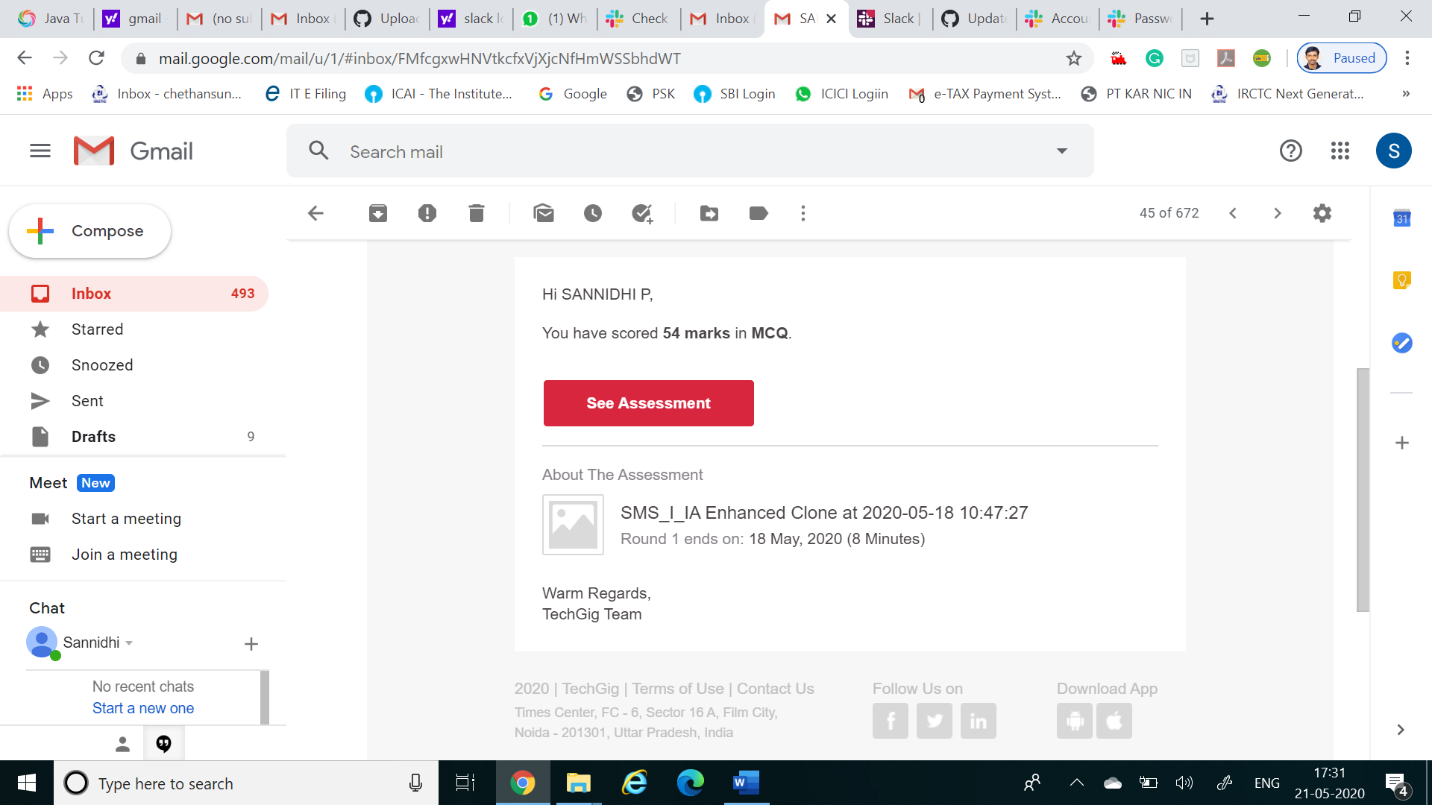
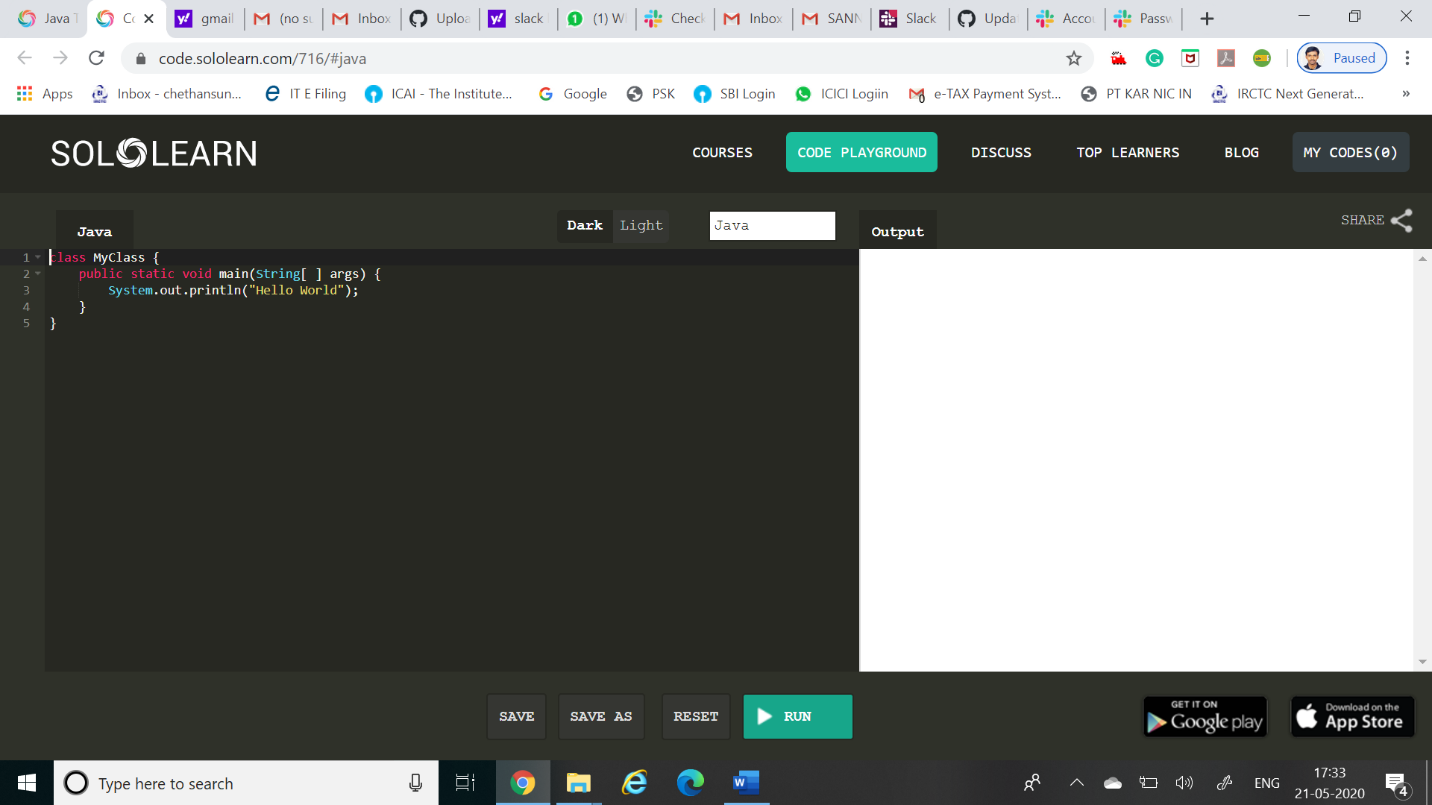
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **18/05/2020** | | | | | **Name:** | **Sannidhi P** | |
| **Sem & Sec** | **8th B** | | | | | **USN:** | **4AL16CS084** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **SMS** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **54** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **JAVASCRIPT Tutorial** | | | | | | | |
| **Certificate Provider** | | | **SOLOLEARN** | | **Duration** | | | **30min** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  **prob1- Using methods charAt() & length() of String class, write a**  **program to print the frequency of each character in a string. “Hello friend”**  **Prob2- Write down a java program to print even and odd numbers series**  **respectively from two threads: t1 and t2 synchronizing on a shared object Let t1**  **print message “ping — >” and t2 print message “,—-pong”.** | | | | | | | | |
| **Status: https://github.com/alvas-education-foundation/Sannidhi1** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | **Sannidhi1** | | | |
| **Uploaded the report in slack** | | | | | **Yes** | | | |

Online Test Details:



Certification Course Details:



Coding Challenges Details:

Program 1 :

import java.util.Scanner;

public class Main

{

public static void main(String[] args) {

int i;

String str;

int count[] = new int[256];

Scanner s = new Scanner(System.in);

System.out.print("Enter a String : ");

str=s.nextLine();

for (i = 0; i < str.length(); i++) {

count[(int) str.charAt(i)]++;

}

for (i = 0; i < 256; i++) {

if (count[i] != 0) {

System.out.println( (char) i + " : " + count[i]);

}

}

}

}

Program 2 :

public class PingPong extendsThread {

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");

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();

}

}

}

}

}