

## BÁO CÁO THỰC HÀNH LAB 1- LAB 1 REPORT

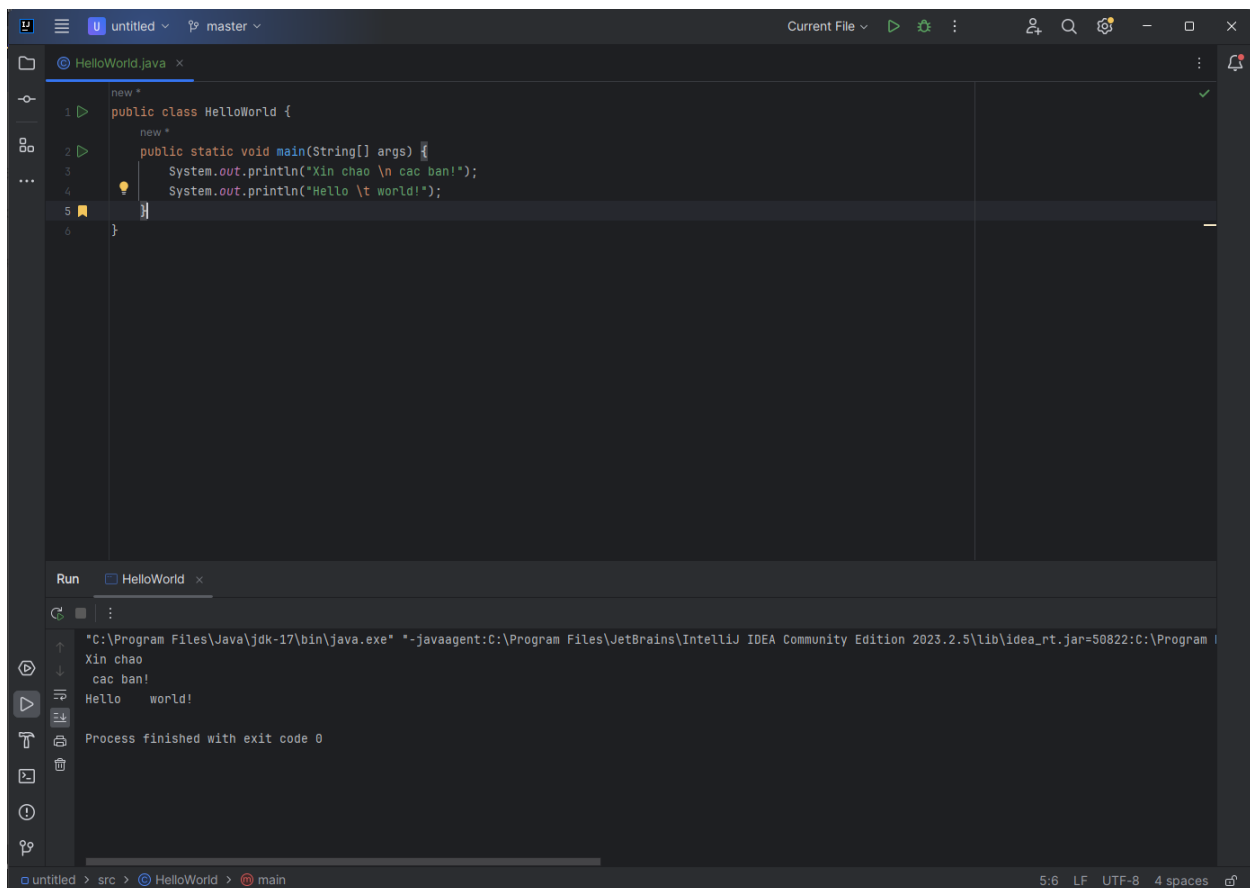
### LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG - Object oriented programming

#### The Very First Java Programs

##### 2.2.1 Write, compile the first Java application:

```
1 //Example 1: HelloWorld.java
2 //Text-printing program
3 public class HelloWorld {
4
5     public static void main(String args[]){
6         System.out.println("Xin chao \n cac ban!");
7         System.out.println("Hello \t world!");
8
9     } // end of method main
10 }
```

#### Result-Kết quả



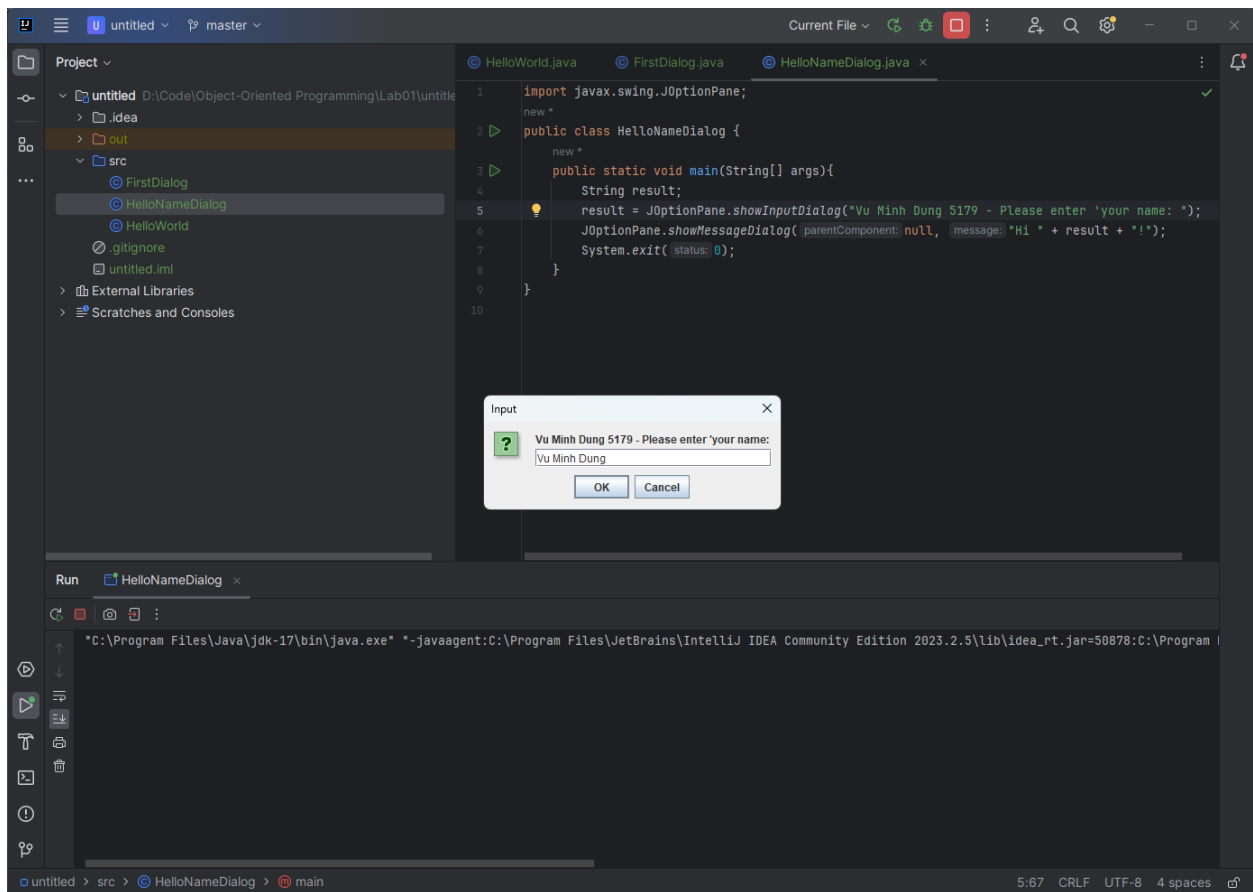
The screenshot displays the IntelliJ IDEA IDE interface. The top pane shows the source code for `HelloWorld.java`, which is a simple Java program with a `main` method that prints two lines of text. The bottom pane shows the output of the program's execution, which matches the printed text in the code. The status bar at the bottom indicates the file is in the `src` directory, the class is `HelloWorld`, and the method is `main`.

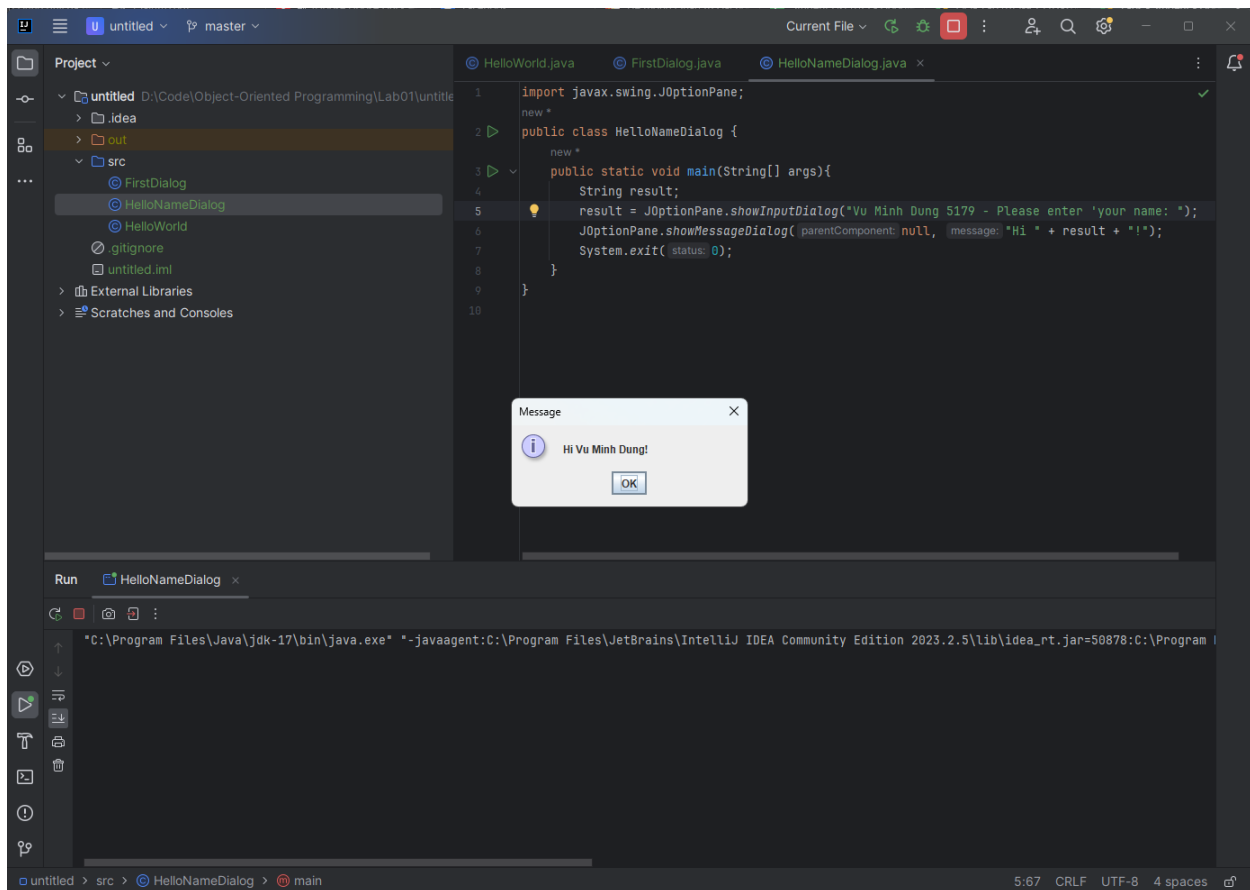
```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Xin chao \n cac ban!");
        System.out.println("Hello \t world!");
    }
}
```

Run HelloWorld

```
"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\lib\idea_rt.jar=50822:C:\Program Files\Java\jdk-17\bin" -Didea.config.path=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\conf -Didea.system.path=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\lib\idea_rt.jar -Didea.version=2023.2.5 -Xmx1024m -Xms64m -XX:MaxPermSize=256m -XX:+UseG1GC -XX:+UseStringDeduplication -XX:HeapDumpPath=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\bin\idea.log -jar C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\bin\idea_rt.jar 50822
Xin chao
cac ban!
Hello    world!
Process finished with exit code 0
```





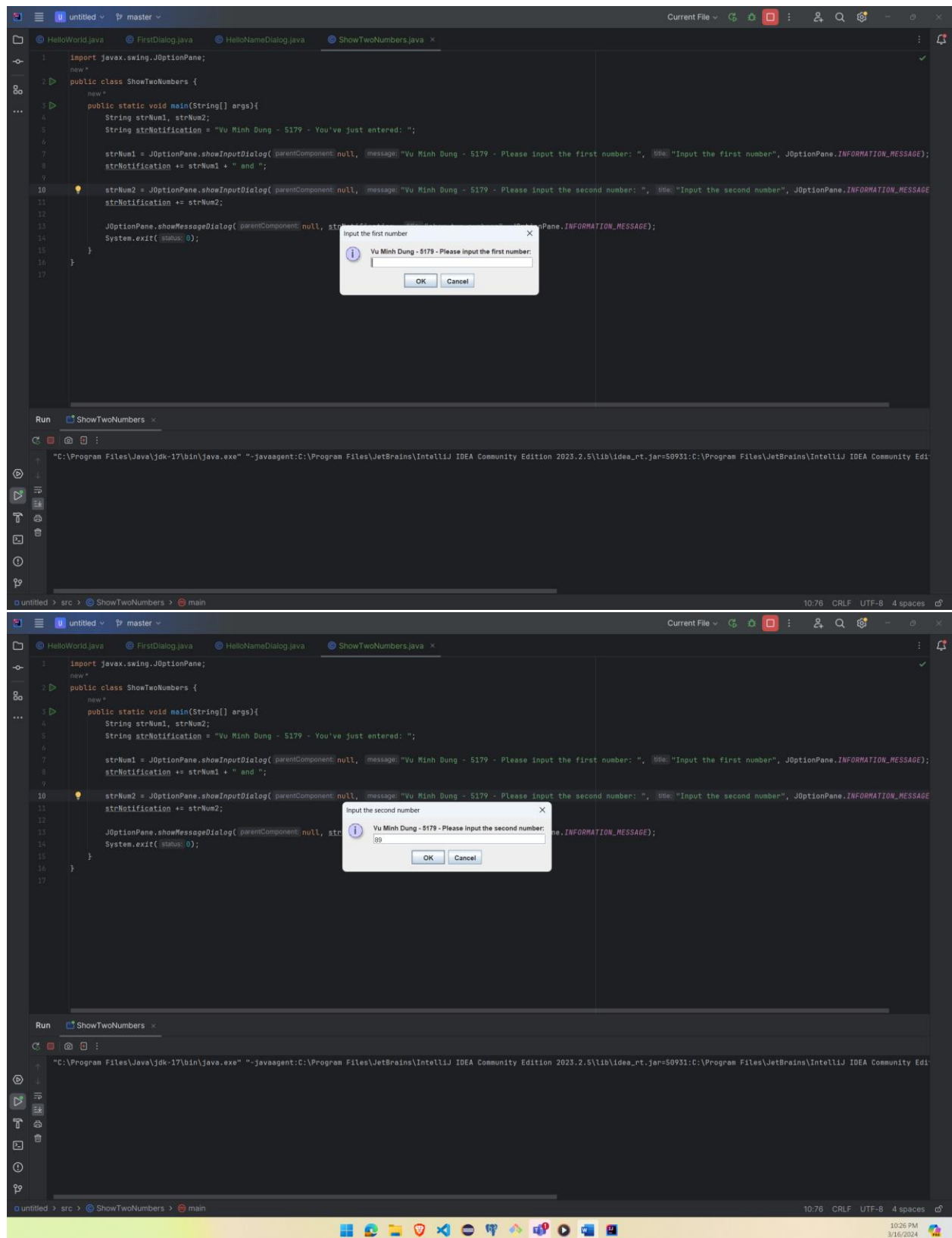


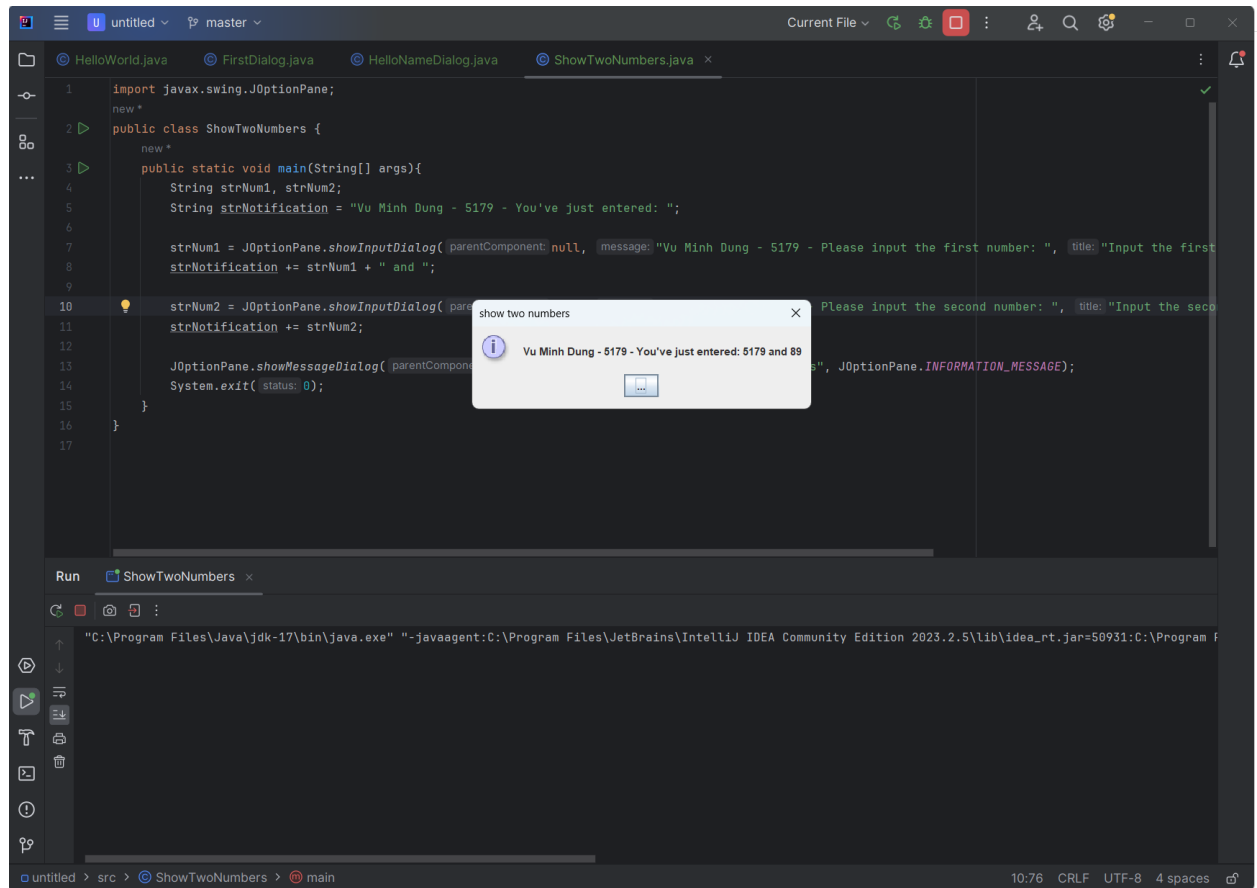
2.2.4 Write, compile, and run the following example:

```

1 // Example 5: ShowTwoNumbers.java
2 import javax.swing.JOptionPane;
3 public class ShowTwoNumbers {
4     public static void main(String[] args){
5         String strNum1, strNum2;
6         String strNotification = "You've just entered: ";
7
8         strNum1 = JOptionPane.showInputDialog(null,
9             "Please input the first number: ", "Input the first number",
10            JOptionPane.INFORMATION_MESSAGE);
11         strNotification += strNum1 + " and ";
12
13         strNum2 = JOptionPane.showInputDialog(null,
14             "Please input the second number: ", "Input the second number",
15            JOptionPane.INFORMATION_MESSAGE);
16         strNotification += strNum2;
17
18         JOptionPane.showMessageDialog(null, strNotification,
19             "Show two numbers", JOptionPane.INFORMATION_MESSAGE);
20         System.exit(0);
21     }
22 }

```

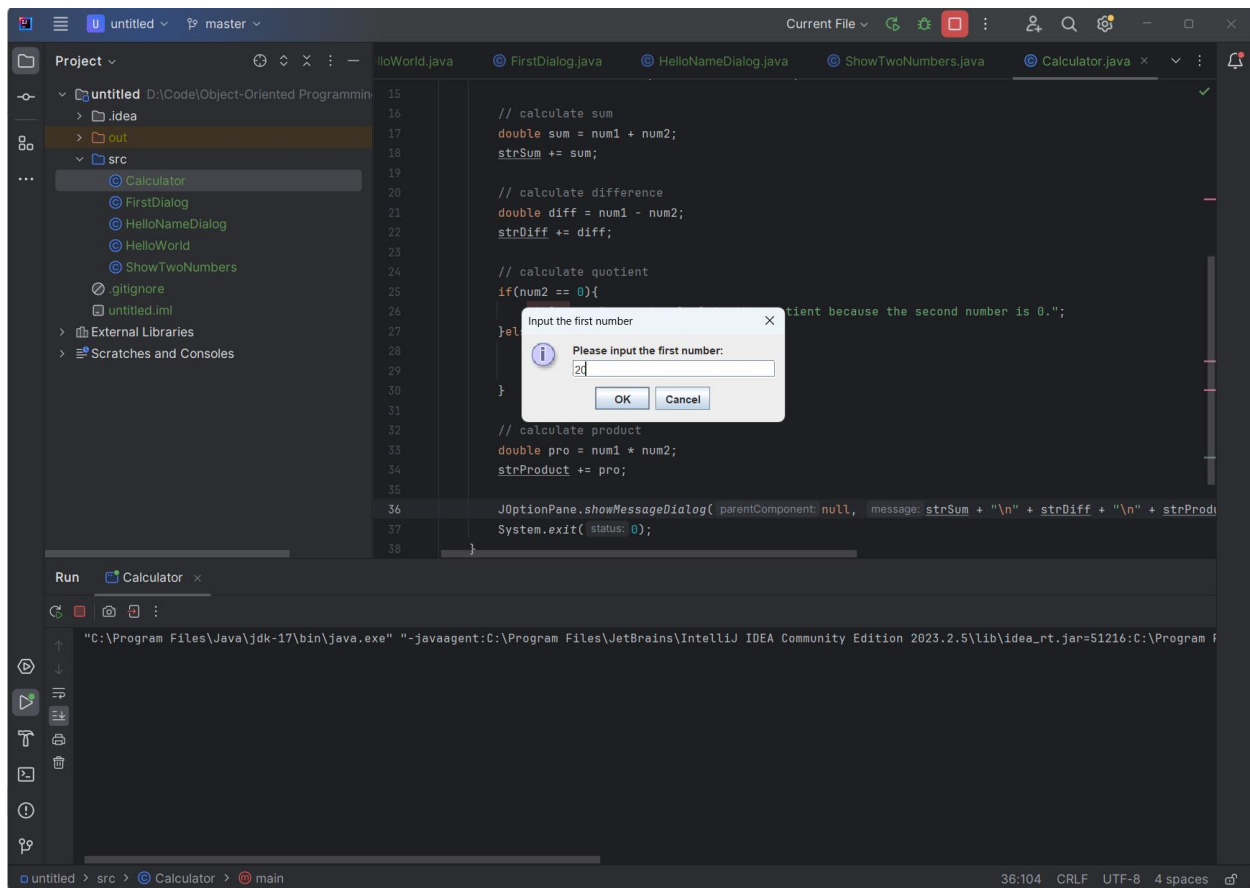


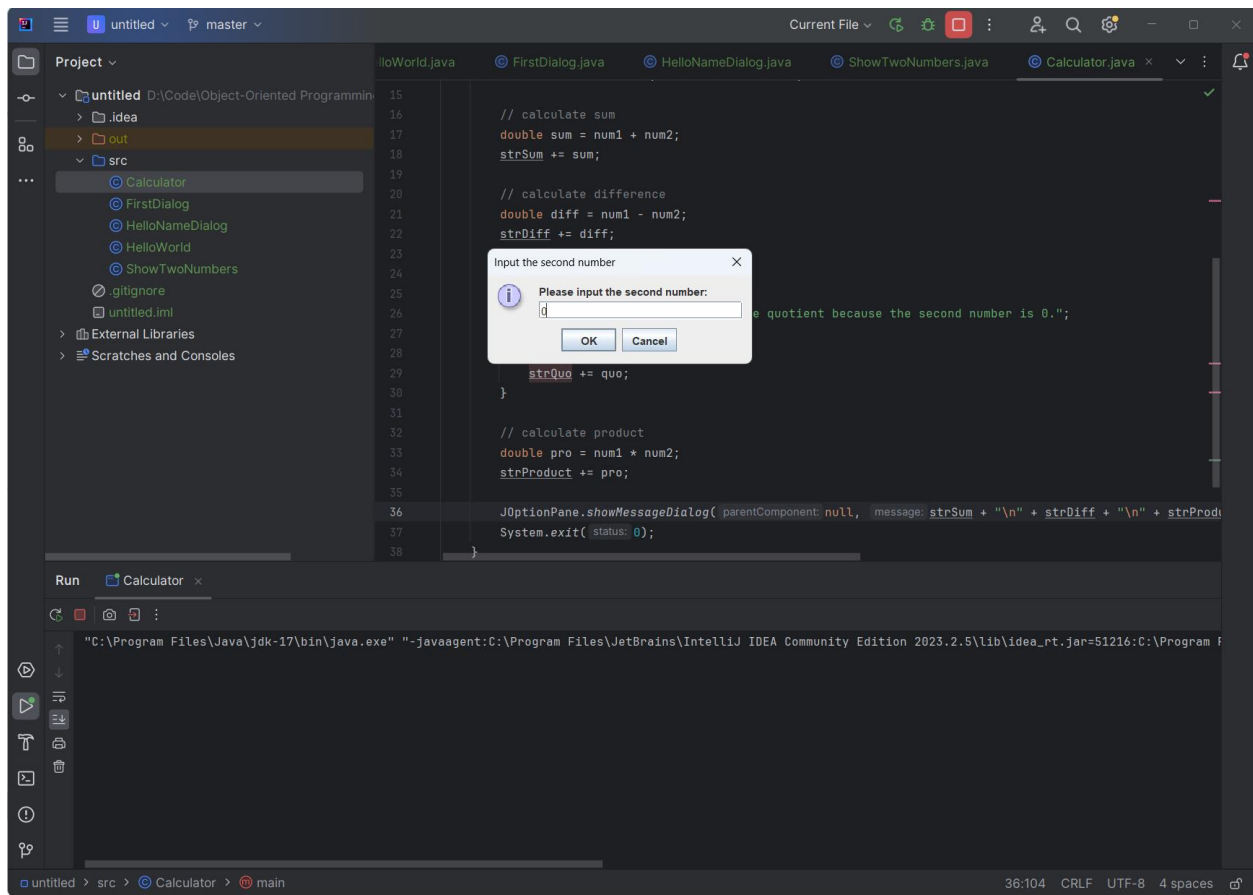


2.2.5 Write a program to calculate sum, difference, product, and quotient of 2 double numbers which are entered by users.

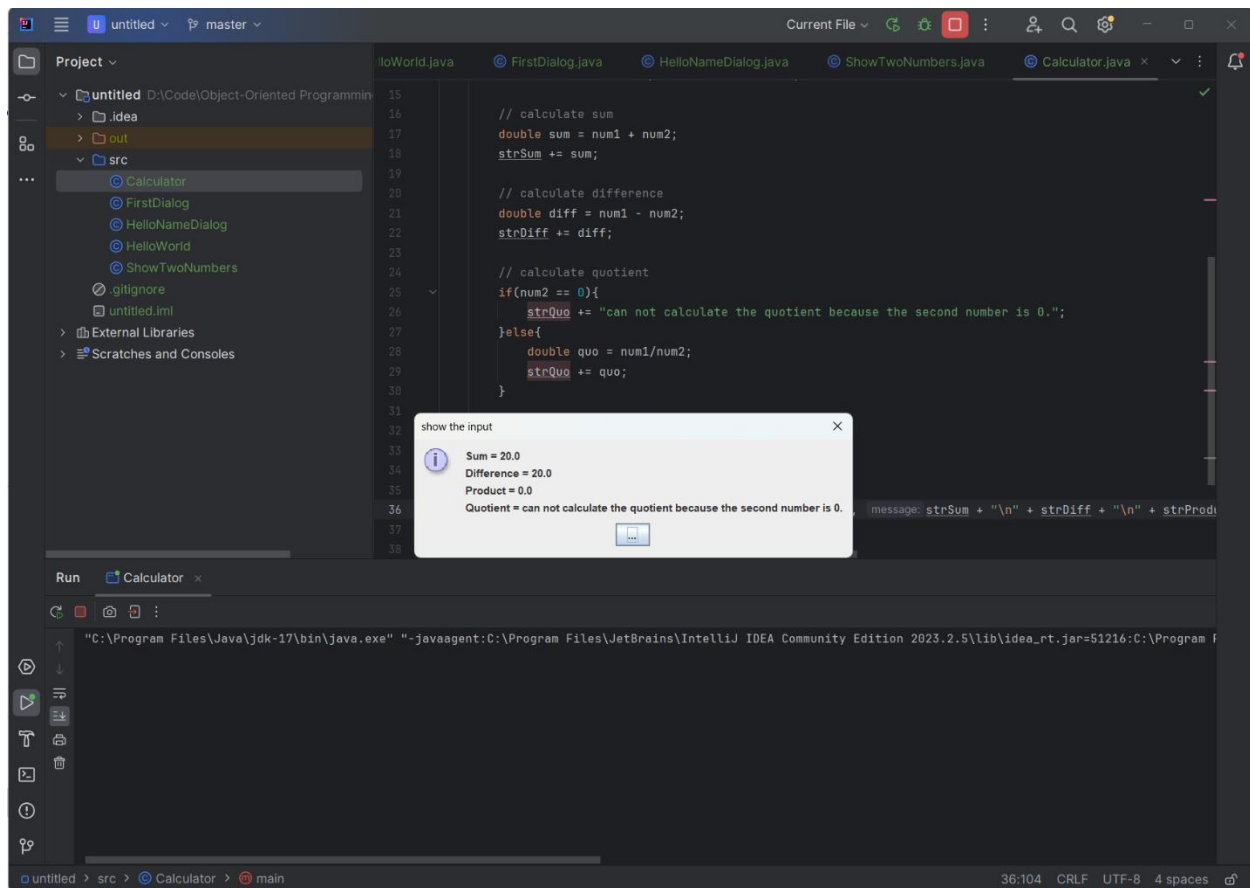
### Notes

- To convert from String to double, you can use  
**double num1 = Double.parseDouble(strNum1)**
- Check the divisor of the division









## 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

```

1  import javax.swing.JOptionPane;
2  new *
3  /> public class SolveEquation {
4      new *
5      public static void main(String[] args){
6          String choice;
7          choice = JOptionPane.showInputDialog( parentComponent: null, message: "Choose the type of equation: \n1.First-degree equation with one variable. (ax + b = 0)\n2.Second-degree equation with one
8
9          int choice_int = Integer.parseInt(choice);
10         if(choice_int == 1){
11             String a = JOptionPane.showInputDialog( parentComponent: null, message: "Input a = ", title: "Input the first coefficient", JOptionPane.INFORMATION_MESSAGE);
12             double a_dou = Double.parseDouble(a);
13
14             String b = JOptionPane.showInputDialog( parentComponent: null, message: "Input b = ", title: "Input the second coefficient", JOptionPane.INFORMATION_MESSAGE);
15             double b_dou = Double.parseDouble(b);
16
17             if(a_dou == 0){
18                 if(b_dou != 0){
19                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has no solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
20                 }else{
21                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has infinite solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
22                 }
23             }else{
24                 double solution = (-1) * b_dou / a_dou;
25                 JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 1 solution: \n x = " + solution, title: "solution", JOptionPane.INFORMATION_MESSAGE);
26             }
27         }else if(choice_int == 2){
28             String a = JOptionPane.showInputDialog( parentComponent: null, message: "Input a = ", title: "Input the first coefficient", JOptionPane.INFORMATION_MESSAGE);
29             double a_dou = Double.parseDouble(a);
30
31             String b = JOptionPane.showInputDialog( parentComponent: null, message: "Input b = ", title: "Input the second coefficient", JOptionPane.INFORMATION_MESSAGE);
32             double b_dou = Double.parseDouble(b);
33
34             String c = JOptionPane.showInputDialog( parentComponent: null, message: "Input c = ", title: "Input the third coefficient", JOptionPane.INFORMATION_MESSAGE);
35             double c_dou = Double.parseDouble(c);
36
37             if(a_dou == 0){
38                 if(b_dou == 0){
39                     if(c_dou == 0){
40                         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has infinite solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
41                     }else{
42                         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has no solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
43                     }
44                 }else{
45                     double result = (-1) * c_dou / b_dou;
46                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 1 solution: \n x = " + result, title: "solution", JOptionPane.INFORMATION_MESSAGE);
47                 }
48             }else{
49                 double delta = b_dou * b_dou - 4 * a_dou * c_dou;
50                 if(delta < 0){
51                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has no solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
52                 }else if(delta == 0){
53                     double result = (-1) * b_dou / (2 * a_dou);
54                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 1 solution: \n x = " + result, title: "solution", JOptionPane.INFORMATION_MESSAGE);
55                 }else{
56                     double x1 = ((-1) * b_dou + Math.sqrt(delta))/(2 * a_dou);
57                     double x2 = ((-1) * b_dou - Math.sqrt(delta))/(2 * a_dou);
58                     JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 2 solution: \n x1 = " + x1 + "\n x2 = " + x2, title: "solution", JOptionPane.INFORMATION_MESSAGE);
59                 }
60             }
61         }else{
62             JOptionPane.showMessageDialog( parentComponent: null, message: "Invalid choice");
63         }
64     }
65 }

```

```

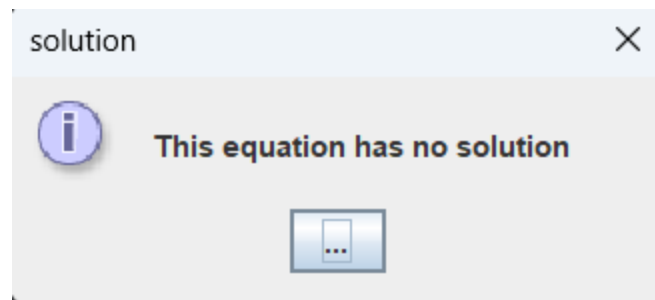
32 String c = JOptionPane.showInputDialog( parentComponent: null, message: "Input c = ", title: "Input the third coefficient", JOptionPane.INFORMATION_MESSAGE);
33 double c_dou = Double.parseDouble(c);
34
35 if(a_dou == 0){
36     if(b_dou == 0){
37         if(c_dou == 0){
38             JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has infinite solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
39         }else{
40             JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has no solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
41         }
42     }else{
43         double result = (-1) * c_dou / b_dou;
44         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 1 solution: \n x = " + result, title: "solution", JOptionPane.INFORMATION_MESSAGE);
45     }
46 }else{
47     double delta = b_dou * b_dou - 4 * a_dou * c_dou;
48     if(delta < 0){
49         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has no solution", title: "solution", JOptionPane.INFORMATION_MESSAGE);
50     }else if(delta == 0){
51         double result = (-1) * b_dou / (2 * a_dou);
52         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 1 solution: \n x = " + result, title: "solution", JOptionPane.INFORMATION_MESSAGE);
53     }else{
54         double x1 = ((-1) * b_dou + Math.sqrt(delta))/(2 * a_dou);
55         double x2 = ((-1) * b_dou - Math.sqrt(delta))/(2 * a_dou);
56         JOptionPane.showMessageDialog( parentComponent: null, message: "This equation has 2 solution: \n x1 = " + x1 + "\n x2 = " + x2, title: "solution", JOptionPane.INFORMATION_MESSAGE);
57     }
58 }
59 }else{
60     JOptionPane.showMessageDialog( parentComponent: null, message: "Invalid choice");
61 }
62 }
63 }
64

```

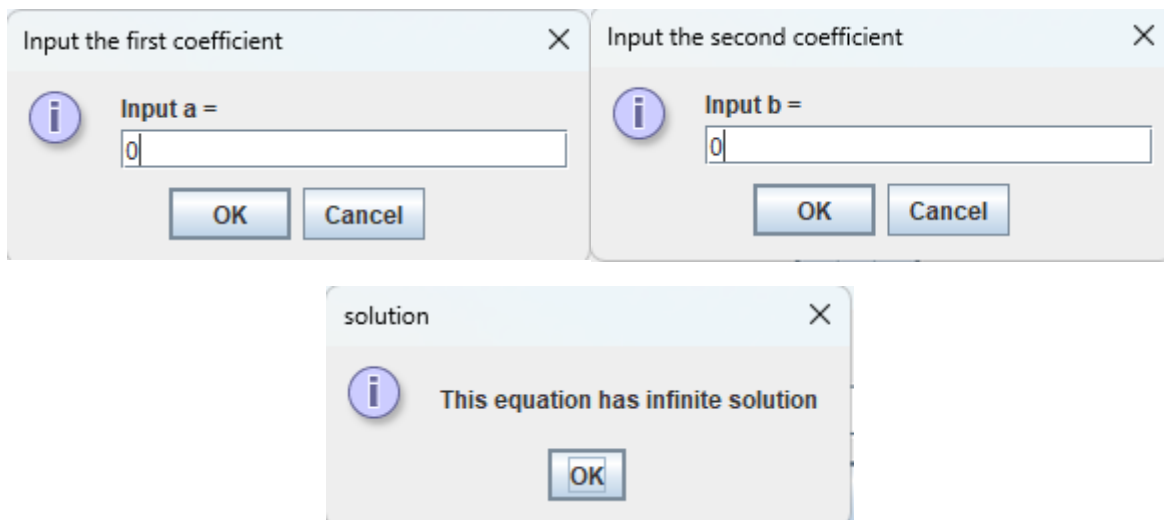
## Result

1. First equation:
  - a. 1 solution

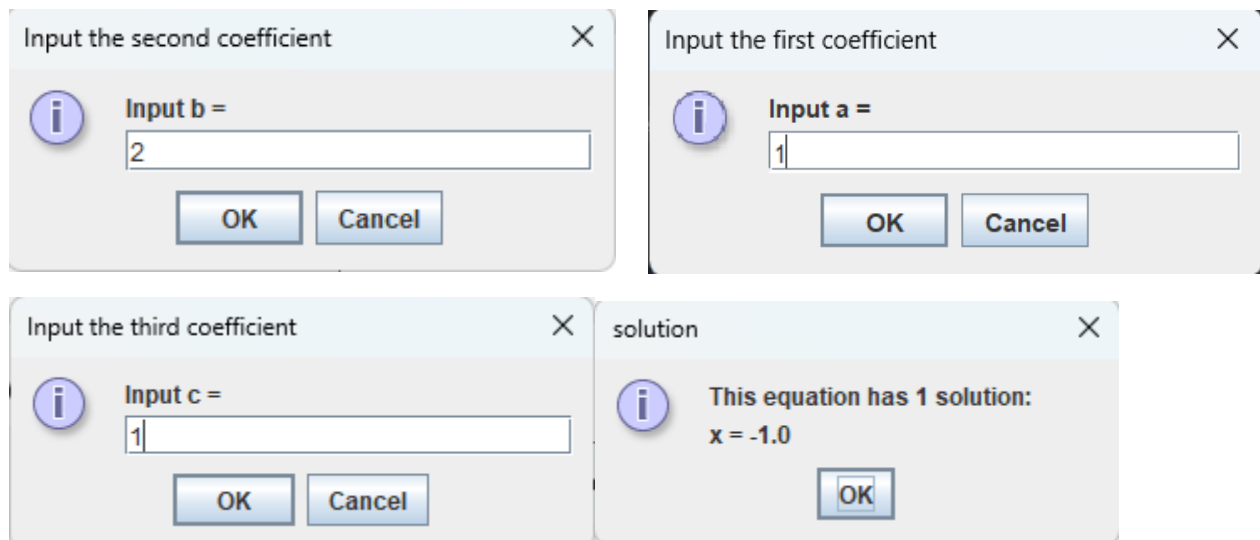
- b. No solution



c. Infinite solution



2. Second degree with one variable





```

tDialog.java  HelloNameDialog.java  ShowTwoNumbers.java  Calculator.java  SolveEquation.java  Triangle.java  DayOfMonth.java
1  import java.util.Scanner;
   new *
2  public class DayOfMonth {
   new *
3  public static void main(String[] args) {
4      Scanner scanner = new Scanner(System.in);
5
6      int month, year;
7      boolean validInput = false;
8
9      // Loop until valid input is provided
10     do {
11         System.out.print("Enter the month (1-12): ");
12         month = scanner.nextInt();
13         System.out.print("Enter the year: ");
14         year = scanner.nextInt();
15
16         // Check if the month and year are valid
17         if (month < 1 || month > 12 || year < 1) {
18             System.out.println("Invalid month or year. Please enter again.");
19             continue;
20         } else {
21             validInput = true;
22         }
23
24         // System.out.print("Enter the year: ");
25         // year = scanner.nextInt();
26     } while (!validInput);
27
28     // Determine the number of days in the entered month
29     int daysInMonth;
30     switch (month) {
31         case 2: // February
32             if (isLeapYear(year)) {
33                 daysInMonth = 29;

```

```

31         case 2: // February
32             if (isLeapYear(year)) {
33                 daysInMonth = 29;
34             } else {
35                 daysInMonth = 28;
36             }
37             break;
38         case 4:
39         case 6:
40         case 9:
41         case 11: // April, June, September, November
42             daysInMonth = 30;
43             break;
44         default: // January, March, May, July, August, October, December
45             daysInMonth = 31;
46             break;
47     }
48
49     System.out.println("Number of days in the entered month: " + daysInMonth);
50
51     scanner.close();
52 }
53
54 // Method to check if a year is a leap year
55 // usage new *
56 public static boolean isLeapYear(int year) {
57     return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
58 }
59

```

```

Run DayOfMonth
"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2021.2.5\lib\idea_rt.jar=51870:C:\Program Files\JetBrains\IntelliJ IDEA Community Ed
Enter the month (1-12): 20
Enter the year: 2000
Invalid month or year. Please enter again.
Enter the month (1-12): 20
Enter the year: -1
Invalid month or year. Please enter again.
Enter the month (1-12): 2
Enter the year: 2004
Number of days in the entered month: 29

Process finished with exit code 0

```

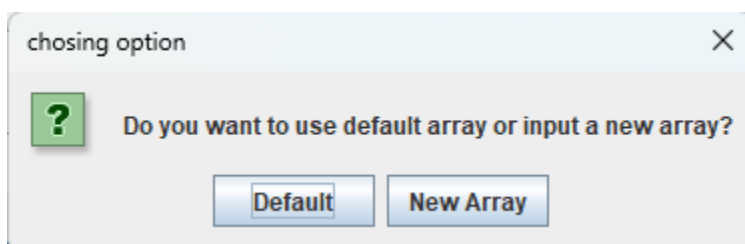
6.5 Write a Java program to sort a numeric array, and calculate the sum and average value of array elements.

```

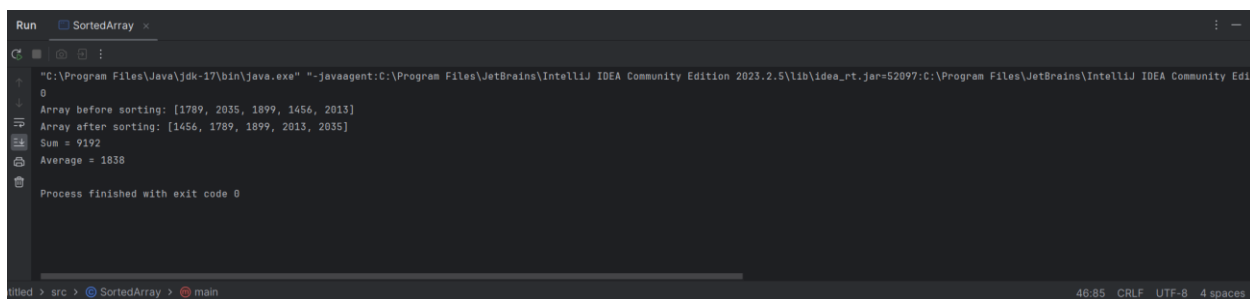
1  import java.util.Scanner;
2  import javax.swing.JOptionPane;
3  import java.util.Arrays;
4  import java.util.jar.Attributes;
5
6  new *
7  public class SortedArray {
8      new *
9      public static void main(String[] args){
10         Object[] options = {"Default", "New Array"};
11         int option = JOptionPane.showOptionDialog( parentComponent: null, message: "Do you want to use default array or input a new array?",
12             title: "choosing option", JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE, icon: null, options, initialValue: null);
13         System.out.println(option);
14         int sum = 0;
15         if(option == 0){
16             int[] defaultArray = {1789, 2035, 1899, 1456, 2013};
17             int n = defaultArray.length;
18             System.out.println("Array before sorting: " + Arrays.toString(defaultArray));
19             Arrays.sort(defaultArray);
20             for(int i = 0; i < n; i++){
21                 sum += defaultArray[i];
22             }
23             System.out.println("Array after sorting: " + Arrays.toString(defaultArray));
24             System.out.println("Sum = " + sum);
25             System.out.println("Average = " + sum/n);
26         }else{
27             boolean isValid = false;
28             int n = 0;
29             do{
30                 String input_n = JOptionPane.showInputDialog( parentComponent: null, message: "Input the size of your array: n = ", title: "Input size", JOp
31             }try{
32                 n = Integer.parseInt(input_n);
33             }catch (Exception e){
34                 JOptionPane.showMessageDialog( parentComponent: null, message: "The size must be an integer!!!");
35             }finally {
36                 if(n > 0){
37                     isValid = true;
38                 }
39             }
40         }
41     }
42 }

```

```
33         }finally {
34             if(n > 0){
35                 isValid = true;
36             }
37         }
38     }while (!isValid);
39
40     int[] inputArray = new int[n];
41     Scanner sc = new Scanner(System.in);
42     for(int i = 0; i < n; i++){
43         inputArray[i] = sc.nextInt();
44         sum += inputArray[i];
45     }
46     System.out.println("Array before sorting: " + Arrays.toString(inputArray));
47     Arrays.sort(inputArray);
48     System.out.println("Array after sorting: " + Arrays.toString(inputArray));
49     System.out.println("Sum = " + sum);
50     System.out.println("Average = " + sum/n);
51 }
52 }
53 }
54 }
```

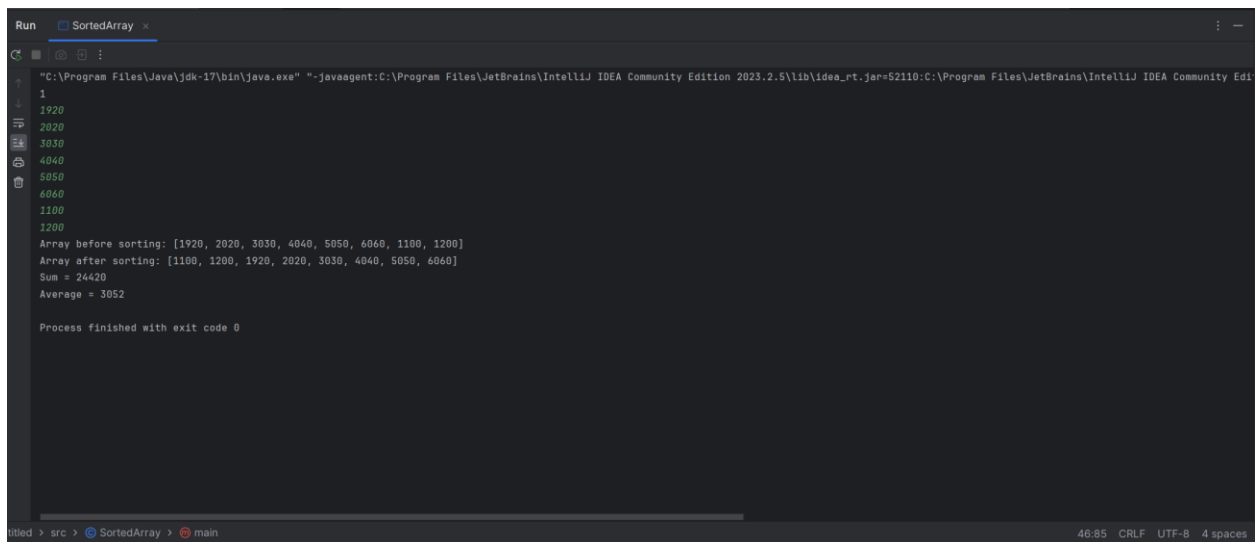


## 1. Default Array



## 2. Input Array





```
Run SortedArray x
1
1920
2020
3030
4040
5050
6060
1100
1200
Array before sorting: [1920, 2020, 3030, 4040, 5050, 6060, 1100, 1200]
Array after sorting: [1100, 1200, 1920, 2020, 3030, 4040, 5050, 6060]
Sum = 24420
Average = 3052
Process finished with exit code 0
```

src > SortedArray > main 46:85 CRLF UTF-8 4 spaces