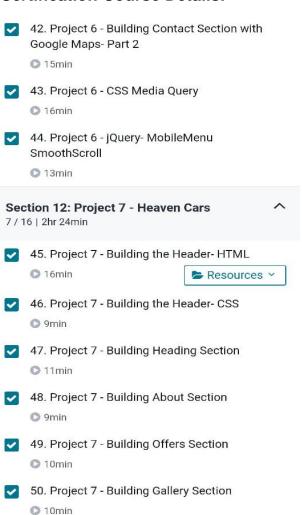
DAILY ONLINE ACTIVITIES SUMMARY

		Name:	Deepa		
8 th Sem	ı	USN:	4AL16CS029		
Online Test Summary					
Big Data Analytics					
30		Score	22		
Certification Course Summary					
Course Seven to Heaven- HTML5 CSS3 jQuery course					
	udemy.com/	Duration		10 hrs	
Coding Challenges					
Problem Statement: 1)TRIPLY LINKED LIST DESCRIPTION. The Programming Question will be posted Tomorrow.					
Status: Completed					
Uploaded the report in Github			Yes		
If yes Repository name			Daily_report		
Uploaded the report in slack			yes		
	Big Da 30 Seven to atement: ill be pos mpleted the repor	Big Data Analytics Certification Co Seven to Heaven- HTML5 CS udemy.com/ Coding Co atement: 1)TRIPLY LINKED LI ill be posted Tomorrow. Inpleted the report in Github ository name	Online Test Summary Big Data Analytics Certification Course Summa Seven to Heaven- HTML5 CSS3 jQuery co udemy.com/ Duration Coding Challenges attement: 1)TRIPLY LINKED LIST DESCRIP ill be posted Tomorrow. Impleted The report in Github Yes Daily_repo	Online Test Summary Big Data Analytics Certification Course Summary Seven to Heaven- HTML5 CSS3 jQuery course udemy.com/ Duration Coding Challenges attement: 1)TRIPLY LINKED LIST DESCRIPTION. The ill be posted Tomorrow. Impleted the report in Github Yes Daily_report	

Online Test Details:



Certification Course Details:



51. Project 7 - Building Contact Section

- 52. Project 7 Building Preloader
 - C 6min
- 53. Project 7 Adding Animations
 - 10min
- 54. Project 7 Making the Menu Works
 - 0 8min
- 55. Project 7 jQuery- SmoothScroll
 - 3min
- 56. Project 7 Making Filterable Gallery
 - O 8min
- 57. Project 7 CSS Media Query
 - 12min
- 58. Project 7 Adding Favicon and Meta Tags
 - 7min
- 59. Project 7 Beautify Code and Adding Auto Prefixes
 - 3min
- 60. Project 7 Launching Website
 - 10min

Teach the world online

Create an online video course, reach students across the globe, and earn money



Coding Challenges Details:

```
Program 1:
#include<stdio.h>
struct SLL;
struct TLL {
struct TLL *top;
struct TLL *bottom;
struct SLL *next;
};
typedef struct TLL tnode;
typedef struct SLL {
char ch;
struct SLL *link;
};
typedef struct SLL snode;
snode *newnode, *ptr, *prev, *temp;
snode *first = NULL, *last = NULL;
tnode *newt, *tlast = NULL, *ttemp;
//--- TLL node---
tnode* create_tnode()
```

```
{
 newt=(tnode*)malloc(sizeof(tnode));
 if (newt == NULL)
 {
   printf("\nMemory was not allocated");
   return 0;
 }
  else
 {
   newt->top = NULL;
   newt->bottom = NULL;
   newt->next = NULL;
   return newt;
 }
}
//---SLL---
snode* create_node(char c)
{
 newnode = (snode *)malloc(sizeof(snode));
 if (newnode == NULL)
 {
   printf("\nMemory was not allocated");
   return 0;
 }
```

```
else
    newnode->ch = c;
    newnode->link = NULL;
    return newnode;
 }
}
//--- insert SLL---
void insert_node_first(char c)
{
  newnode = create_node(c);
  if(tlast->next == NULL)
    tlast->next = newnode;
  if (first == last && first == NULL)
 {
    first = last = newnode;
    first->link = NULL;
    last->link = NULL;
 }
  else
  {
    temp = first;
```

```
first = newnode;
   first->link = temp;
 }
 printf("\n----", c);
}
//---insert TLL---
void insert_Tnode()
{
 newt=create_tnode();
 if (tlast == NULL)
 {
   tlast = newt;
   tlast->next = NULL;
   tlast->top = NULL;
   tlast->bottom = NULL;
 }
 else
 {
   ttemp = tlast;
   tlast = newt;
   tlast->next = NULL;
   tlast->top = ttemp;
```

```
tlast->bottom = NULL;
    ttemp->bottom = tlast;
  }
  printf("\n----CREATED NEW TLL--- ");
}
void main()
{
  char s[100], n;
  int i;
  scanf("%[^;]s",s);
  insert_Tnode();
  for(i = 0; s[i] != '\0'; i++)
  {
    n = s[i];
    if(n == '\n')
      insert_Tnode();
    else
      insert_node_first(n);
  }
  printf("\n%s\n",s);
}
```