DAILY ONLINE ACTIVITIES SUMMARY

Date:	**17/6/2020**	**Name:**	**Sourabh Kakade**
Sem & Sec	**8th Sem**	**USN:**	**4AL16CS104**

Online Test Summary

Subject	**NO TEST TAKEN**		
Max. Marks	**NIL**	**Score**	**NIL**

Certification Course Summary

Course	**Complete Java Bootcamp.**		
Certificate Provider	**Udemy**	**Duration**	**37hr**

Coding Challenges

Problem Statement:

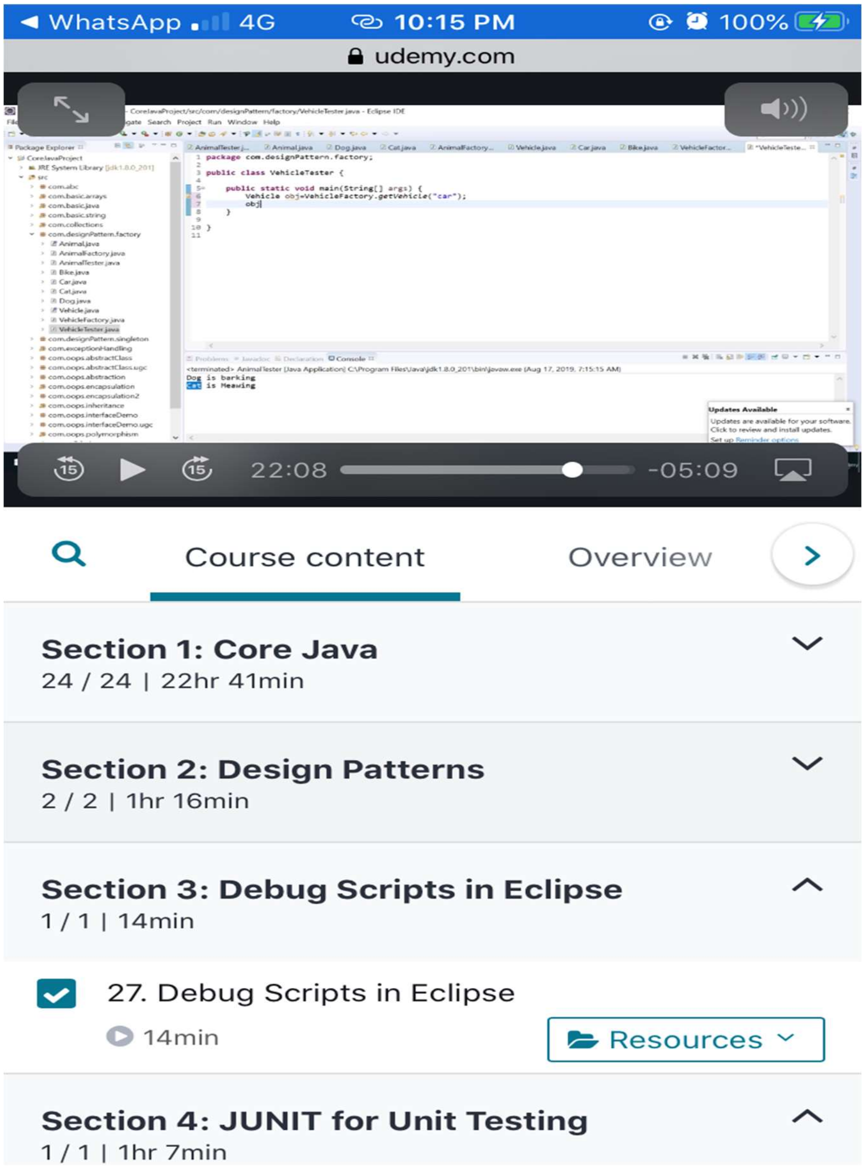
1: Python program to check if a binary tree is bst or not.

Status: COMPLETED

Uploaded the report in Github	**yes**
If yes Repository name	**Sourabh Kakade**
Uploaded the report in slack	**yes**

Online Test Details: (**Attach the snapshot and briefly write the report for the same**)

Certification Course Details: (**Attach the snapshot and briefly write the report for the same**)



◀ WhatsApp 4G
10:15 PM
100%
udemy.com
CoreJavaProject/src/com/designPattern/factory/VehicleTester.java - Eclipse IDE
AnimalTester.j... Animal.java Dog.java Cat.java AnimalFactory... Vehicle.java Car.java Bike.java VehicleFactor... *VehicleTeste...

```
package com.designPattern.factory;

public class VehicleTester {

    public static void main(String[] args) {
        Vehicle obj=VehicleFactory.getVehicle("car");
        obj
    }

}
```

```
<terminated> AnimalTester [Java Application] C:\Program Files\Java\jdk1.8.0_201\bin\javaw.exe (Aug 17, 2019, 7:15:15 AM)
Dog is barking
Cat is Meawing
```

Updates Available
Updates are available for your software. Click to review and install updates.
Set up Reminder options

Section 1: Core Java
24 / 24 | 22hr 41min

Section 2: Design Patterns
2 / 2 | 1hr 16min

Section 3: Debug Scripts in Eclipse
1 / 1 | 14min

☑ 27. Debug Scripts in Eclipse
14min
Resources

Section 4: JUNIT for Unit Testing
1 / 1 | 1hr 7min

Coding Challenges Details: (**Attach the snapshot and briefly write the report for the same**)

INT_MAX=4294967296

```
INT_MIN=-4294967296

classNode:
    def__init__(self,data):
        self.data=data
        self.left=None
        self.right=None
    def isBST(node):
        return(isBSTUtil(node,INT_MIN,INT_MAX))
    def isBSTUtil(node,mini,maxi):
        if node is None:
            return True
        if node.data<miniornode.data>maxi:
            return False
        return(isBSTUtil(node.left,mini,node.data-
    1)andisBSTUtil(node.right,node.data+1,maxi))
    root=Node(4)
    root.left=Node(2)
    root.right=Node(5)
    root.left.left=Node(1)
    root.left.right=Node(3)
    if(isBST(root)):
        print("isBST")
    else:
        print("Not a BST")
```