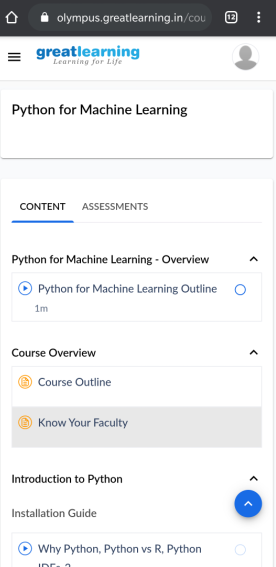
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **24-06-2020** | | | | | **Name:** | **Rakesh M Kotian** | |
| **Sem & Sec** | **8 th sec-b** | | | | | **USN:** | **4al16cs072** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | |  | | | | | | |
| **Max. Marks** | |  | | **Score** | | |  | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Python for machine learning** | | | | | | | |
| **Certificate Provider** | | | **Great learning** | | **Duration** | | | **6 hours** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  Double linked list | | | | | | | | |
| **Status:solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | **Rakeshkotian08** | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

Online Test Details: (Attach the snapshot and briefly write the report for the same)

Certification Course Details: (Attach the snapshot and briefly write the report for the same)



|  |  |
| --- | --- |
| |  | | --- | | #include<iostream>  #include<conio.h>  class node{  public:  int data;  node\* next;  node\* prev;  };  node\* head;  node\* newnode(int x)  {  node\* new\_node=new node();  new\_node->data=x;  new\_node->next=NULL;  new\_node->prev=NULL;  return new\_node;  }  /\*  void insertathead(int x)  {  node\* new\_node=newnode(x);  if (head==NULL)  {  head=new\_node;  return;  }  head->prev=new\_node;  new\_node->next=head;  head=new\_node;    }  \*/  node\* insertattail( node\* head,int x)  {  node\* new\_node=newnode(x);  if(head==NULL)  {  head=new\_node;  return head;  }  else  {  node\* temp2=head;  while(temp2->next!=NULL)  {  temp2=temp2->next;  }  temp2->next=newnode(x);  new\_node->prev=temp2;  new\_node->next=NULL;    }  return head;  }  void printt()  {  node\* temp =head;  while(temp!=NULL)  {  printf("%d",temp->data);  temp=temp->next;  }  printf("\n");  }  void print(node\* head)  {  while(head !=NULL)  {  printf("%d",head->data);  head=head->next;  }  }  void reverseprint()  {  node\* temp=head;  if(temp==NULL) return;  while(temp->next!=NULL) temp=temp->next;  while(temp!=NULL)  {  printf("%d",temp->data);  temp=temp->prev;  }  }  int main()  {  head=NULL;  /\*insertathead(1);  insertathead(2);  insertathead(3);\*/  head=insertattail(head,1);  head=insertattail(head,2);  head=insertattail(head,3);  print(head);  // printt();  // reverseprint();  getch();  return 0;  } | |