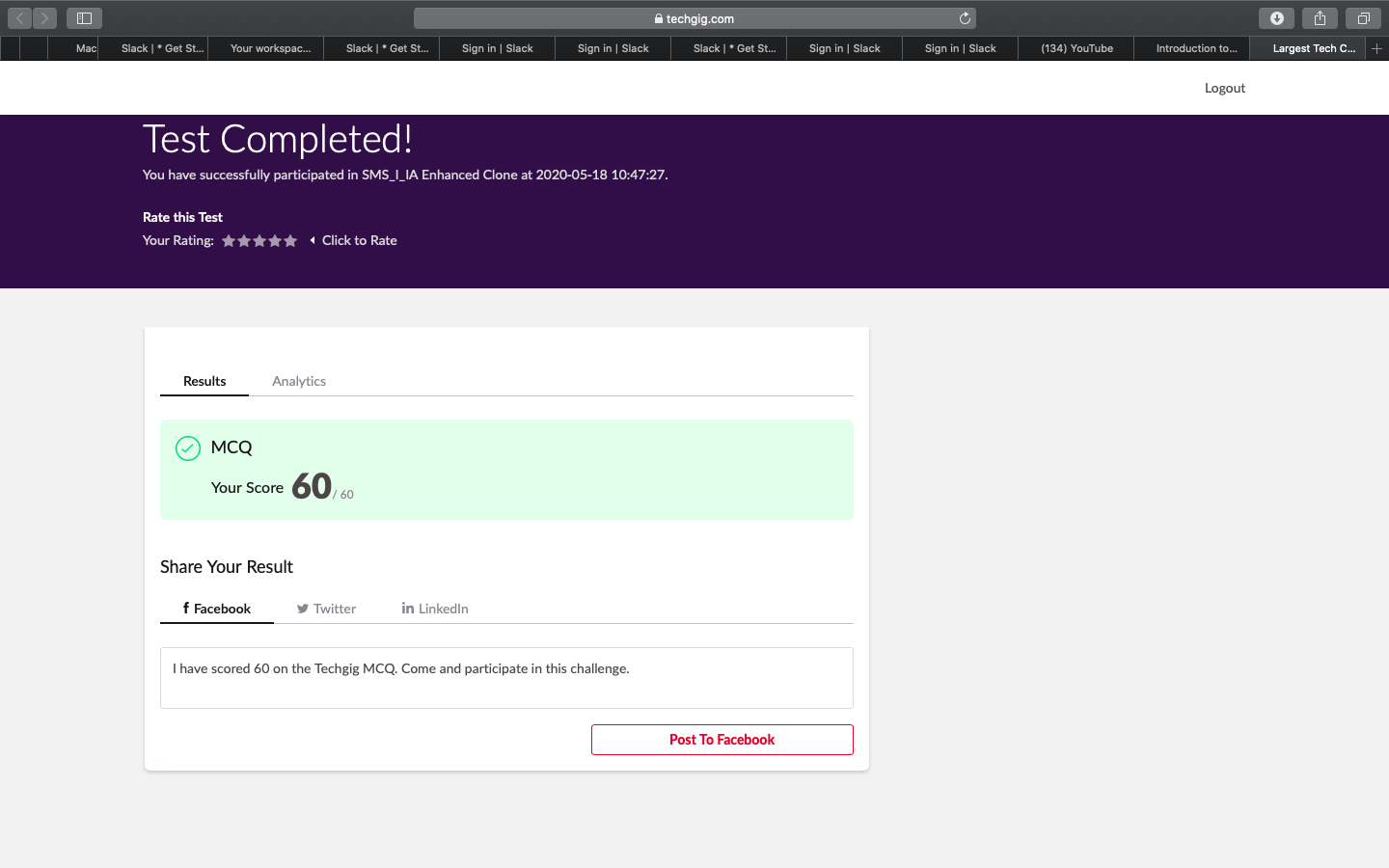
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **21/05/2020** | | | | | **Name:** | **Varshitha V Makam** | |
| **Sem & Sec** | **8th B** | | | | | **USN:** | **4AL16CS118** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **SMS** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **60** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | Introduction to Hadoop | | | | | | | |
| **Certificate Provider** | | | **Great learning** | | **Duration** | | | **15 mins** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** | | | | | | | | |
| **Status: COMPLETED** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **YES** | | | |
| **If yes Repository name** | | | | | **varshitha-123** | | | |
| **Uploaded the report in slack** | | | | | **YES** | | | |

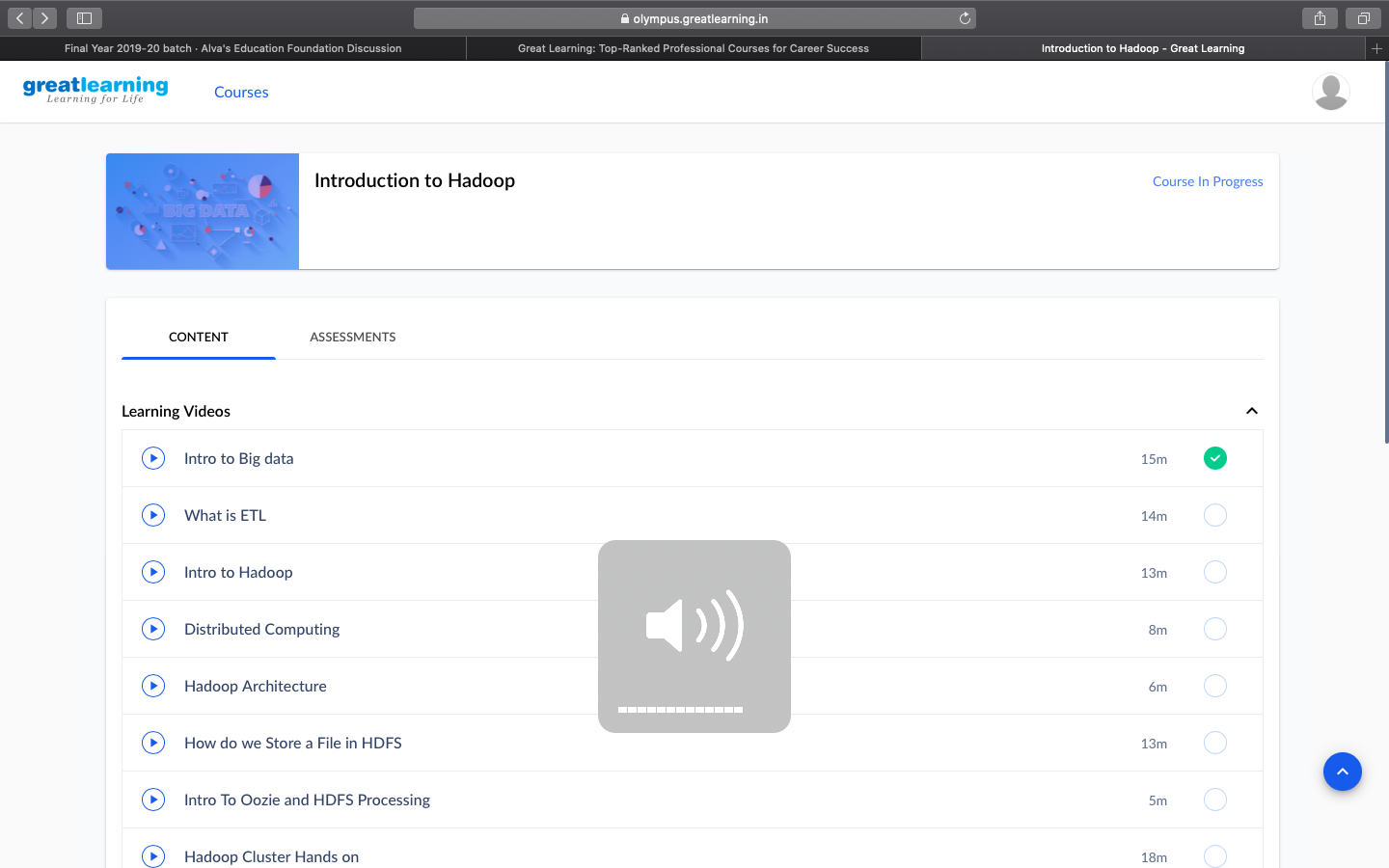
Online Test Details:

Test on module 3

Snapshot of test



Certification Course Details:



#### Introduction to BigData

Big Data is also **data** but with a **huge size**. 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

**Program no:1**

#include <stdio.h>  
#include <stdlib.h>  
struct node  
{  
int data;  
struct node *next;  
};  
struct Node reverse(struct Node head,int k)  
{  
struct Node current= head;  
struct Node next= Null;  
struct Node prev= Null;  
int count = 0;  
while(current!=Null && count<k)  
{  
next= current->next;  
current->next = prev;  
prev= current;  
current= next;  
count++;  
}  
if ( next!=Null)  
head->next= reverse( next,k);  
return prev;  
}  
void push( struct Node ==head\_ref,int new\_data)  
{  
struct Node= new\_node= (struct Node*) malloc(sizeof(struct Node));  
}  
}  
int main()  
{  
Struct node \*prev,\*head,\*p;  
int n,i;  
printf ("number of elements:");  
scanf("%d",&n);  
head=NULL;  
for(i=0;i<n;i++)  
{  
p=malloc(sizeof(struct node));  
scanf("%d",&p->data);  
p->next=NULL;  
if(head==NULL)  
head=p;  
else  
prev->next=p;  
prev=p;  
}  
return 0;  
}