6. Exercises.

6.1 Write, compile and run the ChoosingOption program:

Questions:

* What happens if users choose “Cancel”?

Answers: Automatically choose “No” option.

* How to customize the options to users, e.g. only two options: “Yes” and “No”, OR “I do”

and “I don’t”

Answers: Use the following code

import javax.swing.JOptionPane;

public class ChoosingOption {

    public static void main(String[] args) {

        String[] options = { "Scissors", "Rock", "Paper" };

        int selection = JOptionPane.showOptionDialog(null, "Select one:", "Let's play a game!",

                0, 3, null, options, options[0]);

        if (selection == 0)

        {

            JOptionPane.showMessageDialog(null, "You have chose Scissors !!!");

        }

        else if (selection == 1)

        {

            JOptionPane.showMessageDialog(null, "You have chose Rock !!!");

        }

        else if (selection == 2)

        {

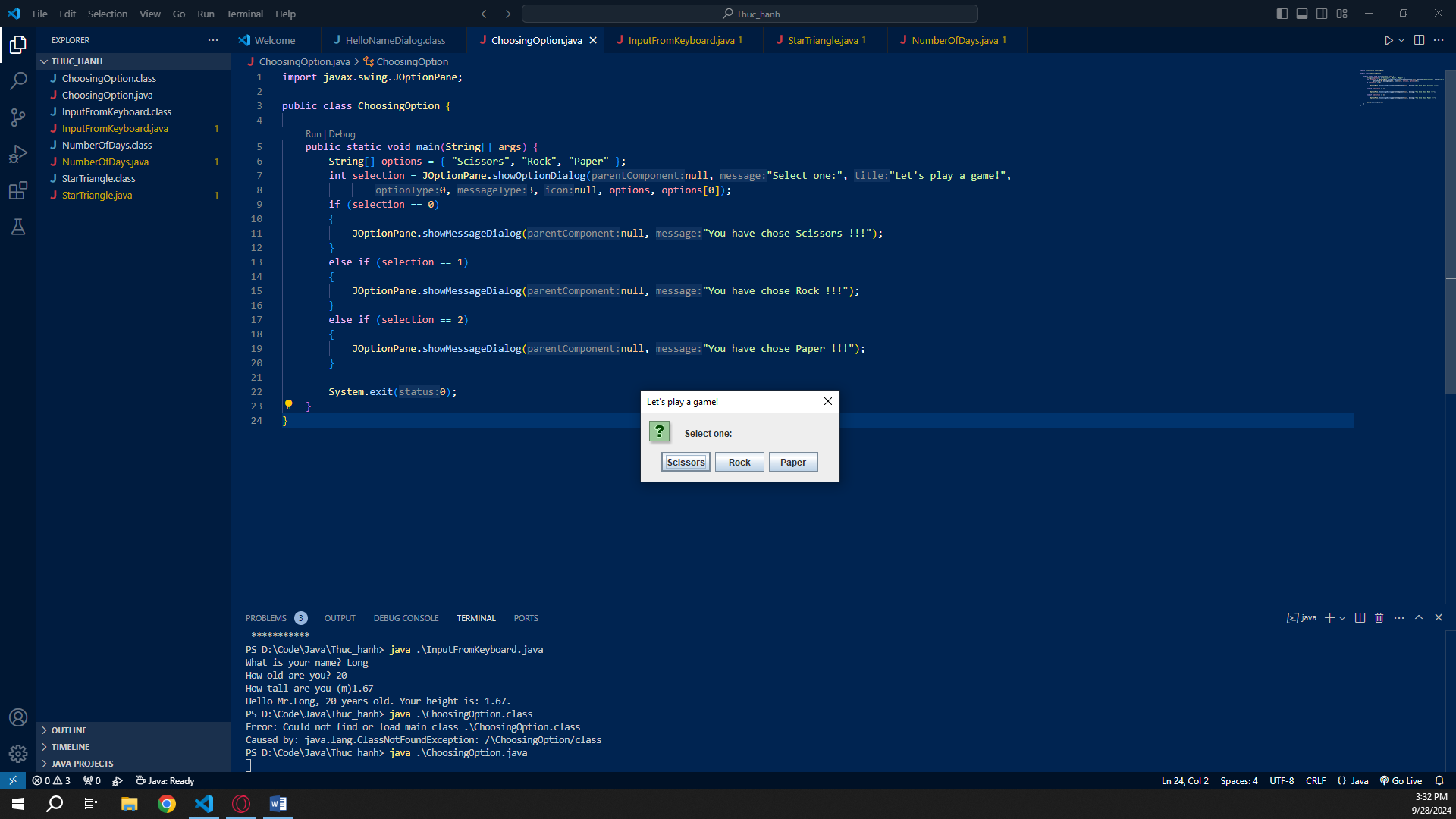
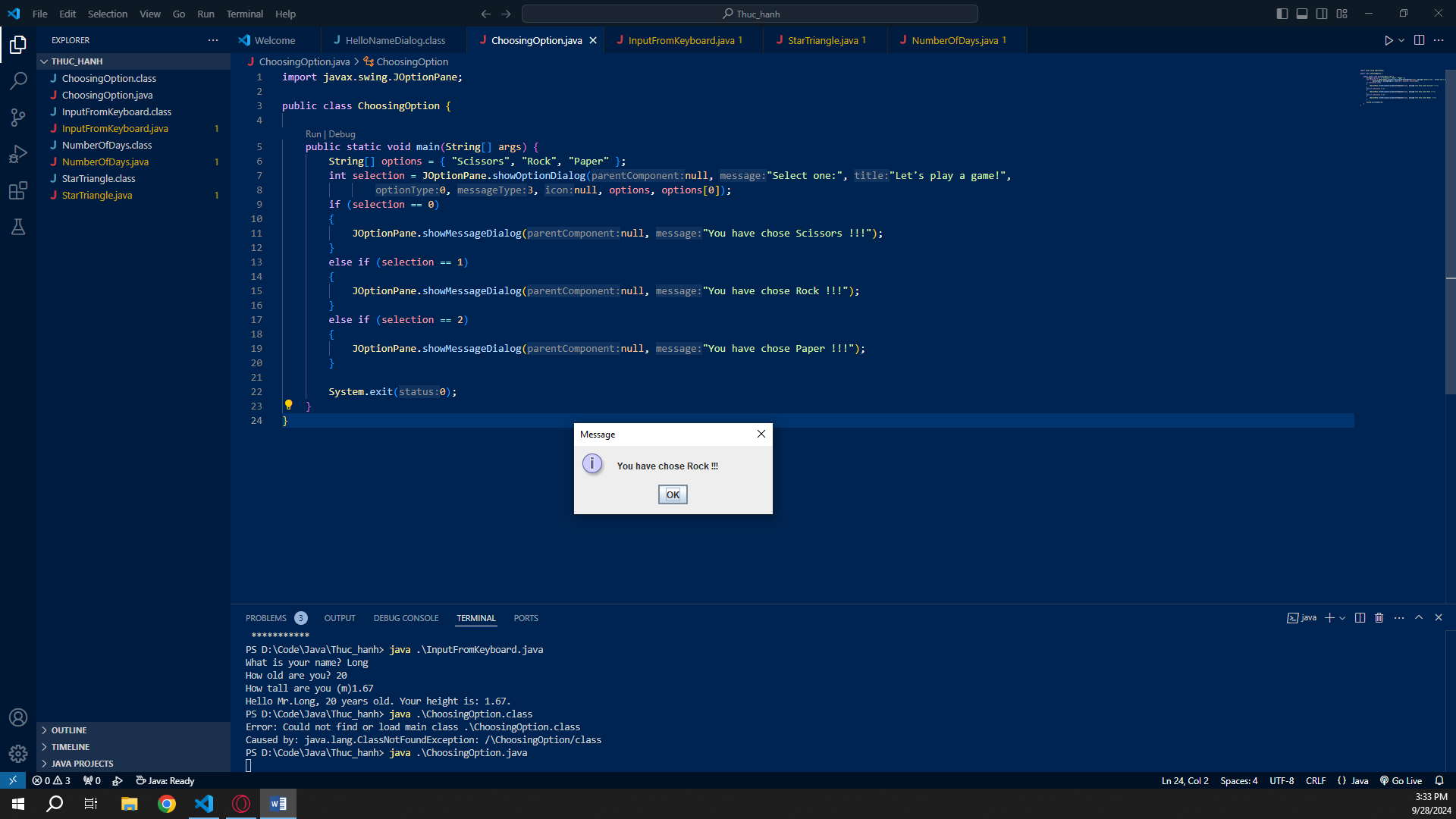
            JOptionPane.showMessageDialog(null, "You have chose Paper !!!");

        }

        System.exit(0);

    }

Output:

6.2 Write a program for input/output from keyboard

Code:

import java.util.Scanner;

public class InputFromKeyboard {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("What is your name? ");

        String name = input.nextLine();

        System.out.print("How old are you? ");

        int age = input.nextInt();

        System.out.print("How tall are you (m)");

        double height = input.nextDouble();

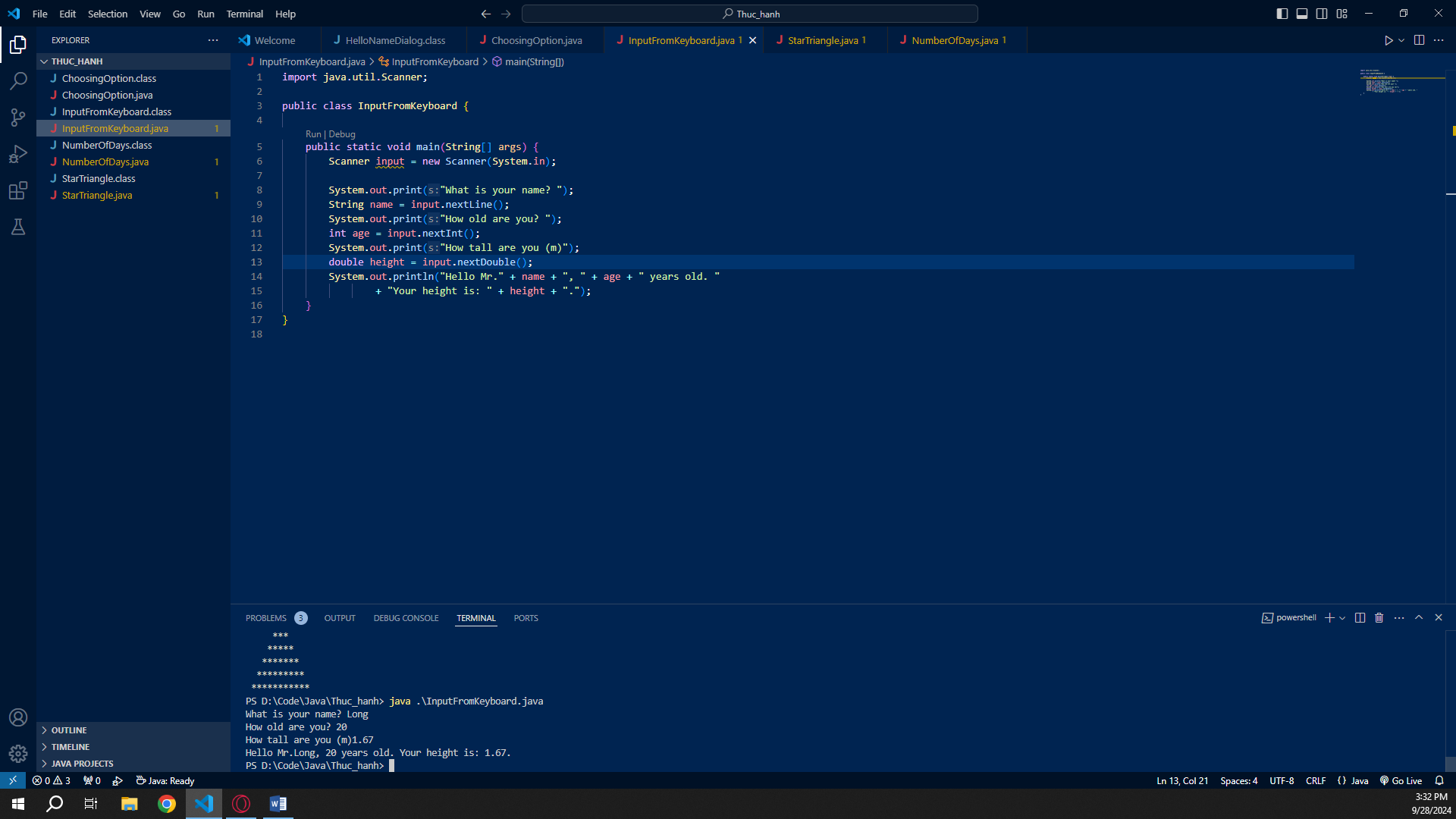
        System.out.println("Hello Mr." + name + ", " + age + " years old. "

                + "Your height is: " + height + ".");

    }

}

Output:



6.3 Write a program to display a triangle with a height of n stars (\*), n is entered by users.

E.g. n=5:

Code:

import java.util.Scanner;

public class StarTriangle

{

    public static void main(String[] args)

    {

        Scanner input = new Scanner(System.in);

        System.out.print("Your triangle height: ");

        int n = input.nextInt();

        for (int i = 0 ; i < n ; i++)

        {

            String triangleToLine = "";

            for (int j = 0 ; j < n\*2 ; j++)

            {

                if (j < n - i || j > n + i)

                {

                    triangleToLine += " ";

                }

                else

                {

                    triangleToLine += "\*";

                }

            }

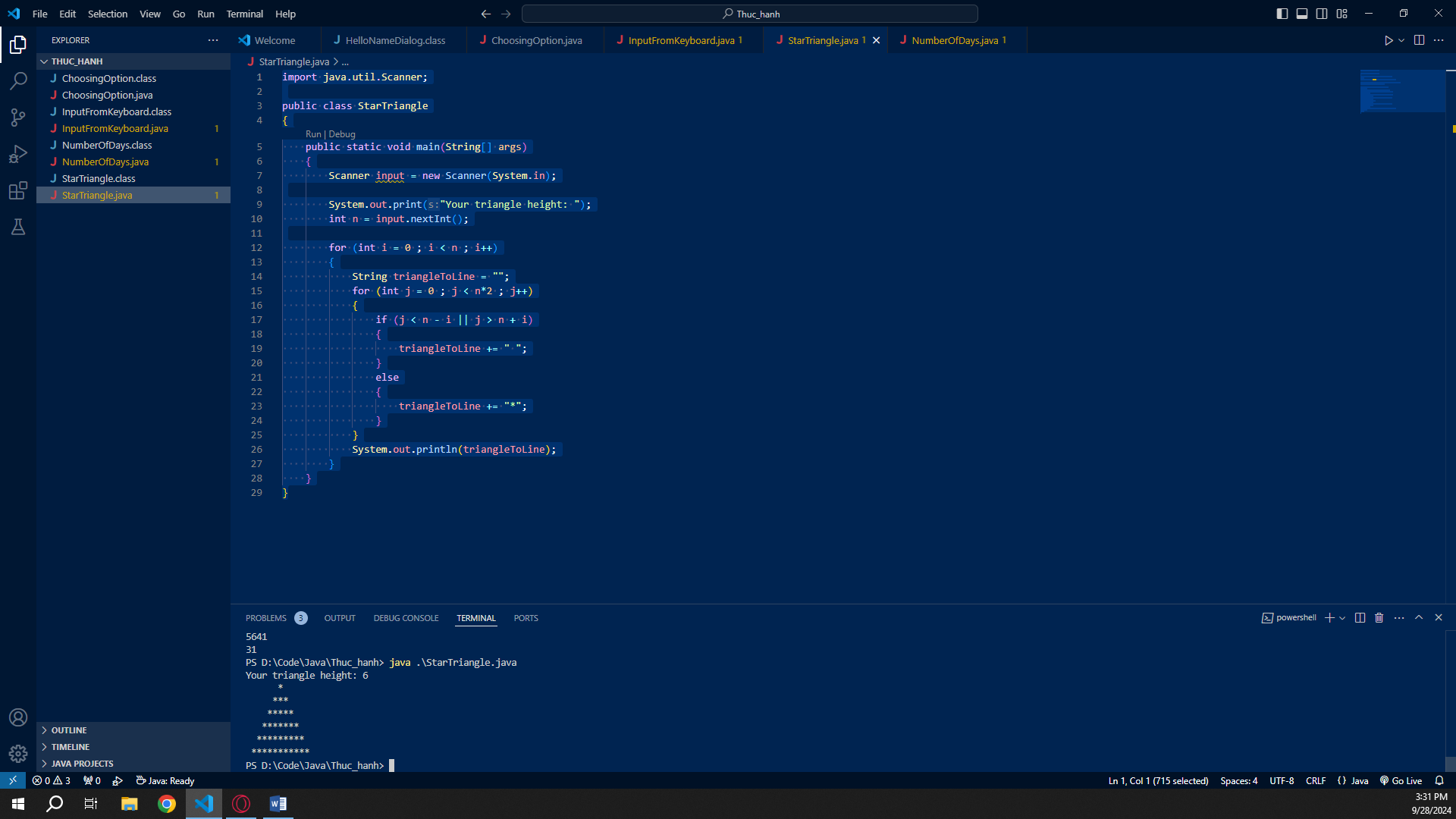
            System.out.println(triangleToLine);

        }

    }

}

Output:



6.4 Write a program to display the number of days of a month, which is entered by users

(both month and year). If it is an invalid month/year, ask the user to enter again.

Code:

import java.util.\*;

public class NumberOfDays {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Get month: ");

        String month = input.nextLine();

        System.out.println("Get Year: ");

        int year = 0;

        try {

            year = input.nextInt();

        } catch (Exception e) {

            System.out.println(

                    "valid inputs of year 1999 is only 1999, but not 99, \"one thousand nine hundred ninety-" + //

                            "nine\", or anything else.");

            System.exit(0);

        }

        String[] Month = { "January", "February", "March", "April", "May", "June", "July", "August", "September",

                "October", "November", "December" };

        String[] Abbreviation = { "Jan.", "Feb.", "Mar.", "Apr.", "May", "June", "Jul", "Aug.", "Sept.",

                "Oct.", "Nov.", "Dec." };

        String[] In3letters = { "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep",

                "Oct", "Nov", "Dec" };

        String[] InNumber = { "1", "2", "3", "4", "5", "6", "7", "8", "9",

                "10", "11", "12" };

        List<String> MonthToList = Arrays.asList(Month), AbbreviationToList = Arrays.asList(Abbreviation),

                In3lettersToList = Arrays.asList(In3letters), InNumberToList = Arrays.asList(InNumber);

        int[] DayOfMonthInCommonYear = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

        int[] DayOfMonthInLeapYear = { 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

        if (MonthToList.contains(month)) {

            if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {

                System.out.println(DayOfMonthInLeapYear[MonthToList.indexOf(month)]);

            } else {

                System.out.println(DayOfMonthInCommonYear[MonthToList.indexOf(month)]);

            }

        } else if (AbbreviationToList.contains(month)) {

            if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {

                System.out.println(DayOfMonthInLeapYear[AbbreviationToList.indexOf(month)]);

            } else {

                System.out.println(DayOfMonthInCommonYear[AbbreviationToList.indexOf(month)]);

            }

        } else if (In3lettersToList.contains(month)) {

            if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {

                System.out.println(DayOfMonthInLeapYear[In3lettersToList.indexOf(month)]);

            } else {

                System.out.println(DayOfMonthInCommonYear[In3lettersToList.indexOf(month)]);

            }

        } else if (InNumberToList.contains(month)) {

            if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {

                System.out.println(DayOfMonthInLeapYear[InNumberToList.indexOf(month)]);

            } else {

                System.out.println(DayOfMonthInCommonYear[InNumberToList.indexOf(month)]);

            }

        } else {

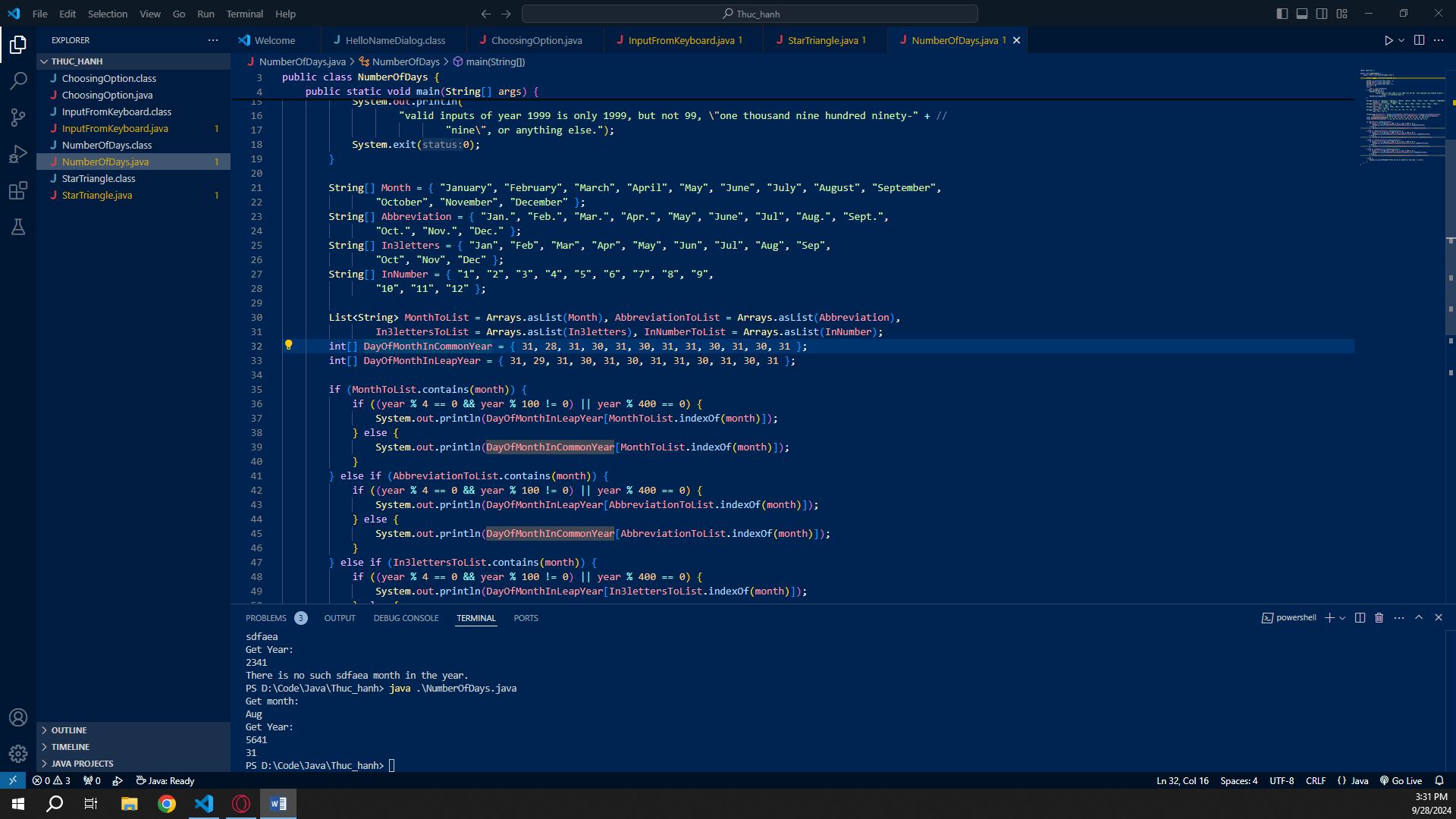
            System.out.printf("There are no %s months in the year.", month);

        }

    }

}

Output:



6.5 Write a Java program to sort a numeric array, and calculate the sum and average value of array elements.

Code:

import java.util.\*;

public class SortArray {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("Get the size of the array: ");

        int n = input.nextInt(), sum = 0;

        int[] inputArray = new int[n];

        System.out.print("Get the elements of array: ");

        for (int i = 0; n > i; ++i) {

            inputArray[i] = input.nextInt();

            sum += inputArray[i];

        }

        Arrays.sort(inputArray);

        System.out.print("Sorted array: ");

        for (int e : inputArray)

        {

            System.out.print(e + " ");

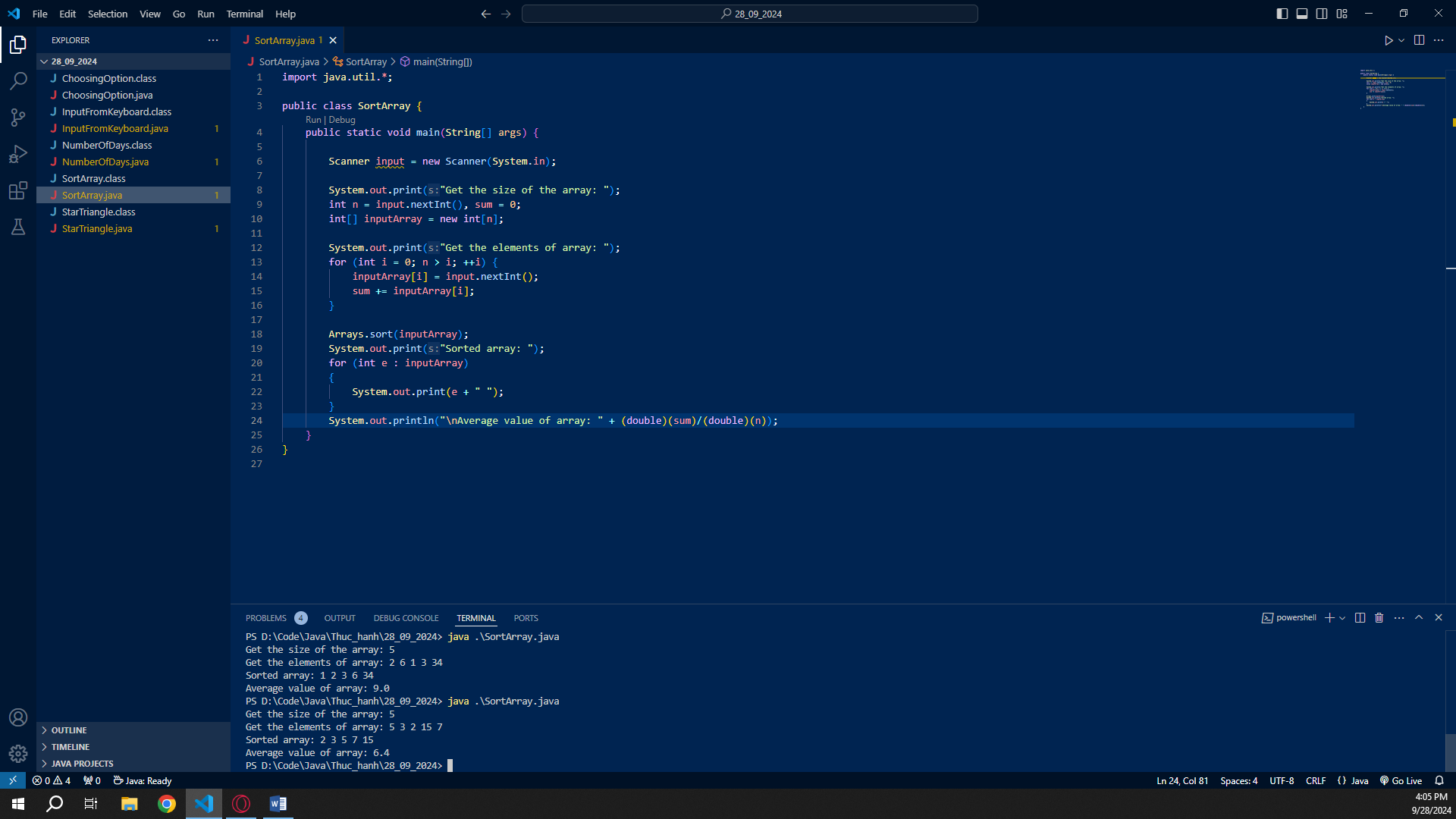
        }

        System.out.println("\nAverage value of array: " + (double)(sum)/(double)(n));

    }

}

Output:



6.6 Write a Java program to add two matrices of the same size.

Code:

import java.util.\*;

public class addMatrices {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("Get matrix size: ");

        int n = input.nextInt(), m = input.nextInt();

        int[][] matrix1 = new int[n][m], matrix2 = new int[n][m];

        System.out.printf("Get the elements of matrix1 height = %d, width = %d: \n", n, m);

        for (int i = 0; i < n; ++i) {

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

                matrix1[i][j] = input.nextInt();

            }

        }

        System.out.printf("Get the elements of matrix2 height = %d, width = %d: \n", n, m);

        for (int i = 0; i < n; ++i) {

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

                matrix2[i][j] = input.nextInt();

            }

        }

        System.out.println("Matrix after add: \n");

        for (int i = 0; i < n; ++i) {

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

                int sum = matrix1[i][j] + matrix2[i][j];

                System.out.print(sum + " ");

            }

            System.out.print("\n");

        }

    }

}

Output:

