**C-DAC Mumbai Date 26/09/2024**

**Subject: Algorithm and Data Structure**

**Assignment 1**

**Solve the assignment with following thing to be added in each question.**

-Program

-Flow chart

-Explanation

-Output

-Time and Space complexity

1. Printing Patterns

Problem: Write a Java program to print patterns such as a right triangle of stars.

Test Cases:

Input: n = 3

Output:

\*

\*\*

\*\*\*

Input: n = 5

Output:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

import java.util.Scanner;

class StarPattern {

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

System.out.println("Enter no. of rows");

int n=sc.nextInt();

for(int i=1;i<=n;i++){

for(int j=1;j<=i;j++){

System.out.print("\*");

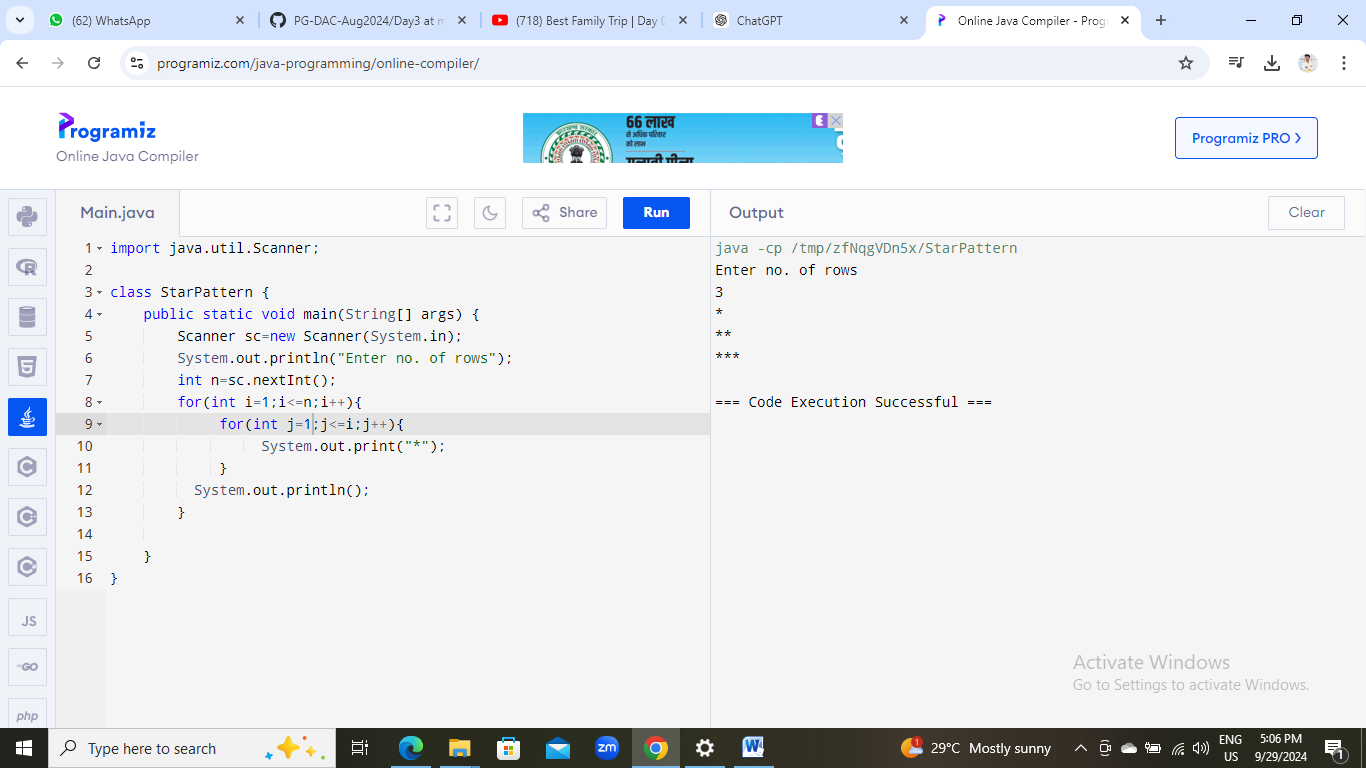
}

System.out.println();

}

}

}



2. Remove Array Duplicates

Problem: Write a Java program to remove duplicates from a sorted array and return the new length of the array.

Test Cases:

Input: arr = [1, 1, 2]

Output: 2

Input: arr = [0, 0, 1, 1, 2, 2, 3, 3]

Output: 4

class RemoveDuplicates {

public static void main(String[] args) {

int arr[] = {0, 0, 1, 1, 2, 2, 3, 3};

int j=0;

for(int i=0;i<arr.length-1;i++){

if(arr[i]!=arr[i+1])

arr[j++]=arr[i];

}

arr[j++]=arr[arr.length - 1];

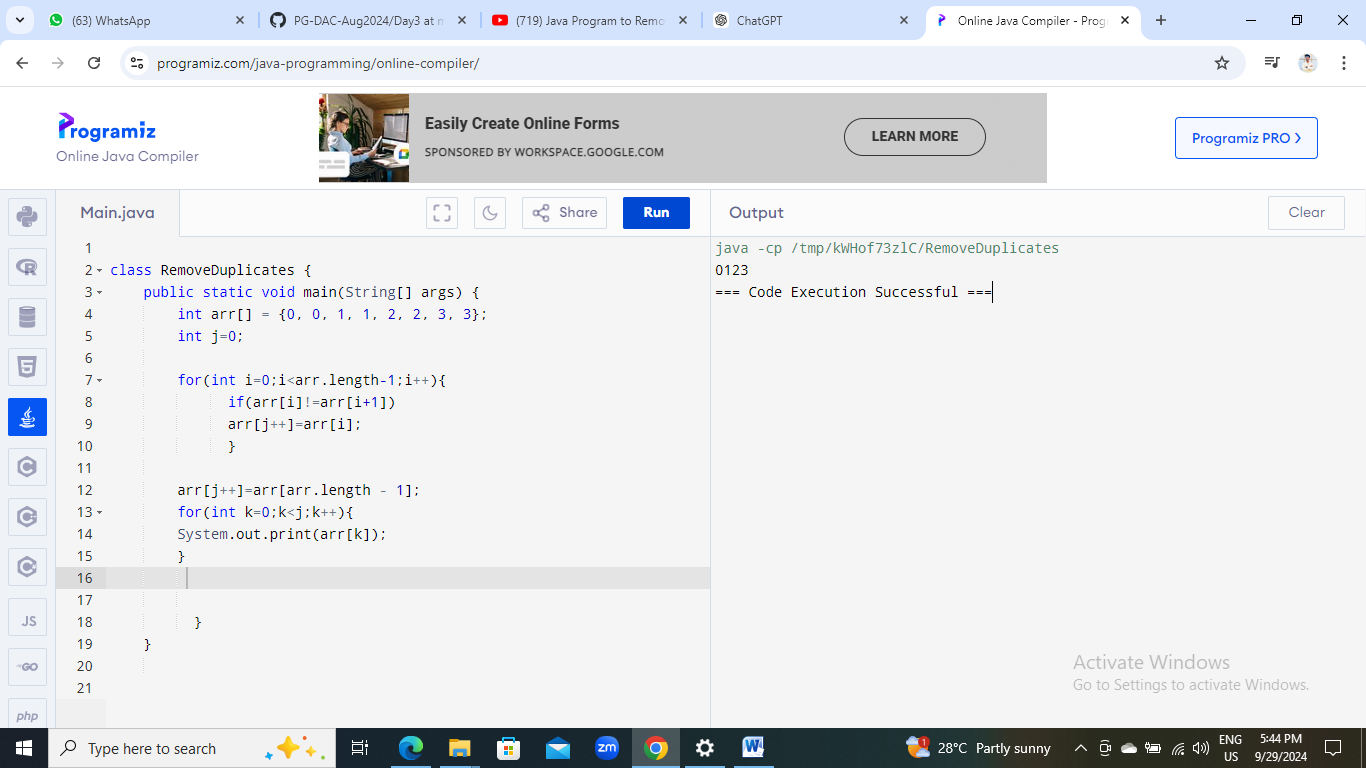
for(int k=0;k<j;k++){

System.out.print(arr[k]);

}

}

}



3. Remove White Spaces from String

Problem: Write a Java program to remove all white spaces from a given string.

Test Cases:

Input: "Hello World"

Output: "HelloWorld"

Input: " Java Programming "

Output: "JavaProgramming"

class RemoveSpaces{

public static void main(String[] args){

removeSpaces("Hello World Java Programming");

}

public static void removeSpaces(String s){

for(char c:s.toCharArray()){

if(c !=' '){

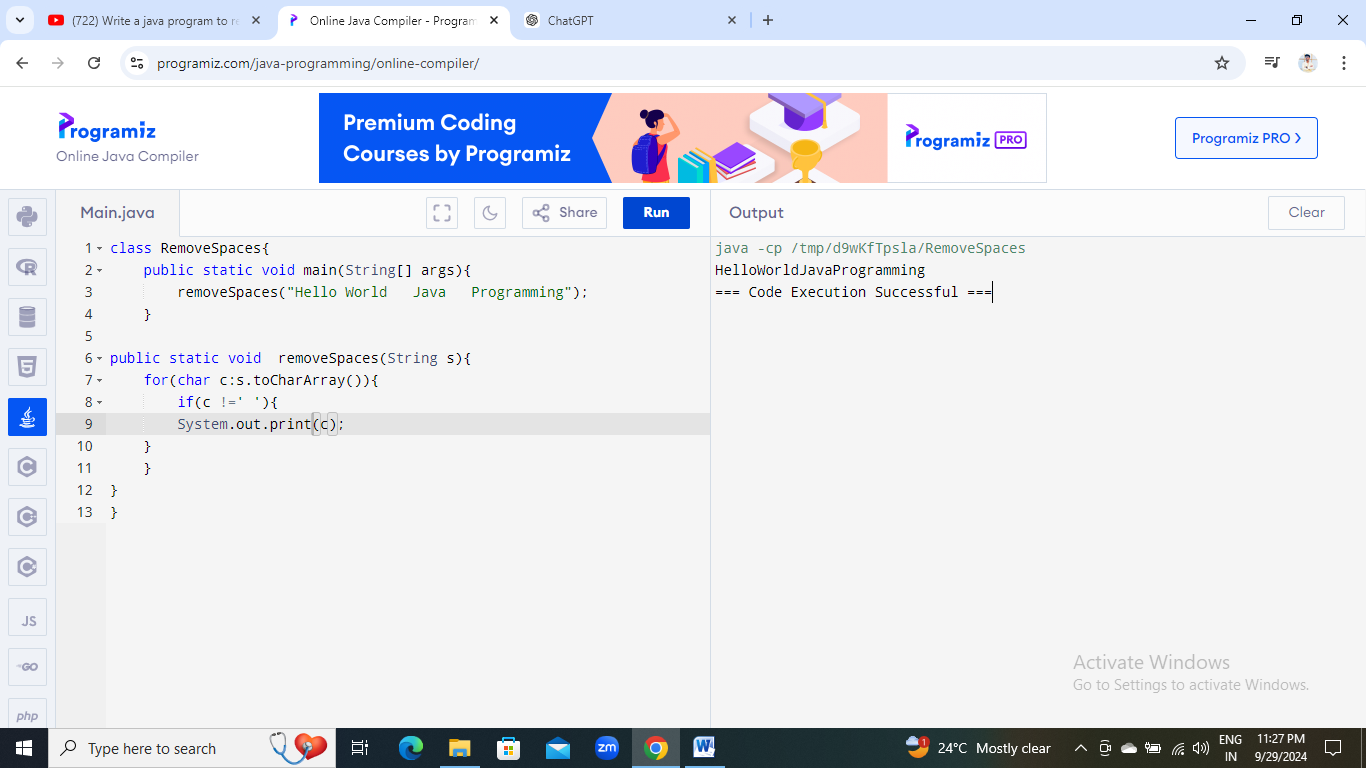
System.out.print(c);

}

}

}

}



4. Reverse a String

Problem: Write a Java program to reverse a given string.

Test Cases:

Input: "hello"

Output: "olleh"

Input: "Java"

Output: "avaJ"

class ReverseString{

public static void main(String[] args){

String s="hello";

System.out.println(reverseString(s));

}

private static String reverseString(String s){

if(s==null || s.length()<=1){

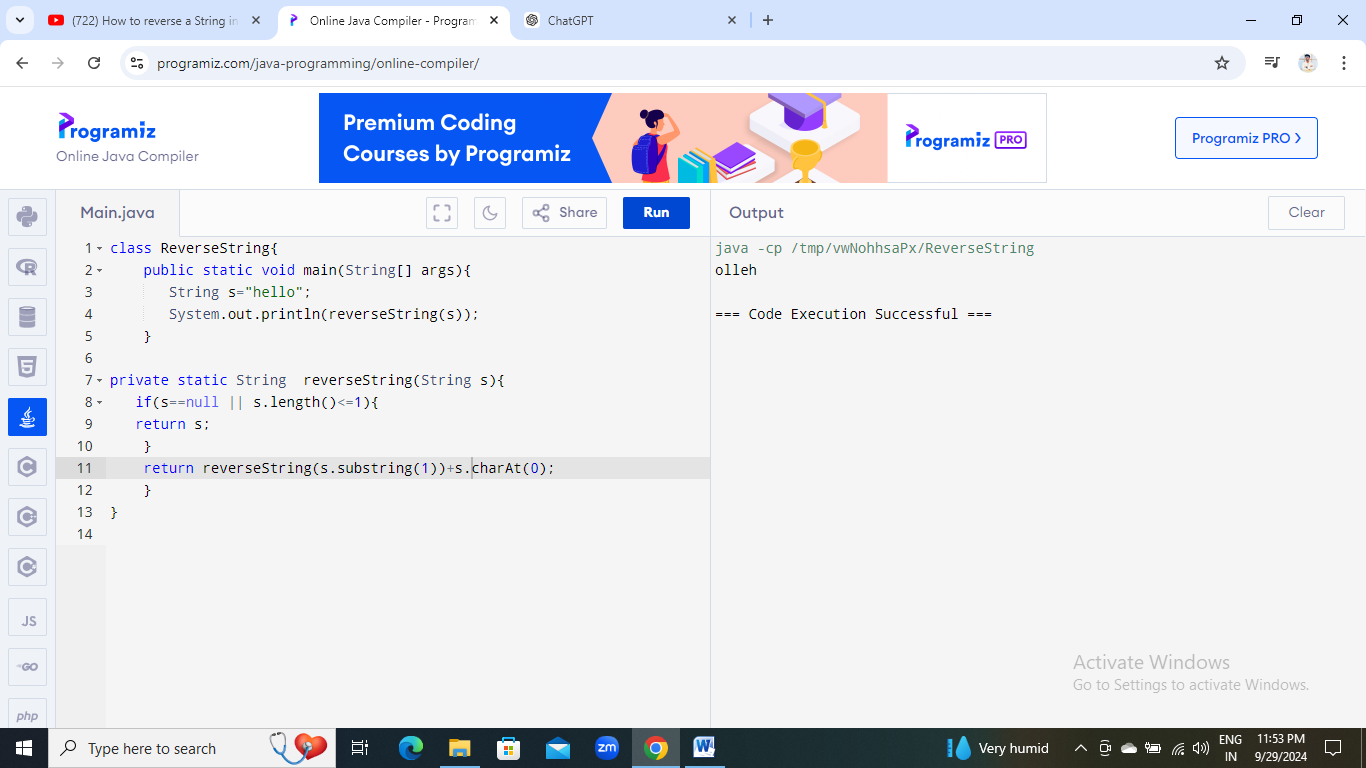
return s;

}

return reverseString(s.substring(1))+s.charAt(0);

}

}



5. Reverse Array in Place

Problem: Write a Java program to reverse an array in place.

Test Cases:

Input: arr = [1, 2, 3, 4]

Output: [4, 3, 2, 1]

Input: arr = [7, 8, 9]

Output: [9, 8, 7]

class ReverseArray{

public static void main(String[] args){

int[] arr={1,2,3,4};

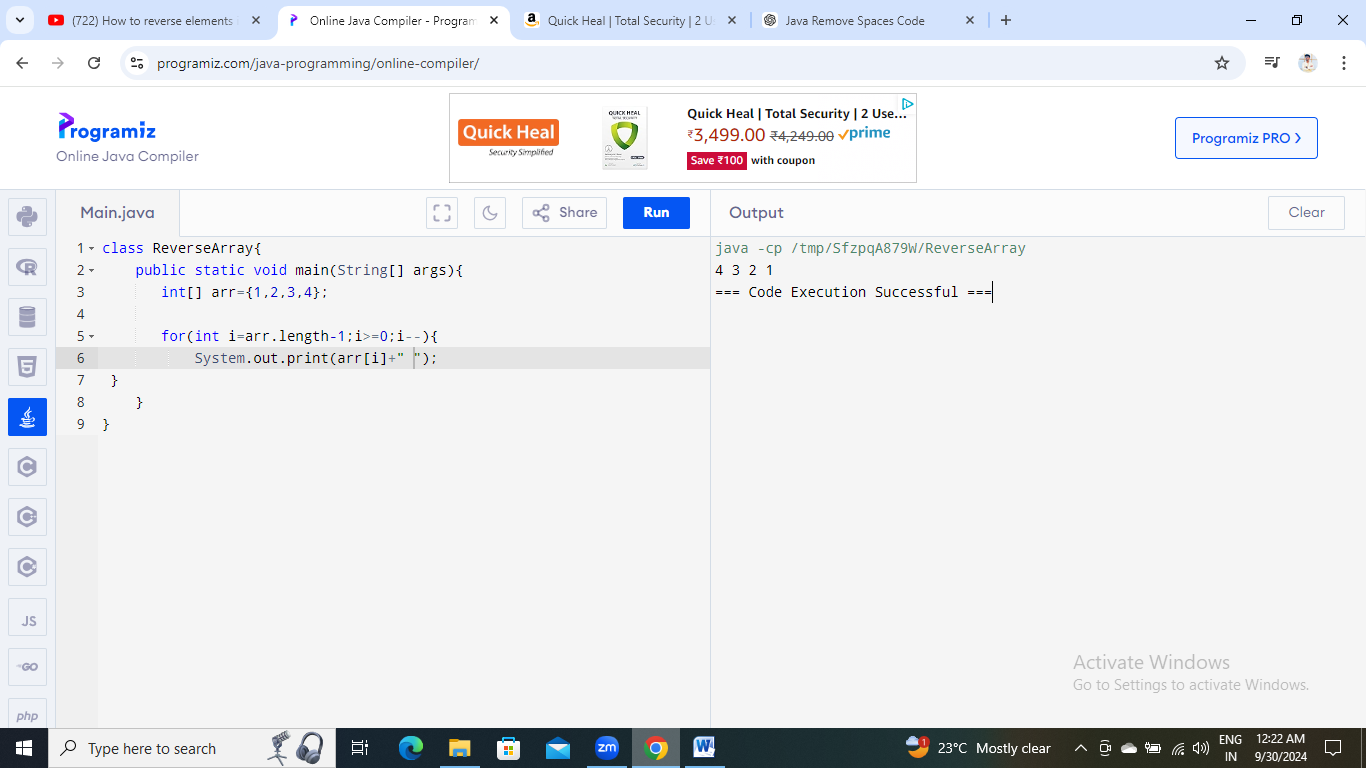
for(int i=arr.length-1;i>=0;i--){

System.out.print(arr[i]+" ");

}

}

}



6. Reverse Words in a String

Problem: Write a Java program to reverse the words in a given sentence.

Test Cases:

Input: "Hello World"

Output: "World Hello"

Input: "Java Programming"

Output: "Programming Java"

7. Reverse a Number

Problem: Write a Java program to reverse a given number.

Test Cases:

Input: 12345

Output: 54321

Input: -9876

Output: -6789

import java.util.Scanner;

class Reversenum{

static int n,r;

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter the number");

n=sc.nextInt();

while(n>0){

r=n%10;

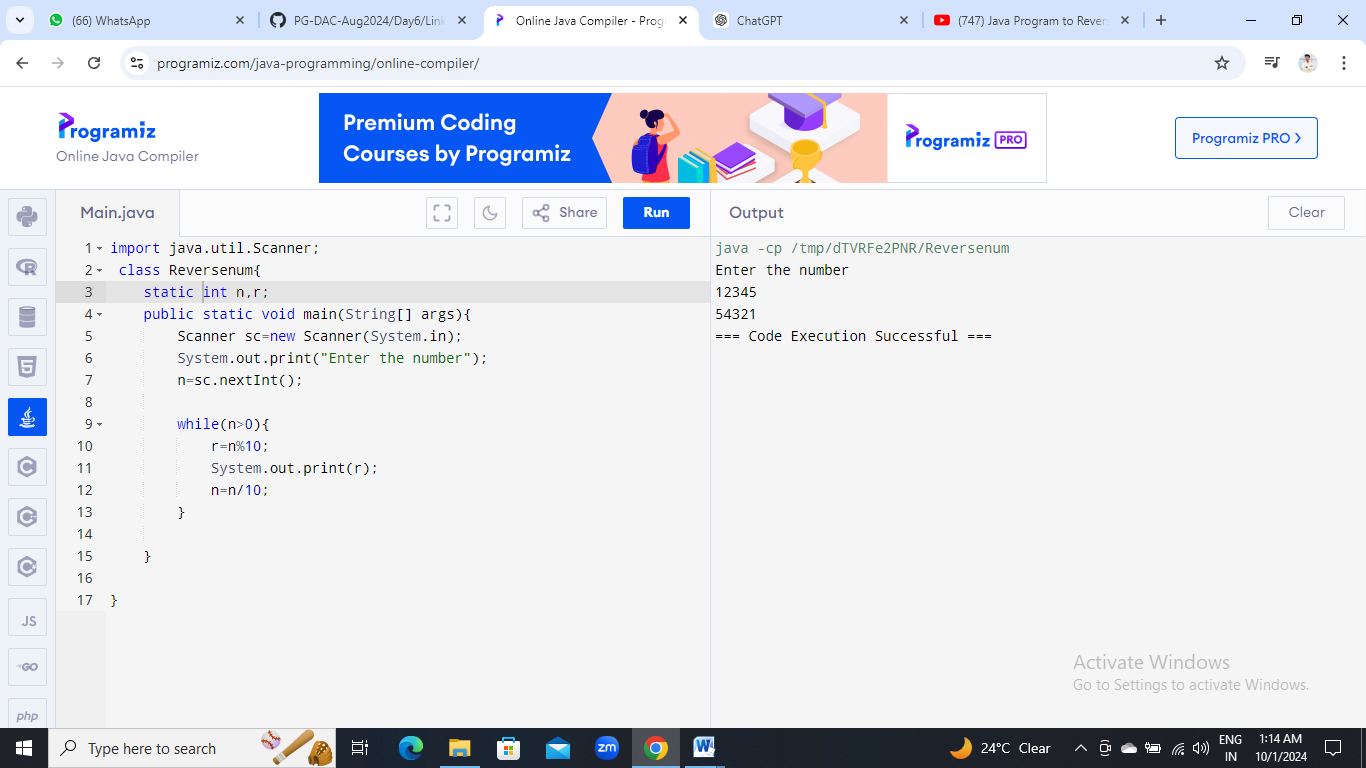
System.out.print(r);

n=n/10;

}

}

}



8. Array Manipulation

Problem: Perform a series of operations to manipulate an array based on range update queries. Each query adds a value to a range of indices.

Test Cases:

Input: n = 5, queries = [[1, 2, 100], [2, 5, 100], [3, 4, 100]]

Output: 200

Input: n = 4, queries = [[1, 3, 50], [2, 4, 70]]

Output: 120

9. String Palindrome

Problem: Write a Java program to check if a given string is a palindrome.

Test Cases:

Input: "madam"

Output: true

Input: "hello"

Output: false

import java.util.Scanner;

class Palindrome{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.println("Enter the String");

String str=sc.nextLine();

String org=str;

String rev="";

int len=str.length();

for(int i=len-1;i>=0;i--){

rev=rev+str.charAt(i);

}

if(org.equals(rev)){

System.out.println("yes");

}

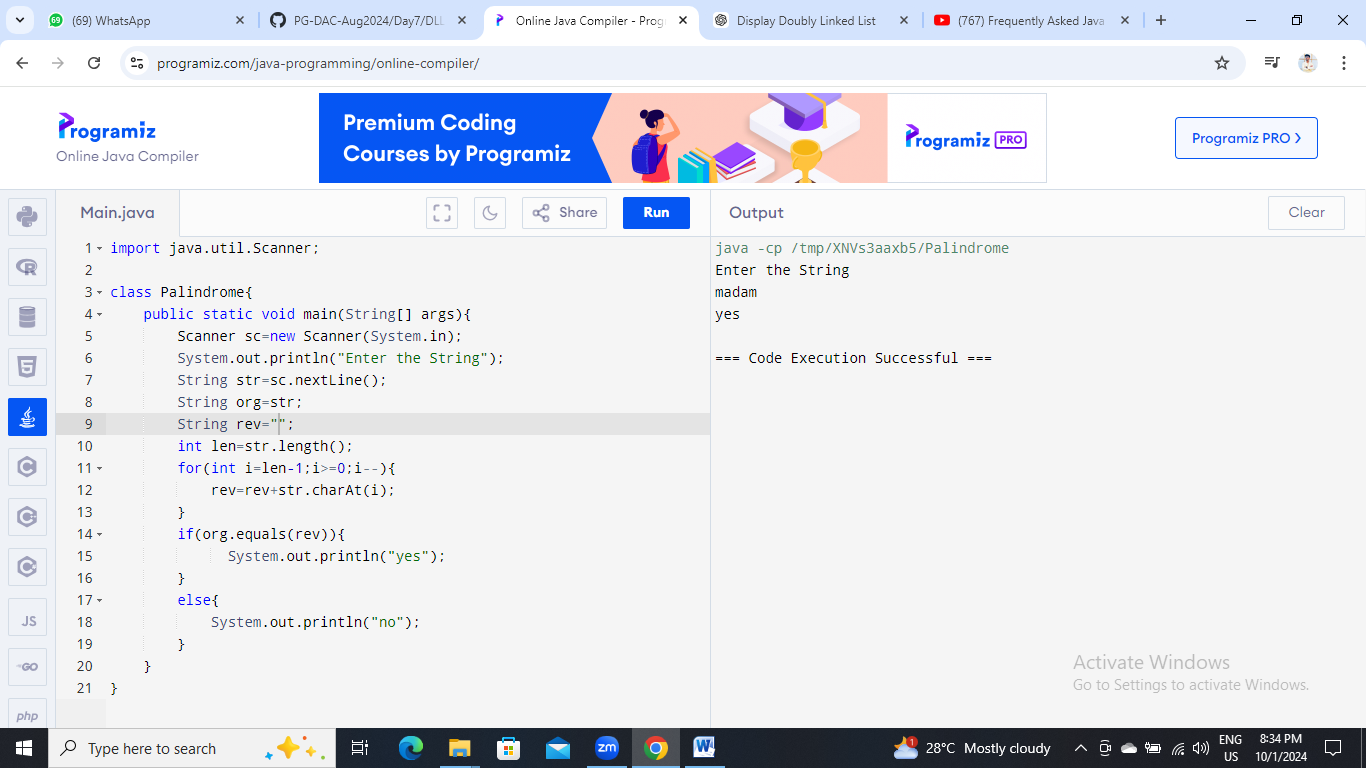
else{

System.out.println("no");

}

}

}



Here’s a continuation of the list of assignment questions starting from question 21, with two test cases for each:

10. Array Left Rotation

Problem: Write a Java program to rotate an array to the left by d positions.

Test Cases:

Input: arr = [1, 2, 3, 4, 5], d = 2

Output: [3, 4, 5, 1, 2]

Input: arr = [10, 20, 30, 40], d = 1

Output: [20, 30, 40, 10]

class ShiftArray{

public static void main(String[] args){

int[] arr=new int[] {1,2,3,4,5};

int n=2;

for(int i=0;i<n;i++){

int first=arr[0],j;

for(j=0;j<arr.length-1;j++){

arr[j]=arr[j+1];

}

arr[j]=first;

}

for(int i=0;i<arr.length;i++){

System.out.print(arr[i]+" ");

}

}

}

