

DAILY ONLINE ACTIVITIES SUMMARY

Date:	21/06/2020	Name:	Divyashree Naik
Sem & Sec	8 th sem A	USN:	4AL16CS034
Online Test Summary			
Subject	---		
Max. Marks	---	Score	---
Certification Course Summary			
Course	Java Programming		
Certificate Provider	Great Learning	Duration	3.5 hrs
Coding Challenges			
Problem Statement: Python program to check whether a given a binary tree is a valid binary search tree (BST) or not.			
Status: Complete			
Uploaded the report in Github		Yes	
If yes Repository name		Alvas-Report	
Uploaded the report in slack		Yes	

Online certification:

The screenshot shows the Great Learning website interface. The top navigation bar includes 'Home', 'Live Sessions', and 'Certificates'. A 'My Courses' button and a user profile icon are on the right. The main content area is titled 'CONTENT' and 'ASSESSMENTS'. Under 'CONTENT', there is a section for 'Learning Videos' with a list of seven videos, each with a play button icon, a title, a duration, and a green checkmark indicating completion.

Video Title	Duration	Status
Agenda	51s	Completed
What is Java	1m	Completed
Install Java & Java IDE	5m	Completed
First Java Program	3m	Completed
Variables and Data Types	6m	Completed
Operators in Java	7m	Completed

Coding Challenge:

The screenshot shows the Spyder Python IDE interface. The main editor window displays a Python script for a Binary Search Tree (BST) problem. The script defines a `Node` class and a `isBSTUtil` function to check if a given tree is a BST. The script also includes a driver program to test the function with a specific tree structure.

```
9 class Node:
10     def __init__(self, data):
11         self.data = data
12         self.left = None
13         self.right = None
14 # Returns true if the given tree is a binary search tree
15 # (efficient version)
16 def isBST(node):
17     return isBSTUtil(node, INT_MIN, INT_MAX)
18
19 # Return true if the given tree is a BST and its values
20 # >= min and <= max
21 def isBSTUtil(node, mini, maxi):
22     # An empty tree is BST
23     if node is None:
24         return True
25
26     # False if this node violates min/max constraint
27     if node.data < mini or node.data > maxi:
28         return False
29
30     # Otherwise check the subtrees recursively
31     # tightening the min or max constraint
32     return (isBSTUtil(node.left, mini, node.data - 1) and
33           isBSTUtil(node.right, node.data + 1, maxi))
34
35 # Driver program to test above function
36 root = Node(4)
37 root.left = Node(2)
38 root.right = Node(5)
39 root.left.left = Node(1)
40 root.left.right = Node(3)
41
42 if (isBST(root)):
43     print ("Is BST")
44 else:
45     print ("Not a BST")
```

The IPython console shows the execution of the script, which outputs "Is BST". The console also displays the error message: "SyntaxError: Missing parentheses in call to 'print'. Did you mean print('Is BST')?".