

## DAILY ONLINE ACTIVITIES SUMMARY

<b>Date:</b>	18/5/2020	<b>Name:</b>	Vleena Mascarenhas
<b>Sem &amp; Sec</b>	8 <sup>th</sup> & B	<b>USN:</b>	4AL16CS121
<b>Online Test Summary</b>			
<b>Subject</b>	System Modeling and Simulation		
<b>Max. Marks</b>	60	<b>Score</b>	36
<b>Certification Course Summary</b>			
<b>Course</b>	Introduction to Ethical Hacking		
<b>Certificate Provider</b>	Great learning academy	<b>Duration</b>	6hrs
<b>Coding Challenges</b>			
<b>Problem Statement:</b> 1.Using methods charAt() & length() of string class 2.print even and odd numbers series respectively from two threads:t1 and t2.			
<b>Status:</b> Solved			
<b>Uploaded the report in Github</b>		yes	
<b>If yes Repository name</b>		vleena	
<b>Uploaded the report in slack</b>		yes	

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

The banner shows a 'Challenge Over' status with a play button icon. It is attributed to 'by TechGig' and dated 'SMS\_I\_IA Enhanced Clone At 2020-05-18 10:47:27'. Below the banner, there are two summary boxes. The left box is for 'MCQ' type, showing 'Your Highest Score 36' and 'Max Score 60', with a 'Start Test' button. The right box is a 'Summary' box showing 'Skills' as 'Problem Solving Skills' and 'Ends On' as '18 May'.

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

The section is titled 'Learning Videos' with an upward arrow icon. It contains a video entry titled 'Career and Growth Ladder in Ethical Hacking' with a duration of '18m'. A green checkmark icon is visible to the right of the title.

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

```
prog 1:
import java.util.Scanner;

public class Frequency1
{
    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]);
    }
}
}
}

```

prog 2:

```

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");

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();  
}  
}  
}  
}  
}
```