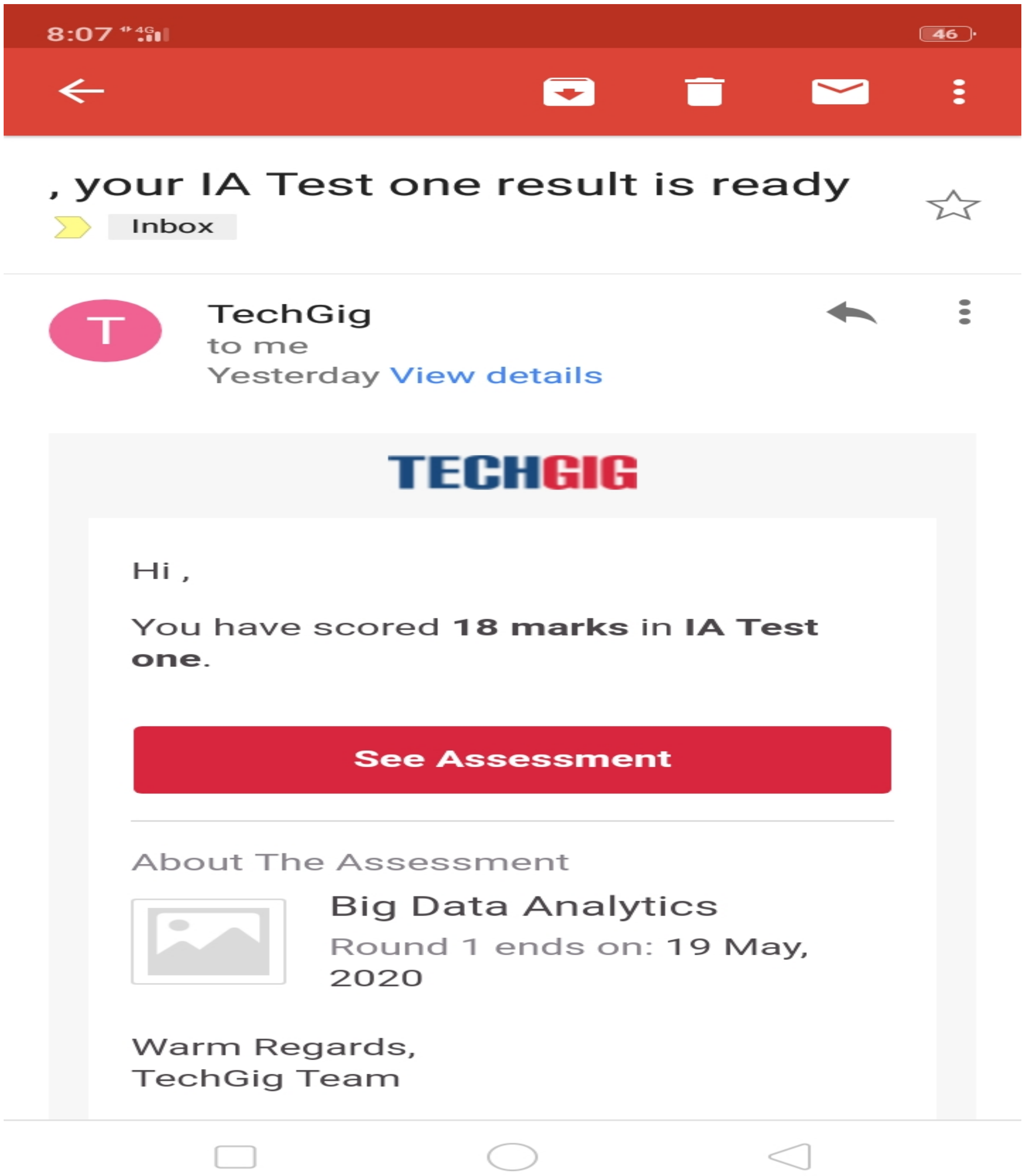
DAILY ONLINE ACTIVITIES SUMMARY

Date:	**19/05/2020**	**Name:**	**MANVITHA Rao**
Sem & Sec	**8th A**	**USN:**	**4AL16CS051**
Online Test Summary			
Subject	**BDA**		
Max. Marks	**30**	**Score**	**18**
Certification Course Summary			
Course	**Introduction to Ethical hacking**		
Certificate Provider	**Great learning**	**Duration**	**6 hours**
Coding Challenges			
Problem Statement:			
Status: COMPLETED			
Uploaded the report in Github	**YES**		
If yes Repository name	**alvas-education-foundation/Manvitha_Rao**		
Uploaded the report in slack	**YES**		

Online Test Details:

Test on module 1

Snapshot of test



8:07

, your IA Test one result is ready

Inbox

TechGig
to me
Yesterday View details

TECHGIG

Hi ,

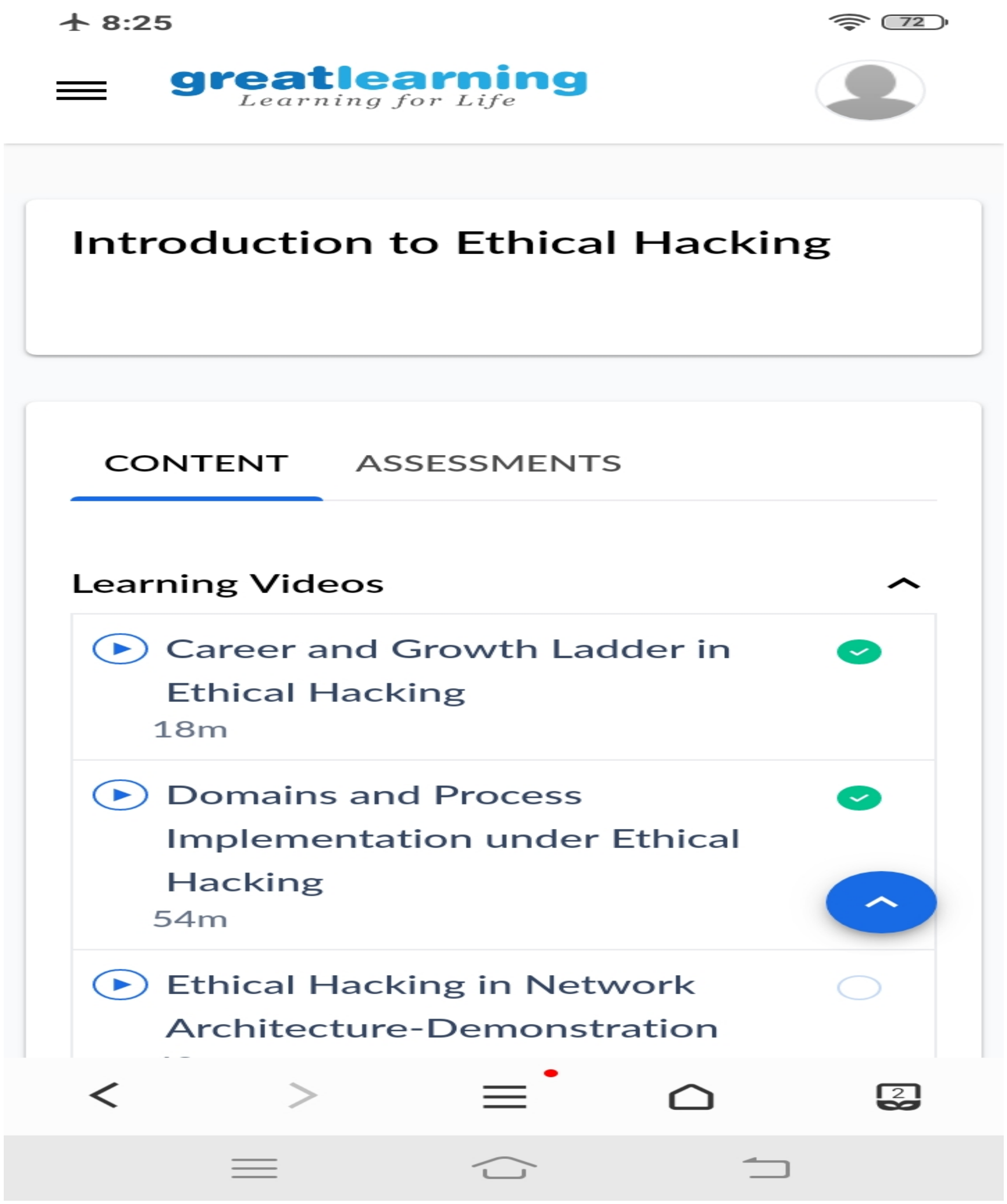
You have scored **18 marks** in **IA Test one**.

About The Assessment

Big Data Analytics
Round 1 ends on: 19 May, 2020

Warm Regards,
TechGig Team

Certification Course Details:


8:25

72

greatlearning
Learning for Life

Introduction to Ethical Hacking

CONTENT ASSESSMENTS

Learning Videos

Career and Growth Ladder in Ethical Hacking ✓
18m

Domains and Process Implementation under Ethical Hacking ✓
54m

Ethical Hacking in Network Architecture-Demonstration

Introduction to Ethical Hacking

Coding Challenges Details

Program1

```
package shortestpalindromeexample.java;
import java.util.Scanner;

public class ShortestPalindromeDemo {

public static String shortestPalindrome(String str) {

int x=0;
int y=str.length()-1;

  while(y>=0){
     if(str.charAt(x)==str.charAt(y)){
          x++;
         }
            y--;
  }

if(x==str.length())
return str;

String suffix = str.substring(x);
String prefix = new StringBuilder(suffix).reverse().toString();
String mid = shortestPalindrome(str.substring(0, x));

return prefix+mid+suffix;
}

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("Enter a String to find out shortest palindrome");

String str=in.nextLine();

System.out.println("Shortest palindrome of "+str+" is "+shortestPalindrome(str));

}
```

Program2

```
import java.util.Stack;

// Data Structure to store a linked list node
class Node {
	int data;
	Node next;

	Node(int i)
	{
		this.data = i;
		this.next = null;
	}
};

class Main
{
	// Function to determine if a given linked list is palindrome or not
	public static boolean isPalindrome(Node head)
	{
		// construct an empty stack
		Stack<Integer> s = new Stack<>();

		// push all elements of the linked list into the stack
		Node node = head;
		while (node != null) {
			s.push(node.data);
			node = node.next;
		}

		// traverse the linked list again
		node = head;
		while (node != null)
		{
			// pop the top element from the stack
			int top = s.pop();

			// compare the popped element with current node's data
			// return false if mismatch happens
			if (top != node.data) {
				return false;
			}

			// advance to the next node
			node = node.next;
```

```
        }

        // we reach here only when the linked list is palindrome
        return true;
    }

    public static void main(String[] args)
    {
        Node head = new Node(1);
        head.next = new Node(2);
        head.next.next = new Node(3);
        head.next.next.next = new Node(2);
        head.next.next.next.next = new Node(1);

        if (isPalindrome(head)) {
            System.out.print("Linked List is a palindrome.");
        } else {
            System.out.print("Linked List is not a palindrome.");
        }
    }
}
```