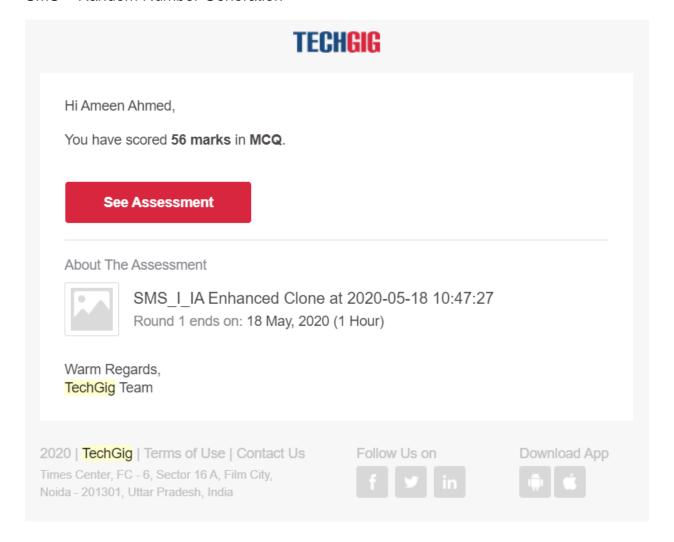
# **DAILY ONLINE ACTIVITIES SUMMARY**

Date:	18/05/202	20	Name:	AMEEN AHMED		
Sem & Sec	8 A		USN:	4AL16CS009		
		Online 1	Test Summary	,		
Subject SMS						
Max. Marks 60			Score 56			
Certification Course Summary						
Course INTRODUCTION TO HADOOP						
Certificate Provider		GREAT LEARNING	Duration		30 MINS	
Coding Challenges						
odd for serie	es. 2) Ping F	Pong Java program	of each character i	n a string a	and to print even and	
Status: CON	MPLETED					
Uploaded the report in Github			YES	YES		
If yes Repository name			Ameen	Ameen		
Uploaded th	e report ir	ı slack	YES	YES		

### **Online Test Details:**

#### SMS – Random Number Generation



#### **Certification Course Details:**

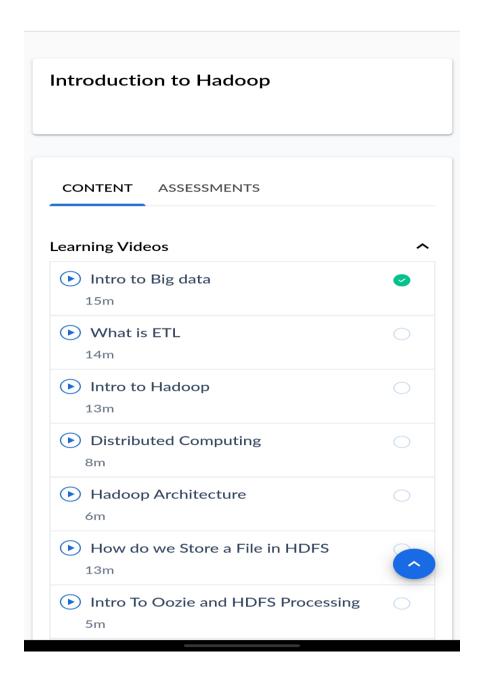
## Introduction to BigData

Big Data is a term used to describe a collection of data that is huge in volume and yet growing exponentially with time. In short such data is so large and complex that none of the traditional data management tools are able to store it or process it efficiently.

# **Types Of Big Data**

BigData could be found in three forms:

- 1. Structured
- 2. Unstructured
- 3. Semi-structured



# **Coding Challenges Details:**

## Program1

```
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");
}
}
}
```

## Program2

```
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();
}
}
```