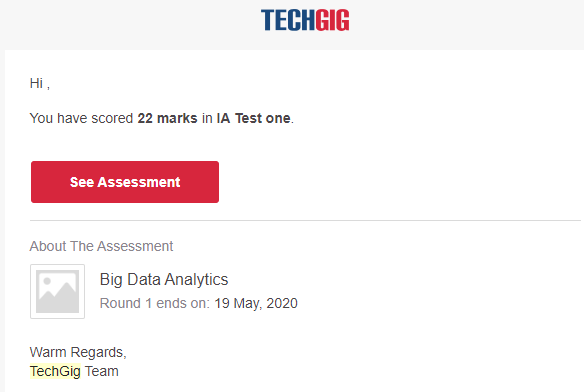
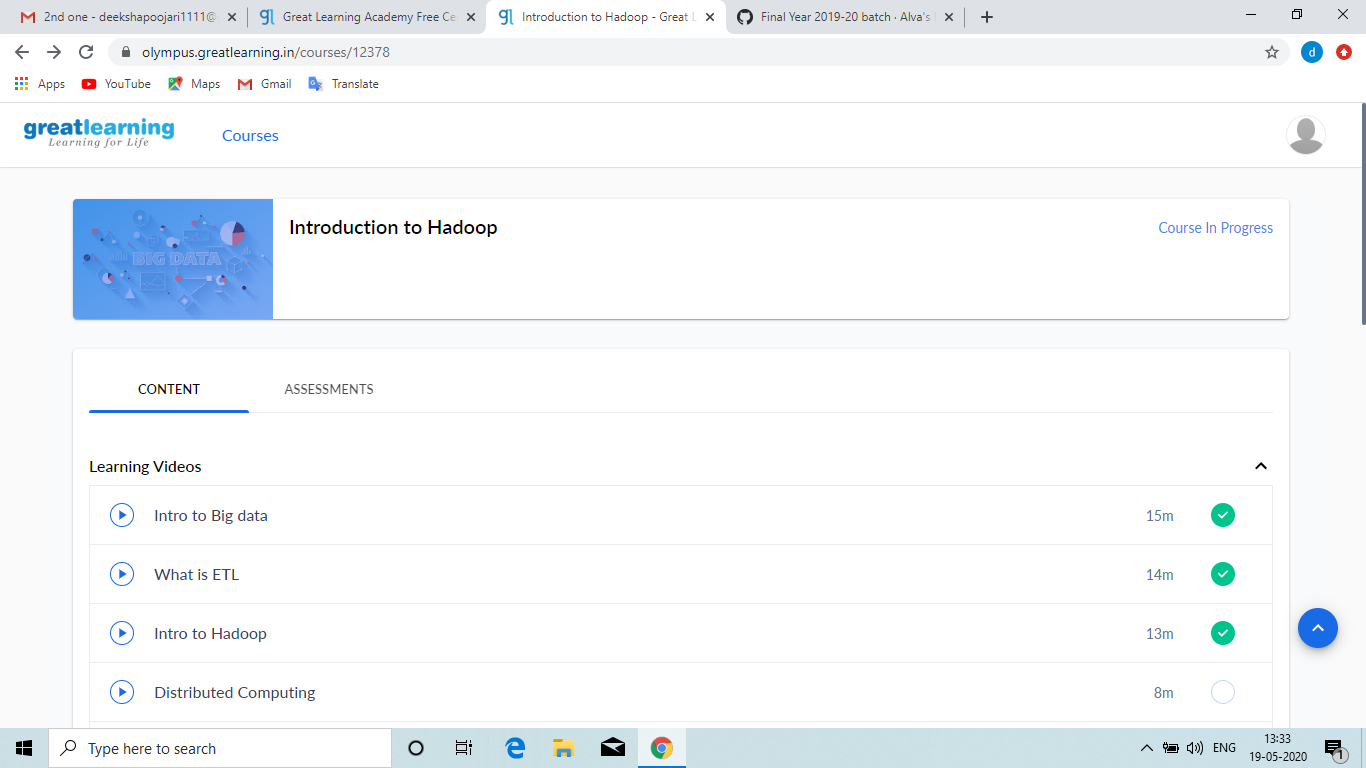
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **19/05/2020** | | | | | **Name:** | **Deeksha D Poojary** | |
| **Sem & Sec** | **VIII Semester & A section** | | | | | **USN:** | **4AL16CS026** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **Big Data Analytics** | | | | | | |
| **Max. Marks** | | **30** | | **Score** | | | **22** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Introduction to Hadoop** | | | | | | | |
| **Certificate Provider** | | | **Great Learning** | | **Duration** | | | **One Video (15 min)** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** **Have a Letter or word then we need add some letters to it and need to find out shortest palindrome & identify linked list is palindrome or not by using stack.** | | | | | | | | |
| **Status: COMPLETED** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **YES** | | | |
| **If yes Repository name** | | | | | **deekshapoojari** | | | |
| **Uploaded the report in slack** | | | | | **YES** | | | |

Online Test Details:



Certification Course Details:



**Intoduction to Hadoop**

MapReduce is the default programming framework used to analyse the data in the Hadoop. Its written in java language. Hadoop came in existence in the year 2005, took the idea from the goggle and created this framework and it is given to Apache. Hadoop is open source.

Coding Challenges Details:

**Program no:1**

**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));**

**}**

**Program no:2**

**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 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.");**

**}**

**}**