DAY-17

Employee ID:

109883998

Name- Rishabh Mnachanda  
  
  
Create node in AVL  
  
class Node {

int key; Here Value stored in the node

int height; Height of the node

Node left; Pointer to left child

Node right; Pointer to right child

Node(int key) {

this.key = key;

this.height = 1; This New node is a leaf, so height = 1

this.left = null;

this.right = null;

}

}

2. check if tree is empty or not

Boolean isempty(MyTree root)

{

if(root==null)

return true;

else

return false;

}  
  
  
3. if tree is empty the inserted node will be the root node.  
  
class Node:

def \_\_init\_\_(self, data):

self.data = data

self.left = None

self.right = None

self.height = 1

class AVLTree:

def insert(self, root, data):

# If tree is empty, create new node as root

if not root:

return Node(data)

# Normal BST insertion

if data < root.data:

root.left = self.insert(root.left, data)

elif data > root.data:

root.right = self.insert(root.right, data)

return root

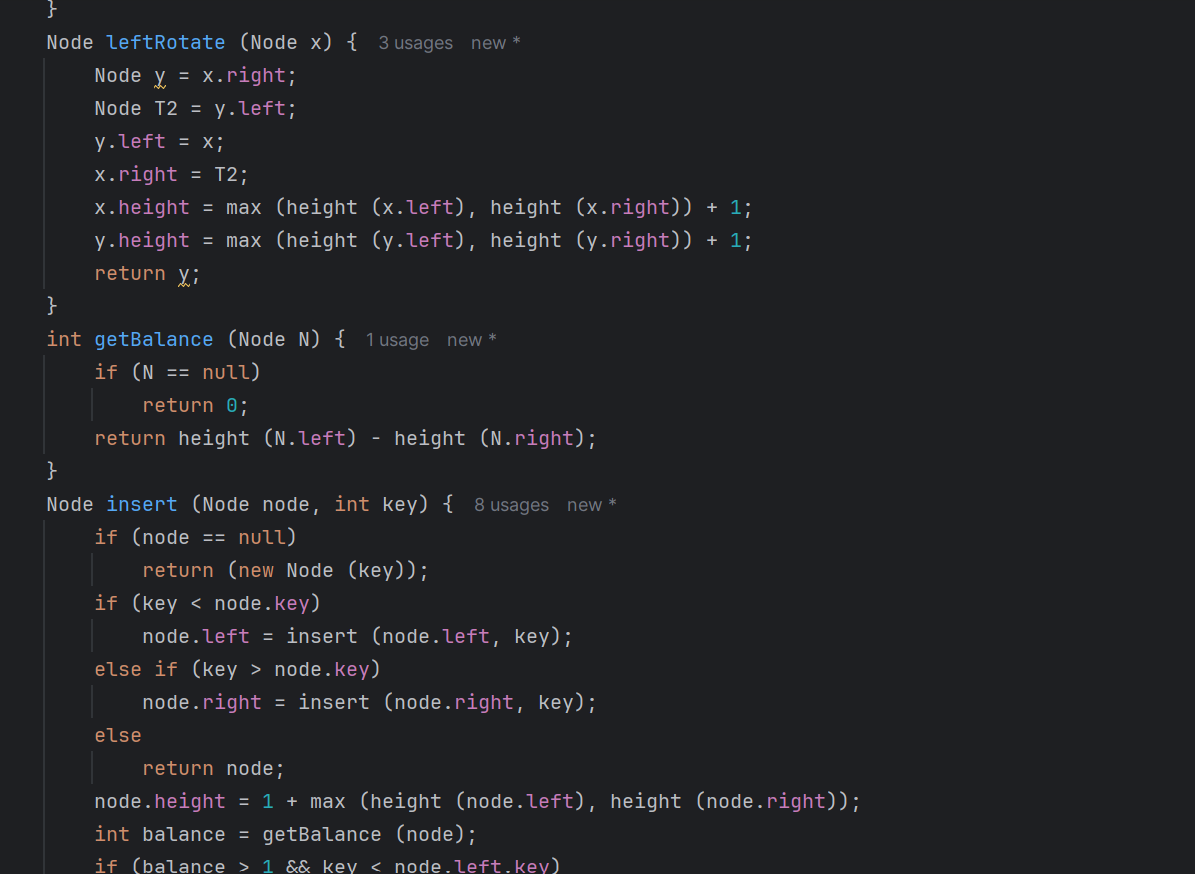
# Example usage:

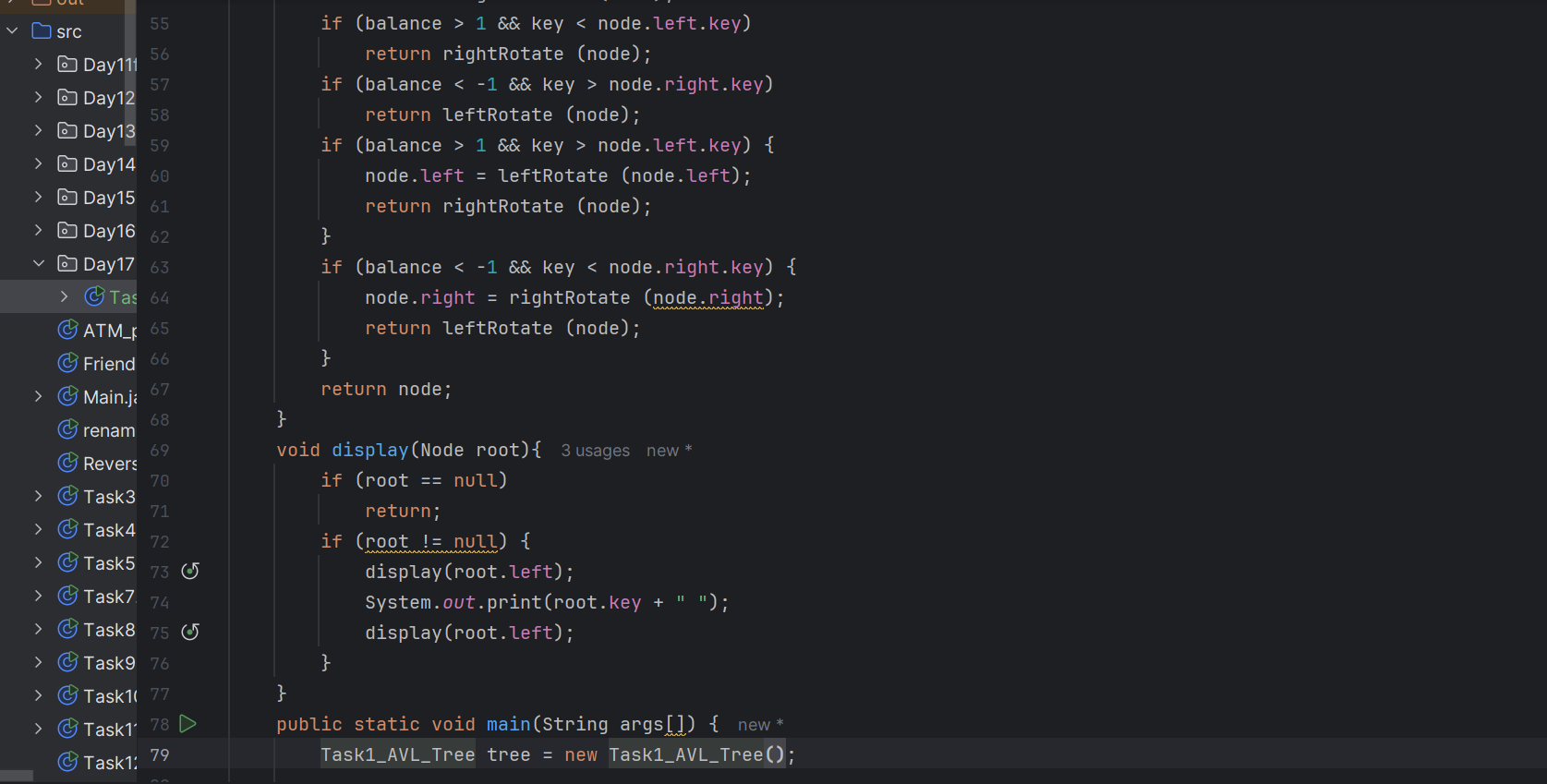
avl = AVLTree()

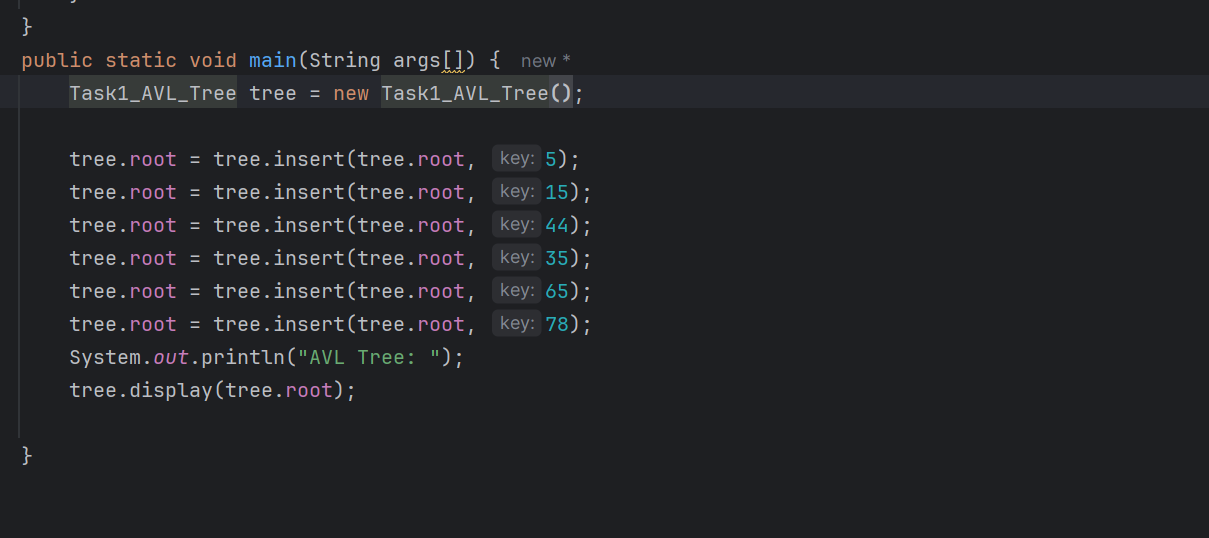
root = None

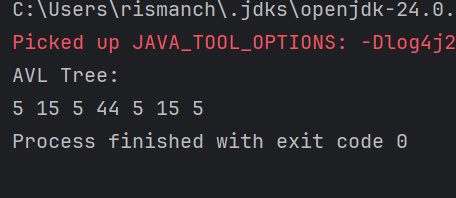
# First insertion - becomes root

root = avl.insert(root, 10) # 10 becomes root  
  
Task 1  
  







Task- Red black tree

