tree
Question	https://leetcode.com/problems/kth-smallest-element-in-a-bst/
Solution	https://www.youtube.com/watch?v=5LUXSvjmGCw
Status	Done

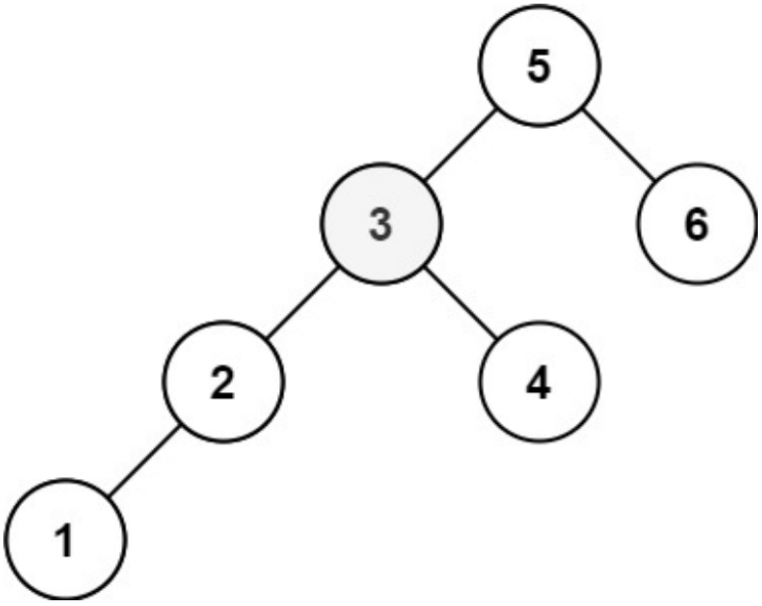
Question

> Given the `root` of a binary search tree, and an integer `k`,
> return *the* `kth` *smallest value (**1-indexed**) of all the values of the nodes in the tree*.



> **Input:** root = [3,1,4,null,2], k = 1
> **Output:** 1

Example 2:



> **Input:** root = [5,3,6,2,4,null,null,1], k = 3

Solution

```
class Solution:
    def kthSmallest(self, root: TreeNode, k: int) -> int:
        # Helper function to count the number of nodes in a subtree.
        def count_nodes(node):
            if not node:
                return 0
            return 1 + count_nodes(node.left) + count_nodes(node.right)

        # If the root is None, return None as the tree is empty.
        if not root:
            return None

        # Calculate the number of nodes in the left subtree.
        left_count = count_nodes(root.left)

        # Depending on the value of k and the number of nodes in the left subtree, make a decision.
        if k <= left_count:
            # If k is less than or equal to the count of nodes in the left subtree,
            # move to the left subtree to find the kth smallest element.
            return self.kthSmallest(root.left, k)
        elif k == left_count + 1:
            # If k is equal to left_count + 1, the current node is the kth smallest element.
            return root.val
        else:
            # If k is greater than left_count + 1, move to the right subtree and adjust k accordingly.
            return self.kthSmallest(root.right, k - left_count - 1)
```

Explanation