Challenge 1: Find minimum value in Binary Search Tree

Given the root to a Binary Search Tree, write a function to find the minimum value in that tree. A solution is placed in the "solution" section for your help, but we would suggest you solve it on your own first.

We'll cover the following

Problem Statement#

Implement the `findMin(root)` function which will find the minimum value in a given Binary Search Tree. Remember, a Binary Search Tree is a Binary Tree which satisfies the following property. An illustration is also provided to jog your memory.

$$NodeValues(LeftSubtree) <= CurrentNodeValue < NodeValues(RightSubTree)$$

Output#

Returns minimum integer value from a given binary search

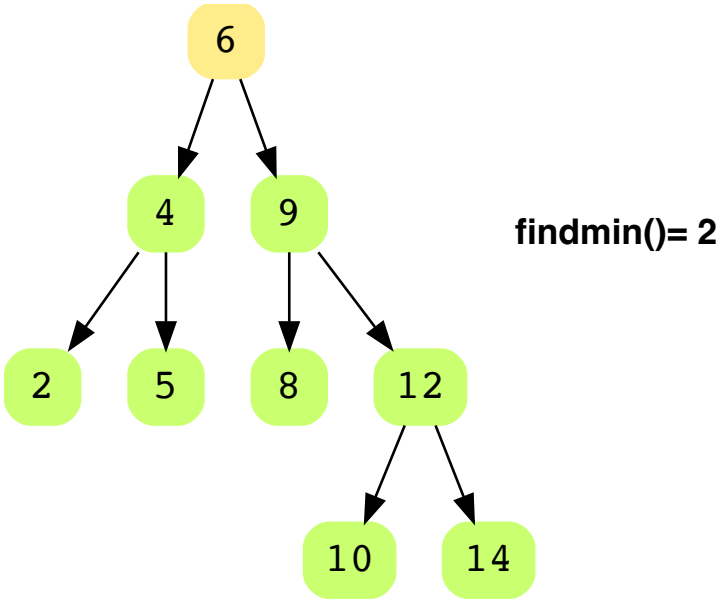
Sample Input#

The root of an object of the BST class which contains data such as.

```
bst = {
    6 -> 4,9
    4 -> 2,5
    9 -> 8,12
    12 -> 10,14
}
where parent -> leftChild,rightChild
```

Sample Output#

```
2
```



Coding Exercise

Take a close look and design a step-by-step algorithm first before jumping to the implementation. This problem is designed for your practice, so try to solve it on your own first. If you get stuck, you can always refer to the solution provided in the solution section. Good Luck!

main.py

BinarySearchTree.py

Node.py

```
from Node import Node
from BinarySearchTree import BinarySearchTree


def findMin(root):
    # Write your code here
    current = root
    while current:
        if current.leftChild:
            current = current.leftChild
        else:
            return current.val
```


Back

Overview of Trees

Next

Solution: Find minimum value in Bina

Report an Issue