

# IT3100E Object-Oriented Programming Lab Report

Name: Nguyễn Đức Anh

Student ID: 20235890

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

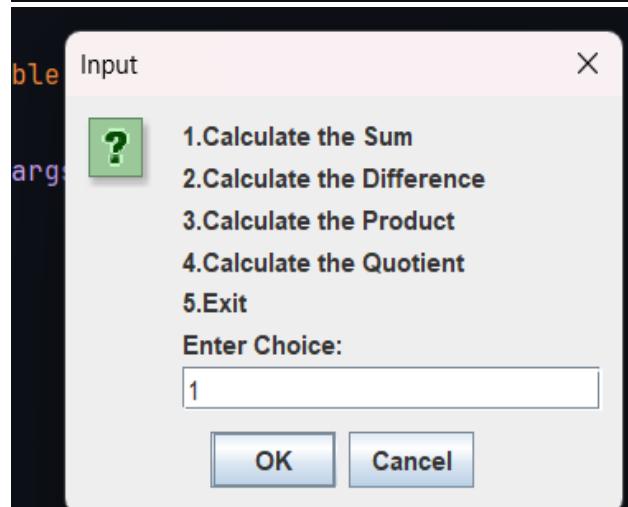
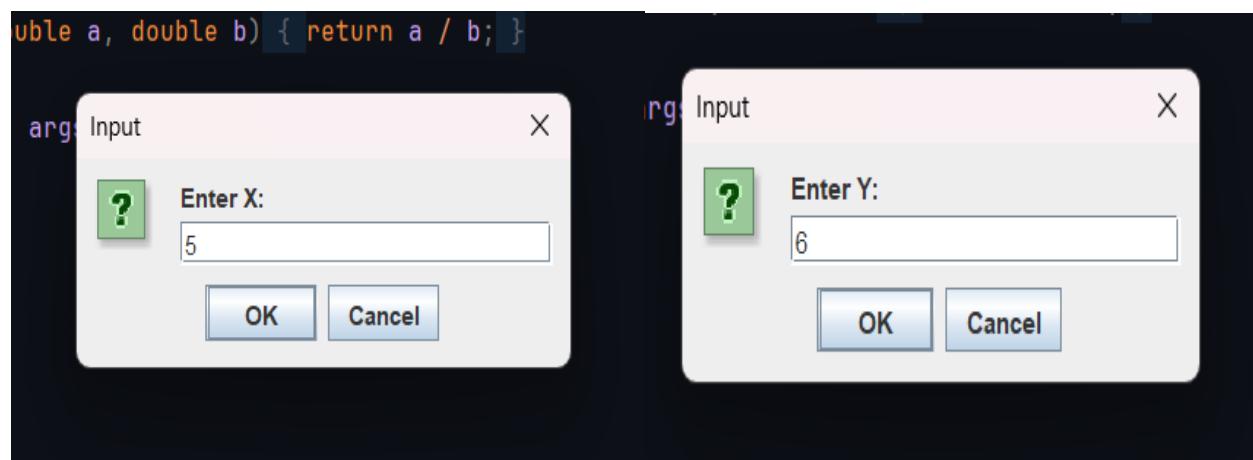
**Source code:**

```
import javax.swing.*;  
public class Main  
{  
    public static double Sum(double a, double b) { return a + b; }  
    public static double Difference(double a, double b) { return a - b; }  
    public static double Product(double a, double b) { return a * b; }  
    public static double Quotient(double a, double b) { return a / b; }  
    public static void main(String[] args)  
    {  
        double sum;  
        double difference;  
        double product;  
        double quotient;  
        String x;  
        String y;  
  
        x = JOptionPane.showInputDialog("Enter X: ");  
        y = JOptionPane.showInputDialog("Enter Y: ");  
  
        double X = Double.parseDouble(x);  
        double Y = Double.parseDouble(y);  
  
        int choice;  
        choice = Integer.parseInt(JOptionPane.showInputDialog("1.Calculate the Sum\n" +  
                "2.Calculate the Difference\n" +  
                "3.Calculate the Product\n" +  
                "4.Calculate the Quotient\n" +  
                "5.Exit\n" +  
                "Enter Choice: "));  
        switch (choice)  
        {  
            case 1:  
                sum = Sum(X, Y);  
                JOptionPane.showMessageDialog(null, message: "The Sum of " + x + " and " + y + " is " + sum);  
                break;  
  
            case 2:  
                difference = Difference(X, Y);  
                JOptionPane.showMessageDialog(null, message: "The Difference of " + x + " and " + y + " is " + difference);  
                break;  
  
            case 3:  
                product = Product(X, Y);  
                JOptionPane.showMessageDialog(null, message: "The Product of " + x + " and " + y + " is " + product);  
                break;  
  
            case 4:  
                quotient = Quotient(X, Y);  
                JOptionPane.showMessageDialog(null, message: "The Quotient of " + x + " and " + y + " is " + quotient);  
                break;  
  
            case 5:  
                break;  
  
            default:  
                JOptionPane.showMessageDialog(null, message: "Invalid Choice");  
        }  
    }  
}
```

```
public static void main(String[] args)  
{  
    "5.Exit\n" +  
    "Enter Choice: "));  
    switch (choice)  
    {  
        case 1:  
            sum = Sum(X, Y);  
            JOptionPane.showMessageDialog(null, message: "The Sum of " + x + " and " + y + " is " + sum);  
            break;  
  
        case 2:  
            difference = Difference(X, Y);  
            JOptionPane.showMessageDialog(null, message: "The Difference of " + x + " and " + y + " is " + difference);  
            break;  
  
        case 3:  
            product = Product(X, Y);  
            JOptionPane.showMessageDialog(null, message: "The Product of " + x + " and " + y + " is " + product);  
            break;  
  
        case 4:  
            quotient = Quotient(X, Y);  
            JOptionPane.showMessageDialog(null, message: "The Quotient of " + x + " and " + y + " is " + quotient);  
            break;  
  
        case 5:  
            break;  
  
        default:  
            JOptionPane.showMessageDialog(null, message: "Invalid Choice");  
    }  
}
```

**Result:**

```
ouble a, double b) { return a / b; }
```



## 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:

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Tree:** The project is named "Lab01" and contains a module named "Lab01 EASOICT-HUST IC". The "Code" directory under "src" contains a file named "FirstDegreeEquation.java".
- Code Editor:** The file "FirstDegreeEquation.java" is open. The code implements a class "FirstDegreeEquation" that takes two double parameters "a" and "b" and prints the solution. It handles the case where "a" is zero by displaying an error message. If "a" is not zero, it calculates the solution as  $x = (-b) / a$  and displays it.
- Status Bar:** The status bar at the bottom shows the path "Lab01 > Code > 2.2.6 > src > FirstDegreeEquation", the current time "12:14", and other system information like CPU usage.

```
1 * Print out the solution
2 */
3
4 public class FirstDegreeEquation { 2 usages
5 {
6     double a; 1 usage
7     double b; 1 usage
8
9     public FirstDegreeEquation(double a, double b) 1 usage
10    {
11        this.a = a;
12        this.b = b;
13    }
14
15    void solve(double a, double b) 1 usage
16    {
17        if (a == 0)
18        {
19            JOptionPane.showMessageDialog( parentComponent: null, message: "The equation has no solution.");
20        }
21        else
22        {
23            double x = (-b) / a;
24            JOptionPane.showMessageDialog( parentComponent: null, message: "x = " + x);
25        }
26    }
27
28 }
29
30 }
31
32 }
33
34 }
```

The screenshot shows a Java IDE interface with the following details:

- Project View:** Shows a project structure under "Lab01 E:\SOICT-HUST IC". The "SecondDegreeEquation.java" file is selected.
- Code Editor:** Displays the content of "SecondDegreeEquation.java". The code defines a class "SecondDegreeEquation" with a constructor, a "solve" method, and a main method for testing.
- Output:** Shows the output of the main method, indicating the equation has no solution and printing the roots x1 and x2.
- File Explorer:** Shows files like "Report.docx", "Report.pdf", and "Scratches and Consoles".
- System Status:** Top right shows system status with icons for battery, signal, and network.

```
public class SecondDegreeEquation {  
    public SecondDegreeEquation(double a, double b, double c) {  
        this.a = a;  
        this.b = b;  
        this.c = c;  
    }  
  
    void solve(double a, double b, double c) {  
        double delta = b * b - 4 * a * c;  
  
        if (delta < 0)  
        {  
            JOptionPane.showMessageDialog(null, "The equation has no solution.");  
        }  
        else if (delta == 0)  
        {  
            double x = -b / (2 * a);  
            JOptionPane.showMessageDialog(null, "x1 = x2 = " + x);  
        }  
        else  
        {  
            double x1 = (-b + Math.sqrt(delta)) / (2 * a);  
            double x2 = (-b - Math.sqrt(delta)) / (2 * a);  
            JOptionPane.showMessageDialog(null, "x1 = " + x1 + "\nx2 = " + x2);  
        }  
    }  
}  
  
public static void main(String[] args) {  
    SecondDegreeEquation equation = new SecondDegreeEquation(1, 2, 1);  
    equation.solve();  
}
```

The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E:\SOICT-HUST IC
- Current File:** SystemOfFirstDegreeEquations.java
- Code Content:**

```
public class SystemOfFirstDegreeEquations {  
    public SystemOfFirstDegreeEquations(double a11, double a12, double b1, double a21, double a22, double b2) {  
        this.a11 = a11;  
        this.a12 = a12;  
        this.b1 = b1;  
        this.a21 = a21;  
        this.a22 = a22;  
        this.b2 = b2;  
    }  
  
    void solve(double a11, double a12, double b1, double a21, double a22, double b2) {  
        double D = a11 * a12 - a21 * a22;  
        double D1 = b1 * a22 - b2 * a12;  
        double D2 = a11 * b2 - a21 * b1;  
  
        if (D != 0) {  
            double x1 = D1 / D;  
            double x2 = D2 / D;  
  
            JOptionPane.showMessageDialog(null, "x1 = " + x1 + "\nx2 = " + x2);  
        } else if ((D1 != 0) || (D2 != 0)) {  
            JOptionPane.showMessageDialog(null, "The equation has no solution!");  
        } else {  
            JOptionPane.showMessageDialog(null, "The equation has infinity solution!");  
        }  
    }  
}
```

- Toolbars and Status Bar:** The status bar at the bottom shows "14:14 CRLF UTF-8 4 spaces" and the system tray indicates "CPU 2% LAT N/A".

The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E:\SOICT-HUST IC
- Current File:** SystemOfFirstDegreeEquations.java
- Code Content:**

```
public class SystemOfFirstDegreeEquations {  
    public SystemOfFirstDegreeEquations(double a11, double a12, double b1, double a21, double a22, double b2) {  
    }  
  
    void solve(double a11, double a12, double b1, double a21, double a22, double b2) {  
        double D = a11 * a12 - a21 * a22;  
        double D1 = b1 * a22 - b2 * a12;  
        double D2 = a11 * b2 - a21 * b1;  
  
        if (D != 0) {  
            double x1 = D1 / D;  
            double x2 = D2 / D;  
  
            JOptionPane.showMessageDialog(null, "x1 = " + x1 + "\nx2 = " + x2);  
        } else if ((D1 != 0) || (D2 != 0)) {  
            JOptionPane.showMessageDialog(null, "The equation has no solution!");  
        } else {  
            JOptionPane.showMessageDialog(null, "The equation has infinity solution!");  
        }  
    }  
}
```

- Toolbars and Status Bar:** The status bar at the bottom shows "14:14 CRLF UTF-8 4 spaces" and the system tray indicates "CPU 2% LAT N/A".

The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E\SOICT-HUST IC
- Current File:** Main.java
- Code Content:**

```
import javax.swing.*;  
public class Main  
{  
    public static void main(String[] args)  
    {  
        int choice = Integer.parseInt(JOptionPane.showInputDialog("1.Solve the first degree equation with one variable\\n2.Solve the system of first-degree equations"));  
        switch(choice)  
        {  
            case 1:  
                double firstDegree_a = Double.parseDouble(JOptionPane.showInputDialog("Enter the first parameter (a): "));  
                double firstDegree_b = Double.parseDouble(JOptionPane.showInputDialog("Enter the second parameter (b): "));  
  
                FirstDegreeEquation first_degree_equation = new FirstDegreeEquation(firstDegree_a, firstDegree_b);  
  
                first_degree_equation.solve(firstDegree_a, firstDegree_b);  
                break;  
  
            case 2:  
                double a11 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (a11): "));  
                double a12 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (a12): "));  
                double b1 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (b1): "));  
                double a21 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (a21): "));  
                double a22 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (a22): "));  
                double b2 = Double.parseDouble(JOptionPane.showInputDialog("Enter the parameter (b2): "));  
  
                SystemOfFirstDegreeEquations system_of_first_degree_equation = new SystemOfFirstDegreeEquations(a11, a12, b1, a21, a22, b2);  
  
                system_of_first_degree_equation.solve(a11, a12, b1, a21, a22, b2);  
                break;  
        }  
    }  
}
```

- System Tray:** Shows various icons including a battery at 100%, network, volume, and system status.

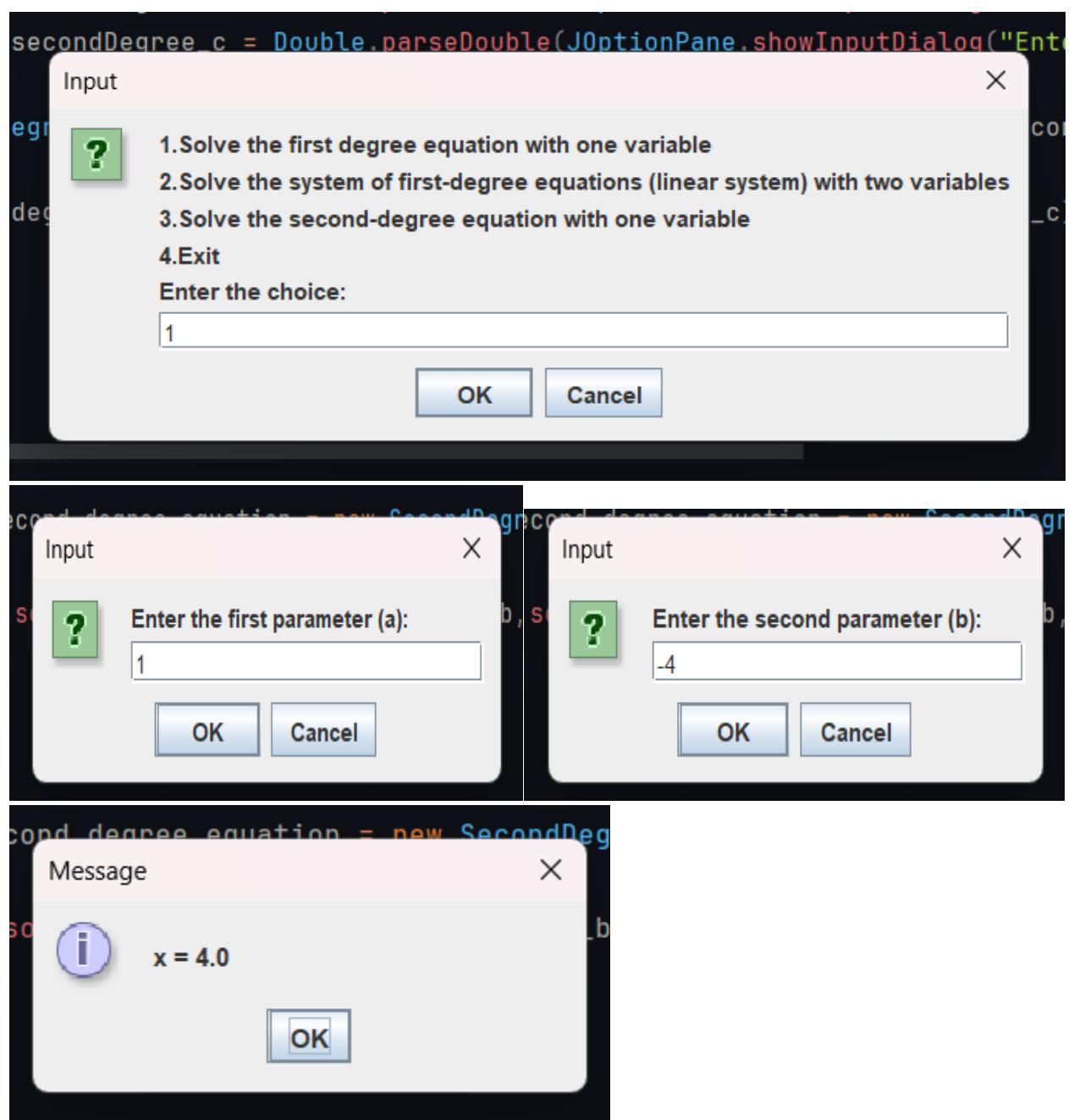
The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E\SOICT-HUST IC
- Current File:** Main.java
- Code Content:**

```
public class Main  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Welcome to the quadratic equation solver!");  
        System.out.println("Please enter the coefficients for the quadratic equation ax^2 + bx + c = 0.");  
        System.out.print("Enter coefficient a: ");  
        double a = scanner.nextDouble();  
        System.out.print("Enter coefficient b: ");  
        double b = scanner.nextDouble();  
        System.out.print("Enter coefficient c: ");  
        double c = scanner.nextDouble();  
  
        SecondDegreeEquation secondDegreeEquation = new SecondDegreeEquation(a, b, c);  
  
        secondDegreeEquation.solve();  
        System.out.println("The solutions are: " + secondDegreeEquation.getResults());  
    }  
}
```

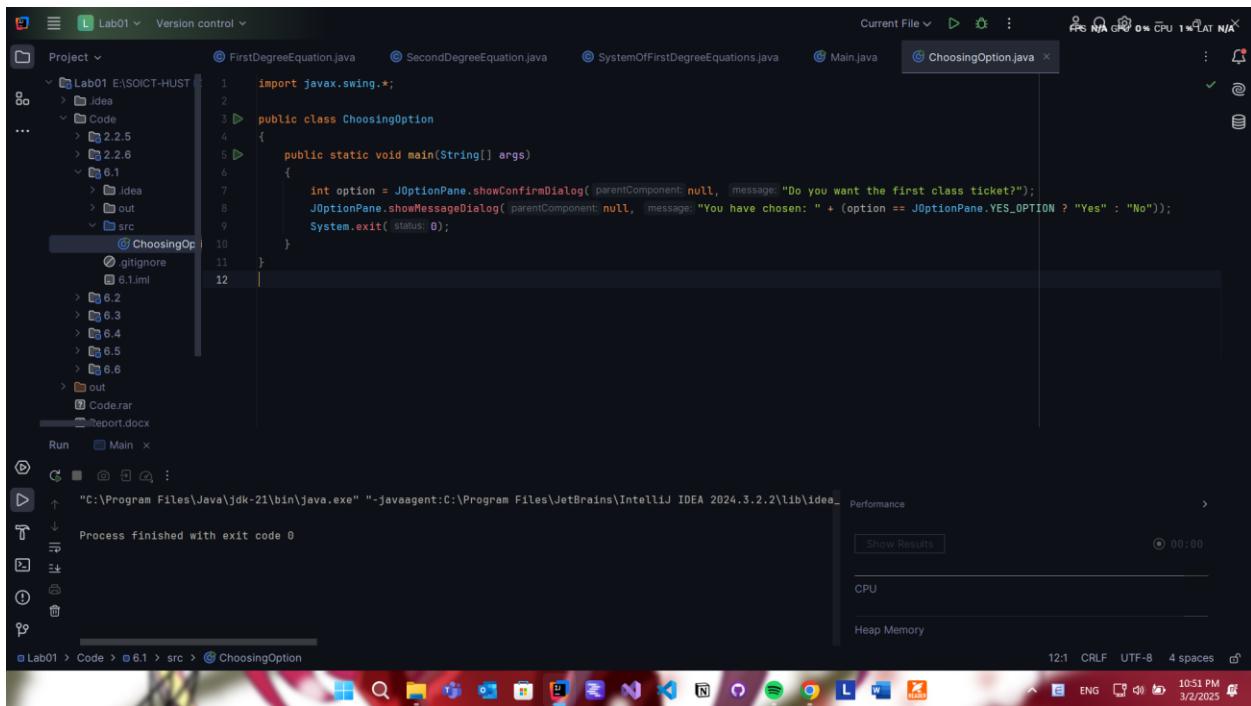
- System Tray:** Shows various icons including a battery at 100%, network, volume, and system status.

**Result:**



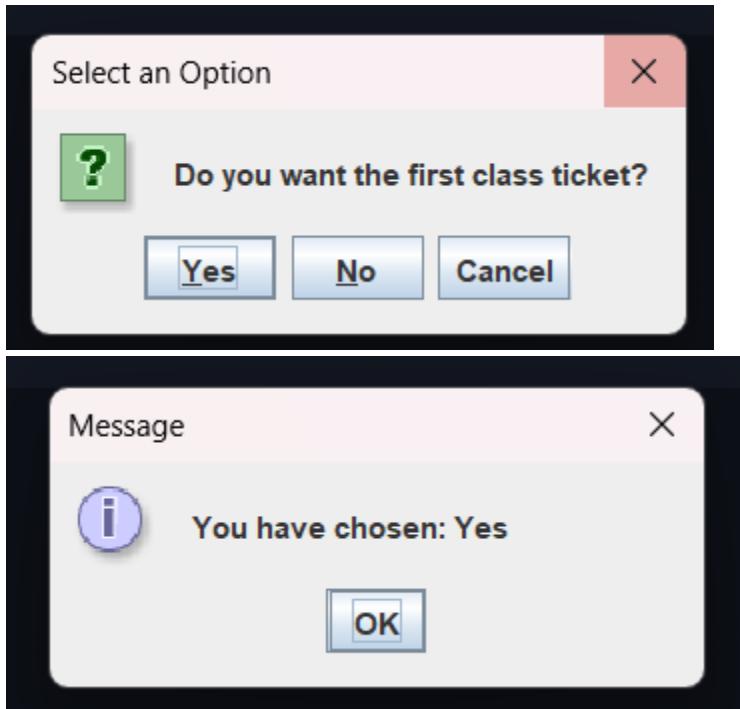
## 6.1 Write, compile and run the ChoosingOption program:

Source code:



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

Result:



## 6.2 Write a program for input/output from keyboard

### Source code:

The screenshot shows the IntelliJ IDEA interface with the project navigation bar at the top. The current file is `InputFromKeyboard.java`. The code itself is as follows:

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

        System.out.println("What is 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();

        //similar to other data types
        //nextByte(), nextShort(), nextLong(), nextFloat(), nextBoolean().

        System.out.println("Mrs/ms. " + strName + ", " + iAge + " years old. " + "Your age is " + dHeight + ".");
    }
}
```

The bottom status bar shows the time as 3:14 PM, date as 3/2/2025, and system information including CPU usage.

### Result:

The screenshot shows the IntelliJ IDEA interface with the run output window open. The command run was `"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\lib\idea_rt.jar=6136,C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\bin" -Dfile.encoding=UTF-8 InputFromKeyboard`. The output window displays the following interaction:

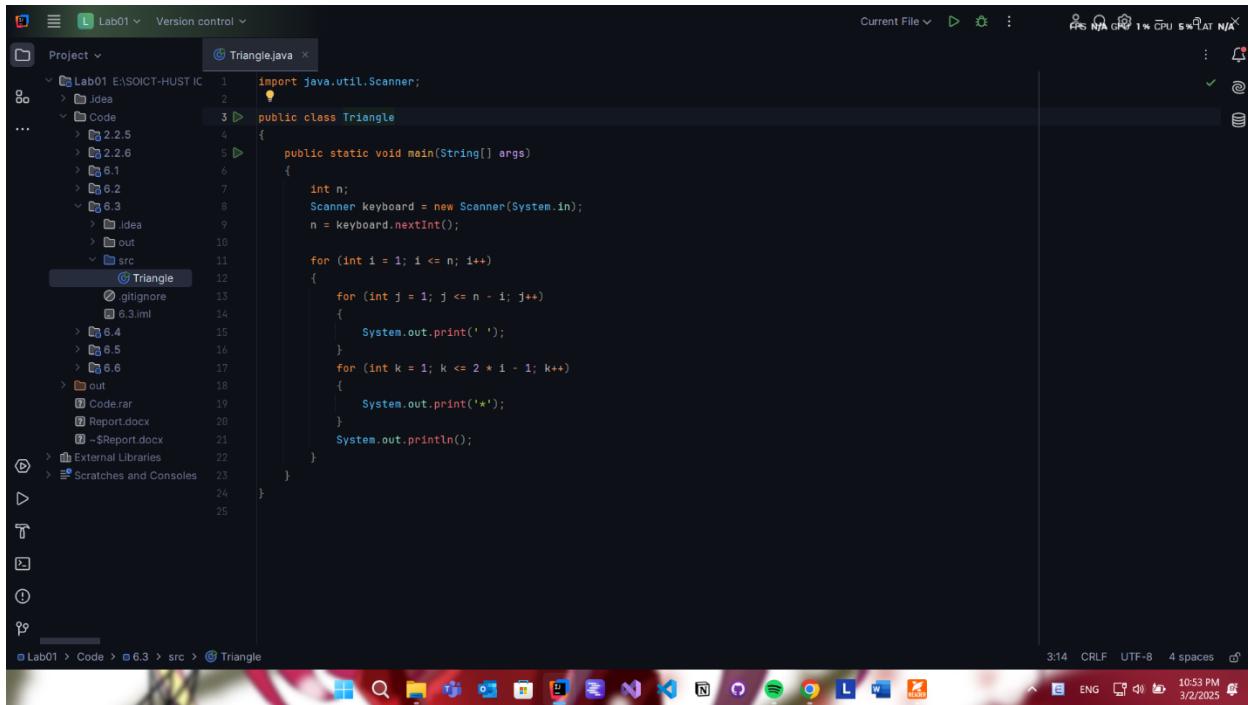
```
What is your name?
Duc Anh
How old are you?
20
How tall are you? (m)
1.75
Mrs/ms. Duc Anh, 20 years old. Your age is 1.75.

Process finished with exit code 0
```

The bottom status bar shows the time as 3:14 PM, date as 3/2/2025, and system information including CPU usage.

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

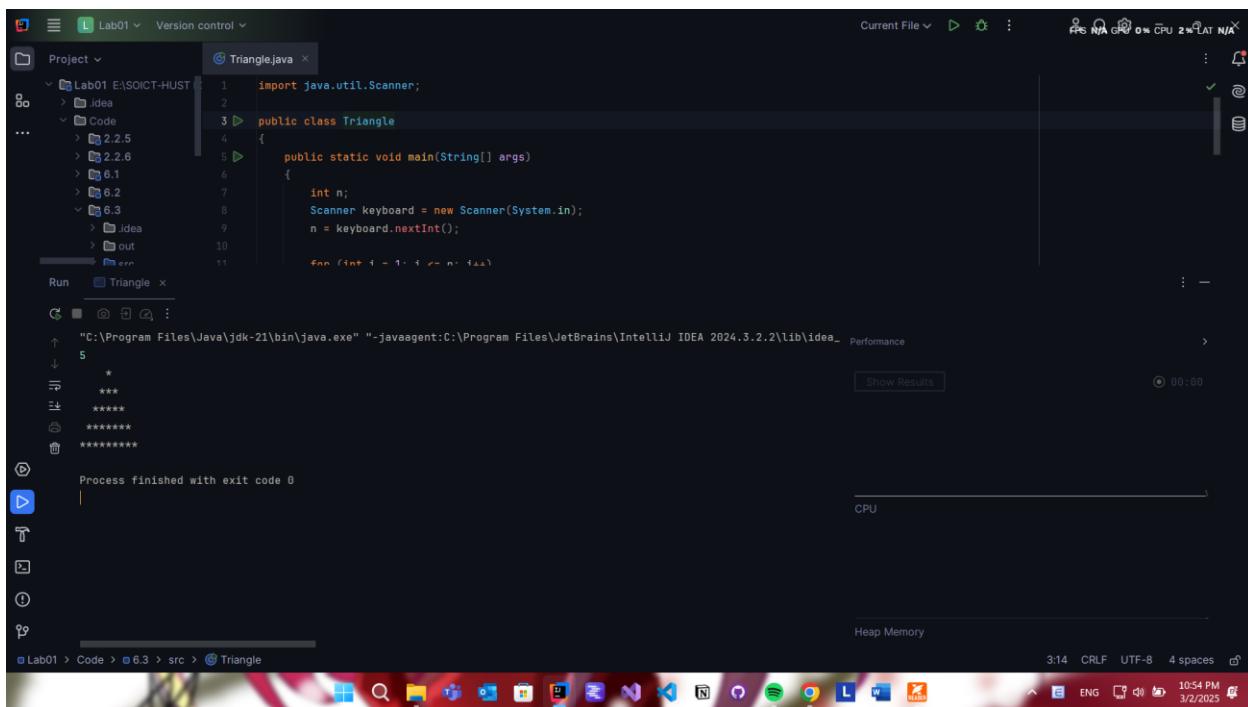
**Source code:**



```
import java.util.Scanner;
public class Triangle
{
    public static void main(String[] args)
    {
        int n;
        Scanner keyboard = new Scanner(System.in);
        n = keyboard.nextInt();

        for (int i = 1; i <= n; i++)
        {
            for (int j = 1; j <= n - i; j++)
            {
                System.out.print(' ');
            }
            for (int k = 1; k <= 2 * i - 1; k++)
            {
                System.out.print('*');
            }
            System.out.println();
        }
    }
}
```

**Result:**



```
import java.util.Scanner;
public class Triangle
{
    public static void main(String[] args)
    {
        int n;
        Scanner keyboard = new Scanner(System.in);
        n = keyboard.nextInt();

        for (int i = 1; i <= n; i++)
        {
            for (int j = 1; j <= n - i; j++)
            {
                System.out.print(' ');
            }
            for (int k = 1; k <= 2 * i - 1; k++)
            {
                System.out.print('*');
            }
            System.out.println();
        }
    }
}

"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\lib\idea_Performance
5
*
***
*****
*****
*****
```

**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:**

```
import java.util.HashMap;
import java.util.Scanner;

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

        HashMap<String, Integer> monthMap = new HashMap<>();
        String[][] monthInputs = {
            {"january", "jan", "1"}, // January
            {"february", "feb", "2"}, // February
            {"march", "mar", "3"}, // March
            {"april", "apr", "4"}, // April
            {"may", "may", "5"}, // May
            {"june", "june", "6"}, // June
            {"july", "july", "7"}, // July
            {"august", "aug", "8"}, // August
            {"september", "sept", "9"}, // September
            {"october", "oct", "10"}, // October
            {"november", "nov", "11"}, // November
            {"december", "dec", "12"} // December
        };

        for (int i = 0; i < monthInputs.length; i++)
        {
            for (String input : monthInputs[i])
            {
                monthMap.put(input, i + 1);
            }
        }
    }
}

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

String monthInput;
Integer monthNumber;
do
{
    System.out.println("Enter the month: ");
    monthInput = scanner.nextLine().toLowerCase();
    monthNumber = monthMap.get(monthInput);
    if (monthNumber == null)
    {
        System.out.println("Invalid month. Please enter again.");
    }
} while (monthNumber == null);

int year = 0;
boolean validYear = false;
do
{
    System.out.println("Enter the year: ");
}
```

The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E\SOICT-HUST IC
- File:** NumberOfDaysOfAMonth.java
- Code Content:**

```
public class NumberOfDaysOfAMonth
{
    public static void main(String[] args)
    {
        int year = 0;
        boolean validYear = false;
        do
        {
            System.out.println("Enter the year: ");
            String yearInput = scanner.nextLine();
            try
            {
                year = Integer.parseInt(yearInput);
                if (year >= 0)
                {
                    validYear = true;
                }
                else
                {
                    System.out.println("Year must be non-negative. Please enter again.");
                }
            }
            catch (NumberFormatException e)
            {
                System.out.println("Invalid year. Please enter a non-negative integer.");
            }
        } while (!validYear);

        int days;
        if (monthNumber == 2 && isLeapYear(year))
        {
            days = 29;
        }
    }
}
```

- IDE Status Bar:** Current File ▾ 4:14 CRLF UTF-8 4 spaces ⌂
- System Taskbar:** Shows various application icons.

The screenshot shows a Java IDE interface with the following details:

- Project:** Lab01 E\SOICT-HUST IC
- File:** NumberOfDaysOfAMonth.java
- Code Content:**

```
public class NumberOfDaysOfAMonth
{
    public static void main(String[] args)
    {
        ...
        catch (NumberFormatException e)
        {
            System.out.println("Invalid year. Please enter a non-negative integer.");
        }
    }
}

while (!validYear);

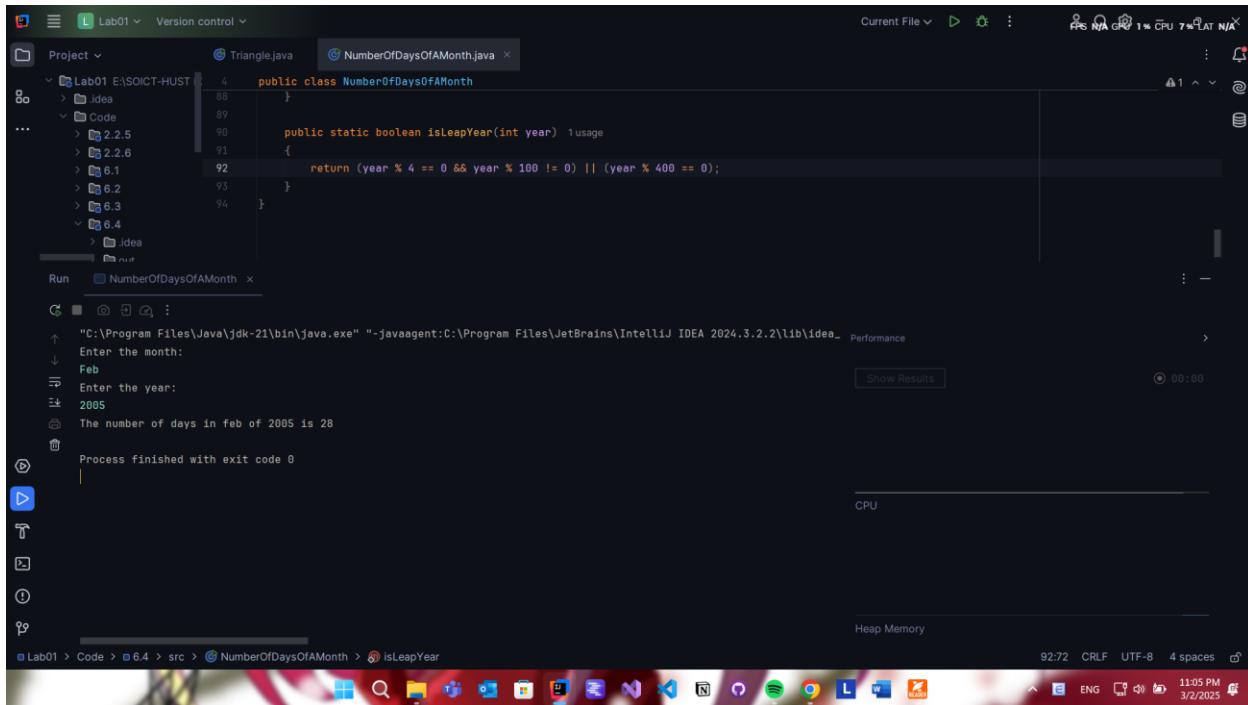
int days;
if (monthNumber == 2 && isLeapYear(year))
{
    days = 29;
}
else
{
    days = daysInMonth[monthNumber - 1];
}

System.out.println("The number of days in " + monthInput + " of " + year + " is " + days);
scanner.close();
}

public static boolean isLeapYear(int year) { return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0); }
}
```

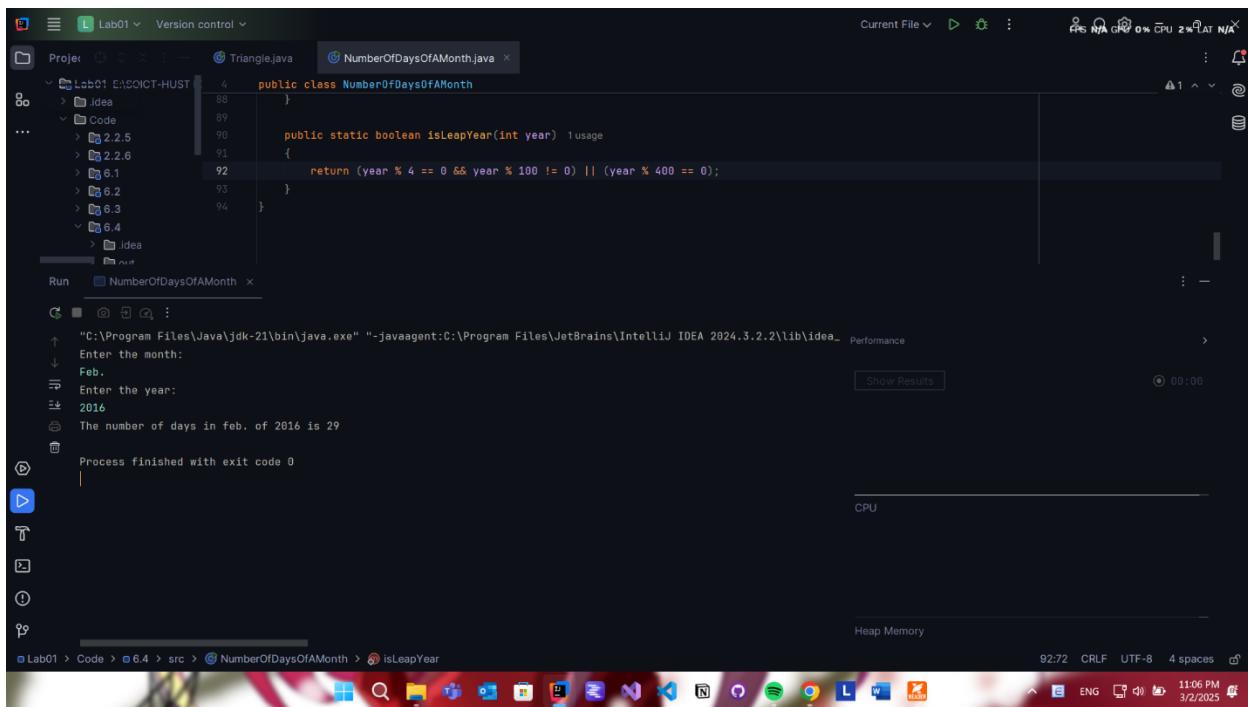
- IDE Status Bar:** Current File ▾ 4:14 CRLF UTF-8 4 spaces ⌂
- System Taskbar:** Shows various application icons.

## Result:



```
public class NumberOfDaysOfMonth {
    public static boolean isLeapYear(int year) { usage
        return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
    }
}

Enter the month:
Feb
Enter the year:
2005
The number of days in feb of 2005 is 28
```

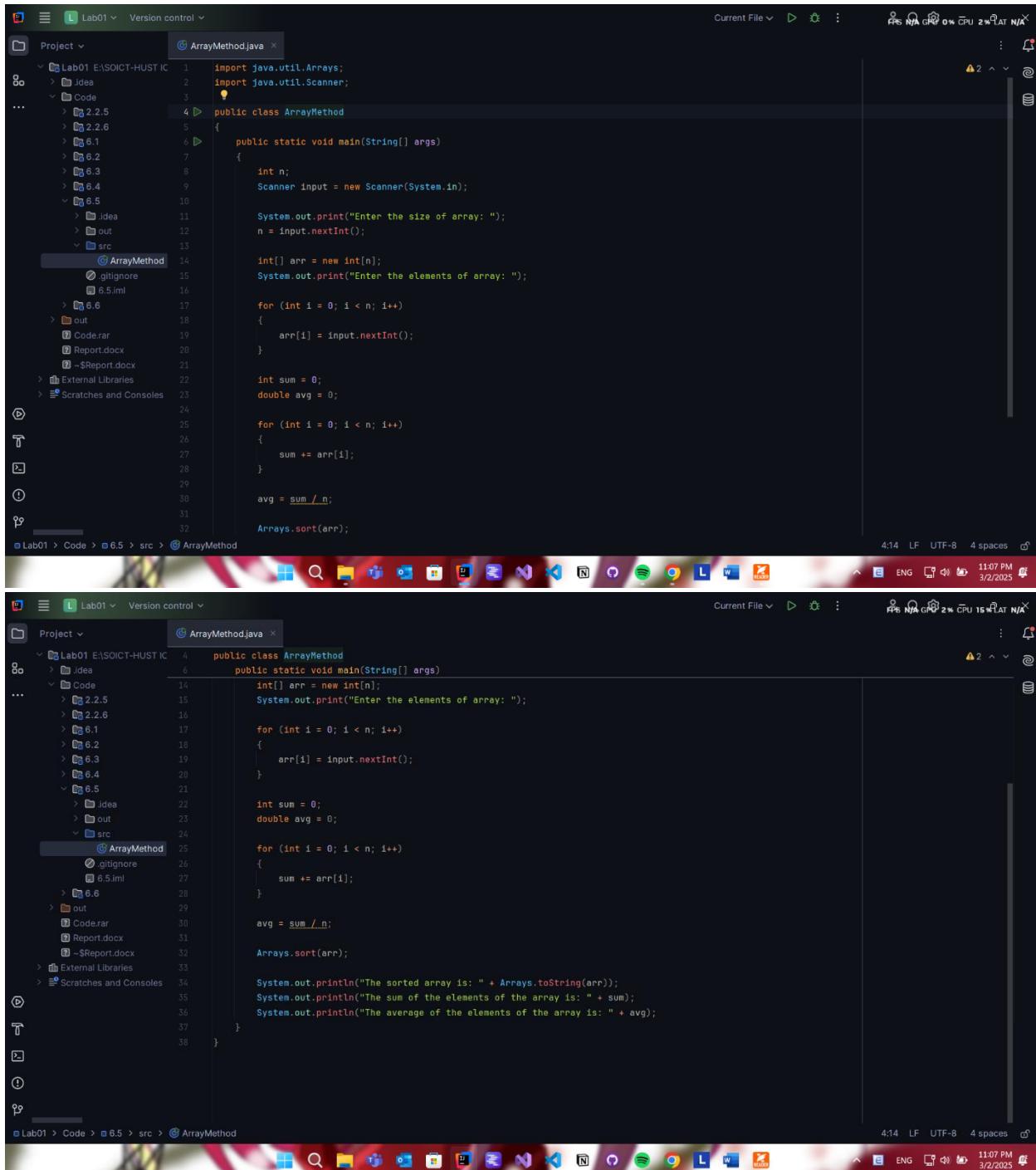


```
public class NumberOfDaysOfMonth {
    public static boolean isLeapYear(int year) { usage
        return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
    }
}

Enter the month:
Feb.
Enter the year:
2016
The number of days in feb. of 2016 is 29
```

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

### Source code:



```
import java.util.Arrays;
import java.util.Scanner;

public class ArrayMethod
{
    public static void main(String[] args)
    {
        int n;
        Scanner input = new Scanner(System.in);

        System.out.print("Enter the size of array: ");
        n = input.nextInt();

        int[] arr = new int[n];
        System.out.print("Enter the elements of array: ");

        for (int i = 0; i < n; i++)
        {
            arr[i] = input.nextInt();
        }

        int sum = 0;
        double avg = 0;

        for (int i = 0; i < n; i++)
        {
            sum += arr[i];
        }

        avg = sum / n;

        Arrays.sort(arr);

        System.out.println("The sorted array is: " + Arrays.toString(arr));
        System.out.println("The sum of the elements of the array is: " + sum);
        System.out.println("The average of the elements of the array is: " + avg);
    }
}
```

## Result:

The screenshot shows the IntelliJ IDEA interface with a Java project named 'Lab01'. The code editor displays 'ArrayMethod.java' with the following content:

```
public class ArrayMethod
{
    public static void main(String[] args)
    {
        int[] arr = new int[n];
        System.out.print("Enter the elements of array: ");

        for (int i = 0; i < n; i++)
        {
            arr[i] = input.nextInt();
        }

        int sum = 0;
        for (int i = 0; i < n; i++)
        {
            sum += arr[i];
        }

        System.out.println("The sorted array is: " + Arrays.toString(arr));
        System.out.println("The sum of the elements of the array is: " + sum);
        System.out.println("The average of the elements of the array is: " + (sum / n));
    }
}
```

The run tool window shows the following output:

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\lib\idea.jar" -Dfile.encoding=UTF-8 ArrayMethod
Enter the size of array: 5
Enter the elements of array: 1789
2035
1899
1456
2013
The sorted array is: [1456, 1789, 1899, 2013, 2035]
The sum of the elements of the array is: 9192
The average of the elements of the array is: 1838.0
```

The status bar at the bottom right shows the time as 4:14 PM, date as 3/27/2024, and system information including CPU usage.

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

## Source code:

The screenshot shows a Java code editor with the following details:

- Project:** Lab01
- File:** Matrices.java
- Code Content:**

```
import java.util.Scanner;
public class Matrices {
    public static int[][] readMatrix(Scanner scanner, int rows, int cols) {
        int[][] matrix = new int[rows][cols];
        for (int i = 0; i < rows; i++) {
            while (true) {
                System.out.print("Row " + (i + 1) + ": ");
                String line = scanner.nextLine();
                String[] parts = line.trim().split("\\s+");
                if (parts.length != cols) {
                    System.out.println("Invalid number of elements. Please enter " + cols + " numbers.");
                    continue;
                }
                try {
                    for (int j = 0; j < cols; j++) {
                        matrix[i][j] = Integer.parseInt(parts[j]);
                    }
                    break;
                }
                catch (NumberFormatException e) {
                    System.out.println("Invalid number. Please enter integers only.");
                }
            }
        }
        return matrix;
    }
}
```

- Toolbars and Status:** The top bar shows "Current File" and system status like "CPU 4%". The bottom bar shows "3:14 LF UTF-8 4 spaces" and a taskbar with various icons.

```
public class Matrices
{
    public static int[][] addMatrices(int[][] a, int[][] b) 1 usage
    {
        int rows = a.length;
        int cols = a[0].length;
        int[][] sum = new int[rows][cols];
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                sum[i][j] = a[i][j] + b[i][j];
            }
        }
        return sum;
    }

    public static void printMatrix(int[][] matrix) 1 usage
    {
        for (int[] row : matrix)
        {
            for (int elem : row)
            {
                System.out.print(elem + " ");
            }
            System.out.println();
        }
    }

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

        System.out.print("Enter the number of rows: ");
        int rows = Integer.parseInt(input.nextLine());
        System.out.print("Enter the number of columns: ");
        int cols = Integer.parseInt(input.nextLine());

        System.out.println("Enter the elements of the first matrix, row by row:");
        int[][] matrix1 = readMatrix(input, rows, cols);

        System.out.println("Enter the elements of the second matrix, row by row:");
        int[][] matrix2 = readMatrix(input, rows, cols);

        int[][] sum = addMatrices(matrix1, matrix2);

        System.out.println("The sum of the two matrices is:");
        printMatrix(sum);
    }
}
```

```
public class Matrices
{
    public static void printMatrix(int[][] matrix) 1 usage
    {
        System.out.println();
    }

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

        System.out.print("Enter the number of rows: ");
        int rows = Integer.parseInt(input.nextLine());
        System.out.print("Enter the number of columns: ");
        int cols = Integer.parseInt(input.nextLine());

        System.out.println("Enter the elements of the first matrix, row by row:");
        int[][] matrix1 = readMatrix(input, rows, cols);

        System.out.println("Enter the elements of the second matrix, row by row:");
        int[][] matrix2 = readMatrix(input, rows, cols);

        int[][] sum = addMatrices(matrix1, matrix2);

        System.out.println("The sum of the two matrices is:");
        printMatrix(sum);
    }
}
```

## Result:

The screenshot shows the IntelliJ IDEA interface during the execution of a Java program named `Matrices.java`. The code defines a class `Matrices` with methods for printing a matrix and adding two matrices. The IDE's run tool window displays the standard output of the program, which includes user prompts for matrix dimensions and elements, and the resulting sum matrix.

```
public class Matrices
{
    public static void printMatrix(int[][] matrix) {
        for (int i = 0; i < matrix.length; i++) {
            for (int j = 0; j < matrix[i].length; j++) {
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }

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

        System.out.print("Enter the number of rows: ");
        int rows = input.nextInt();

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

        System.out.print("Enter the elements of the first matrix, row by row:");
        int[][] matrix1 = new int[rows][cols];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix1[i][j] = input.nextInt();
            }
        }

        System.out.print("Enter the elements of the second matrix, row by row:");
        int[][] matrix2 = new int[rows][cols];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix2[i][j] = input.nextInt();
            }
        }

        int[][] result = addMatrices(matrix1, matrix2);
        printMatrix(result);
    }

    static int[][] addMatrices(int[][] matrix1, int[][] matrix2) {
        int[][] result = new int[matrix1.length][matrix1[0].length];
        for (int i = 0; i < matrix1.length; i++) {
            for (int j = 0; j < matrix1[i].length; j++) {
                result[i][j] = matrix1[i][j] + matrix2[i][j];
            }
        }
        return result;
    }
}
```

The terminal output shows:

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\lib\idea_rt.jar=5334,C:\Program Files\JetBrains\IntelliJ IDEA 2024.3.2.2\bin" -Dfile.encoding=UTF-8 Matrices
Enter the number of rows: 3
Enter the number of columns: 3
Enter the elements of the first matrix, row by row:
Row 1: 4 3 5
Row 2: 5 7 8
Row 3: 9 1 3
Enter the elements of the second matrix, row by row:
Row 1: 1 2 3
Row 2: 4 5 6
Row 3: 7 8 9
The sum of the two matrices is:
5 5 8
9 12 14
16 9 12
Process finished with exit code 0
```