

Lab 01: Environment Setup and Java Basics

Student name: Trịnh Lương Việt

Student ID: 20236012

Assignment 2.2.5: Write a program to calculate sum, difference, product, and quotient of 2 double numbers which are entered by users.

Source code:

```
package twonumbers;
import javax.swing.*;

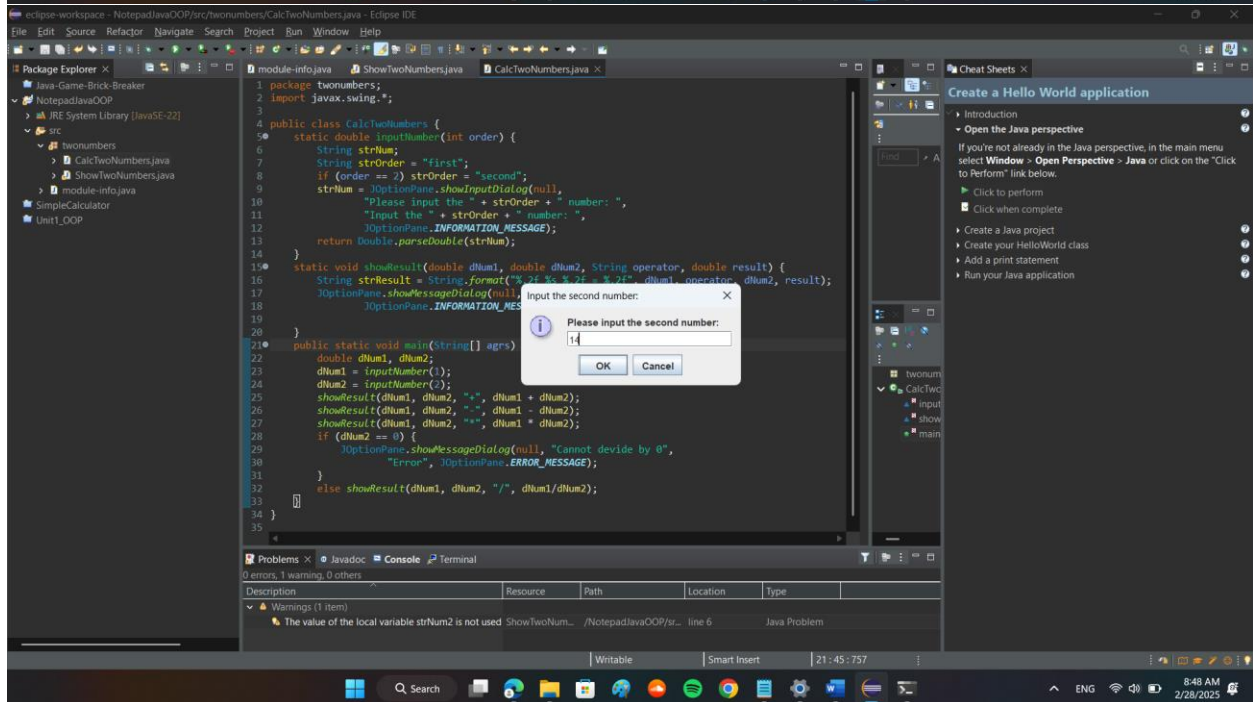
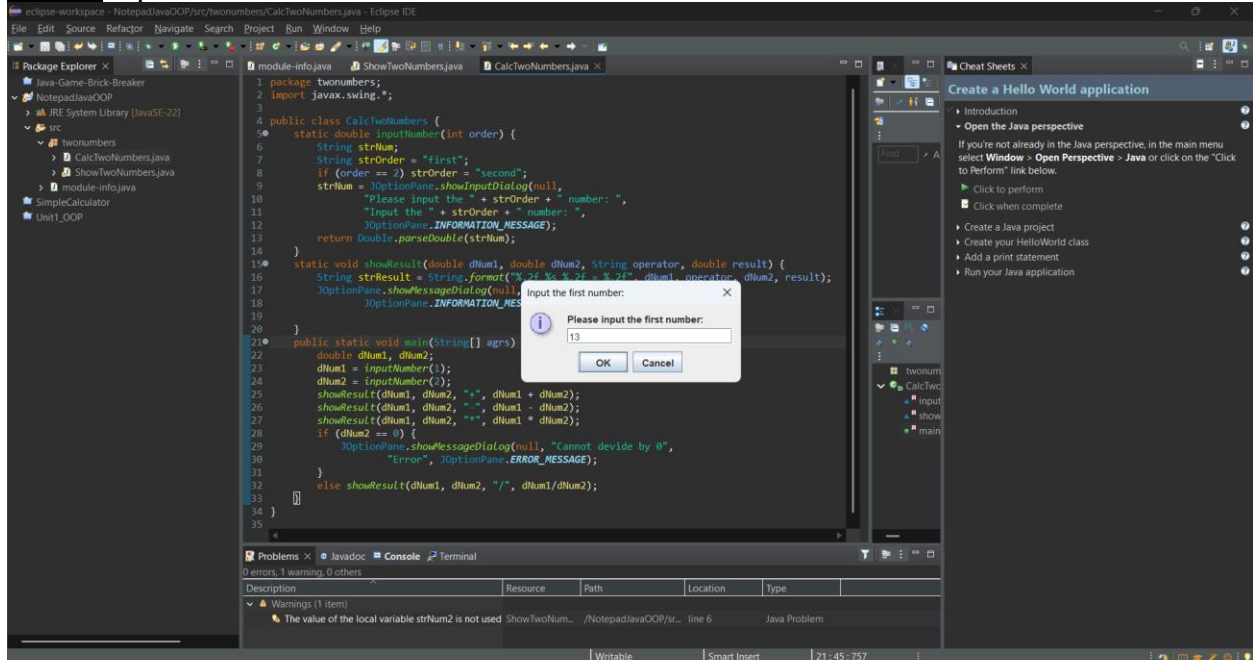
public class CalcTwoNumbers {
    static double inputNumber(int order) {
        String strNum;
        String strOrder = "first";
        if (order == 2) strOrder = "second";
        strNum = JOptionPane.showInputDialog(null,
            "Please input the " + strOrder + " number: ",
            "Input the " + strOrder + "number: ",
            JOptionPane.INFORMATION_MESSAGE);
        return Double.parseDouble(strNum);
    }

    static void showResult(double dNum1, double dNum2, String operator, double result) {
        String strResult = String.format("%.2f %s %.2f = %.2f",
            dNum1, operator, dNum2, result);
        JOptionPane.showMessageDialog(null, strResult, "Result",
            JOptionPane.INFORMATION_MESSAGE);
    }

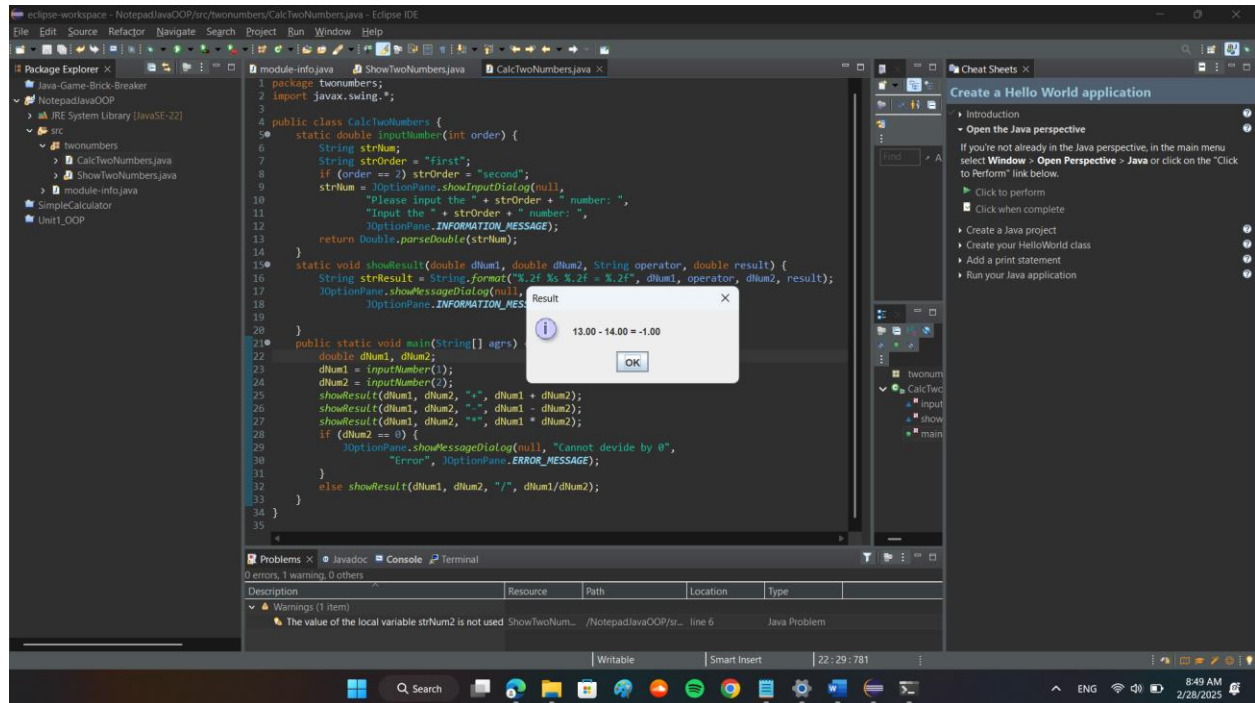
    public static void main(String[] args) {
        double dNum1, dNum2;
        dNum1 = inputNumber(1);
        dNum2 = inputNumber(2);
        showResult(dNum1, dNum2, "+", dNum1 + dNum2);
        showResult(dNum1, dNum2, "-", dNum1 - dNum2);
        showResult(dNum1, dNum2, "*", dNum1 * dNum2);
        if (dNum2 == 0) {
            JOptionPane.showMessageDialog(null, "Cannot divide by 0",
                "Error", JOptionPane.ERROR_MESSAGE);
        }
        else showResult(dNum1, dNum2, "/", dNum1/dNum2);
    }
}
```

Result:

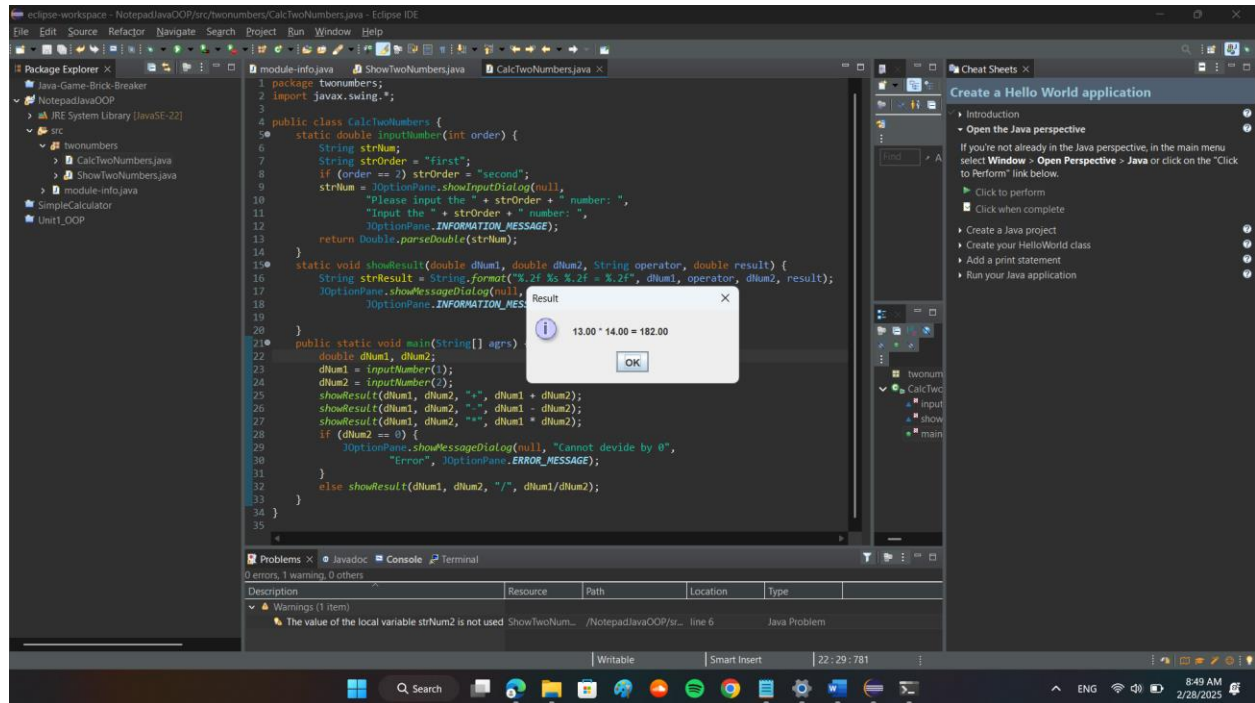
Step 1: Input the number:



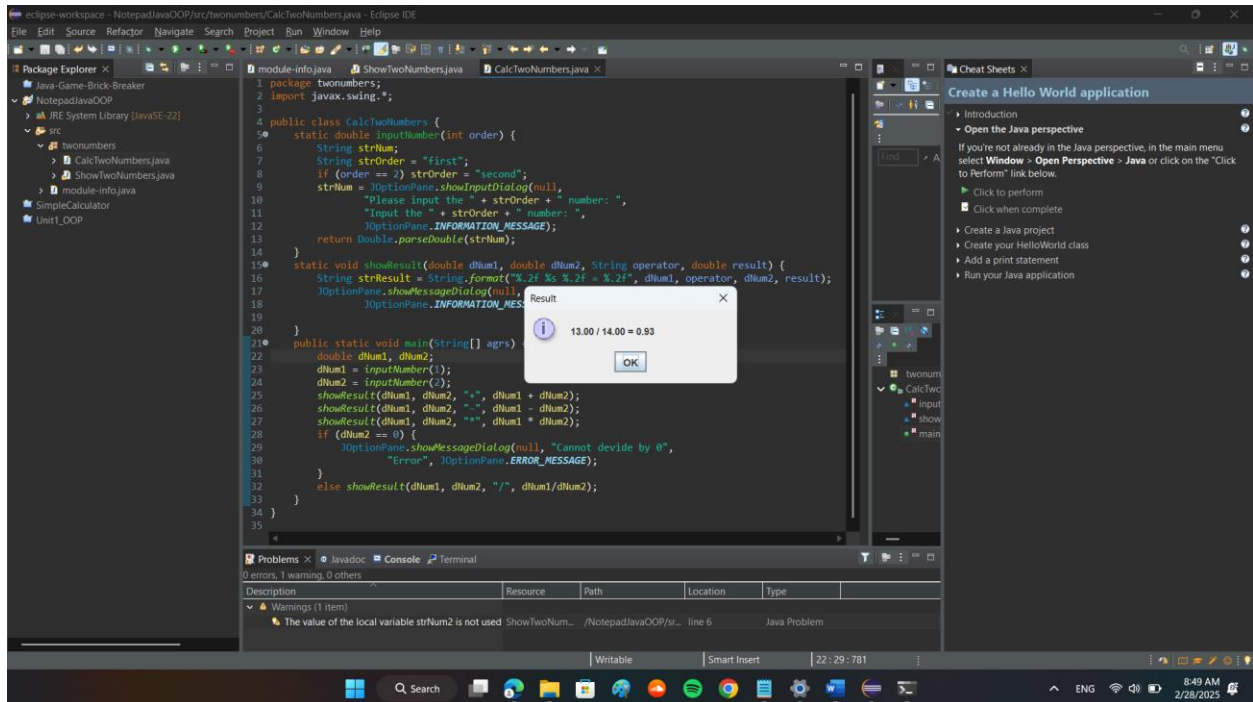
Step 2: Calculate and show the result Sum:



Difference:

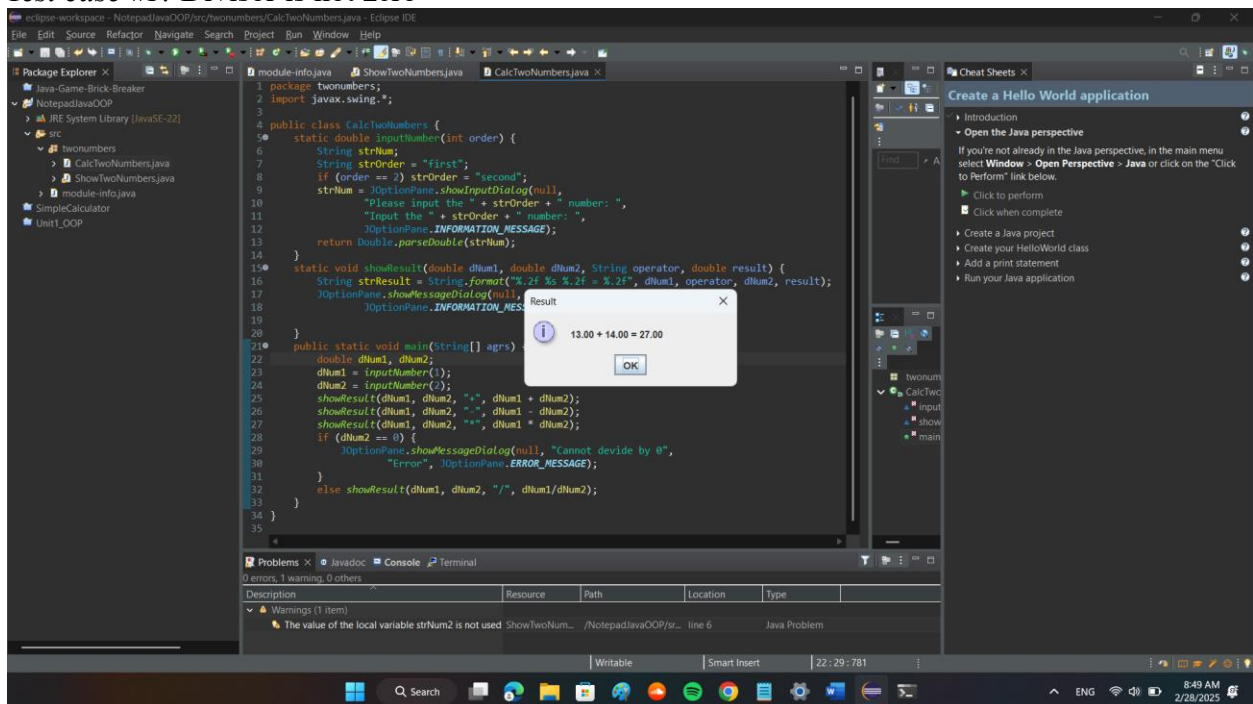


Product:

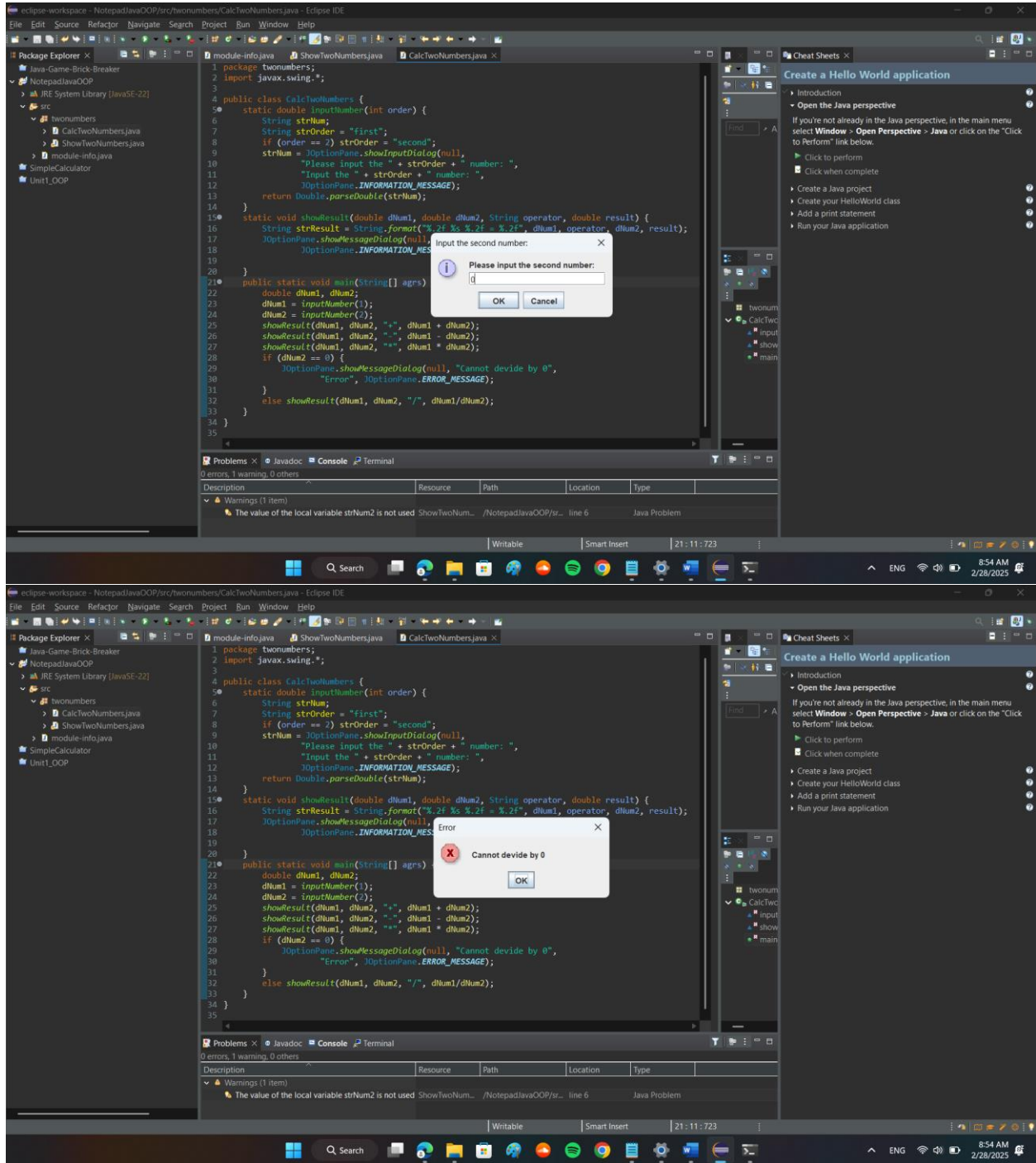


Division:

Test case #1: Divisor is not zero



Test case #2: Divisor is zero



Assignment 2.2.6: Write a program to solve:

- *The first-degree equation (linear equation) with one variable*
- *The system of first-degree equations (linear system) with two variables*
- *The second-degree equation with one variable*

Source code:

```
package implementings;
```

```
import java.util.*;
```

```
public class SolveEquation {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int choice;
        do {
            System.out.println("\nEquation Solver Menu:");
            System.out.println("1. Solve first-degree equation (ax + b = 0)");
            System.out.println("2. Solve system of first-degree equations");
            System.out.println("3. Solve second-degree equation (ax^2 + bx + c = 0)");
            System.out.println("0. Exit");
            System.out.print("Choose an option: ");
            choice = scanner.nextInt();

            switch (choice) {
                case 1:
                    solveLinearEquation(scanner);
                    break;
                case 2:
                    solveLinearSystem(scanner);
                    break;
                case 3:
                    solveQuadraticEquation(scanner);
                    break;
                case 0:
                    System.out.println("Exit!");
                    break;
                default:
                    System.out.println("Invalid option! Try again.");
            }
        } while (choice != 0);

        scanner.close();
    }
}
```

```

private static void solveLinearEquation(Scanner scanner) {
    System.out.print("Enter a (coefficient of x): ");
    double a = scanner.nextDouble();
    System.out.print("Enter b (constant term): ");
    double b = scanner.nextDouble();

    if (a == 0) {
        if (b == 0) {
            System.out.println("Infinite solutions.");
        } else {
            System.out.println("No solution.");
        }
    } else {
        double x = -b / a;
        System.out.println("Solution: x = " + x);
    }
}

```

```

private static void solveLinearSystem(Scanner scanner) {
    System.out.println("Enter coefficients for the system:");
    System.out.print("Enter a1: ");
    double a1 = scanner.nextDouble();
    System.out.print("Enter b1: ");
    double b1 = scanner.nextDouble();
    System.out.print("Enter c1: ");
    double c1 = scanner.nextDouble();
    System.out.print("Enter a2: ");
    double a2 = scanner.nextDouble();
    System.out.print("Enter b2: ");
    double b2 = scanner.nextDouble();
    System.out.print("Enter c2: ");
    double c2 = scanner.nextDouble();

    double D = a1 * b2 - a2 * b1;
    double Dx = c1 * b2 - c2 * b1;
    double Dy = a1 * c2 - a2 * c1;

    if (D == 0) {
        if (Dx == 0 && Dy == 0) {
            System.out.println("Infinite solutions.");
        } else {
            System.out.println("No solution.");
        }
    } else {
        double x = Dx / D;
        double y = Dy / D;
    }
}

```

```

        System.out.println("Solution: x = " + x + ", y = " + y);
    }
}

private static void solveQuadraticEquation(Scanner scanner) {
    System.out.print("Enter a (coefficient of x^2): ");
    double a = scanner.nextDouble();
    System.out.print("Enter b (coefficient of x): ");
    double b = scanner.nextDouble();
    System.out.print("Enter c (constant term): ");
    double c = scanner.nextDouble();

    if (a == 0) {
        System.out.println("This is not a quadratic equation. Switching to linear equation...");
        solveLinearEquation(scanner);
        return;
    }

    double delta = b * b - 4 * a * c;

    if (delta > 0) {
        double x1 = (-b + Math.sqrt(delta)) / (2 * a);
        double x2 = (-b - Math.sqrt(delta)) / (2 * a);
        System.out.println("Two distinct solutions: x1 = " + x1 + ", x2 = " + x2);
    } else if (delta == 0) {
        double x = -b / (2 * a);
        System.out.println("Double root: x = " + x);
    } else {
        System.out.println("No real solution.");
    }
}
}

```


Result:

Solve first-degree equation: (3 test case: Unique solution, No solution, Infinite solutions)

```
34 } while (choice != 0);
35 scanner.close();
36 }
37 }
38
39 private static void solveLinearEquation(Scanner scanner) {
40     System.out.print("Enter a (coefficient of x): ");
41     double a = scanner.nextDouble();
42     System.out.print("Enter b (constant term): ");
43     double b = scanner.nextDouble();
44
45     if (a == 0) {
46         if (b == 0) {
47             System.out.println("Infinite solutions.");
48         } else {
49             System.out.println("No solution.");
50         }
51     } else {
52         double x = -b / a;
53         System.out.println("Solution: x = " + x);
54     }
55 }
```

Equation Solver Menu:
1. Solve first-degree equation ($ax + b = 0$)
2. Solve system of first-degree equations
3. Solve second-degree equation ($ax^2 + bx + c = 0$)
0. Exit
Choose an option: 1
Enter a (coefficient of x): 6.89
Enter b (constant term): 45
Solution: x = -1.0812772133526851

Equation Solver Menu:
1. Solve first-degree equation ($ax + b = 0$)
2. Solve system of first-degree equations
3. Solve second-degree equation ($ax^2 + bx + c = 0$)
0. Exit
Choose an option: 1
Enter a (coefficient of x): 0
Enter b (constant term): 0
No solution.

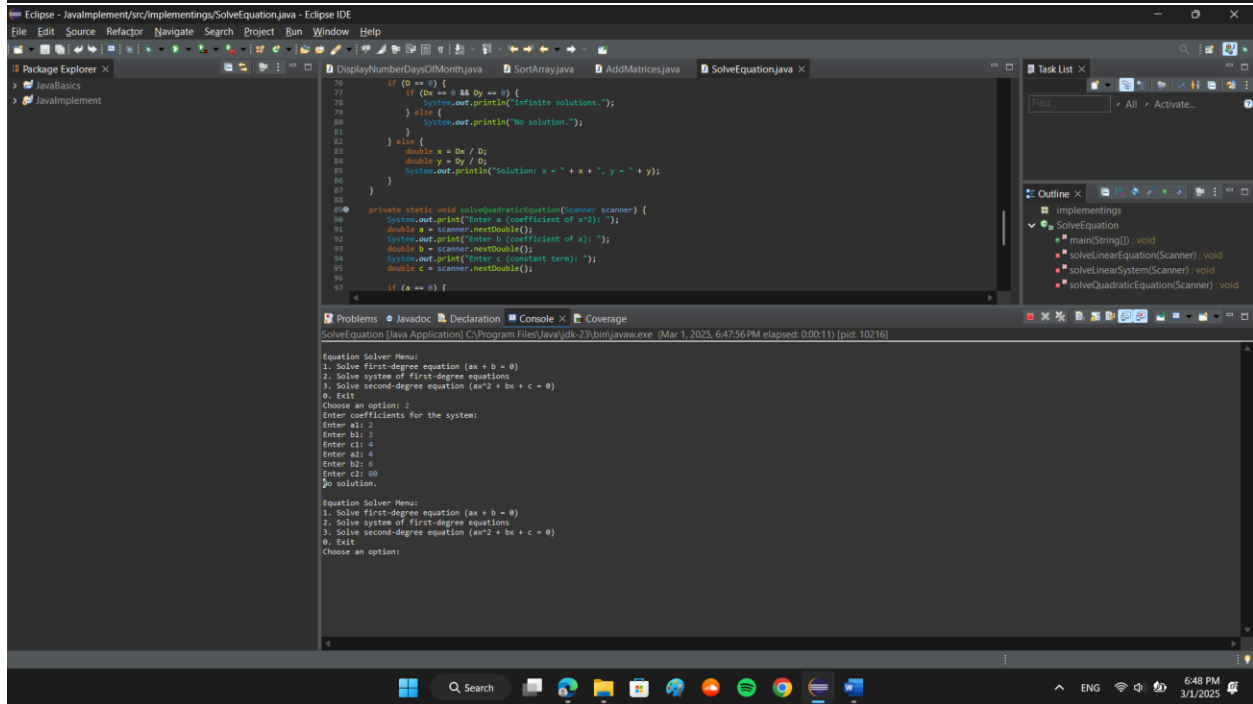
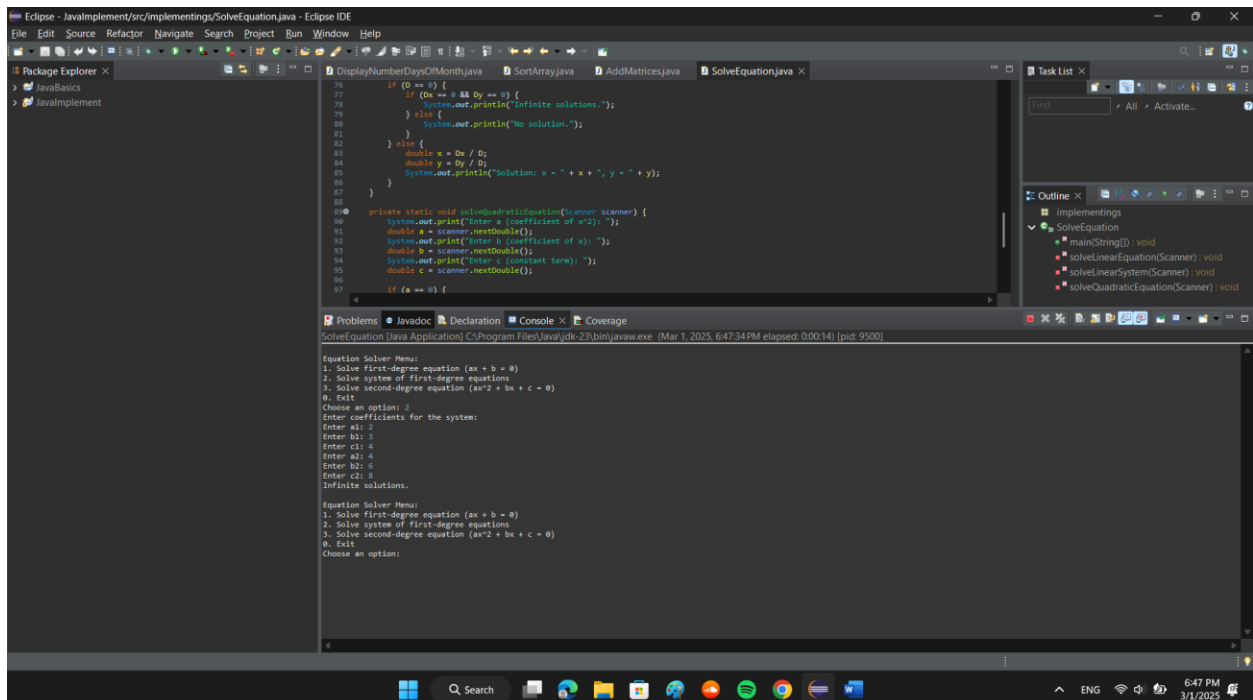
Equation Solver Menu:
1. Solve first-degree equation ($ax + b = 0$)
2. Solve system of first-degree equations
3. Solve second-degree equation ($ax^2 + bx + c = 0$)
0. Exit
Choose an option: 1
Enter a (coefficient of x): 0
Enter b (constant term): 0
Infinite solutions.

Solve system of first-degree equations: (3 test case: Unique solution, No solution, Infinite solutions)

```
77 if (Dd == 0 && Dy == 0) {
78     System.out.println("Infinite solutions.");
79 } else {
80     System.out.println("No solution.");
81 }
82 } else {
83     double x = Dd / D;
84     double y = Dy / D;
85     System.out.println("Solution: x = " + x + ", y = " + y);
86 }
87 }
88
89 private static void solveQuadraticEquation(Scanner scanner) {
90     System.out.print("Enter a (coefficient of x^2): ");
91     double a = scanner.nextDouble();
92     System.out.print("Enter b (coefficient of x): ");
93     double b = scanner.nextDouble();
94     System.out.print("Enter c (constant term): ");
95     double c = scanner.nextDouble();
96
97     if (a == 0) {
98         // This block is not shown in the provided code snippet
99     }
100 }
```

Equation Solver Menu:
1. Solve first-degree equation ($ax + b = 0$)
2. Solve system of first-degree equations
3. Solve second-degree equation ($ax^2 + bx + c = 0$)
0. Exit
Choose an option: 2
Enter coefficients for the system:
Enter a1: 8
Enter b1: 9
Enter c1: 5.6
Enter a2: 4
Enter b2: -6
Enter c2: 9.9796
Solution: x = 1.469242857142857, y = -0.6837714285714286

Equation Solver Menu:
1. Solve first-degree equation ($ax + b = 0$)
2. Solve system of first-degree equations
3. Solve second-degree equation ($ax^2 + bx + c = 0$)
0. Exit
Choose an option:



Solve second-degree equation (3 test case: Two distinct solutions, Double roots, No real solution)

The screenshot shows the Eclipse IDE with the `SolveEquation.java` file open. The code implements a menu-driven program to solve linear and quadratic equations. The console output shows three test cases: two distinct solutions, double roots, and no real solution.

```

76 if (D == 0) {
77     if (Dx == 0 && Dy == 0) {
78         System.out.println("Infinite solutions.");
79     } else {
80         System.out.println("No solution.");
81     }
82 } else {
83     double x = Dx / D;
84     double y = Dy / D;
85     System.out.println("Solution: x = " + x + ", y = " + y);
86 }
87 }
88
89 private static void solveQuadraticEquation(Scanner scanner) {
90     System.out.println("Enter a (coefficient of x^2): ");
91     double a = scanner.nextDouble();
92     System.out.println("Enter b (coefficient of x): ");
93     double b = scanner.nextDouble();
94 }

```

Console Output:

```

Equation Solver Menu:
1. Solve first-degree equation (ax + b = 0)
2. Solve system of first-degree equations
3. Solve second-degree equation (ax^2 + bx + c = 0)
0. Exit
Choose an option: 3
Enter a (coefficient of x^2): 1
Enter b (coefficient of x): 7
Enter c (constant term): 12
Two distinct solutions: x1 = -3.0, x2 = -4.0

Equation Solver Menu:
1. Solve first-degree equation (ax + b = 0)
2. Solve system of first-degree equations
3. Solve second-degree equation (ax^2 + bx + c = 0)
0. Exit
Choose an option: 3
Enter a (coefficient of x^2): 1
Enter b (coefficient of x): 1
Enter c (constant term): 1
No real solution.

Equation Solver Menu:
1. Solve first-degree equation (ax + b = 0)
2. Solve system of first-degree equations
3. Solve second-degree equation (ax^2 + bx + c = 0)
0. Exit
Choose an option: 3
Enter a (coefficient of x^2): 1
Enter b (coefficient of x): 2
Enter c (constant term): 1
Double root: x = -1.0

```

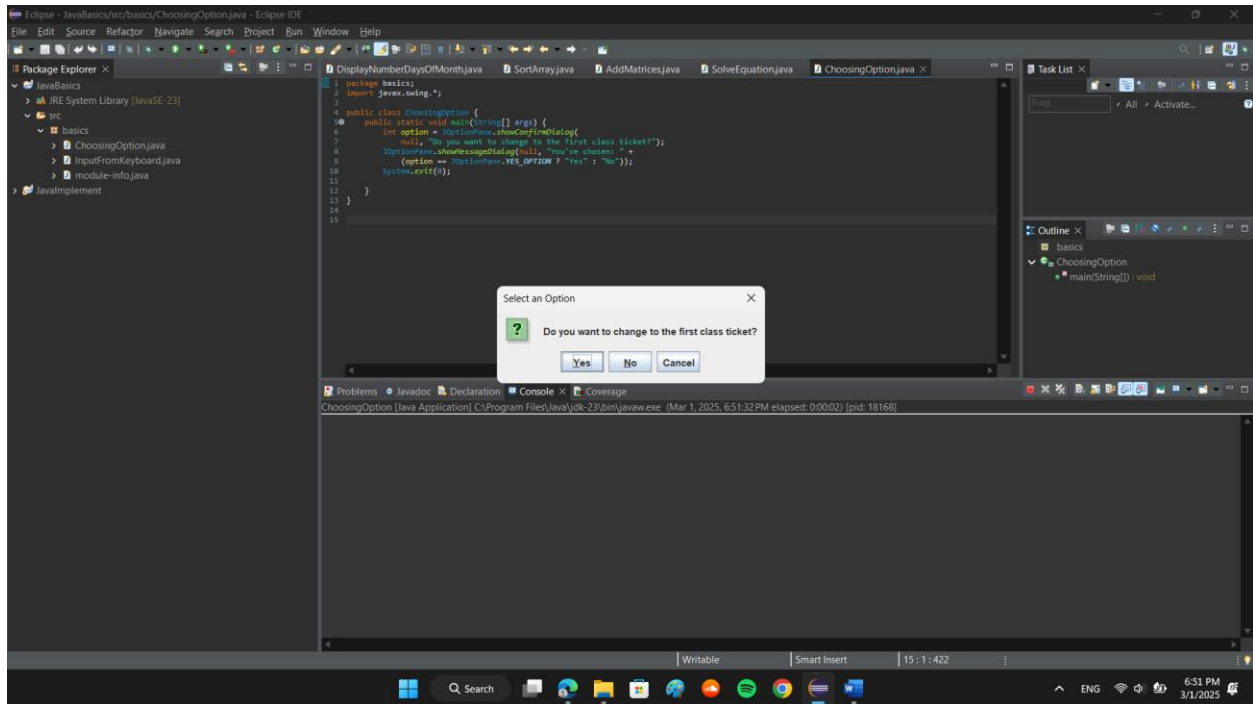
Assignment 6.1: Write, compile and run the ChoosingOption program

Source code:

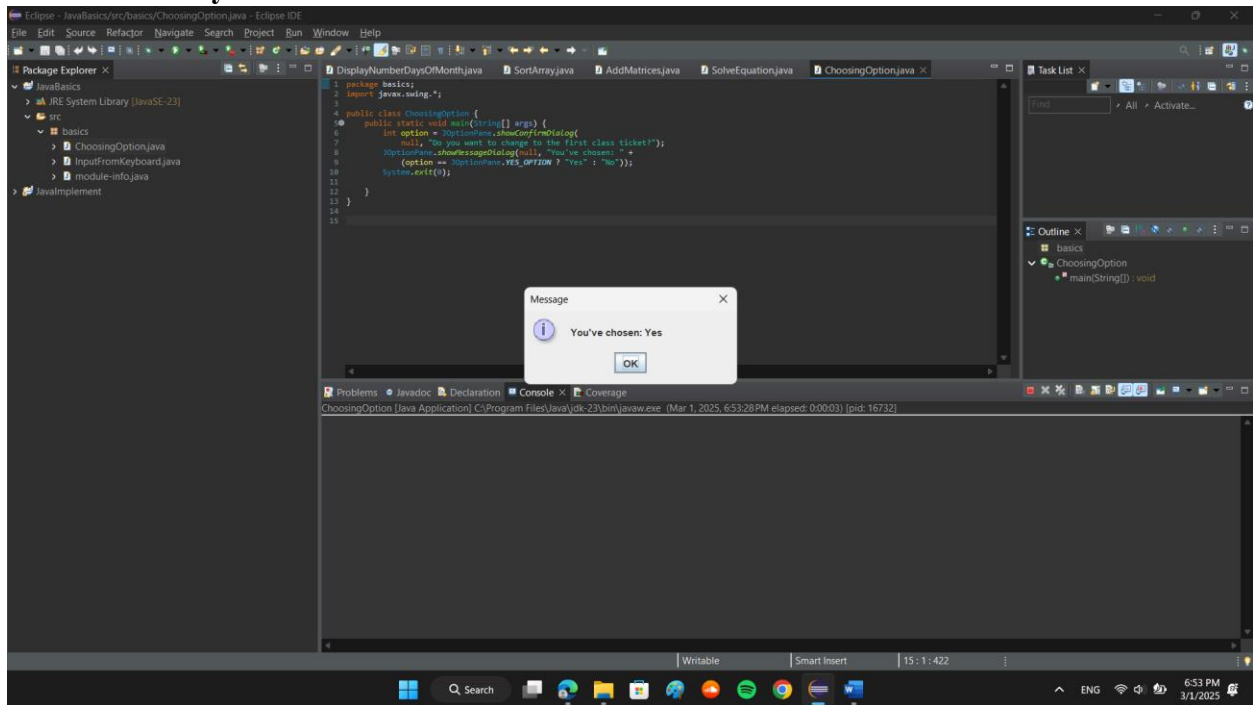
```
package basics;
import javax.swing.*;
```

```
public class ChoosingOption {
    public static void main(String[] args) {
        int option = JOptionPane.showConfirmDialog(
            null, "Do you want to change to the first class ticket?");
        JOptionPane.showMessageDialog(null, "You've chosen: " +
            (option == JOptionPane.YES_OPTION ? "Yes" : "No"));
        System.exit(0);
    }
}
```

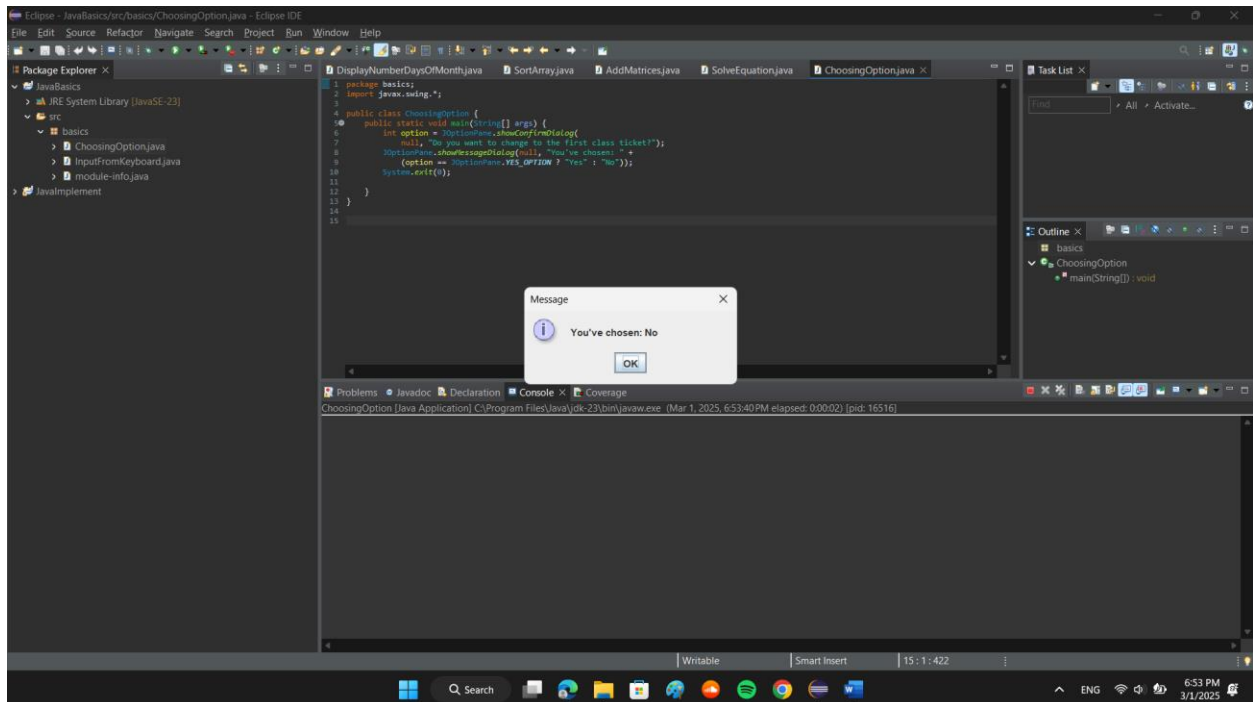
Result:



When choose “yes”:



When choose “no/cancel”:



Assignment 6.2: Write a program for input/output from keyboard

Source code:

```
package basics;
import java.util.*;

public class InputFromKeyboard {
    public static void main(String[] args) {
        Scanner keyboard = new Scanner(System.in);

        System.out.println("What's your name?");
        String strName = keyboard.nextLine();
        System.out.println("How old are you?");
        int iAge = keyboard.nextInt();
        System.out.println("How tall are you (m)?");
        double dHeight = keyboard.nextDouble();
        System.out.println("Mrs/Ms. " + strName + ", " + iAge + " years old. "
            + "Your height is: " + dHeight + ". ");
        keyboard.close();
    }
}
```

Result:

The screenshot displays the Eclipse IDE interface. The Package Explorer on the left shows a project named 'JavaBasics' with a source folder 'src' containing several Java files, including 'InputFromKeyboard.java'. The main editor window shows the code for 'InputFromKeyboard.java', which is a Java class with a 'main' method. The code prompts the user for their name, age, and height, and then prints out the collected information. The Console window at the bottom shows the output of the program, indicating that it was executed successfully and the user input was correctly processed.

```
1 package basics;
2
3 import java.util.*;
4
5 public class InputFromKeyboard {
6     public static void main(String[] args) {
7         Scanner keyboard = new Scanner(System.in);
8
9         System.out.println("What's your name?");
10        String strName = keyboard.nextLine();
11        System.out.println("How old are you?");
12        int Age = keyboard.nextInt();
13        System.out.println("How tall are you (m)?");
14        double dHeight = keyboard.nextDouble();
15        System.out.println("Mrs/Ms. " + strName + ", " + Age + " years old. " +
16            "Your height is: " + dHeight + "m.");
17        keyboard.close();
18    }
19 }
20
```

Console Output:

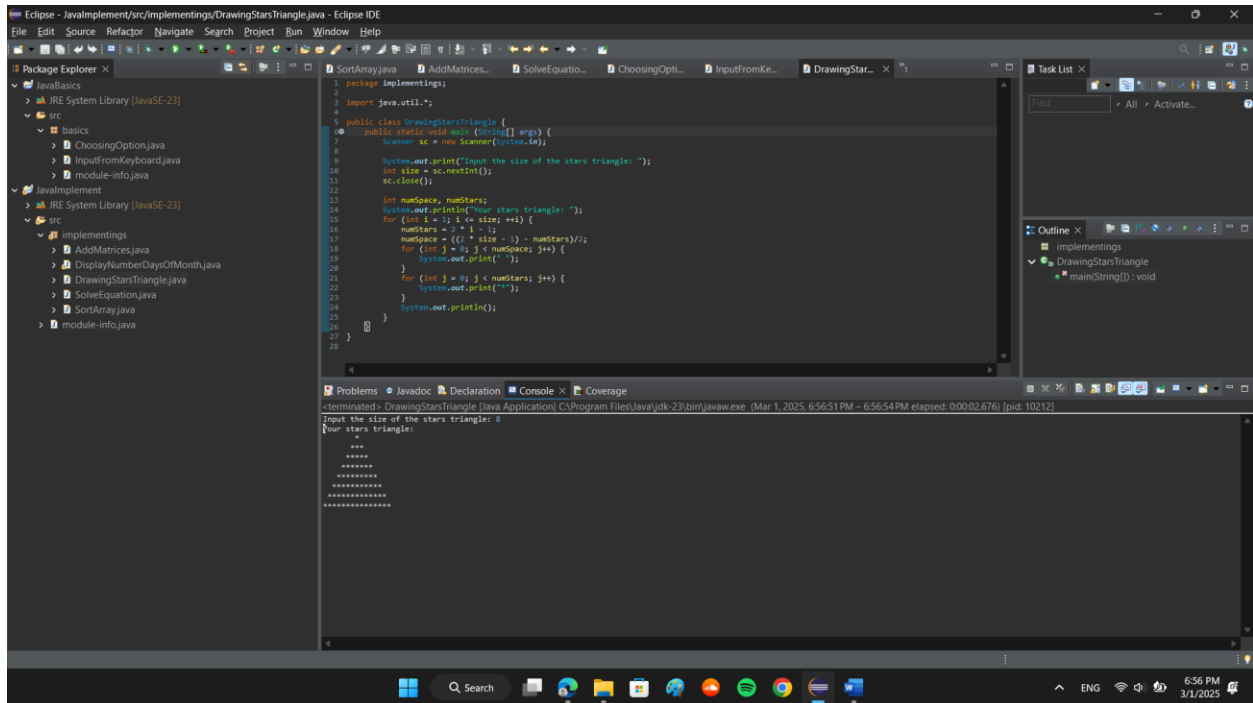
```
<terminated> InputFromKeyboard [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 6:55:37 PM - 6:55:45 PM elapsed: 00:00:07.793) [pid: 15732]
What's your name?
Trang
How old are you?
19
How tall are you (m)?
1.60
Mrs/Ms. Trang, 19 years old. Your height is: 1.6.
```


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

Source code:

```
package implementings;
import java.util.*;
public class DrawingStarsTriangle {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Input the size of the stars triangle: ");
        int size = sc.nextInt();
        sc.close();
        int numSpace, numStars;
        System.out.println("Your stars triangle: ");
        for (int i = 1; i <= size; ++i) {
            numStars = 2 * i - 1;
            numSpace = ((2 * size - 1) - numStars)/2;
            for (int j = 0; j < numSpace; j++) {
                System.out.print(" ");
            }
            for (int j = 0; j < numStars; j++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

Result:



Assignment 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.

Source code:

```
package implementings;
```

```
import java.util.*;
```

```

public class DisplayNumberDaysOfMonth {
    public static void main(String[] args) {
        HashMap<String, Integer> monthDays = new HashMap<String, Integer>();
        int iInputYear = -1, iInputMonth = -1;
        String buffer;
        String[][] monthNames = {
            {"January", "Jan.", "Jan", "1"}, {"February", "Feb.", "Feb", "2"},
            {"March", "Mar.", "Mar", "3"}, {"April", "Apr.", "Apr", "4"},
            {"May", "May", "May", "5"}, {"June", "Jun.", "Jun", "6"},
            {"July", "Jul.", "Jul", "7"}, {"August", "Aug.", "Aug", "8"},
            {"September", "Sep.", "Sep", "9"}, {"October", "Oct.", "Oct", "10"},
            {"November", "Nov.", "Nov", "11"}, {"December", "Dec.", "Dec", "12"}
        };
        int[] dayOfMonths = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
        for (int i = 0; i < 12; i++) {
            for (String names : monthNames[i]) {
                monthDays.put(names.toLowerCase(), i+1);
            }
        }
    }
}

```

```

    }
}

Scanner sc = new Scanner(System.in);
while (iInputYear < 0) {
    System.out.print("Enter a year: ");
    if (sc.hasNextInt()) {
        iInputYear = sc.nextInt();
        if (iInputYear < 0) {
            System.out.println("Invalid year!");
        }
    }
    else {
        System.out.println("Invalid year!");
        sc.next();
    }
}
sc.nextLine();

while(iInputMonth < 0) {
    System.out.print("Enter a month: ");
    buffer = sc.next().toLowerCase();
    if (monthDays.containsKey(buffer)) {
        iInputMonth = monthDays.get(buffer);
    }
    else {
        System.out.println("Invalid month!");
    }
}
int res = dayOfMonths[iInputMonth-1];
if (iInputMonth == 2) {
    if ((iInputYear % 4 == 0 && iInputYear % 100 != 0) || (iInputYear % 400
== 0)) {
        res = 29;
    }
}
System.out.println(iInputMonth + "/" + iInputYear + " has " + res + " days");
return;
}
}

```

Result:

Handle invalid year/month + Jan./1024

```
104 string[] monthNames = {
105     "January", "Jan", "Jan", "1", {"February", "Feb.", "Feb", "2"},
106     "March", "Mar.", "Mar.", "3"}, {"April", "Apr.", "Apr", "4"},
107     "May", "May", "May", "5"}, {"June", "Jun.", "Jun", "6"},
108     "July", "Jul.", "Jul.", "7"}, {"August", "Aug.", "Aug", "8"},
109     "September", "Sep.", "Sep", "9"}, {"October", "Oct.", "Oct", "10"},
110     "November", "Nov.", "Nov", "11"}, {"December", "Dec.", "Dec", "12"}
111 };
112 int[] dayOfMonths = {11, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
113 for (int i = 0; i < 12; i++) {
114     for (String names : monthNames[i]) {
115         monthDays.put(names.toLowerCase(), i+1);
116     }
117 }
118 Scanner sc = new Scanner(System.in);
119 while (inputYear < 0) {
120     System.out.print("Enter a year: ");
121     if (sc.hasNextInt()) {
122         inputYear = sc.nextInt();
123         if (inputYear < 0) {
124             System.out.println("Invalid year!");
125         }
126     }
127     else {
128         System.out.println("Invalid year!");
129     }
130     sc.next();
131 }
132 while (true) {
133     System.out.print("Enter a month: ");
134     if (sc.hasNextString()) {
135         inputMonth = sc.next();
136         if (!monthDays.containsKey(inputMonth.toLowerCase())) {
137             System.out.println("Invalid month!");
138         }
139     }
140     else {
141         System.out.println("Invalid month!");
142     }
143     sc.next();
144 }
145 int year = inputYear;
146 int month = inputMonth;
147 int days = monthDays.get(inputMonth.toLowerCase());
148 System.out.println(year + "/" + month + " has " + days + " days");
149 }
```

Console Output:

```
<terminated> DisplayNumberDaysOfMonth [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 6:58:59 PM - 6:59:17 PM elapsed: 0:00:17.582) [pid: 12816]
Enter a year: 42107
Invalid year!
Enter a year: 2323
Invalid year!
Enter a year: 23.23
Invalid year!
Enter a year: 1024
Invalid month: eurf
Invalid month:
Enter a month: Jan
Invalid month:
Enter a month: Jan.
1/1024 has 31 days
```

Feb/1900:

```
104 string[] monthNames = {
105     "January", "Jan", "Jan", "1", {"February", "Feb.", "Feb", "2"},
106     "March", "Mar.", "Mar.", "3"}, {"April", "Apr.", "Apr", "4"},
107     "May", "May", "May", "5"}, {"June", "Jun.", "Jun", "6"},
108     "July", "Jul.", "Jul.", "7"}, {"August", "Aug.", "Aug", "8"},
109     "September", "Sep.", "Sep", "9"}, {"October", "Oct.", "Oct", "10"},
110     "November", "Nov.", "Nov", "11"}, {"December", "Dec.", "Dec", "12"}
111 };
112 int[] dayOfMonths = {11, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
113 for (int i = 0; i < 12; i++) {
114     for (String names : monthNames[i]) {
115         monthDays.put(names.toLowerCase(), i+1);
116     }
117 }
118 Scanner sc = new Scanner(System.in);
119 while (inputYear < 0) {
120     System.out.print("Enter a year: ");
121     if (sc.hasNextInt()) {
122         inputYear = sc.nextInt();
123         if (inputYear < 0) {
124             System.out.println("Invalid year!");
125         }
126     }
127     else {
128         System.out.println("Invalid year!");
129     }
130     sc.next();
131 }
132 while (true) {
133     System.out.print("Enter a month: ");
134     if (sc.hasNextString()) {
135         inputMonth = sc.next();
136         if (!monthDays.containsKey(inputMonth.toLowerCase())) {
137             System.out.println("Invalid month!");
138         }
139     }
140     else {
141         System.out.println("Invalid month!");
142     }
143     sc.next();
144 }
145 int year = inputYear;
146 int month = inputMonth;
147 int days = monthDays.get(inputMonth.toLowerCase());
148 System.out.println(year + "/" + month + " has " + days + " days");
149 }
```

Console Output:

```
<terminated> DisplayNumberDaysOfMonth [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 7:00:08 PM - 7:00:16 PM elapsed: 0:00:07.906) [pid: 10504]
Enter a year: 1900
Enter a month: Feb
2/1900 has 28 days
```

February/2000:

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like `javaBasics`, `src`, and `module-info.java`.
- Editor:** Displays the source code of `DisplayNumberDaysOfMonth.java`. The code includes a `main` method that takes user input for a year and month, and uses a `switch` statement to determine the number of days in the month. The code is as follows:

```
11 String[] months = {"January", "Jan", "Jan", "February", "Feb", "Feb", "2"},
12 {"March", "Mar", "Mar", "3"}, {"April", "Apr", "Apr", "4"},
13 {"May", "May", "May", "5"}, {"June", "Jun", "Jun", "6"},
14 {"July", "Jul", "Jul", "7"}, {"August", "Aug", "Aug", "8"},
15 {"September", "Sep", "Sep", "9"}, {"October", "Oct", "Oct", "10"},
16 {"November", "Nov", "Nov", "11"}, {"December", "Dec", "Dec", "12"};
17
18 int[] dayOfMonth = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
19
20 for (int i = 0; i < 12; i++) {
21     for (String names : months[i]) {
22         monthDays.put(names.toLowerCase(), i+1);
23     }
24 }
25
26 Scanner sc = new Scanner(System.in);
27 while (true) {
28     System.out.print("Enter a year: ");
29     if (sc.hasNextInt()) {
30         inputYear = sc.nextInt();
31         if (inputYear < 0) {
32             System.out.println("Invalid year!");
33         }
34     } else {
35         System.out.println("Invalid year!");
36     }
37     sc.next();
38 }
39
40 System.out.print("Enter a month: ");
41 if (sc.hasNextInt()) {
42     inputMonth = sc.nextInt();
43     if (inputMonth < 0) {
44         System.out.println("Invalid month!");
45     }
46 } else {
47     System.out.println("Invalid month!");
48 }
49 sc.next();
50
51 int days = 0;
52 for (String names : months[inputMonth-1]) {
53     days = monthDays.get(names.toLowerCase());
54 }
55
56 System.out.println("The number of days in the month of " + inputMonth + " is " + days);
```
- Task List:** Shows the task list with the task `main(String[]) : void`.
- Outline:** Shows the outline of the `main` method.
- Problems:** Shows the problems list with the message `<terminated> DisplayNumberDaysOfMonth [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 7:00:42 PM - 7:00:55 PM elapsed: 0:00:13.633) [pid: 19108]`.
- Console:** Shows the console output:

```
Enter a year: 2000
Enter a month: February
2/2000 has 29 days
```

2/2016:

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like `javaBasics`, `src`, and `module-info.java`.
- Editor:** Displays the source code of `DisplayNumberDaysOfMonth.java`. The code is the same as in the previous screenshot.
- Task List:** Shows the task list with the task `main(String[]) : void`.
- Outline:** Shows the outline of the `main` method.
- Problems:** Shows the problems list with the message `<terminated> DisplayNumberDaysOfMonth [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 7:01:42 PM - 7:01:46 PM elapsed: 0:00:03.925) [pid: 6628]`.
- Console:** Shows the console output:

```
Enter a year: 2016
Enter a month: 2
2/2016 has 29 days
```

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

Soucre code:

```
package implementings;

import java.util.*;

public class SortArray {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] numbers;
        System.out.print("Manually? (yes/no): ");
        String choice = scanner.nextLine().toLowerCase();

        if (choice.equals("yes")) {
            System.out.print("Enter the number of elements: ");
            int n = scanner.nextInt();
            numbers = new int[n];

            System.out.println("Enter " + n + " numbers:");
            for (int i = 0; i < n; i++) {
                numbers[i] = scanner.nextInt();
            }
        } else {
            numbers = new int[]{1789, 2035, 1899, 1456, 2013};
            System.out.println("Default array: " + Arrays.toString(numbers));
        }

        scanner.close();

        Arrays.sort(numbers);

        int sum = 0;
        for (int num : numbers) {
            sum += num;
        }
        double average = (double) sum / numbers.length;

        System.out.println("Sorted Array: " + Arrays.toString(numbers));
        System.out.println("Sum: " + sum);
        System.out.printf("Average: %.2f\n", average);
    }
}
```


Result:

User enter their array:

The screenshot shows the Eclipse IDE with the `SortArray.java` file open. The code defines a `SortArray` class with a `main` method that prompts the user to enter the number of elements and the elements themselves. The console output shows the user entering 4 elements: 1224, 2424, 223123, and 122. The sorted array is [122, 1224, 2424, 223123], with a sum of 224893 and an average of 56723.25.

```
1 package Implementings;
2 import java.util.*;
3
4 public class SortArray {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         int[] numbers;
8         System.out.print("Manually? (yes/no): ");
9         String choice = scanner.nextLine().toLowerCase();
10
11         if (choice.equals("yes")) {
12             System.out.print("Enter the number of elements: ");
13             int n = scanner.nextInt();
14             numbers = new int[n];
15
16             System.out.println("Enter " + n + " numbers:");
17             for (int i = 0; i < n; i++) {
18                 numbers[i] = scanner.nextInt();
19             }
20             numbers = new int[]{1789, 2035, 1899, 1456, 2013};
21             System.out.println("Default array: " + Arrays.toString(numbers));
22         }
23         scanner.close();
24         Arrays.sort(numbers);
25     }
26 }
27
28
29
```

Manually? (yes/no): yes
Enter the number of elements: 4
Enter 4 numbers:
1224 2424 223123 122
Sorted Array: [122, 1224, 2424, 223123]
Sum: 224893
Average: 56723.25

Use default array: [1789, 2035, 1899, 1456, 2013]

The screenshot shows the Eclipse IDE with the `SortArray.java` file open. The code is the same as in the previous screenshot, but the user has chosen the default array. The console output shows the default array [1789, 2035, 1899, 1456, 2013], which is sorted to [1456, 1789, 1899, 2013, 2035], with a sum of 9192 and an average of 1838.40.

```
1 package Implementings;
2 import java.util.*;
3
4 public class SortArray {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         int[] numbers;
8         System.out.print("Manually? (yes/no): ");
9         String choice = scanner.nextLine().toLowerCase();
10
11         if (choice.equals("yes")) {
12             System.out.print("Enter the number of elements: ");
13             int n = scanner.nextInt();
14             numbers = new int[n];
15
16             System.out.println("Enter " + n + " numbers:");
17             for (int i = 0; i < n; i++) {
18                 numbers[i] = scanner.nextInt();
19             }
20             numbers = new int[]{1789, 2035, 1899, 1456, 2013};
21             System.out.println("Default array: " + Arrays.toString(numbers));
22         }
23         scanner.close();
24         Arrays.sort(numbers);
25     }
26 }
27
28
29
```

Manually? (yes/no): no
Default array: [1789, 2035, 1899, 1456, 2013]
Sorted array: [1456, 1789, 1899, 2013, 2035]
Sum: 9192
Average: 1838.40

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

Source code:

```
package implementings;

import java.util.*;

public class AddMatrices {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter number of rows: ");
        int rows = scanner.nextInt();
        System.out.print("Enter number of columns: ");
        int cols = scanner.nextInt();

        int[][] matrix1 = new int[rows][cols];
        int[][] matrix2 = new int[rows][cols];
        int[][] sumMatrix = new int[rows][cols];

        System.out.println("Enter elements of first matrix:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix1[i][j] = scanner.nextInt();
            }
        }

        System.out.println("Enter elements of second matrix:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix2[i][j] = scanner.nextInt();
            }
        }

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                sumMatrix[i][j] = matrix1[i][j] + matrix2[i][j];
            }
        }

        System.out.println("Sum of matrices:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                System.out.print(sumMatrix[i][j] + " ");
            }
        }
    }
}
```

```

    }
    System.out.println();
}

scanner.close();
}
}

```

Result:

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like `src`, `basics`, and `javainplement`.
- Editor:** Displays the `AddMatrices.java` file. The code is as follows:


```

1 package implementings;
2
3 import java.util.*;
4
5 public class AddMatrices {
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in);
8
9         System.out.print("Enter number of rows: ");
10        int rows = scanner.nextInt();
11        System.out.print("Enter number of columns: ");
12        int cols = scanner.nextInt();
13
14        int[][] matrix1 = new int[rows][cols];
15        int[][] matrix2 = new int[rows][cols];
16        int[][] sumMatrix = new int[rows][cols];
17
18        System.out.println("Enter elements of first matrix:");
19        for (int i = 0; i < rows; i++) {
20            for (int j = 0; j < cols; j++) {
21                matrix1[i][j] = scanner.nextInt();
22            }
23        }
24
25        System.out.println("Enter elements of second matrix:");
26        for (int i = 0; i < rows; i++) {
27            for (int j = 0; j < cols; j++) {
28                matrix2[i][j] = scanner.nextInt();
29            }
30        }
31    }
32 }
      
```
- Console:** Shows the output of the program:


```

Enter number of rows: 2
Enter number of columns: 3
Enter elements of first matrix:
1 2 3
0 8 4
Enter elements of second matrix:
12 343 45
12 566 224210
Sum of matrices:
13 345 50
18 574 324210
      
```
- Task List:** Empty.
- Outline:** Shows the `main(String[]) : void` method.
- Problems:** Shows a message: "terminated- AddMatrices [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Mar 1, 2025, 7:06:23 PM - 7:06:35 PM elapsed: 0:00:12:559) [pid: 3832]"