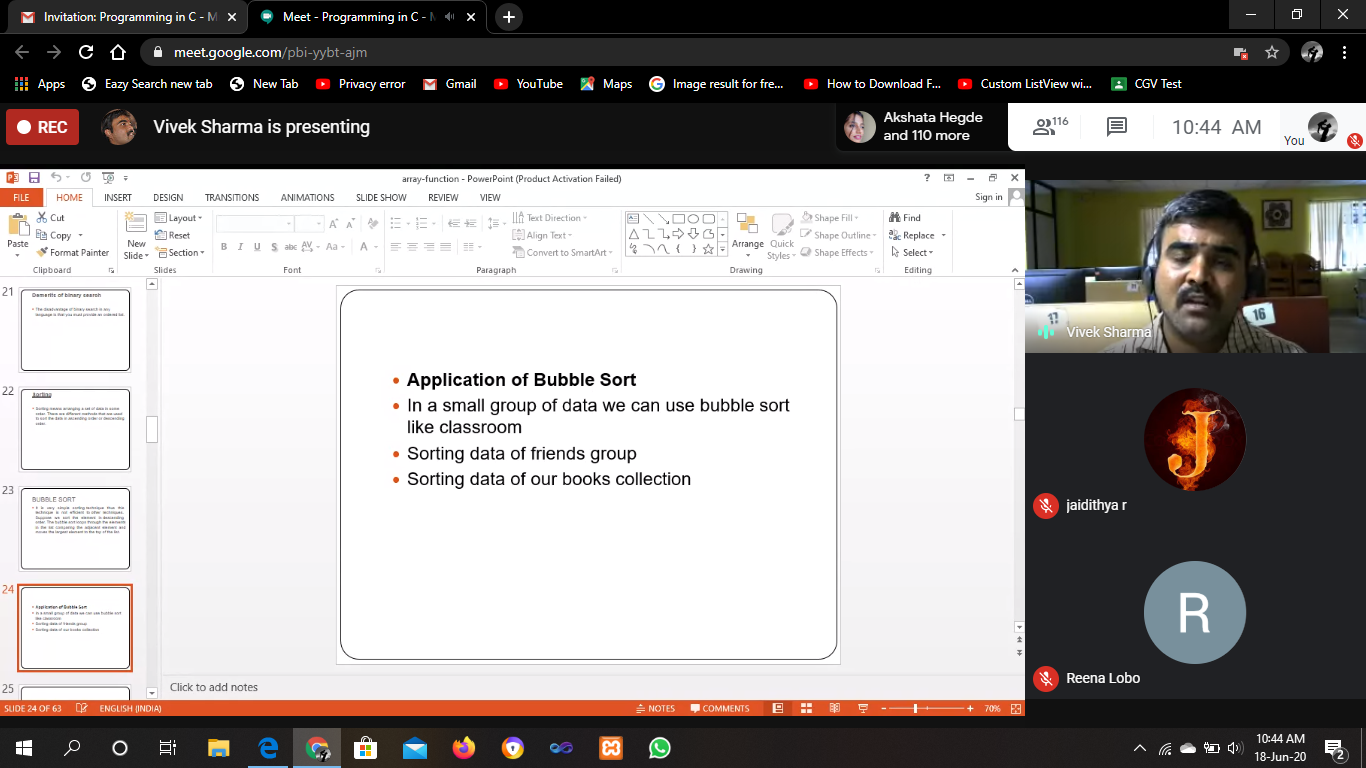
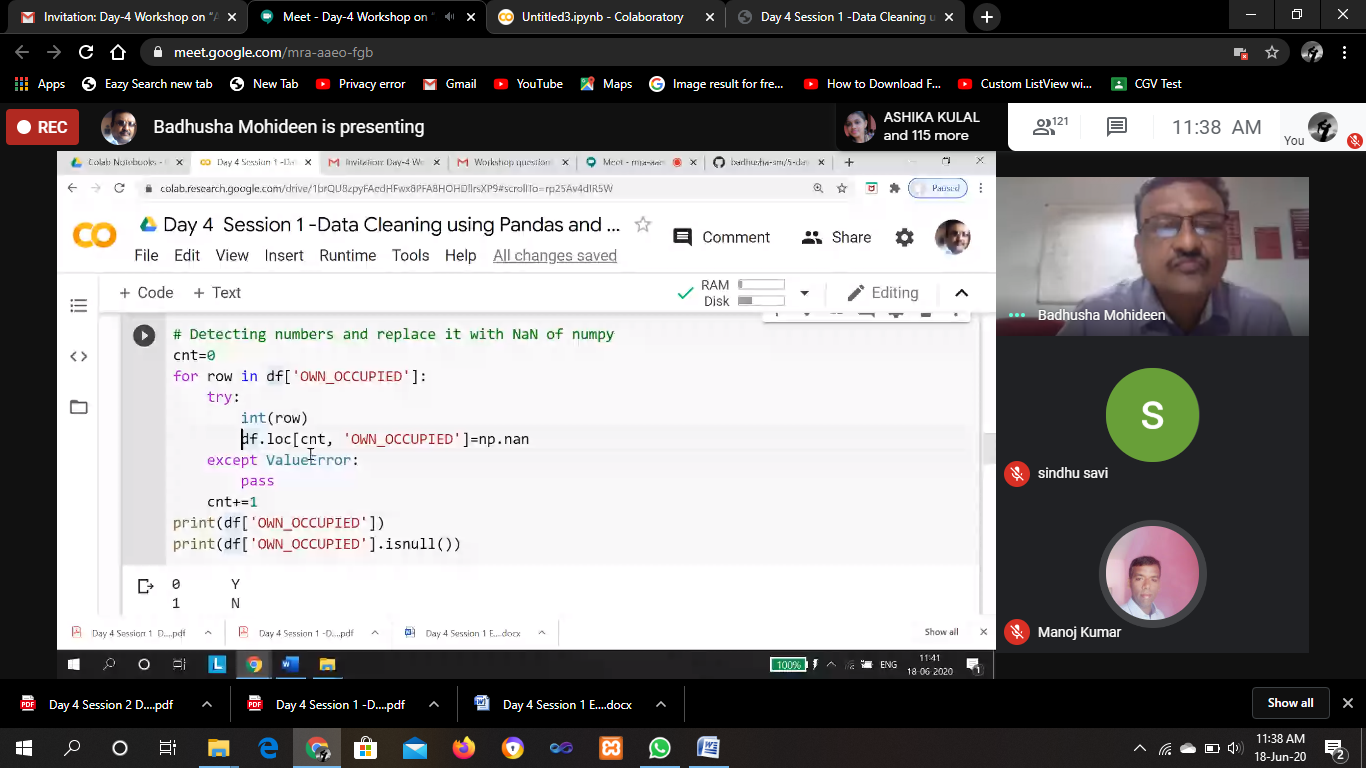
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **18/06/2020** | | | | | **Name:** | **Prajna** | |
| **Sem & Sec** | **6th & A** | | | | | **USN:** | **4al17cs059** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | |  | | | | | | |
| **Max. Marks** | |  | | **Score** | | |  | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | Programming in C  Applications of python in DA and ML | | | | | | | |
| **Certificate Provider** | | | **Vivek Sharma**  **Dr. Mohideen**  **Badusha** | | **Duration** | | | **4hours** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** 3 Programs | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | **Online Coding Repository**  <https://github.com/prajna-nayak-098/Daily-Report>  **Workshop on Application of python**  <https://github.com/prajna-nayak-098/Applications-of-python-in-DA-and-ML1> | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

**SNAPSHOTS**

****

****

ONLINE CODING

**1. Python Program to Count Number of Lowercase Characters in a String**

string=input("Enter string:")

count=0

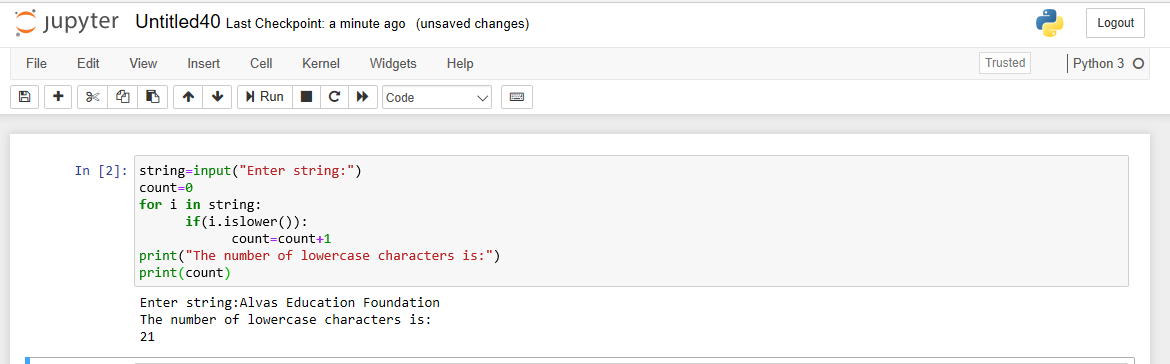
for i in string:

if(i.islower()):

count=count+1

print("The number of lowercase characters is:")

print(count)



**2. Write a Java program to Check if a binary tree is binary search tree or not**

class Node

{

int data;

Node left, right;

public Node(int item)

{

data = item;

left = right = null;

}

}

public class BinaryTree

{

Node root;

boolean isBST() {

return isBSTUtil(root, Integer.MIN\_VALUE,

Integer.MAX\_VALUE);

}

boolean isBSTUtil(Node node, int min, int max)

{

if (node == null)

return true;

if (node.data < min || node.data > max)

return false;

return (isBSTUtil(node.left, min, node.data-1) &&

isBSTUtil(node.right, node.data+1, max));

}

public static void main(String args[])

{

BinaryTree tree = new BinaryTree();

tree.root = new Node(7);

tree.root.left = new Node(2);

tree.root.right = new Node(5);

tree.root.left.left = new Node(1);

tree.root.left.right = new Node(3);

if (tree.isBST())

System.out.println("IS BST");

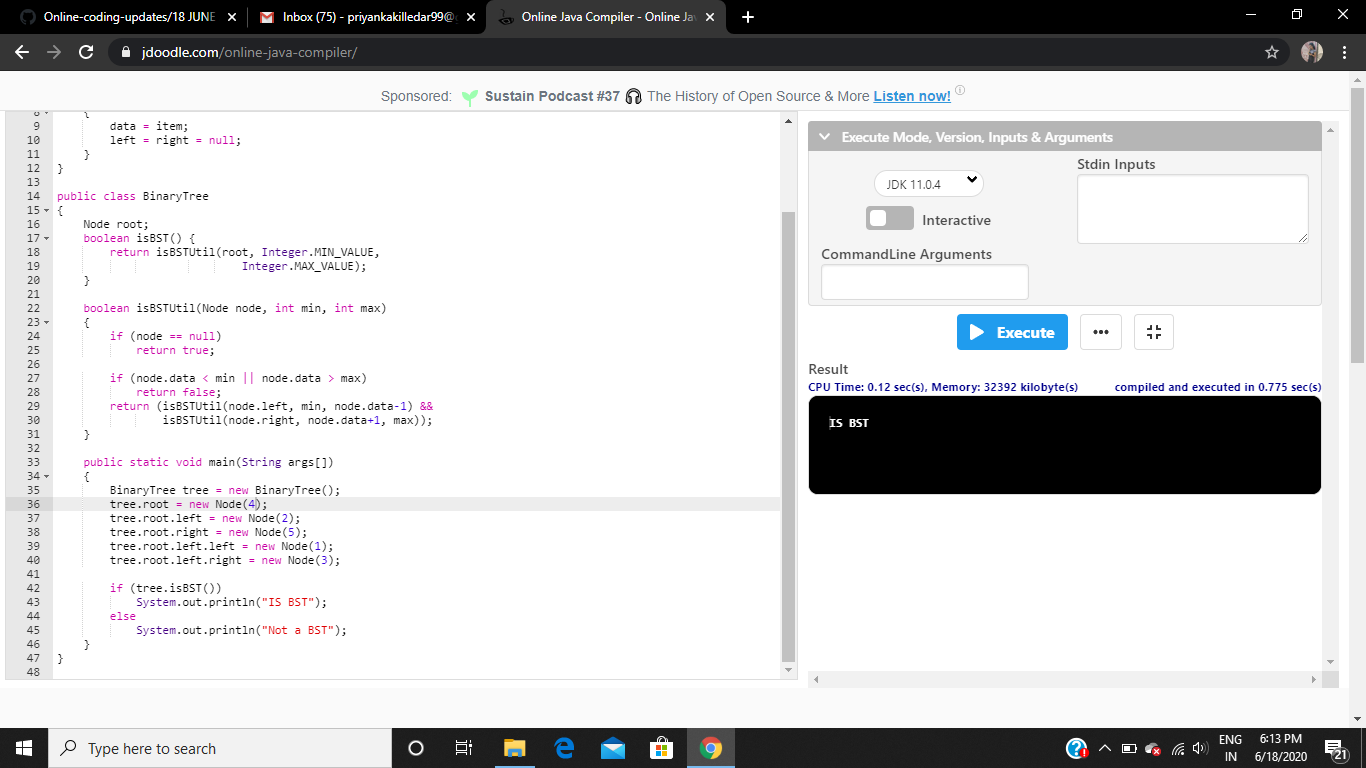
else

System.out.println("Not a BST");

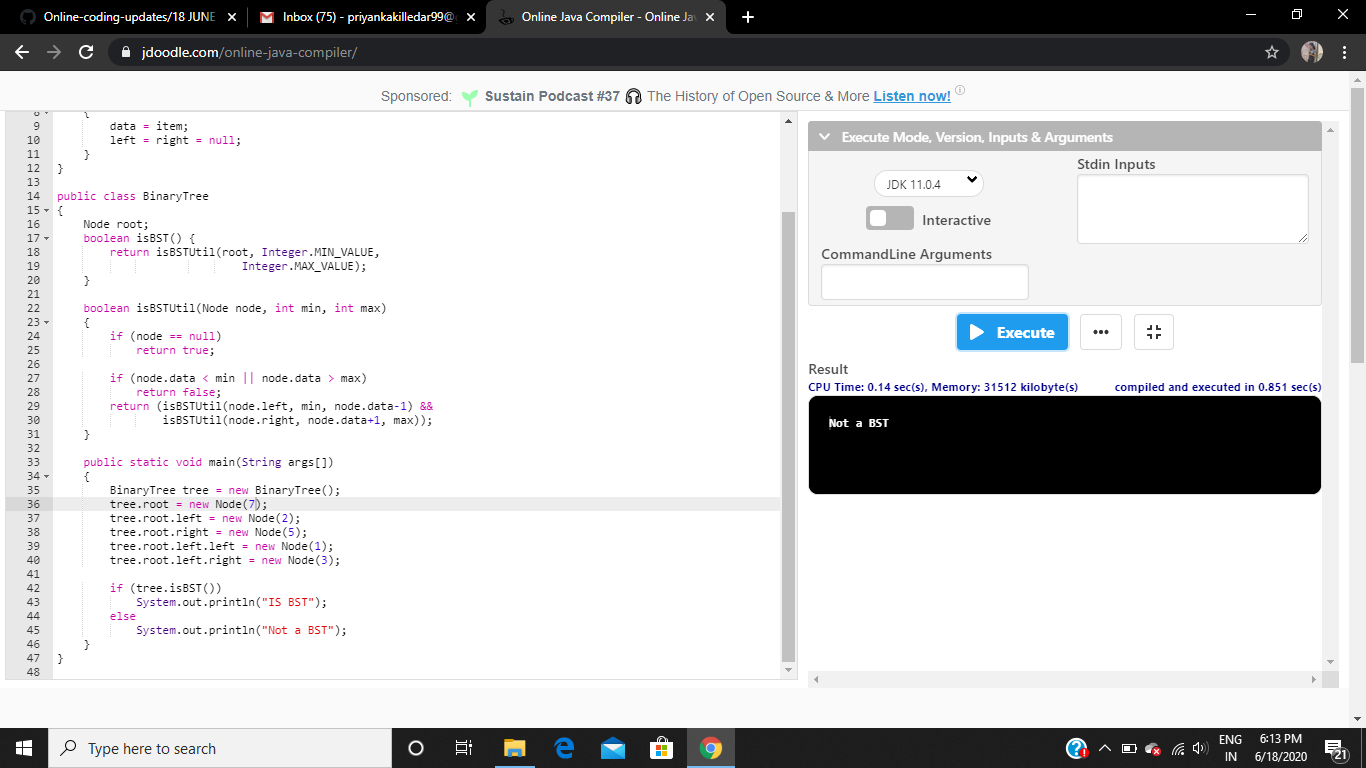
}

}

**Node= 4,2,5,1,3 IS BST**



**Node= 7,2,5,1,3 IS NOT BST**



**3. Write a C Program to generate first N Magic Numbers**.

#include<stdio.h>

int nthMagicNo(int n)

{

int pow = 1, answer = 0;

while (n)

{

pow = pow\*5;

if (n & 1)

answer += pow;

printf("%d ",pow);

n >>= 1;

}

}

int main()

{

printf("Input: n = 1\n");

printf("output: ");

nthMagicNo(1);

printf("\nInput: n = 2\n");

printf("output: ");

nthMagicNo(2);

printf("\nInput: n = 3\n");

printf("output: ");

nthMagicNo(3);

printf("\nInput: n = 8\n");

printf("output: ");

nthMagicNo(8);

return 0;

}

