**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **14-06-2020** | | | | | **Name:** | **Ashwini S Jadamali** | |
| **Sem & Sec** | **6th&A** | | | | | **USN:** | **4AL17CS018** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **No exam.** | | | | | | |
| **Max. Marks** | | **-** | | **Score** | | | **-** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **INTRODUCTION TO FULL STACK DEVELOPMENT.** | | | | | | | |
| **Certificate Provider** | | | Greatlearning  Academy | | **Duration** | | | 22hours |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**1.write a java program to remove specific characters in the String.  2. Write a C Program to implement the Binary Reversal. . | | | | | | | | |
| **Status: Done** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **YES** | | | |
| **If yes Repository name** | | | | | [**https://github.com/ashwinijadamali/online-coding-activites**](https://github.com/ashwinijadamali/online-coding-activites) | | | |
| **Uploaded the report in slack** | | | | | **YES** | | | |

Online Test Details:

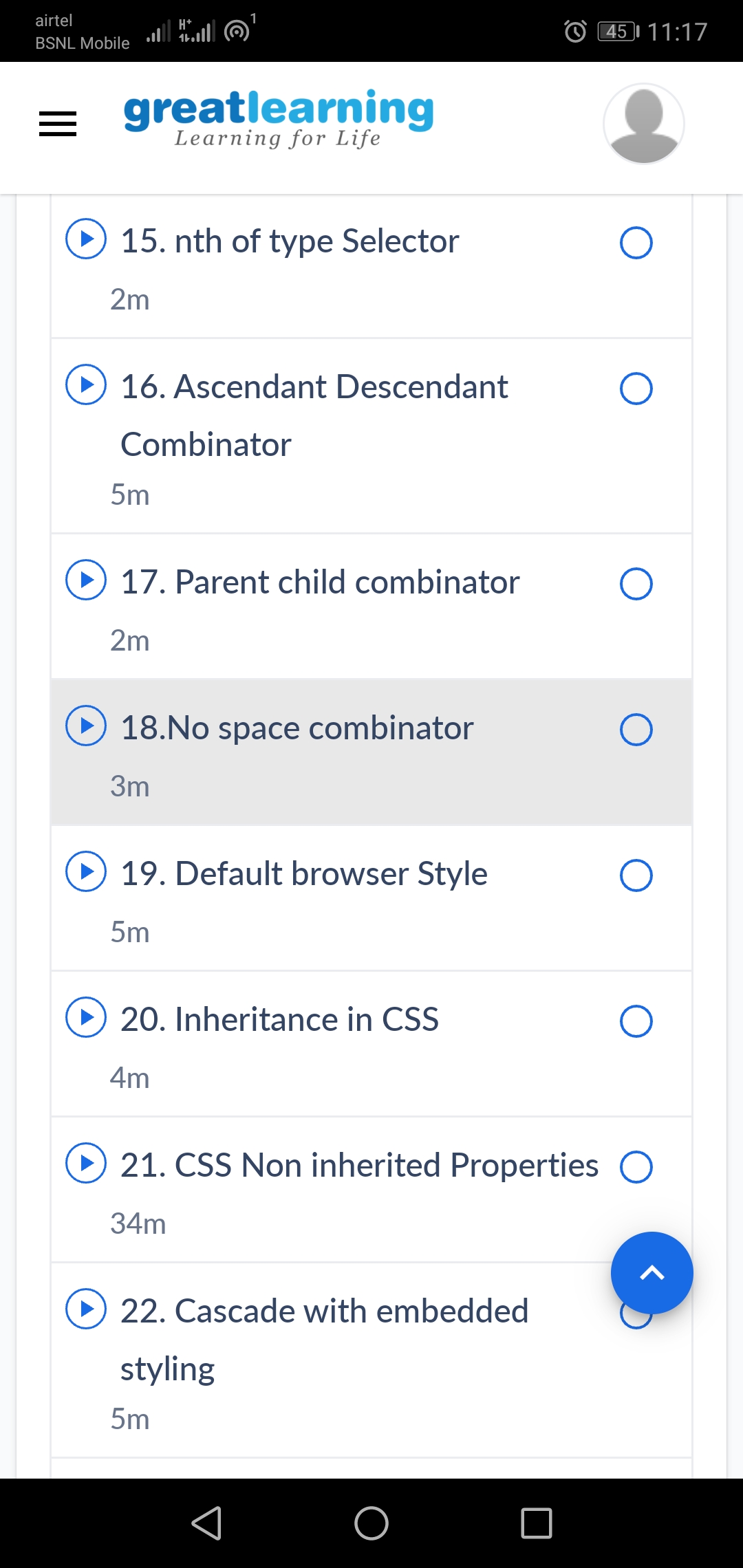
Subject:- **No exam.**

Certification Course Details:

**INTRODUCTION TO FULL STACK DEVELOPMENT.**

Today I have studied :

* Nth of type selector.
* Ascendant Descendant combinator.
* Parent child combinator.
* No space combiner.
* Default browser style.
* Inheritance in css.
* Css non inherited properties.
* Cascade with embedded styling.

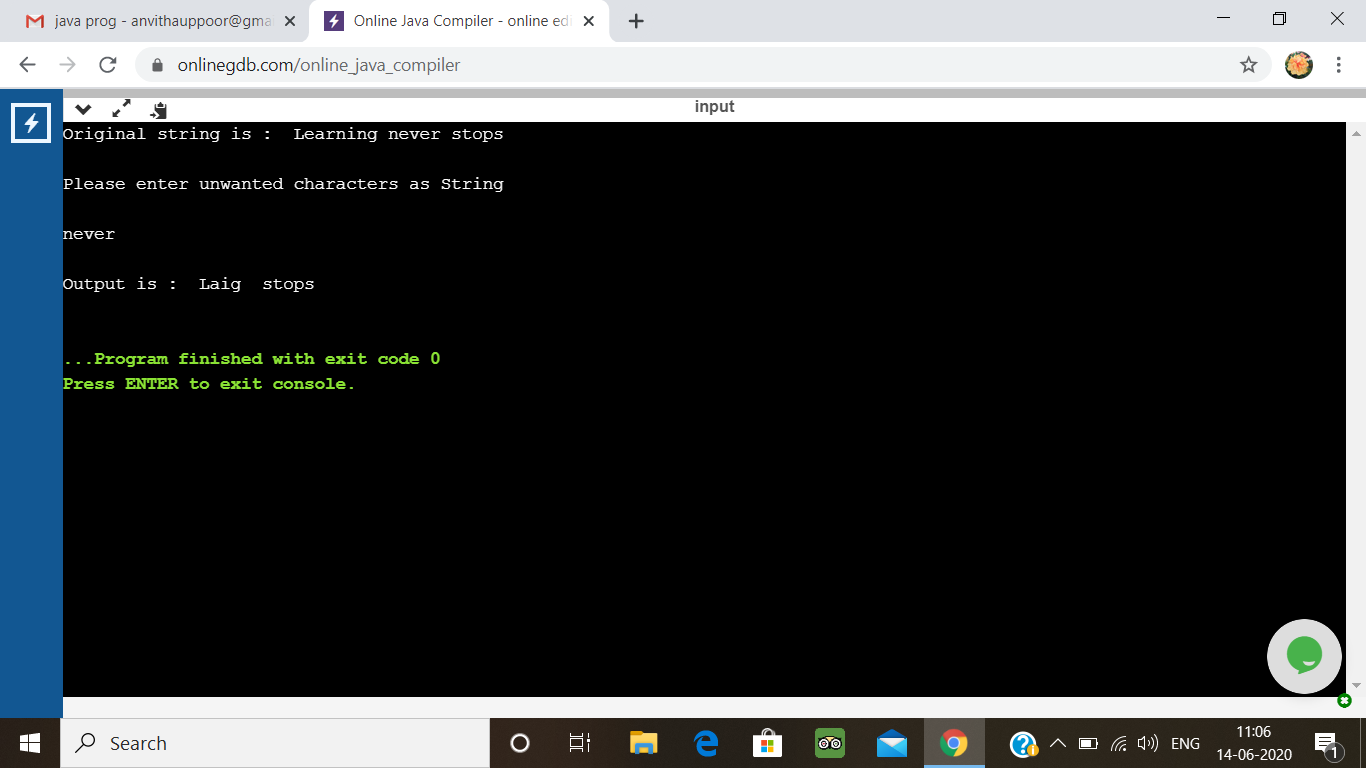


Coding Challenges Details:

1.write a java program to remove specific characters in the String  
If the original string is "Learning never stops" and the user inputs string to remove "estp" then the it should print "Larning nvr o" as output .

import java.util.Scanner;  
public class Main{  
    public static void main(String[] args)  
    {  
        String originalstring="Learning never stops";  
        System.out.println("Original string is :  "+ originalstring);  
        System.out.println("");  
        System.out.println("Please enter unwanted characters as String");  
        System.out.println("");  
        Scanner in =new Scanner(System.in);  
        String removecharacterstring=in.nextLine();  
        String output=removeSpecificChars(originalstring, removecharacterstring);  
        System.out.println("");  
        System.out.print("Output is :  " );  
        System.out.println(output);  
    }  
     
    public static String removeSpecificChars(String originalstring ,String removecharacterstring)  
    {  
        char[] orgchararray=originalstring.toCharArray();  
        char[] removechararray=removecharacterstring.toCharArray();  
        int start,end=0;  
        boolean[]  tempBoolean = new boolean[128];  
        for(start=0;start < removechararray.length;++start)  
        {  
            tempBoolean[removechararray[start]]=true;  
        }  
        for(start=0;start < orgchararray.length;++start)  
        {  
            if(!tempBoolean[orgchararray[start]])  
            {  
                orgchararray[end++]=orgchararray[start];  
            }  
        }  
        return new String(orgchararray,0,end);  
    }  
}

**output:**



2. Write a C Program to implement the Binary

Have the function BinaryReversal(str) take the str parameter being passed, which will be a positive integer, take its binary representation, reverse that string of bits, and then finally return the new reversed string in decimal form. For example: if str is 47 then the binary version of this integer is 101111 but we pad it to be 00101111 (Total number of bits must be multiples of 4). Your program should reverse this binary string which then becomes: 11110100 and then finally return the decimal version of this string, which is 244.  
Examples

Input: 213  
Output: 171

Input: 4567  
Output: 60296

#include <stdio.h>

int main ()

{

int n = 0, num = 0, count = 0, rev\_bits = 0;

printf ("Enter the number: ");

scanf ("%d", &n);

while (n > 0)

{

rev\_bits = rev\_bits << 1;

if (n & 1 == 1)

{

rev\_bits = rev\_bits ^ 1;

}

n = n >> 1;

}

printf ("\nThe reversed resultant = %d\n", rev\_bits);

return 0;

}

**Output:**

