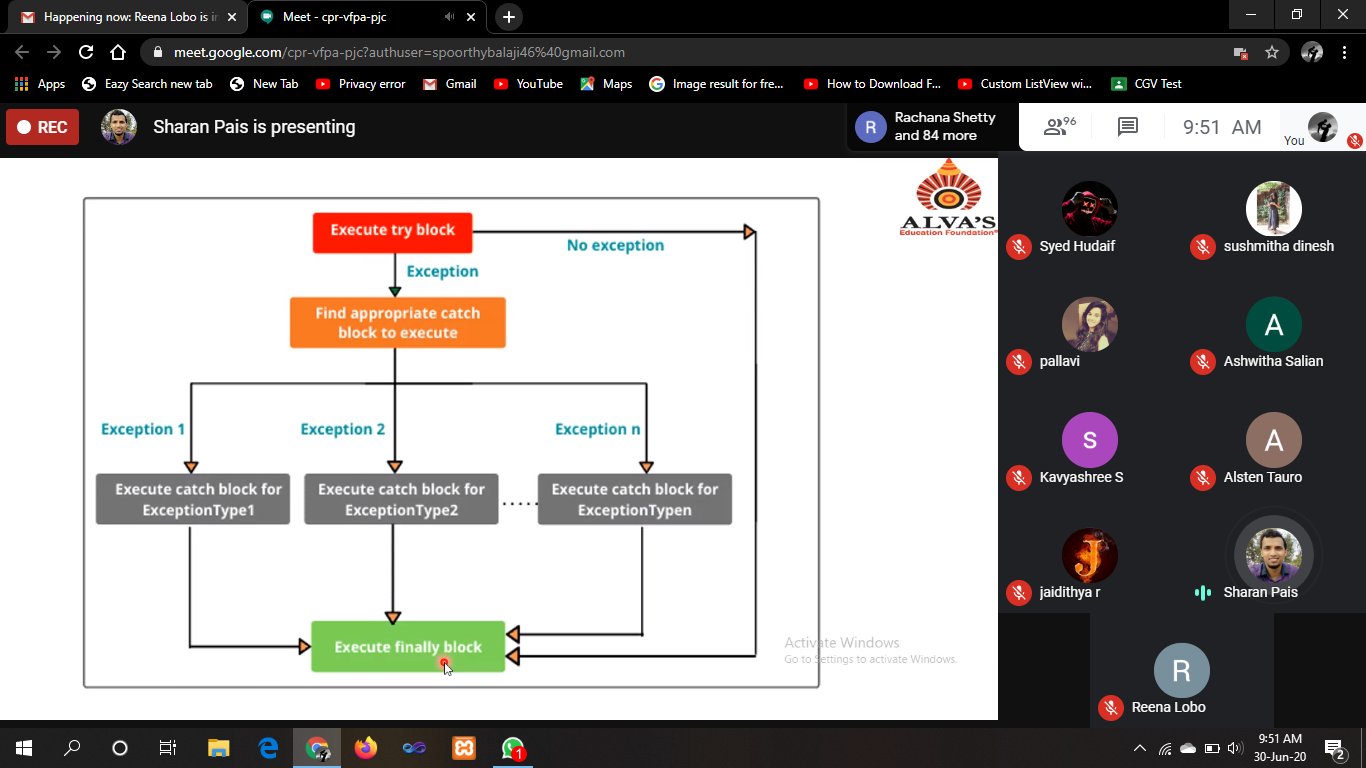
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | | **30/06/2020** | | | | **Name:** | **Prajna** | |
| **Sem & Sec** | | **6th & A** | | | | **USN:** | **4al17cs059** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | | **JAVA** | | | | | |
| **Max. Marks** | | | **-** | **Score** | | | **-** | |
| **Pre-placement Training Summary** | | | | | | | | |
| **Topic** | **JAVA** | | | | | | | |
| **Faculty** | Sharan Pais | | | | **Duration** | | | **4hours** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement**: 3 programs | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | <https://github.com/prajna-nayak-098/Daily-Report> | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

**SNAPSHOTS**

****

**ONLINE CODING**

**1. Python Program to Read a Linked List in Reverses**

class Node:

def \_\_init\_\_(self, data):

self.data = data

self.next = None

class LinkedList:

def \_\_init\_\_(self):

self.head = None

self.last\_node = None

def append(self, data):

if self.last\_node is None:

self.head = Node(data)

self.last\_node = self.head

else:

self.last\_node.next = Node(data)

self.last\_node = self.last\_node.next

def display(self):

current = self.head

while current:

print(current.data, end = ' ')

current = current.next

def reverse\_llist(llist):

before = None

current = llist.head

if current is None:

return

after = current.next

while after:

current.next = before

before = current

current = after

after = after.next

current.next = before

llist.head = current

a\_llist = LinkedList()

data\_list = input('Enter the elements in the linked list:\n ').split()

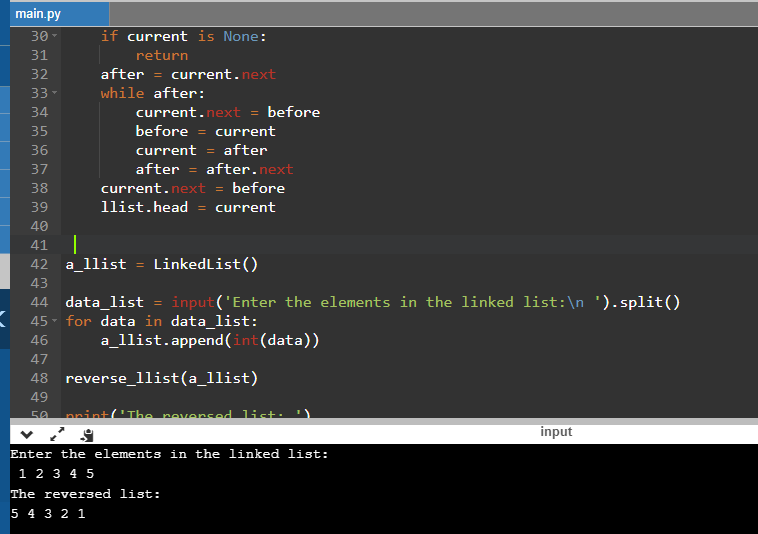
for data in data\_list:

a\_llist.append(int(data))

reverse\_llist(a\_llist)

print('The reversed list: ')

a\_llist.display()



**2. Write a Java Program to determine whether one string is a rotation of another.**

public class Main {

public static booleancheckRotation(String st1, String st2) {

if (st1.length() != st2.length()) {

return false;

}

String st3 = st1 + st1;

if (st3.contains(st2))

return true;

else

return false;

}

public static void main(String[] args) {

String str1 = "avajava";

String str2 = "javaava";

System.out.println("Checking if a string is rotation of another");

if (checkRotation(str1, str2)) {

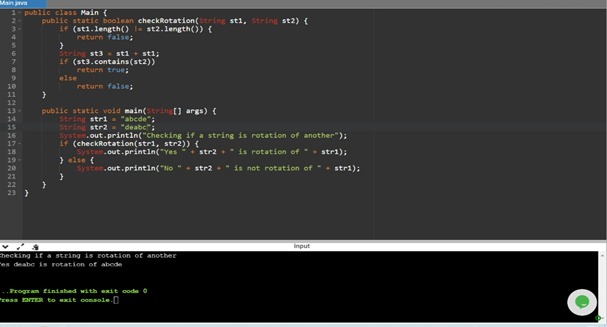
System.out.println("Yes " + str2 + " is rotation of " + str1);

} else {

System.out.println("No " + str2 + " is not rotation of " + str1);

}

}



**3. Write a C Program to generate first n Ugly Numbers**

# include <stdio.h>

# include <string.h>

int main()

{

intn,x=0;

printf("Input an integer number: ");

scanf("%d",&n);

if (n <= 0) {

printf("Input a correct number.");

}

while (n != 1)

{

if (n % 5 == 0)

{

n /= 5;

}

else if (n % 3 == 0)

{

n /= 3;

}

else if (n % 2 == 0)

{

n /= 2;

}

else

{

printf("It is not an ugly number.\n");

x = 1;

break;

}

}

if (x==0)

{

printf("It is an ugly number.\n");

}

}

