DAILY ONLINE ACTIVITIES SUMMARY

Date:	18/05/2020	Name:	Deekshith T R
Sem & Sec	8th A	USN:	4AL16CS027

Online Test Summary			
Subject	SMS		
Max. Marks	60	Score	60

Certification Course Summary			
Course	Getting Started Hadoop		
Certificate Provider	Great learning	Duration	24 mins

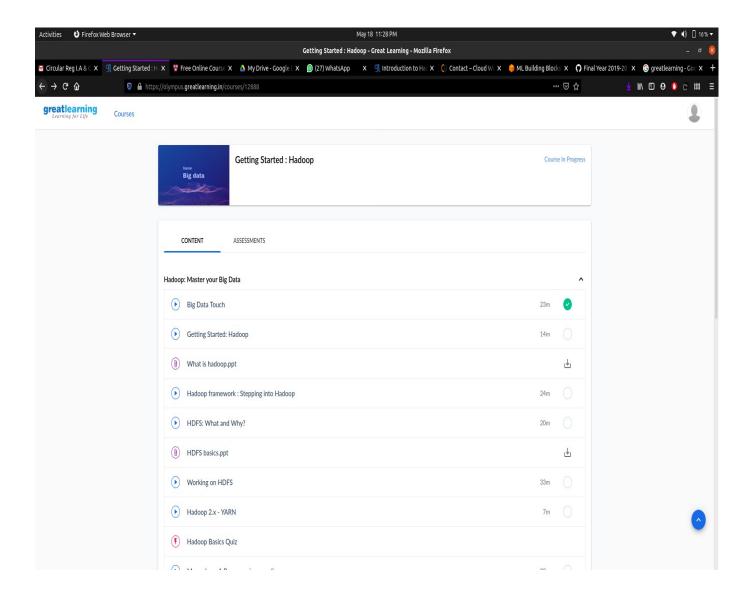| Coding Challenges	
Problem Statement:	
Status: COMPLETED	
Uploaded the report in Github	YES
If yes Repository name	Deekshithtr_16cs027
Uploaded the report in slack	YES

Online Test Details:

Test on module 3 (Random number generation)

Snapshot of test

Certification Course Details:

https://olympus.greatlearning.in/courses/12888

greatlearning
Learning for Life   Courses

Master
Big data

Getting Started : Hadoop

Course In Progress

CONTENT	ASSESSMENTS

Hadoop: Master your Big Data   ^

▶ Big Data Touch	23m	✓
▶ Getting Started: Hadoop	14m	○
◉ What is hadoop.ppt		⤓
▶ Hadoop framework : Stepping into Hadoop	24m	○
▶ HDFS: What and Why?	20m	○
◉ HDFS basics.ppt		⤓
▶ Working on HDFS	33m	○
▶ Hadoop 2.x - YARN	7m	○
⬢ Hadoop Basics Quiz		

1. **Semi-structured**

Coding Challenges Details

Program no:1
```java
package pk;
import java.util.Scanner;
public class StringOperators
{
public static void main(String args[])
{
int i;
String str;

   int counter[] = new int[256];
   Scanner in = new Scanner(System.in);

   System.out.print("Enter a String : ");
   str=in.nextLine();

    for (i = 0; i < str.length(); i++) {
      counter[(int) str.charAt(i)]++;
   }
   // Print Frequency of characters
   for (i = 0; i < 256; i++) {
      if (counter[i] != 0) {
          System.out.println((char) i  + ":-" + counter[i] + " times");
      }
   }
}
}
```

Program no:2

```java
public class PingPong extends Thread {
static StringBuilder object = new StringBuilder("");

public static void main(String[] args) throws InterruptedException {

Thread t1 = new PingPong();
Thread t2 = new PingPong();

t1.setName("\nping");
t2.setName(" pong");
```

```java
        t1.start();
        t2.start();
    }

    @override
    public void run() {
        working();
    }

    void working() {
        while (true) {
            synchronized (object) {
                try {
                    System.out.print(Thread.currentThread().getName());
                    object.notify();
                    object.wait();
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}
```