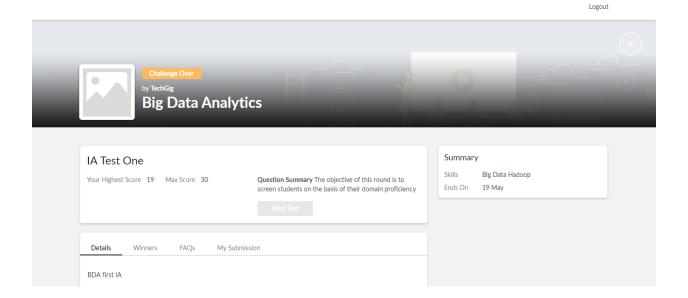
DAILY ONLINE ACTIVITIES SUMMARY

Date:	19/05/201	19	Name:	KIRAN K		
Sem & Sec	8 th A		USN:	4AL16CS046		
Online Test Summary						
Subject BDA						
Max. Marks	s 30		Score	19		
Certification Course Summary						
Course	Introduction To Hadoop					
Certificate Provider		GreatLearning	Duration	Duration		
Coding Challenges						
Problem Statement:						
Status: Completed						
Uploaded th	e report ii	ı Github	yes			
If yes Repos	itory nam	е	KiranK27751			
Uploaded th	e report ii	ı slack	yes			
			1			

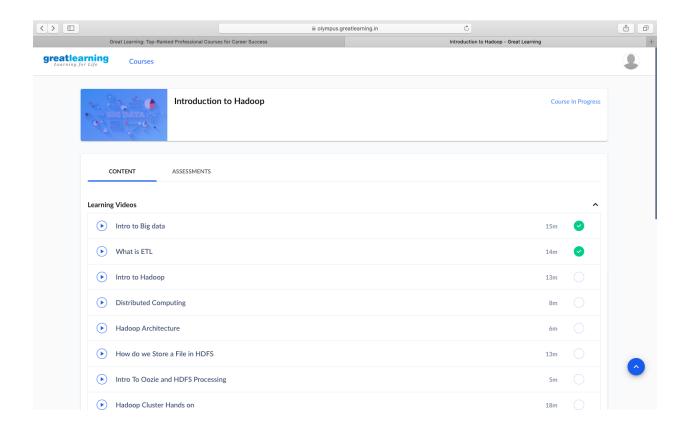
Online Test Details:



Certification Course Details:

Organizations can optimize IoT data, quickly and cost-effectively deriving its business value by developing expertise in ETL (extract, transfer, load) technologies, such as stream processing and data lakes.

At many organizations, though, this may lead to IT bottlenecks, long project delays, and data science being deferred. Result: IoT projects – in which predictive analytics data is meant to play a critical role in improving operational efficiency and spurring innovation – *still* haven't crossed the proof-of-concept threshold and definitely cannot demonstrate ROI.



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));
}
program 2
import java.util.Stack;
class Node {
int data;
Node next:
Node(int i)
{
          this.data = i;
          this.next = null;
};
class Main
public static boolean isPalindrome(Node head)
// construct an empty stack
Stack s = new Stack<>();
Node node = head;
while (node != null) {
s.push(node.data);
node = node.next;
          node = head;
          while (node != null)
                    int top = s.pop();
                    if (top != node.data) {
                             return false;
```

```
node = node.next;
}

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 = new Node(2);
    head.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.");
    }
}
```