

Lab 01: Environment Setup and Java Basics

Full Name: Trịnh Diễm Quỳnh

Student ID: 20210737

1. Getting Started

```
[(base) trindhdiemquynh@Quinn-2 ~ % javac -version
javac 20
(base) trindhdiemquynh@Quinn-2 ~ % █
```

2. First Programs

2.1. Java Programming Steps

Step 1: Write the source code such as the code shown in Figure 3. and save in, e.g., “HelloWorld.java” file.

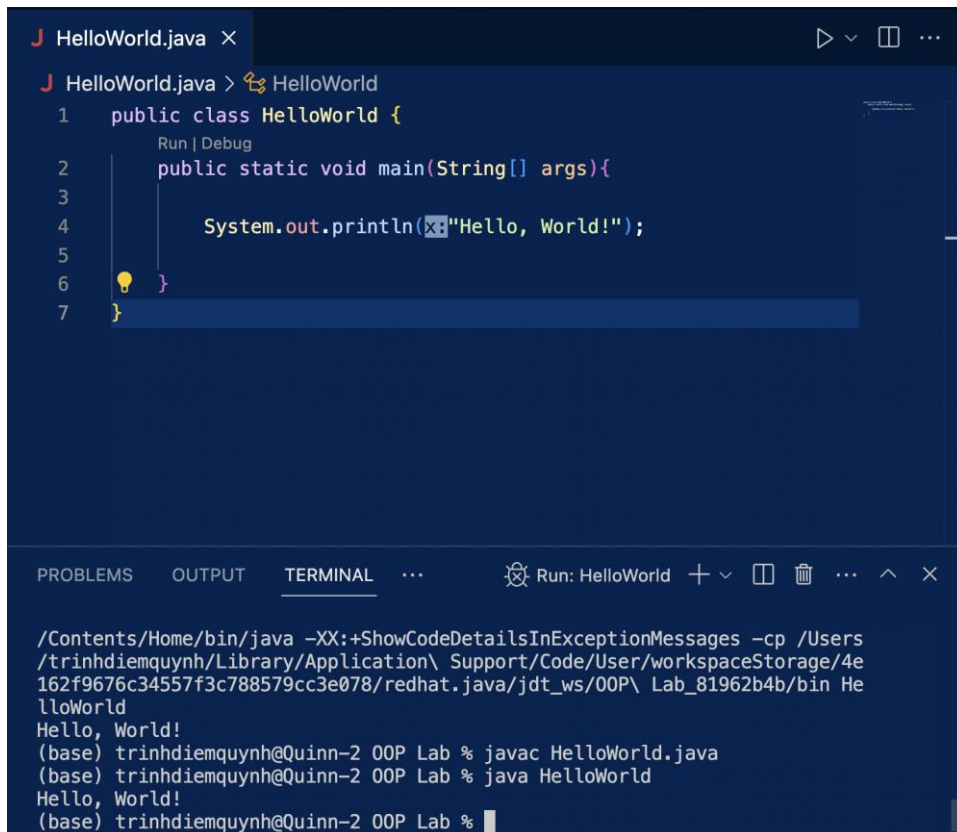
A screenshot of an IDE window titled 'HelloWorld.java'. The code editor shows the following Java code:

```
1 public class HelloWorld {
2     public static void main(String[] args){
3
4         System.out.println(x:"Hello, World!");
5     }
6 }
7
```

The code is highlighted with syntax coloring. A lightbulb icon is visible in the left margin next to line 6. The IDE interface includes a toolbar with 'Run' and 'Debug' buttons.

Step 2: Compile the source code into Java portable bytecode (or machine code) using the JDK's Java compiler.

Step 3: Run the compiled bytecode using the JDK's Java Runtime.



The screenshot shows an IDE window with a tab for 'HelloWorld.java'. The code editor displays the following Java code:

```
1 public class HelloWorld {  
2     public static void main(String[] args){  
3  
4         System.out.println("Hello, World!");  
5  
6     }  
7 }
```

Below the code editor is a terminal window titled 'Run: HelloWorld'. It shows the execution of the program, including the compilation and running commands, and the output 'Hello, World!'.

```
/Contents/Home/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/  
trinhdiemquynh/Library/Application\ Support/Code/User/workspaceStorage/4e  
162f9676c34557f3c788579cc3e078/redhat.java/jdt_ws/00P\ Lab_81962b4b/bin He  
lloWorld  
Hello, World!  
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac HelloWorld.java  
(base) trinhdiemquynh@Quinn-2 OOP Lab % java HelloWorld  
Hello, World!  
(base) trinhdiemquynh@Quinn-2 OOP Lab %
```

2.2. The Very First Java Programs

2.2.1. Write, compile the first Java application:

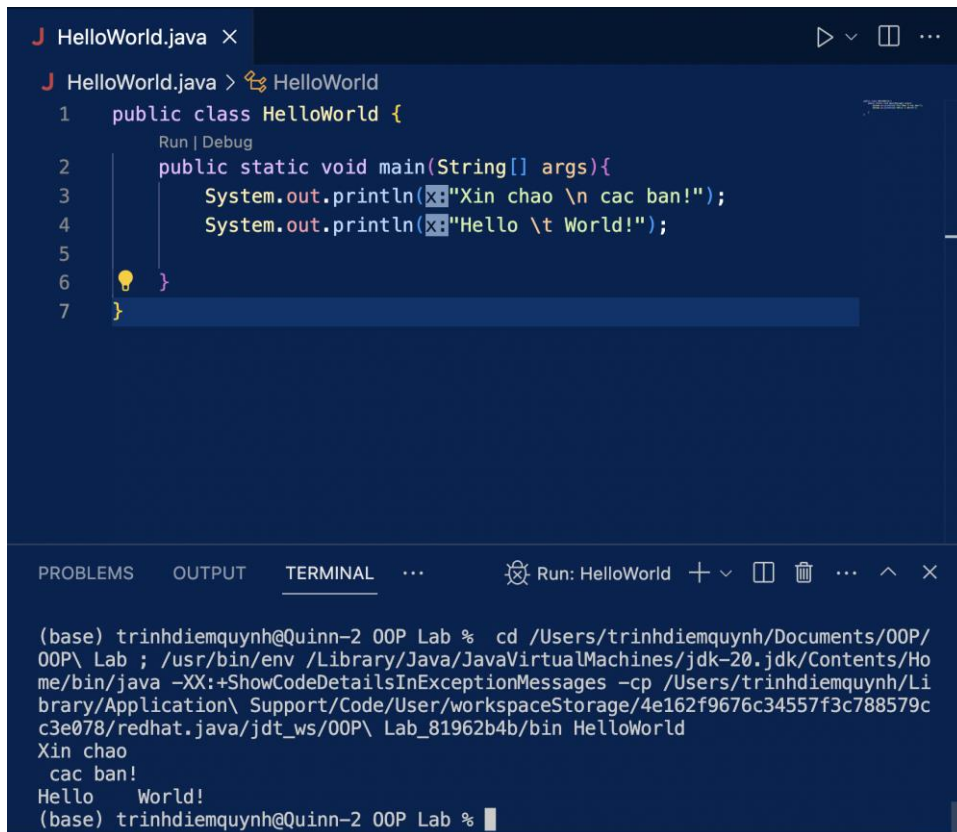
Step 1: Create a new file. From the Notepad interface, choose File → New File.

I use Macbook M1 so I cannot download Notepad because it is not available.

Step 2: Save the file. From the Notepad interface, choose File → Save. Browse the desired directory, change the file name to “ HelloWorld.java” and hit the “Save” button.

Step 3: Write the source code.

Step 4: Compile. On a Command Prompt or a Terminal, change the current working directory into the directory where we have saved the source code. Then issue the following commands.



```

J HelloWorld.java x
J HelloWorld.java > HelloWorld
1 public class HelloWorld {
2     public static void main(String[] args){
3         System.out.println("Xin chao \n cac ban!");
4         System.out.println("Hello \t World!");
5     }
6 }
7 }

Run | Debug

PROBLEMS OUTPUT TERMINAL ... Run: HelloWorld
(base) trindhdiemquynh@Quinn-2 OOP Lab % cd /Users/trindhdiemquynh/Documents/OOP/OOP\ Lab ; /usr/bin/env /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/Home/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/trindhdiemquynh/Library/Application\ Support/Code/User/workspaceStorage/4e162f9676c34557f3c788579c3e078/redhat.java/jdt_ws/OOP\ Lab_81962b4b/bin HelloWorld
Xin chao
cac ban!
Hello    World!
(base) trindhdiemquynh@Quinn-2 OOP Lab %
```

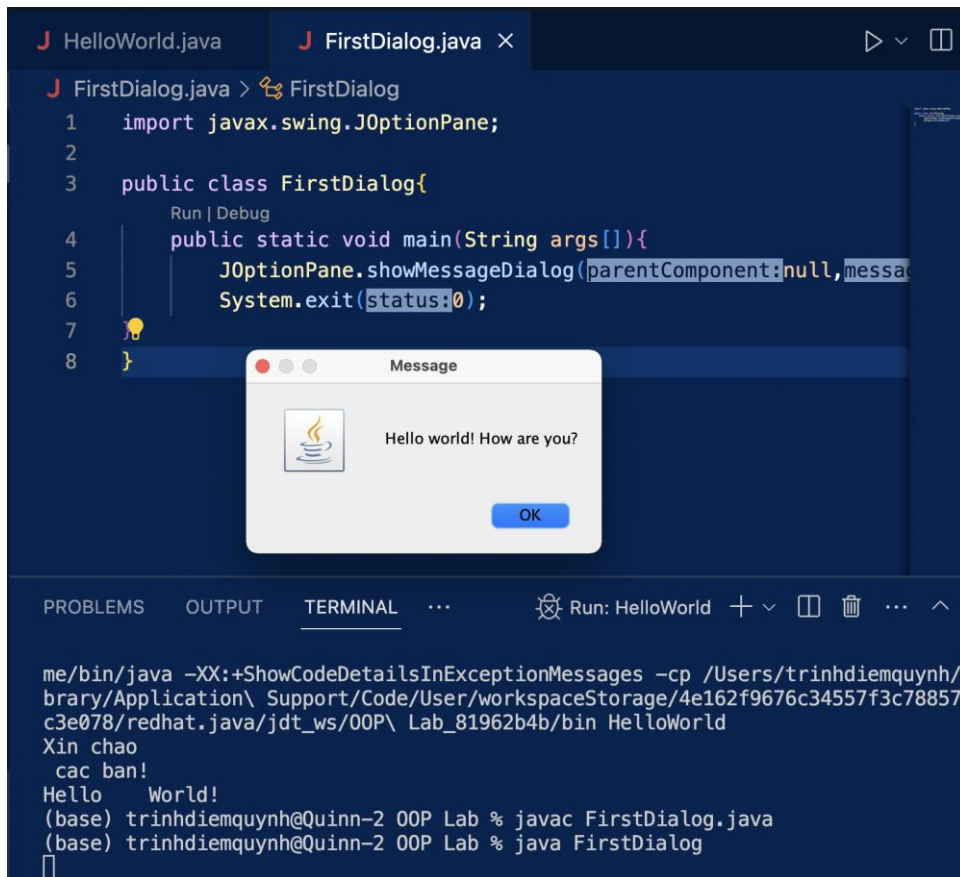
2.2.2. Write, compile the first dialog Java program

Step 1: Create a new file. From the Notepad interface, choose File → New File.

Step 2: Save the file. From the Notepad interface, choose File → Save. Browse the desired directory, change the file name to “ FirstDialog.java,” and click the “Save” button.

Step 3: Write the source code.

Step 4: Compile. On a Command Prompt or a Terminal, change the current working directory into the directory where we have saved the source code.



```
J HelloWorld.java J FirstDialog.java x
J FirstDialog.java > FirstDialog
1 import javax.swing.JOptionPane;
2
3 public class FirstDialog{
4     public static void main(String args[]){
5         JOptionPane.showMessageDialog(parentComponent:null,message:0);
6         System.exit(status:0);
7     }
8 }

Message
Hello world! How are you?
OK

PROBLEMS OUTPUT TERMINAL ... Run: HelloWorld + - - -
me/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/trinhdiemquynh/Library/Application\ Support/Code/User/workspaceStorage/4e162f9676c34557f3c788579c3e078/redhat.java/jdt_ws/00P\ Lab_81962b4b/bin HelloWorld
Xin chao
cac ban!
Hello World!
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac FirstDialog.java
(base) trinhdiemquynh@Quinn-2 OOP Lab % java FirstDialog
```

2.2.3. Write, compile the first input dialog Java application

Step 1: Create a new file. From the Notepad interface, choose File → New File.

Step 2: Save the file. From the Notepad interface, choose File → Save. Browse the desired directory, change the file name to “HelloNameDialog.java,” and click the “Save” button.

Step 3: Write the source code.

Step 4: Compile. On a Command Prompt or a Terminal, change the current working directory into the directory where we have saved the source code.

```
1 import javax.swing.JOptionPane;
2
3 public class HelloNameDialog {
4     public static void main(String[] args){
5         String result;
6         result = JOptionPane.showInputDialog(message:"Please enter your name");
7         JOptionPane.showMessageDialog(parentComponent:null, "Hi " + result);
8         System.exit(status:0);
9     }
10 }
```

Input

Please enter your name

Trịnh Diễm Quỳnh

Cancel OK

```
c3e078/redhat.java/jdt_ws/00P\ Lab_81962b4b/bin HelloWorld
Xin chao
cac ban!
Hello World!
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac FirstDialog.java
(base) trinhdiemquynh@Quinn-2 OOP Lab % java FirstDialog
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac HelloNameDialog.java
(base) trinhdiemquynh@Quinn-2 OOP Lab % java HelloNameDialog
```

```
1 import javax.swing.JOptionPane;
2
3 public class HelloNameDialog {
4     public static void main(String[] args){
5         String result;
6         result = JOptionPane.showInputDialog(message:"Please enter your name");
7         JOptionPane.showMessageDialog(parentComponent:null, "Hi " + result);
8         System.exit(status:0);
9     }
10 }
```

Message

Hi Trịnh Diễm Quỳnh !

OK

```
c3e078/redhat.java/jdt_ws/00P\ Lab_81962b4b/bin HelloWorld
Xin chao
cac ban!
Hello World!
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac FirstDialog.java
(base) trinhdiemquynh@Quinn-2 OOP Lab % java FirstDialog
(base) trinhdiemquynh@Quinn-2 OOP Lab % javac HelloNameDialog.class
(base) trinhdiemquynh@Quinn-2 OOP Lab % java HelloNameDialog
```

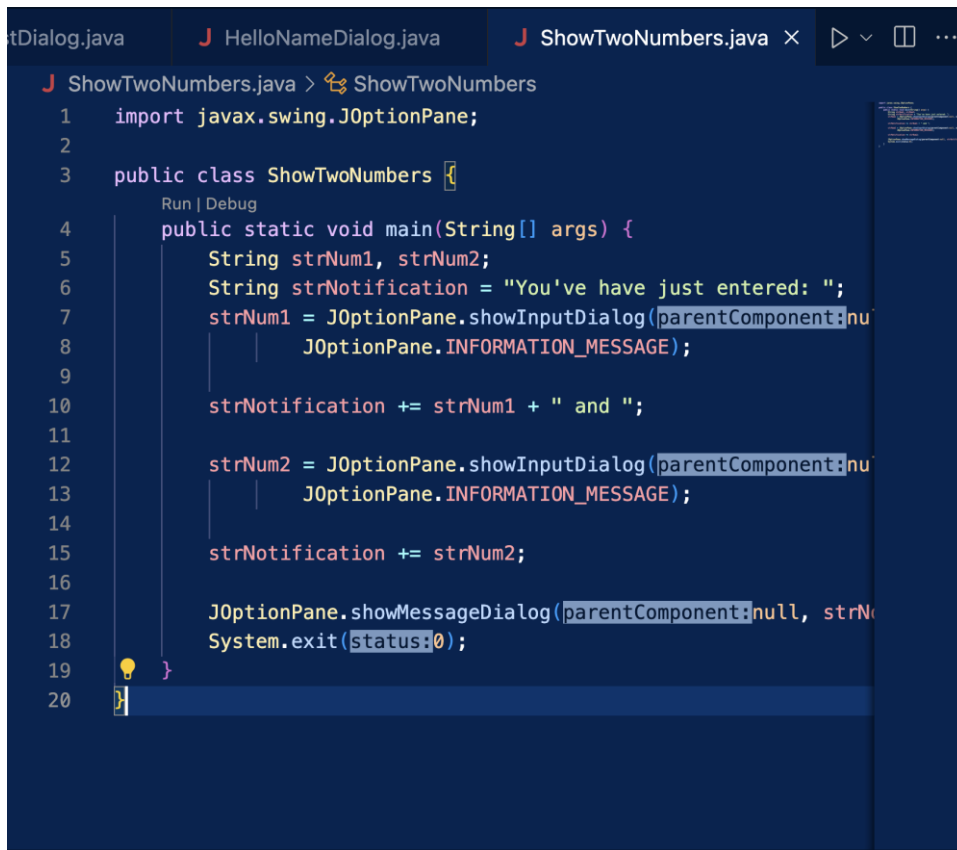
Ln 10, Col 2 Spaces: 4 UTF-8 LF {} Java Prettier

2.2.4. Write, compile, and run the following example:

Step 1: Create a new file. From the Notepad interface, choose File → New File.

Step 2: Save the file. From the Notepad interface, choose File → Save. Browse the desired directory, change the file name to “ShowTwoNumbers.java,” and click the “Save” button.

Step 3: Write the source code.



```
1  import javax.swing.JOptionPane;
2
3  public class ShowTwoNumbers {
4      public static void main(String[] args) {
5          String strNum1, strNum2;
6          String strNotification = "You've have just entered: ";
7          strNum1 = JOptionPane.showInputDialog(parentComponent:null,
8          JOptionPane.INFORMATION_MESSAGE);
9
10         strNotification += strNum1 + " and ";
11
12         strNum2 = JOptionPane.showInputDialog(parentComponent:null,
13         JOptionPane.INFORMATION_MESSAGE);
14
15         strNotification += strNum2;
16
17         JOptionPane.showMessageDialog(parentComponent:null, strN
18         System.exit(status:0);
19     }
20 }
```

Step 4: Compile. On a Command Prompt or a Terminal, change the current working directory into the directory where we have saved the source code.


```
stDialog.java | HelloNameDialog.java | ShowTwoNumbers.java x | Run | Debug | ShowTwoNumbers
1 import javax.swing.JOptionPane;
2
3 public class ShowTwoNumbers {
4     public static void main(String[] args) {
5         String strNum1, strNum2;
6         String strNotification = "You've have just entered: ";
7         strNum1 = JOptionPane.showInputDialog(parentComponent:nu
            INFORMATION_MESSAGE);
            m1 + " and ";
            owInputDialog(parentComponent:nu
            INFORMATION_MESSAGE);
14
15         strNotification += strNum2;
```

Input the first number

Please input the first number:

34

Cancel OK

PROBLEMS | TERMINAL | ... | Run: ShowTwoNumbers + v | ...

```
irtualMachines/jdk-20.jdk/Contents/Home/bin/java -XX:+ShowCodeDetailsInEx
ceptionMessages -cp /Users/trinhdiemquynh/Library/Application\ Support/Co
de/User/workspaceStorage/4e162f9676c34557f3c788579cc3e078/redhat.java/jdt
_ws/00P\ Lab_81962b4b/bin ShowTwoNumbers
(base) trinhdiemquynh@Quinn-2 00P Lab % javac ShowTwoNumbers.java
(base) trinhdiemquynh@Quinn-2 00P Lab % java ShowTwoNumbers
```

```
stDialog.java | HelloNameDialog.java | ShowTwoNumbers.java x | Run | Debug | ShowTwoNumbers
1 import javax.swing.JOptionPane;
2
3 public class ShowTwoNumbers {
4     public static void main(String[] args) {
5         String strNum1, strNum2;
6         String strNotification = "You've have just entered: ";
7         strNum1 = JOptionPane.showInputDialog(parentComponent:nu
            INFORMATION_MESSAGE);
            m1 + " and ";
            owInputDialog(parentComponent:nu
            INFORMATION_MESSAGE);
14
15         strNotification += strNum2;
```

Input the second number

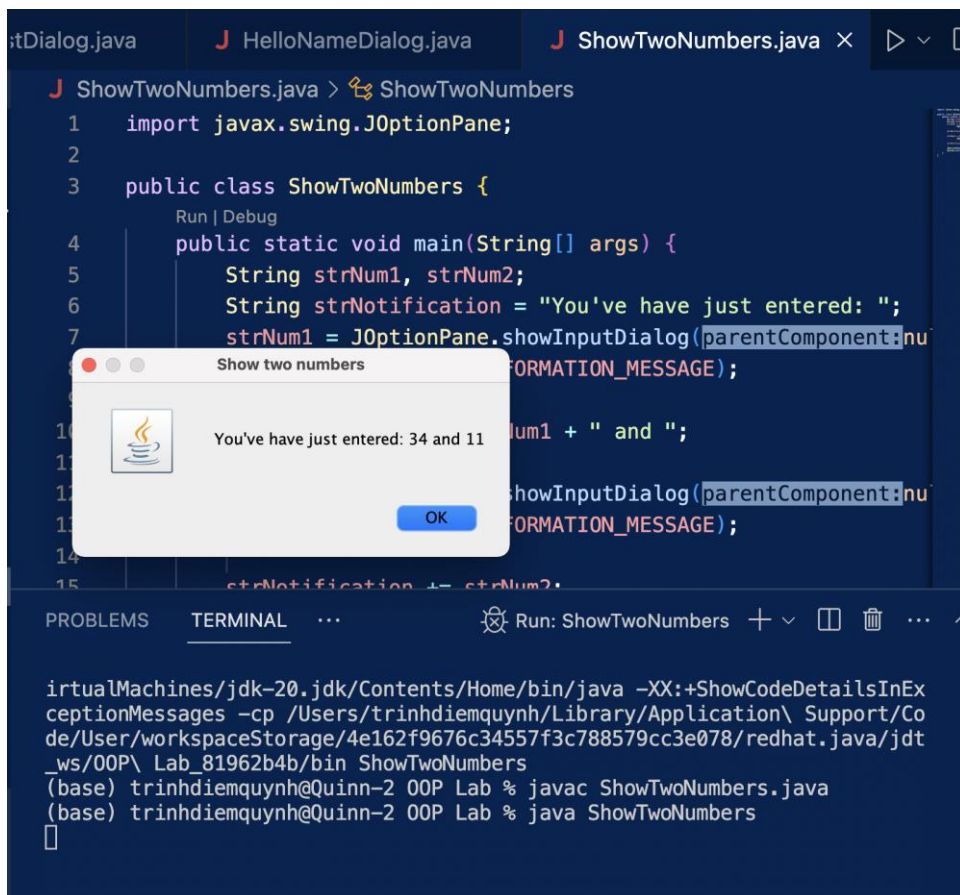
Please input the sencond number

11

Cancel OK

PROBLEMS | TERMINAL | ... | Run: ShowTwoNumbers + v | ...

```
irtualMachines/jdk-20.jdk/Contents/Home/bin/java -XX:+ShowCodeDetailsInEx
ceptionMessages -cp /Users/trinhdiemquynh/Library/Application\ Support/Co
de/User/workspaceStorage/4e162f9676c34557f3c788579cc3e078/redhat.java/jdt
_ws/00P\ Lab_81962b4b/bin ShowTwoNumbers
(base) trinhdiemquynh@Quinn-2 00P Lab % javac ShowTwoNumbers.java
(base) trinhdiemquynh@Quinn-2 00P Lab % java ShowTwoNumbers
```



The screenshot shows an IDE with the file `ShowTwoNumbers.java` open. The code defines a `ShowTwoNumbers` class with a `main` method that uses `JOptionPane` to show a dialog box. The dialog box, titled "Show two numbers", displays the message "You've have just entered: 34 and 11" and has an "OK" button. The terminal at the bottom shows the compilation and execution of the program.

```
1 import javax.swing.JOptionPane;
2
3 public class ShowTwoNumbers {
4     public static void main(String[] args) {
5         String strNum1, strNum2;
6         String strNotification = "You've have just entered: ";
7         strNum1 = JOptionPane.showInputDialog(parentComponent:nu
8         strNum2 = JOptionPane.showInputDialog(parentComponent:nu
9         strNotification = strNum1 + " and " + strNum2;
10        JOptionPane.showMessageDialog(parentComponent:nu, strNotification, "Show two numbers",
11        JOptionPane.INFORMATION_MESSAGE);
12    }
13 }
14
15
```

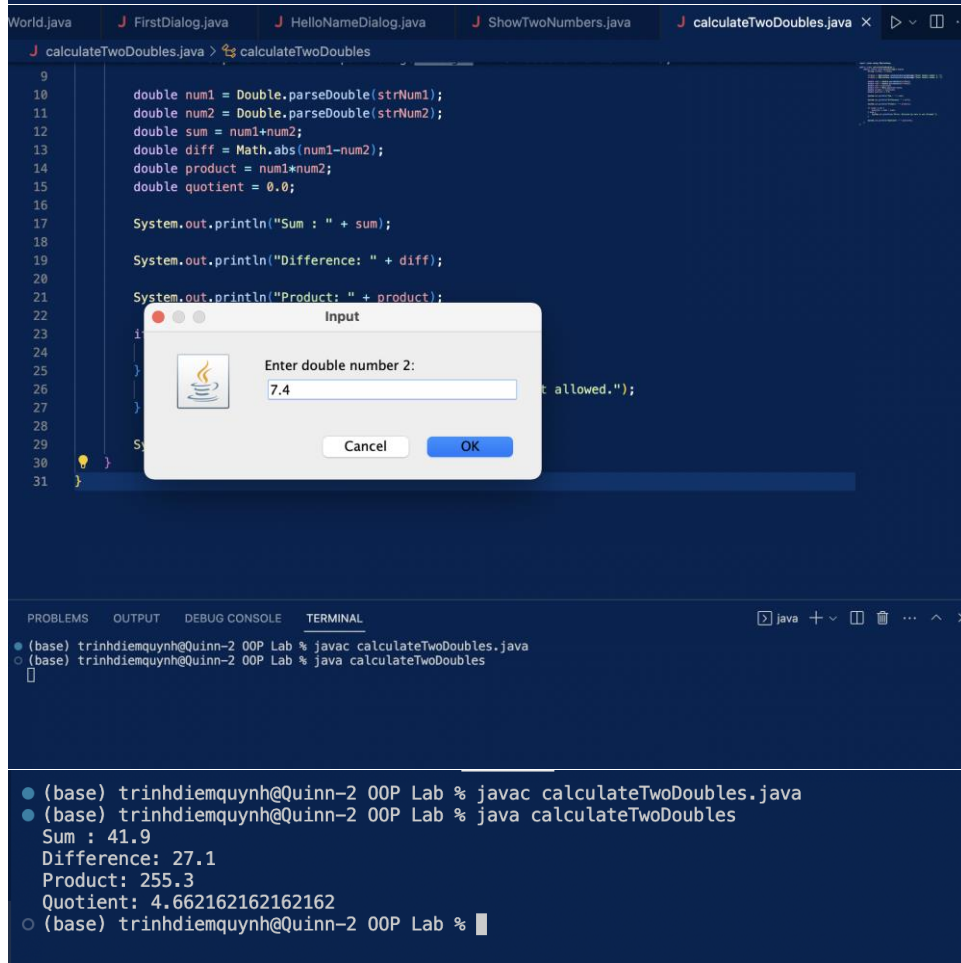
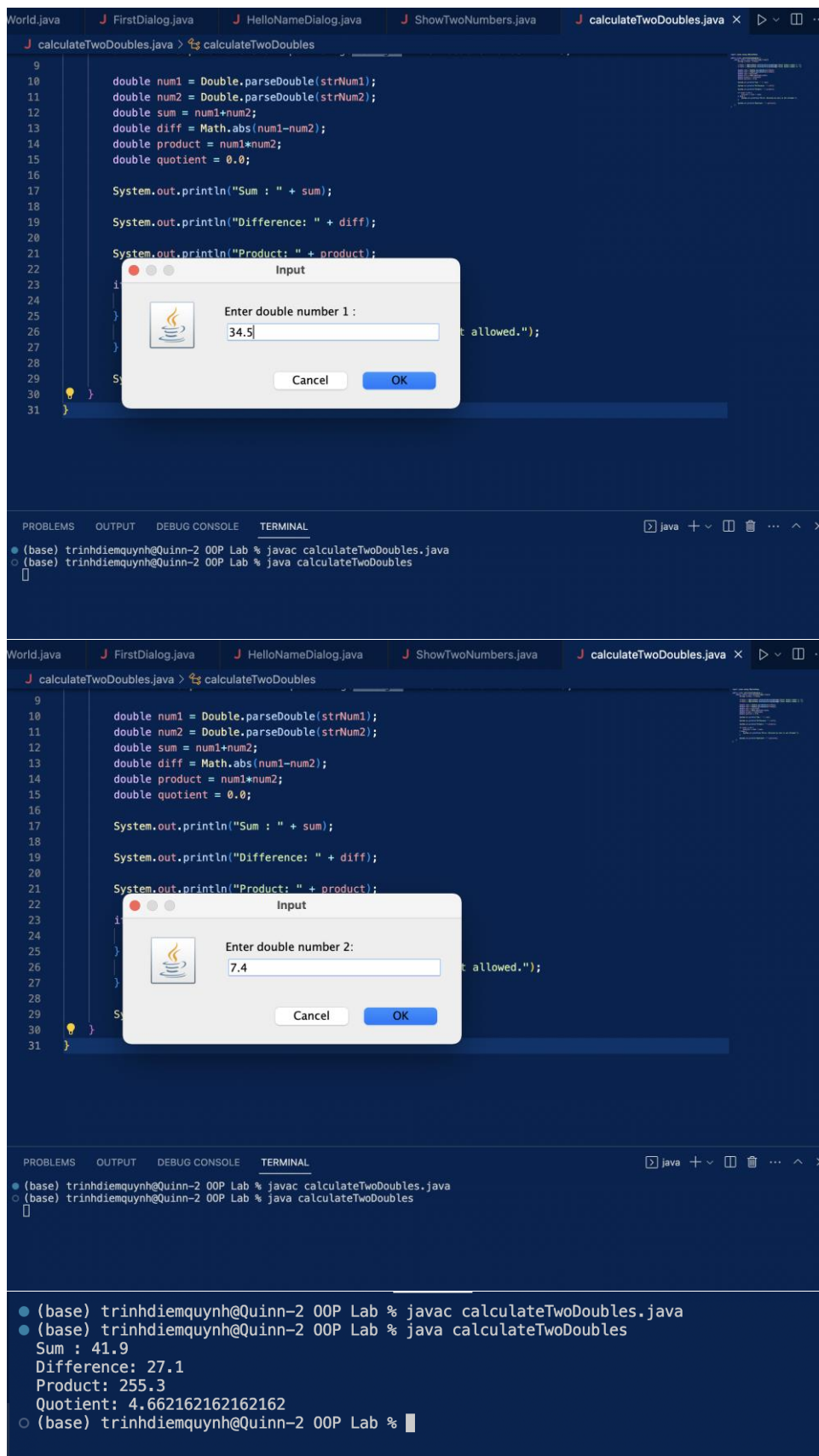
```
Run | Debug
Run: ShowTwoNumbers
virtualMachines/jdk-20.jdk/Contents/Home/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/trinhdiemquynh/Library/Application\ Support/Code/User/workspaceStorage/4e162f9676c34557f3c788579cc3e078/redhat.java/jdt_ws/00P\ Lab_81962b4b/bin ShowTwoNumbers
(base) trinhdiemquynh@Quinn-2 00P Lab % javac ShowTwoNumbers.java
(base) trinhdiemquynh@Quinn-2 00P Lab % java ShowTwoNumbers
```

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



The screenshot shows an IDE with the file `calculateTwoDoubles.java` open. The code defines a `calculateTwoDoubles` class with a `main` method that prompts the user for two double numbers, calculates their sum, difference, product, and quotient, and prints the results. It includes a check for division by zero.

```
1 import javax.swing.JOptionPane;
2
3 public class calculateTwoDoubles {
4     public static void main(String[] args){
5         String strNum1, strNum2;
6
7         strNum1 = JOptionPane.showInputDialog(message:"Enter double number 1 :");
8         strNum2 = JOptionPane.showInputDialog(message:"Enter double number 2: ");
9
10        double num1 = Double.parseDouble(strNum1);
11        double num2 = Double.parseDouble(strNum2);
12        double sum = num1+num2;
13        double diff = Math.abs(num1-num2);
14        double product = num1*num2;
15        double quotient = 0.0;
16
17        System.out.println("Sum : " + sum);
18
19        System.out.println("Difference: " + diff);
20
21        System.out.println("Product: " + product);
22
23        if (num2 != 0) {
24            quotient = num1 / num2;
25        } else {
26            System.out.println("Error: Division by zero is not allowed.");
27        }
28
29        System.out.println("Quotient: " + quotient);
30    }
31 }
```

2.2.6. Write a program to solve:

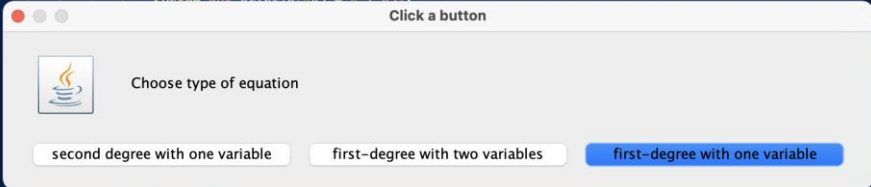
For simplicity, we only consider the real roots of the equations in this task.

- 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

```
J: solveEquation.java > %solveEquation
1 import javax.swing.JOptionPane;
2
3 import java.util.Scanner;
4
5 public class solveEquation {
6     // Run | Debug
7     public static void main(String[] args) {
8         Scanner keyboard = new Scanner(System.in);
9         String[] options = { "first-degree with one variable", "first-degree with two variables",
10             "second degree with one variable" };
11         int x = JOptionPane.showOptionDialog(parentComponent=null, message:"Choose type of equation",
12             title:"Click a button",
13             JOptionPane.DEFAULT_OPTION, JOptionPane.QUESTION_MESSAGE, icon=null, options, options[0]);
14         // first-degree equation with one variable
15         if (x == 0) {
16             System.out.println("Enter a: ");
17             double a = keyboard.nextDouble();
18             System.out.println("Enter b: ");
19             double b = keyboard.nextDouble();
20             if (a == 0) {
21                 if (b == 0)
22                     System.out.println("Infinite solution.");
23                 else
24                     System.out.println("No solution.");
25             } else {
26                 double root = (double) -b / a;
27                 System.out.println("Solution is " + root);
28             }
29         }
30         // first-degree equation with two variables
31         else if (x == 1) {
32             System.out.println("Enter a1: ");
33             double a1 = keyboard.nextDouble();
34             System.out.println("Enter a2: ");
35             double a2 = keyboard.nextDouble();
36             System.out.println("Enter b1: ");
37             double b1 = keyboard.nextDouble();
38             System.out.println("Enter a2: ");
39             double a21 = keyboard.nextDouble();
40             System.out.println("Enter a22: ");
41             double a22 = keyboard.nextDouble();
42             System.out.println("Enter b2: ");
43             double b2 = keyboard.nextDouble();
44             double D = a1 * a22 - a21 * a12;
45             double D1 = b1 * a22 - a12 * b2;
46             double D2 = a11 * b2 - a21 * b1;
47             // Check if the determinant is zero
48             if (D == 0) {
49                 // Check if the system of equations has infinite solutions
50                 if ((D1 == 0 && D2 == 0)) {
51                     System.out.println("Infinite solutions.");
52                 } else {
53                     System.out.println("No solution.");
54                 }
55             } else {
56                 // Calculate the values of x and y
57                 double x1 = (double) D1 / D;
58                 double x2 = (double) D2 / D;
59                 System.out.println("Solution:");
60                 System.out.println("x1 = " + x1);
61                 System.out.println("x2 = " + x2);
62             }
63         }
64         // second-degree equation with one variable
65         else if (x == 2) {
66             System.out.println("Enter a: ");
67             double a = keyboard.nextDouble();
68             System.out.println("Enter b: ");
69             double b = keyboard.nextDouble();
70             System.out.println("Enter c: ");
71             double c = keyboard.nextDouble();
72             if (a == 0) {
73                 if (b == 0) {
74                     if (c == 0)
75                         System.out.println("Infinite solution.");
76                     else
77                         System.out.println("No solution.");
78                 } else {
79                     double root = (double) -c / b;
80                     System.out.println("Solution is " + root);
81                 }
82             } else {
83                 double delta = b * b - 4 * a * c;
84                 if (delta > 0) {
85                     // Two real and distinct roots
86                     double root1 = (-b + Math.sqrt(delta)) / (2 * a);
87                     double root2 = (-b - Math.sqrt(delta)) / (2 * a);
88                     System.out.println("Two real and distinct roots:");
89                     System.out.println("Root 1 = " + root1);
90                     System.out.println("Root 2 = " + root2);
91                 } else if (delta == 0) {
92                     // One real and repeated root
93                     double root = -b / (2 * a);
94                     System.out.println("One real and repeated root:");
95                     System.out.println("Root = " + root);
96                 } else {
97                     System.out.println("No real solution");
98                 }
99             }
100         }
101     }
102 }
103 }
```

```
J: solveEquation.java > %solveEquation
49 // Check if the system of equations has infinite solutions
50 if ((D1 == 0 && D2 == 0)) {
51     System.out.println("Infinite solutions.");
52 } else {
53     System.out.println("No solution.");
54 }
55 } else {
56     // Calculate the values of x and y
57     double x1 = (double) D1 / D;
58     double x2 = (double) D2 / D;
59     System.out.println("Solution:");
60     System.out.println("x1 = " + x1);
61     System.out.println("x2 = " + x2);
62 }
63 }
64 // second-degree equation with one variable
65 else if (x == 2) {
66     System.out.println("Enter a: ");
67     double a = keyboard.nextDouble();
68     System.out.println("Enter b: ");
69     double b = keyboard.nextDouble();
70     System.out.println("Enter c: ");
71     double c = keyboard.nextDouble();
72     if (a == 0) {
73         if (b == 0) {
74             if (c == 0)
75                 System.out.println("Infinite solution.");
76             else
77                 System.out.println("No solution.");
78         } else {
79             double root = (double) -c / b;
80             System.out.println("Solution is " + root);
81         }
82     } else {
83         double delta = b * b - 4 * a * c;
84         if (delta > 0) {
85             // Two real and distinct roots
86             double root1 = (-b + Math.sqrt(delta)) / (2 * a);
87             double root2 = (-b - Math.sqrt(delta)) / (2 * a);
88             System.out.println("Two real and distinct roots:");
89             System.out.println("Root 1 = " + root1);
90             System.out.println("Root 2 = " + root2);
91         } else if (delta == 0) {
92             // One real and repeated root
93             double root = -b / (2 * a);
94             System.out.println("One real and repeated root:");
95             System.out.println("Root = " + root);
96         } else {
97             System.out.println("No real solution");
98         }
99     }
100 }
101 }
102 }
103 }
```

```
50 // Check if the system of equations has infinite solutions
51 if ((D1 == 0) && (D2 == 0)) {
52     System.out.println("Infinite solutions.");
53 } else {
54     System.out.println("No solution.");
55 }
56 } else {
57     // Calculate the values of x and y
58     double x1 = (double) D1 / D;
59     double x2 = (double) D2 / D;
60
61     System.out.println("Solution:");
62     System.out.println("x1 = " + x1);
63     System.out.println("x2 = " + x2);
64 }
65
66 if (a == 0) {
67     if (b == 0) {
68         if (c == 0) {
69             System.out.println("Infinite solution.");
70         } else {
71             System.out.println("No solution.");
72         }
73     } else {
74         System.out.println("Solution:");
75         System.out.println("x = " + (-c / b));
76     }
77 }
```



```
(base) trindhdiemquynh@Quinn-2 OOP Lab % java solveEquation
Enter a:
2
Enter b:
8
Solution is -4.0
(base) trindhdiemquynh@Quinn-2 OOP Lab %
```

3. Introduction to Eclipse / Netbean

4. Javadocs help

5. Your first Java project

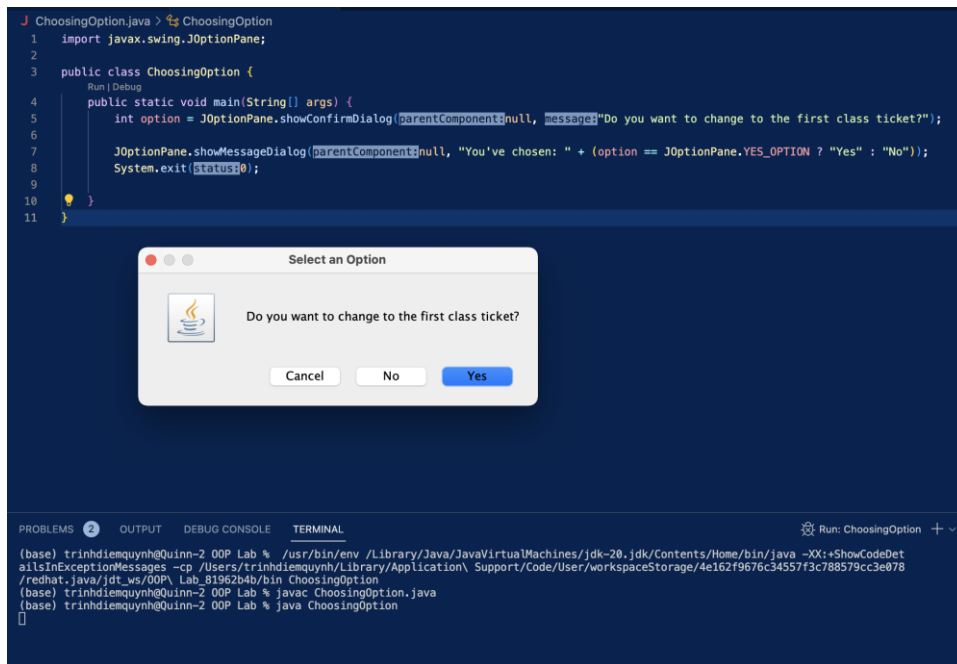
6. Exercises

6.1. Write, compile and run the ChoosingOption program

Step 1: Create a class.

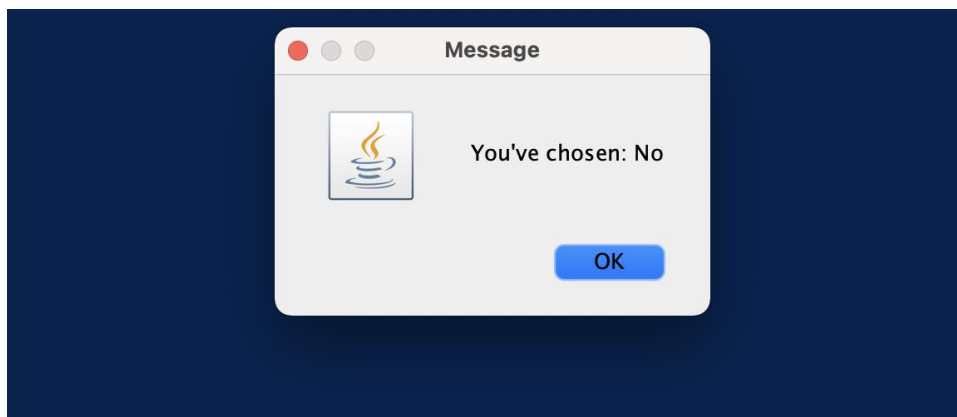
Step 2: Write the program.

Step 3: Save and Launch.



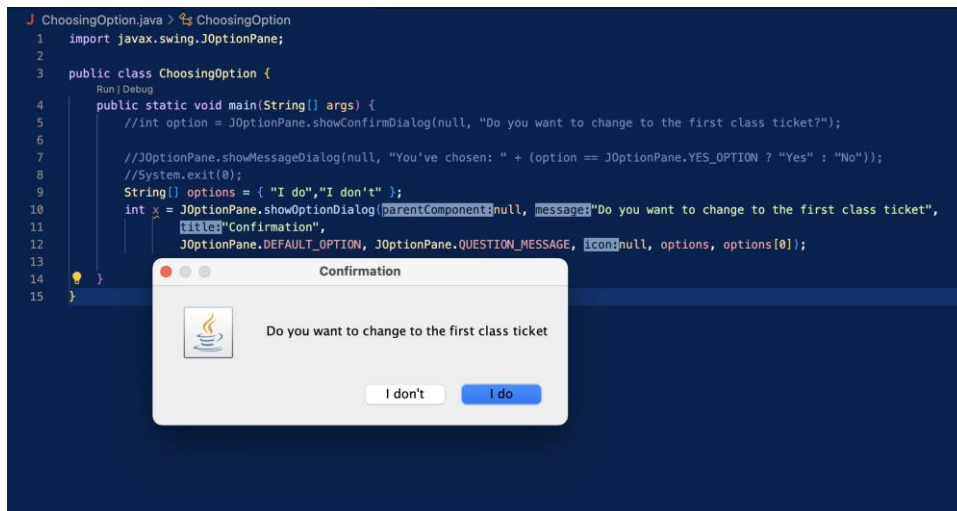
Questions:

- What happens if users choose “Cancel”?



- How to customize the options to users, e.g. only two options: “Yes” and “No”, OR “I do” and “I don’t” (Suggestion: Use Javadocs or using Eclipse/Netbean IDE help).



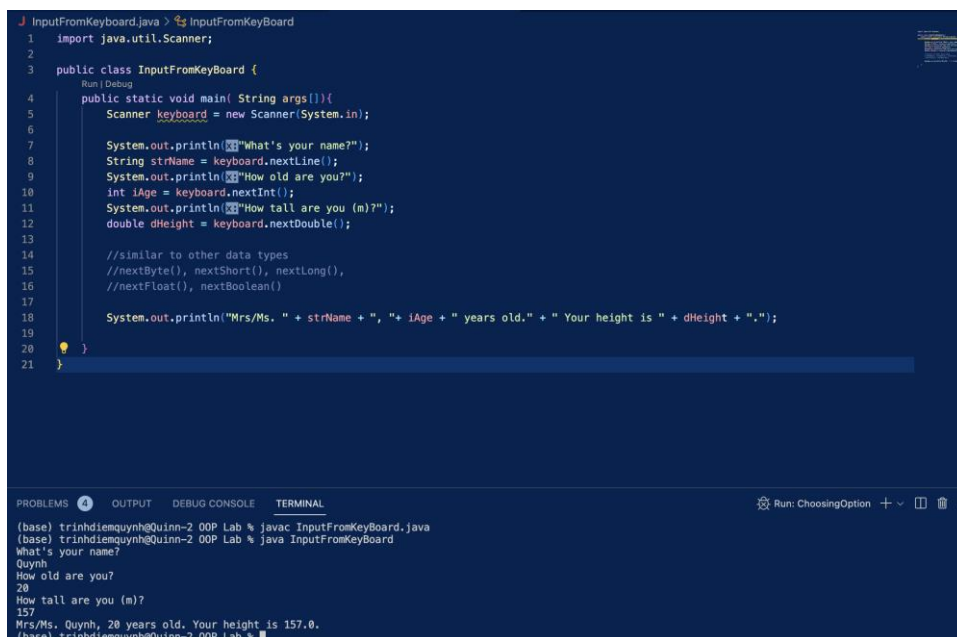


6.2. Write a program for input/output from keyboard

Step 1: Create a class.

Step 2: Write the program.

Step 3: Save and Launch.



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


```

J Triangle.java 1 x
J Triangle.java > Triangle
1 import java.util.Scanner;
2 public class Triangle {
    Run | Debug
3     public static void main(String[] args) {
4         Scanner keyboard = new Scanner(System.in);
5         int n = keyboard.nextInt();
6         for (int i = 1; i <= n; i++) {
7             for (int k = 1; k <= n-i; k++)
8                 System.out.print(" ");
9             for (int j = 1; j <= 2*i-1; j++)
10                {
11                    System.out.print(" *");
12                }
13            System.out.print("\n");
14        }
15    }
16 }

```

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL

```

(base) trindhdiemquynh@Quinn-2 00P Lab % javac Triangle.java
(base) trindhdiemquynh@Quinn-2 00P Lab % java Triangle
3
 *
 ***
*****
(base) trindhdiemquynh@Quinn-2 00P Lab %

```

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.

```

J month.java > month
1 import javax.swing.JOptionPane;
2
3 public class month {
    Run | Debug
4     public static void main(String[] args) {
5         // Get month input from user
6         String monthInput = JOptionPane.showInputDialog(parentComponent:null, message:"Enter a month (full name, abbreviation, or number):",
7             title:"Month Input", JOptionPane.PLAIN_MESSAGE);
8
9         // Get year input from user
10        String yearInput = JOptionPane.showInputDialog(parentComponent:null, message:"Enter a year (non-negative number):", title:"Year Input",
11            JOptionPane.PLAIN_MESSAGE);
12
13        // Check month input
14        int month = parseMonthInput(monthInput);
15        if (month == -1) {
16            JOptionPane.showMessageDialog(parentComponent:null, message:"Invalid month input. Please try again.", title:"Error",
17                JOptionPane.ERROR_MESSAGE);
18            return;
19        }
20
21        // Check year input
22        int year = parseYearInput(yearInput);
23        if (year == -1) {
24            JOptionPane.showMessageDialog(parentComponent:null, message:"Invalid year input. Please try again.", title:"Error",
25                JOptionPane.ERROR_MESSAGE);
26            return;
27        }
28
29        System.out.println(dayOfMonth(year, month));
30    }
31
32    public static int dayOfMonth(int year, int month){
33        if(checkLeapYear(year) == false){
34            if( month == 1 || month == 3 || month == 5 || month == 7 || month == 8 || month == 10 || month ==12 )
35            {
36                return 31;
37            }
38            else if( month == 2) return 28;
39            else return 30;
40        }
41        else {
42            if( month == 1 || month == 3 || month == 5 || month == 7 || month == 8 || month == 10 || month ==12 )
43            {
44                return 31;
45            }

```

```

J month.java > month
42         if( month == 1 || month == 3 || month == 5 || month == 7 || month == 8 || month == 10 || month == 12 )
43         {
44             return 31;
45         }
46         else if( month == 2 ) return 29;
47         else return 30;
48     }
49 }
50 static boolean checkLeapYear(int year){
51     if( year%4 == 0 ){
52         if( year%100 == 0 && year%400 != 0 )
53             return false;
54         else return true;
55     }
56     else{
57         return false;
58     }
59 }
60
61 // Method to parse and validate month input
62 static int parseMonthInput(String input) {
63     input = input.toLowerCase();
64     switch (input) {
65         case "january":
66         case "jan":
67         case "jan.":
68         case "1":
69             return 1;
70         case "february":
71         case "feb":
72         case "feb.":
73         case "2":
74             return 2;
75         case "march":
76         case "mar":
77         case "mar.":
78         case "3":
79             return 3;
80         case "april":
81         case "apr":
82         case "apr.":
83         case "4":
84             return 4;
85         case "may":
86         case "5":

```

```

85         case "may":
86         case "5":
87             return 5;
88         case "june":
89         case "jun":
90         case "jun.":
91         case "6":
92             return 6;
93         case "july":
94         case "jul":
95         case "jul.":
96         case "7":
97             return 7;
98         case "august":
99         case "aug":
100        case "aug.":
101        case "8":
102            return 8;
103        case "september":
104        case "sep":
105        case "sep.":
106        case "9":
107            return 9;
108        case "october":
109        case "oct":
110        case "oct.":
111        case "10":
112            return 10;
113        case "november":
114        case "nov":
115        case "nov.":
116        case "11":
117            return 11;
118        case "december":
119        case "dec":
120        case "dec.":
121        case "12":
122            return 12;
123        default:
124            return -1; // Invalid input
125    }
126 }
127
128 // Method to parse and validate year input
129 static int parseYearInput(String input) {
130     try {

```



```

108 case "october":
109     case "oct":
110     case "oct.":
111     case "10":
112         return 10;
113     case "november":
114     case "nov":
115     case "nov.":
116     case "11":
117         return 11;
118     case "december":
119     case "dec":
120     case "dec.":
121     case "12":
122         return 12;
123     default:
124         return -1;
125 }
126
127 // Method to parse year
128 static int parseYear(String year) {
129     try {
130         int year = Integer.parseInt(year);
131         if (year >= 0) {
132             return year;
133         } else {
134             return -1; // Invalid input
135         }
136     } catch (NumberFormatException e) {
137         return -1;
138     }
139 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

(base) trindhdiemquynh@Quinn-2 OOP Lab % javac Triangle.java
(base) trindhdiemquynh@Quinn-2 OOP Lab % java Triangle
3
*
****
(base) trindhdiemquynh@Quinn-2 OOP Lab % javac month.java
(base) trindhdiemquynh@Quinn-2 OOP Lab % java month

```

```

● (base) trindhdiemquynh@Quinn-2 OOP Lab % javac month.java
● (base) trindhdiemquynh@Quinn-2 OOP Lab % java month
30
○ (base) trindhdiemquynh@Quinn-2 OOP Lab %

```

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.Arrays;
2 import java.util.Scanner;
3
4 public class NumericArray {
5     public static void main(String[] args) {
6         // Input array
7         Scanner keyboard = new Scanner(System.in);
8
9         System.out.print("Enter the size of the array: ");
10        int n = keyboard.nextInt();
11
12        int[] arr = new int[n];
13
14        System.out.println("Enter the elements of the array:");
15        for (int i = 0; i < n; i++) {
16            arr[i] = keyboard.nextInt();
17        }
18        keyboard.close();
19
20        /*System.out.println("The array elements are:");
21        for (int number : arr) {
22            System.out.print(number + " ");
23        }
24        */
25
26        // Sort the array
27        Arrays.sort(arr);
28
29        System.out.println("Sorted Array: " + Arrays.toString(arr));
30
31        // Calculate sum of array elements
32        int sum = 0;
33        for (int i = 0; i < n; i++) {
34            sum += arr[i];
35        }
36        System.out.println("Sum of Array Elements: " + sum);
37
38        // Calculate average of array elements
39        double average = (double) sum / n;
40        System.out.println("Average of Array Elements: " + average);
41    }
42 }

```

```

● (base) trindhdiemquynh@Quinn-2 OOP Lab % javac NumericArray.java
● (base) trindhdiemquynh@Quinn-2 OOP Lab % java NumericArray
Enter the size of the array: 5
Enter the elements of the array:
1789 2035 1899 1456 2013
Sorted Array: [1456, 1789, 1899, 2013, 2035]
Sum of Array Elements: 9192
Average of Array Elements: 1838.4
○ (base) trindhdiemquynh@Quinn-2 OOP Lab % █

```

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

```

J AddMatrix.java > AddMatrix
1  import java.util.Scanner;
2
3  public class AddMatrix {
4      public static void main(String[] args) {
5          // Input matrices size n x m
6          Scanner scanner = new Scanner(System.in);
7
8          int n = scanner.nextInt();
9          int m = scanner.nextInt();
10
11         // matrix 1
12         double[][] a = new double[n][m];
13         // matrix 2
14         double[][] b = new double[n][m];
15         // matrix sum
16         double[][] c = new double[n][m];
17
18         for( int i = 0 ; i<n ; i++)
19             for( int j = 0 ; j<m ; j++){
20                 a[i][j] = scanner.nextDouble();
21             }
22
23         for( int i = 0 ; i<n ; i++)
24             for( int j = 0 ; j<m ; j++){
25                 b[i][j] = scanner.nextDouble();
26             }
27         for( int i = 0 ; i<n ; i++)
28             for( int j = 0 ; j<m ; j++){
29                 c[i][j] = a[i][j]+b[i][j];
30             }
31
32         System.out.println("Matrix 1:");
33         printMatrix(a);
34         System.out.println("Matrix 2:");
35         printMatrix(b);
36         System.out.println("Matrix Sum:");
37         printMatrix(c);
38     }
39
40     // Utility method to print a matrix
41     private static void printMatrix(double[][] matrix) {
42         for (double[] row : matrix) {
43             for (double num : row) {
44                 System.out.print(num + " ");
45             }
46             System.out.println();
47         }
48         System.out.println();
49     }
50 }

```

```

● (base) trindhdiemquynh@Quinn-2 OOP Lab % javac AddMatrix.java
● (base) trindhdiemquynh@Quinn-2 OOP Lab % java AddMatrix
3 2
1 4
2 3
5 6
8 2
9 4
0 3
Matrix 1:
1.0 4.0
2.0 3.0
5.0 6.0

Matrix 2:
8.0 2.0
9.0 4.0
0.0 3.0

Matrix Sum:
9.0 6.0
11.0 7.0
5.0 9.0
○ (base) trindhdiemquynh@Quinn-2 OOP Lab % █

```