ASSIGNMENT-2

1 . Write a program to enter the marks of a student in four subjects. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is 60>= and <75, then the grade is First Division. If aggregate is 50 >= and <60, then the grade is Second Division. If aggregate is 40>= and <50, then the grade is Third Division. Else the grade is Fail.

Sample Input & Output:

Enter the marks in python: 90

Enter the marks in c programming: 91

Enter the marks in Mathematics: 92

Enter the marks in Physics: 93

Total= 366

Aggregate = 91.5

DISTINCTION

Test cases:

a) 18, 76,93,65

b) 73,78,79,75

c) 98,106,120,95

d) 96,73, -85,95

e) 78,59.8,76,79

CODE:

import java.util.Scanner;

public class StudentGrades {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the marks in Python: ");

double pythonMarks = scanner.nextDouble();

System.out.print("Enter the marks in C Programming: ");

double cProgrammingMarks = scanner.nextDouble();

System.out.print("Enter the marks in Mathematics: ");

double mathematicsMarks = scanner.nextDouble();

System.out.print("Enter the marks in Physics: ");

double physicsMarks = scanner.nextDouble();

double total = pythonMarks + cProgrammingMarks + mathematicsMarks + physicsMarks;

double aggregate = total / 4;

String grade;

if (aggregate > 75) {

grade = "DISTINCTION";

} else if (aggregate >= 60 && aggregate < 75) {

grade = "First Division";

} else if (aggregate >= 50 && aggregate < 60) {

grade = "Second Division";

} else if (aggregate >= 40 && aggregate < 50) {

grade = "Third Division";

} else {

grade = "Fail";

}

System.out.println("Total = " + total);

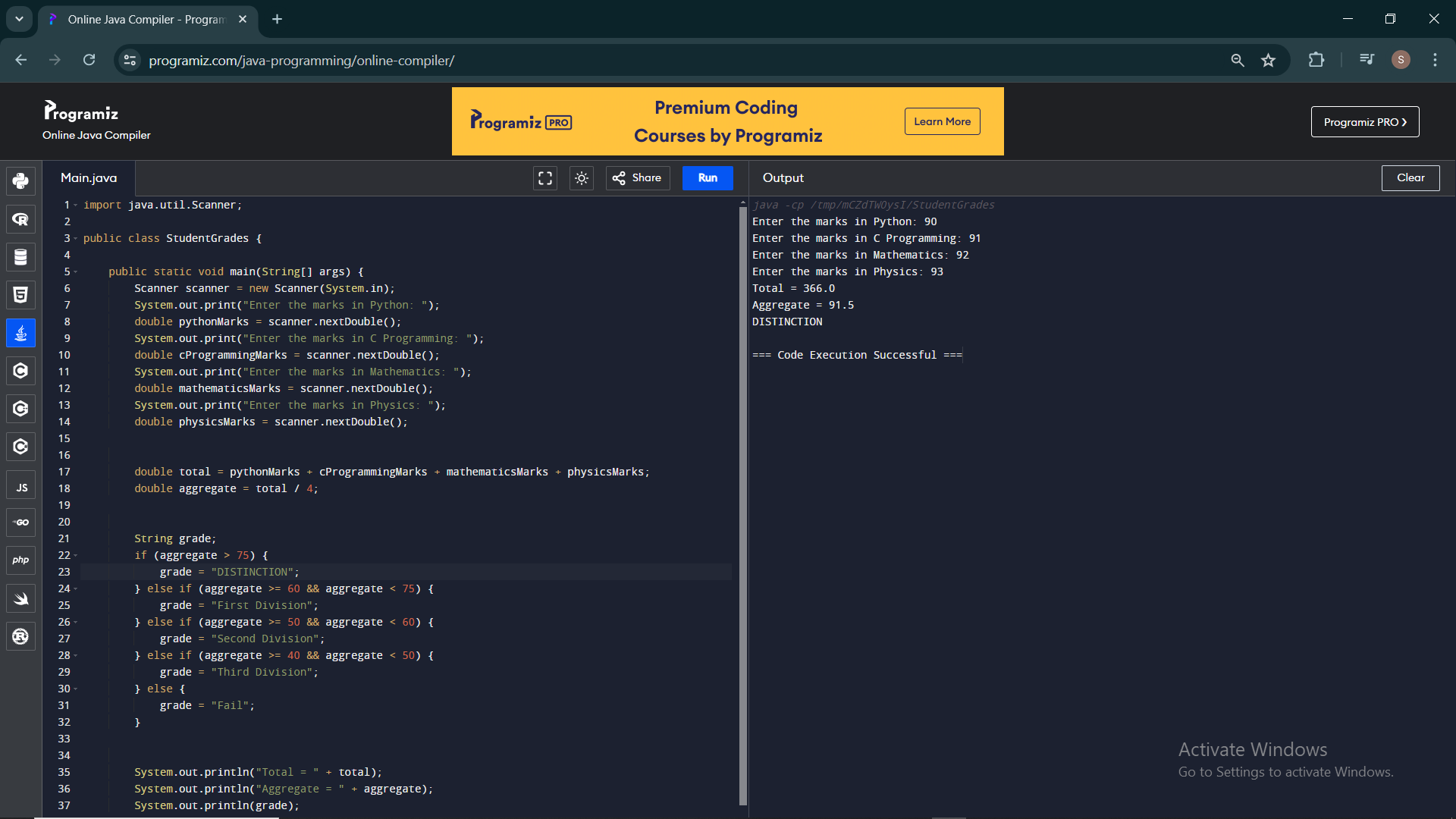
System.out.println("Aggregate = " + aggregate);

System.out.println(grade);

scanner.close();

}

}



2. Write a program to calculate tax given the following conditions:

a. If income is less than or equal to 1,50,000 then no tax

b. If taxable income is 1,50,001 – 3,00,000 the charge 10% tax

c. If taxable income is 3,00,001 – 5,00,000 the charge 20% tax

d. If taxable income is above 5,00,001 then charge 30% tax

Sample Input:

Enter the income:200000

Sample Output:

Tax= 20000

Test cases:

1. 400700

2. 2789239

3. 150000

4. 00000

5. -125486

CODE:

import java.util.Scanner;

public class TaxCalculator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the income: ");

double income = scanner.nextDouble();

double tax = calculateTax(income);

System.out.println("Tax = " + tax);

scanner.close();

}

public static double calculateTax(double income) {

double tax;

if (income <= 150000) {

tax = 0;

} else if (income <= 300000) {

tax = (income - 150000) \* 0.10;

} else if (income <= 500000) {

tax = 15000 + (income - 300000) \* 0.20;

} else {

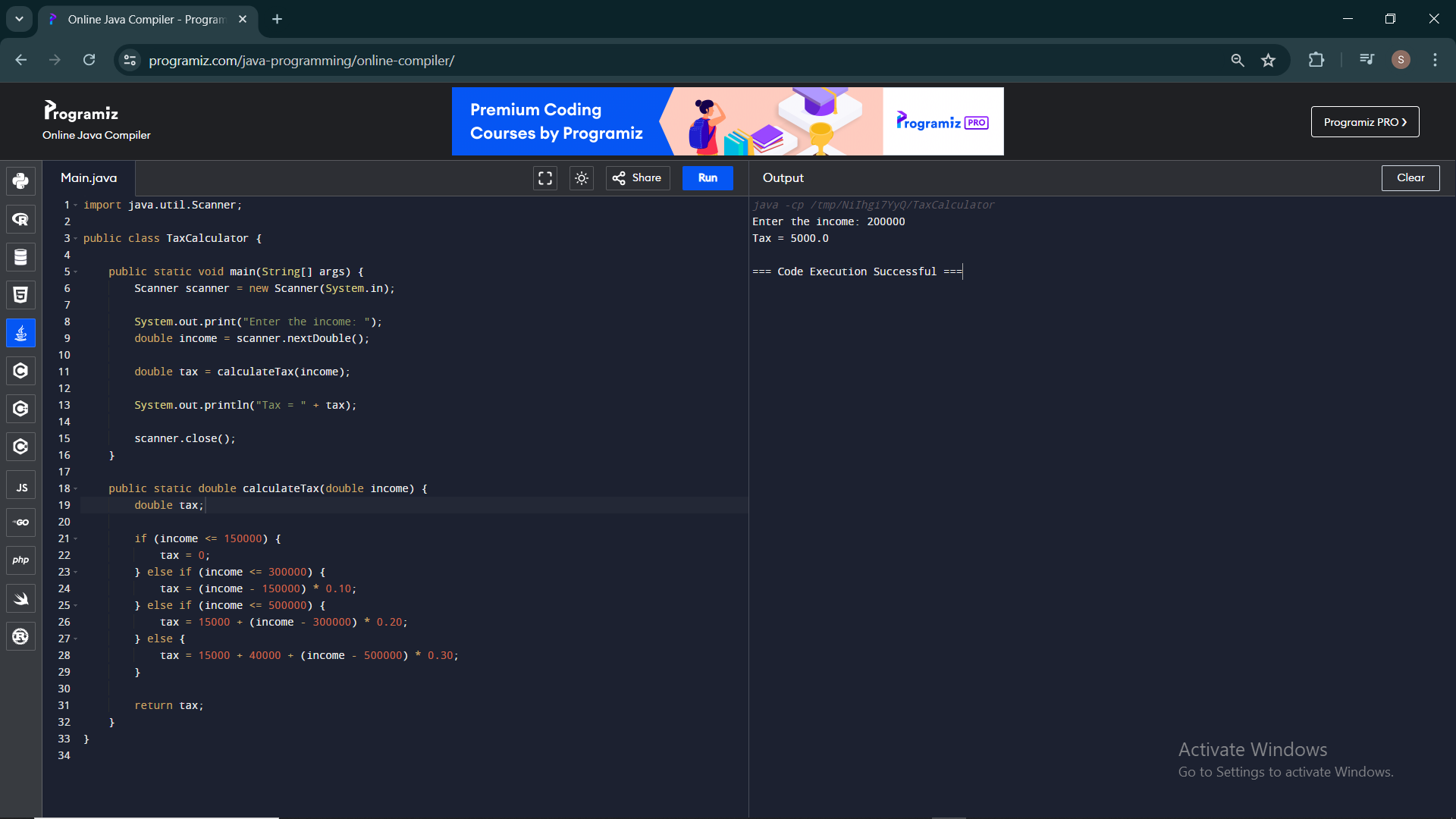
tax = 15000 + 40000 + (income - 500000) \* 0.30;

}

return tax;

}

}



3. Write a program to print the first n perfect numbers. (Hint Perfect number means a positive integer that is equal to the sum of its proper divisors)

Sample Input:

N = 3

Sample Output:

First 3 perfect numbers are: 6 , 28 , 496

Test Cases:

1. N = 0

2. N = 5

3. N = -2

4. N = -5

5. N = 0.2

CODE:

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class PerfectNumbers {

public static boolean isPerfect(int num) {

int sum = 1;

for (int i = 2; i <= Math.sqrt(num); i++) {

if (num % i == 0) {

sum += i;

if (i != num / i) {

sum += num / i;

}

}

}

return sum == num && num != 1;

}

public static List<Integer> getFirstNPerfectNumbers(int n) {

List<Integer> perfectNumbers = new ArrayList<>();

int num = 2;

while (perfectNumbers.size() < n) {

if (isPerfect(num)) {

perfectNumbers.add(num);

}

num++;

}

return perfectNumbers;

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the value of N: ");

double input = scanner.nextDouble();

int n = (int) input;

if (n <= 0) {

System.out.println("N should be a positive integer.");

} else {

List<Integer> perfectNumbers = getFirstNPerfectNumbers(n);

System.out.print("First " + n + " perfect numbers are: ");

for (int i = 0; i < perfectNumbers.size(); i++) {

if (i != 0) {

System.out.print(", ");

}

System.out.print(perfectNumbers.get(i));

}

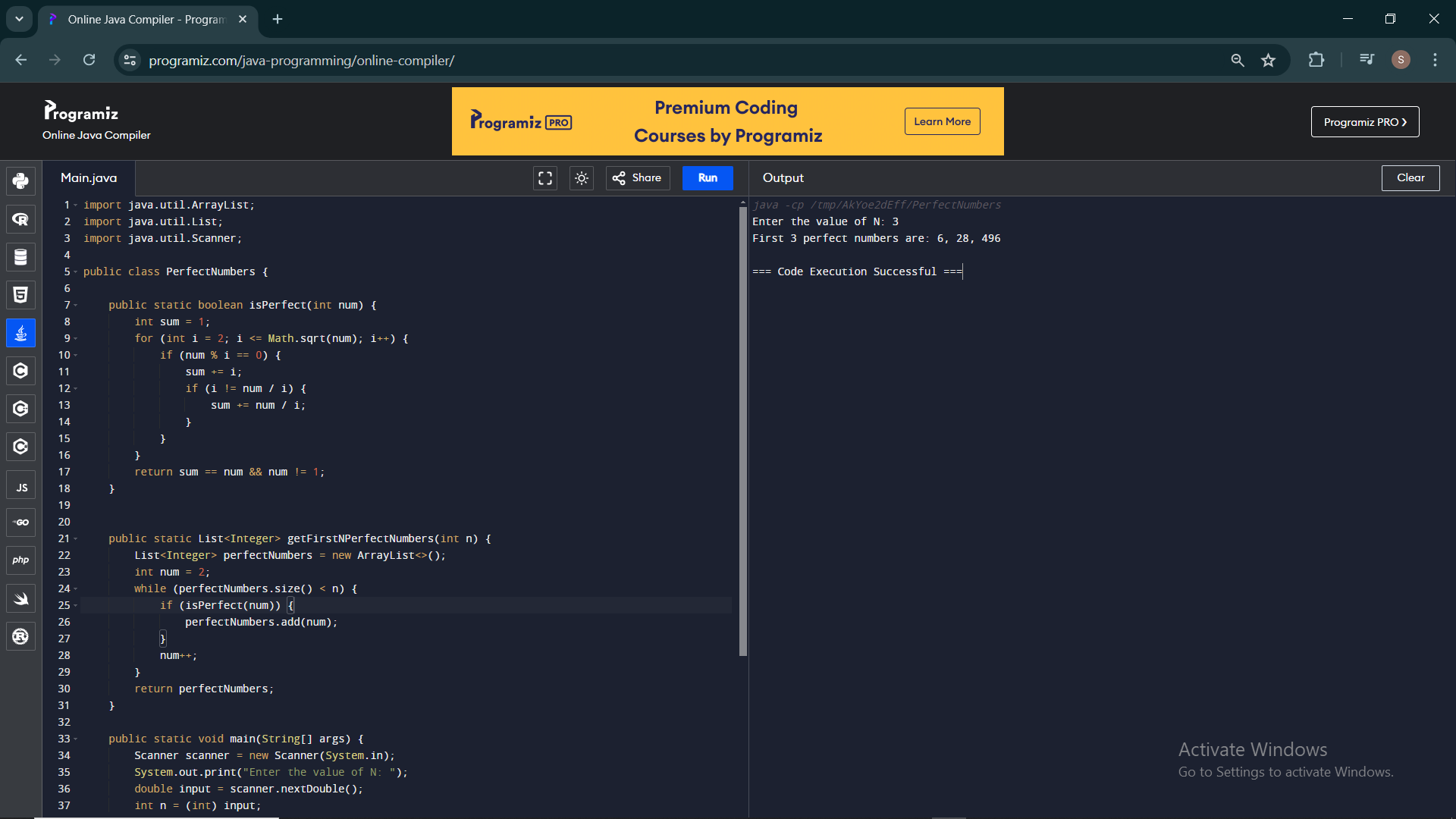
System.out.println();

}

scanner.close();

}

}



4. Write a Program to Find the Nth Largest Number in a array.

Sample Input:

List : {14, 67, 48, 23, 5, 62}

N = 4

Sample Output:

4th Largest number: 23

Test cases:

1. N = 0

2. N = -5

3. N = 1

4. N = M

5. N = %

CODE:

import java.util.Arrays;

import java.util.Collections;

public class NthLargestNumber {

public static void main(String[] args) {

Integer[] array = {14, 67, 48, 23, 5, 62};

int N = 4;

String result = findNthLargest(array, N);

System.out.println(result);

System.out.println(findNthLargest(array, 0));

System.out.println(findNthLargest(array, -5));

System.out.println(findNthLargest(array, 1));

System.out.println(findNthLargest(array, array.length));

System.out.println(findNthLargest(array, array.length + 1));

}

public static String findNthLargest(Integer[] array, int N) {

if (N <= 0) {

return "N should be greater than 0";

}

if (N > array.length) {

return "N is out of range";

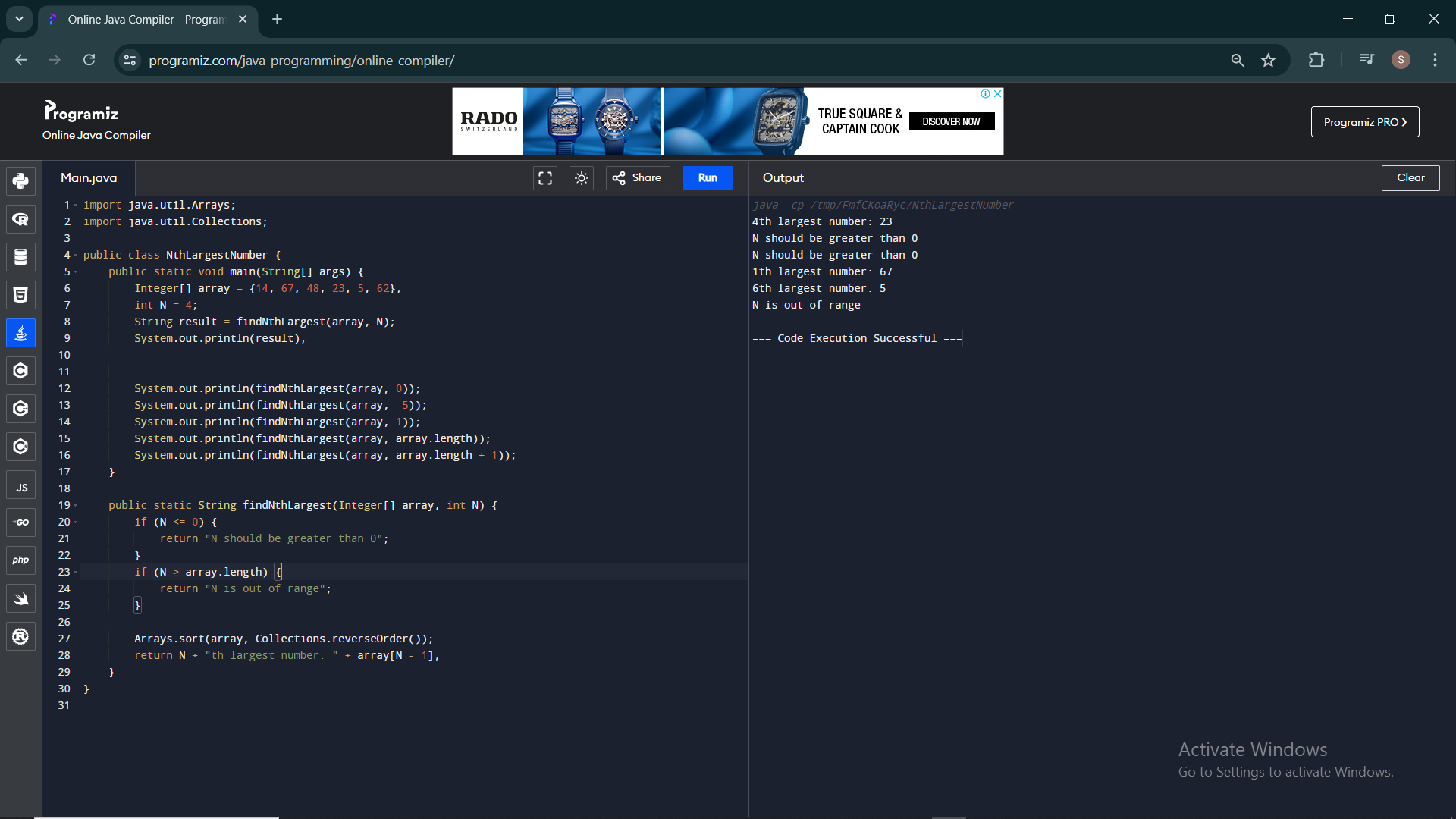
}

Arrays.sort(array, Collections.reverseOrder());

return N + "th largest number: " + array[N - 1];

}

}



5. Write a program to find the number of special characters in the given statement

Sample Input:

Given statement: Modi Birthday @ September 17, #&$% is the wishes code for him.

Sample Output:

Number of special Characters: 5

CODE:

public class SpecialCharacterCounter {

public static void main(String[] args) {

String statement ="Modi Birthday @ September 17, #&$% is the wishes code for him.";

int specialCharCount = countSpecialCharacters(statement);

System.out.println("Number of special characters: " + specialCharCount);

}

public static int countSpecialCharacters(String str) {

int count = 0;

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

char c = str.charAt(i);

if (!Character.isLetterOrDigit(c) && !Character.isWhitespace(c)) {

count++;

}

}

return count;

}

}

. 