DAILY ONLINE ACTIVITIES SUMMARY

Date:	**20/05/2020**	**Name:**	**Safnaaz**
Sem & Sec	**8th B**	**USN:**	**4AL16CS081**

Online Test Summary

Subject	**Internet of Things (IOT)**		
Max. Marks	**30**	**Score**	**24**

Certification Course Summary

Course	**Introduction to Hadoop**		
Certificate Provider	**Great learning**	**Duration**	**30 mins**

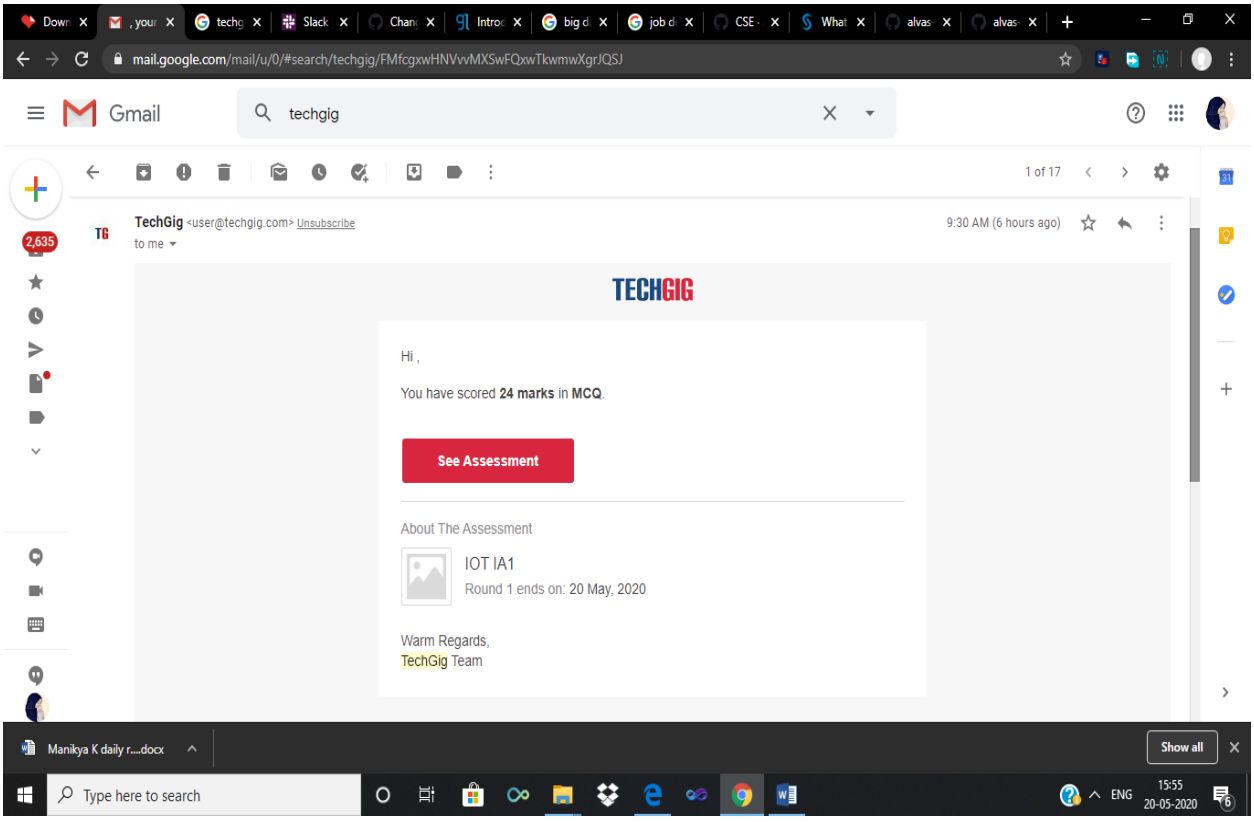
Coding Challenges

Problem Statement:1) finding frequency of each character in a string and to print even and odd for series. 2) java program	
Status: COMPLETED	
Uploaded the report in Github	**YES**
If yes Repository name	**Safnaazsheikh**
Uploaded the report in slack	**YES**

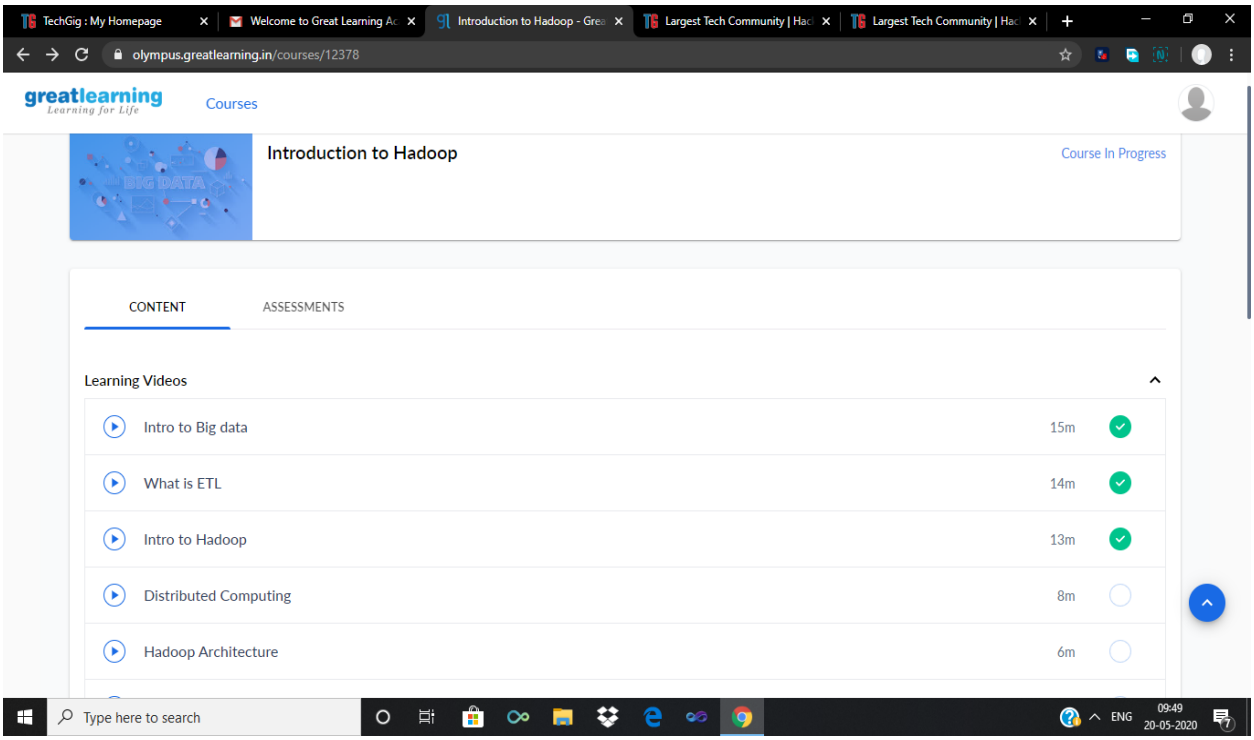
Online Test Details:

Test on module 3 (Random number generation)

Snapshot of test



Certification Course Details:



What is Hadoop and its Ecosystem?

Hadoop Ecosystem is a platform or framework which solves big data problems. You can consider it as a suite which encompasses a number of services (ingesting, storing, analyzing and maintaining) inside it. For storage we use HDFS (Hadoop Distributed Filesystem).

The main components of Hadoop ecosystem

It comprises of different components and services (ingesting, storing, analyzing, and maintaining) inside of it. Most of the services available in the Hadoop ecosystem are to supplement the main four **core** components of Hadoop which include HDFS, YARN, MapReduce and Common.

Coding Challenges Details

Program no:1

```
struct Node
{
```

```
int data;
struct Node* next;
};

pointer to the new head node. */
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));

new_node->data  = new_data;

new_node->next = (*head_ref);

(*head_ref)    = new_node;
}

void printList(struct Node *node)
{
while (node != NULL)
{
   printf("%d  ", node->data);
   node = node->next;
 }
 }
```

```
int main(void)
{
 struct Node* head = NULL;
 push(&head, 8);
 push(&head, 7);
 push(&head, 6);
 push(&head, 5);
 push(&head, 4);
 push(&head, 3);
 push(&head, 2);
 push(&head, 1);

 printf("\nGiven linked list \n");
 printList(head);
 head = reverse(head, 2);

 printf("\nReversed Linked list \n");
 printList(head);

 return(0);
 }
```