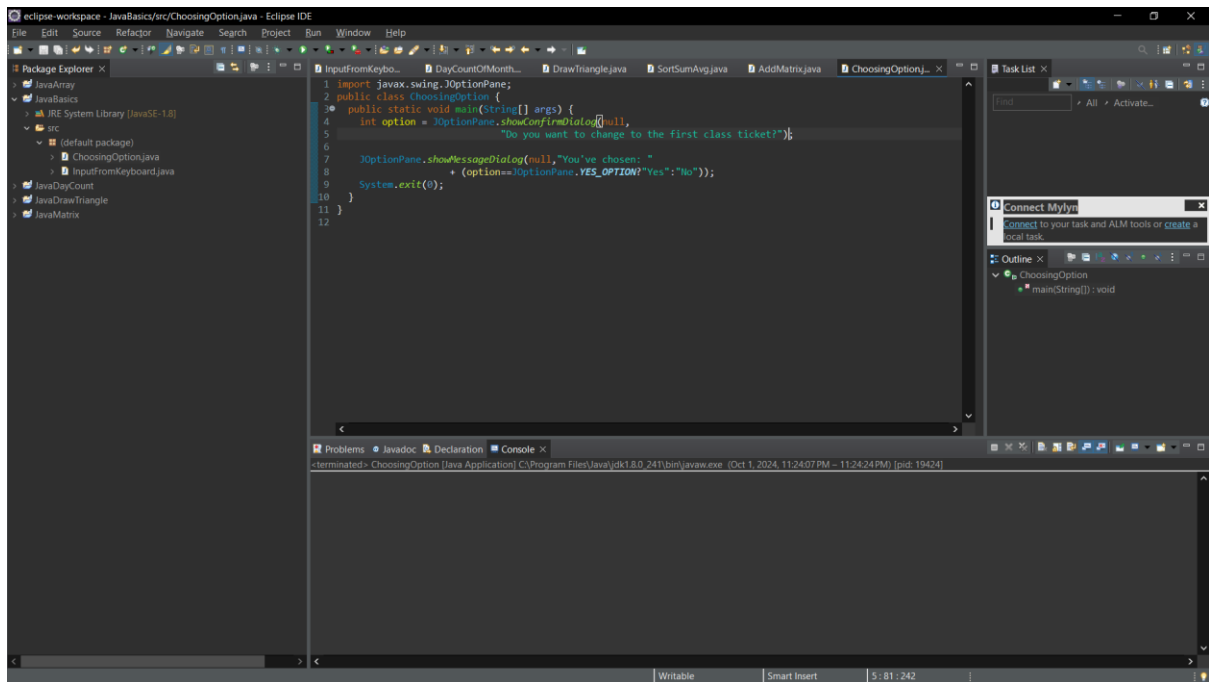


Lab 01: Environment Setup and Java Basics

Họ Tên	Nguyễn Thế Kiên
MSSV	20194597

6. Exercises

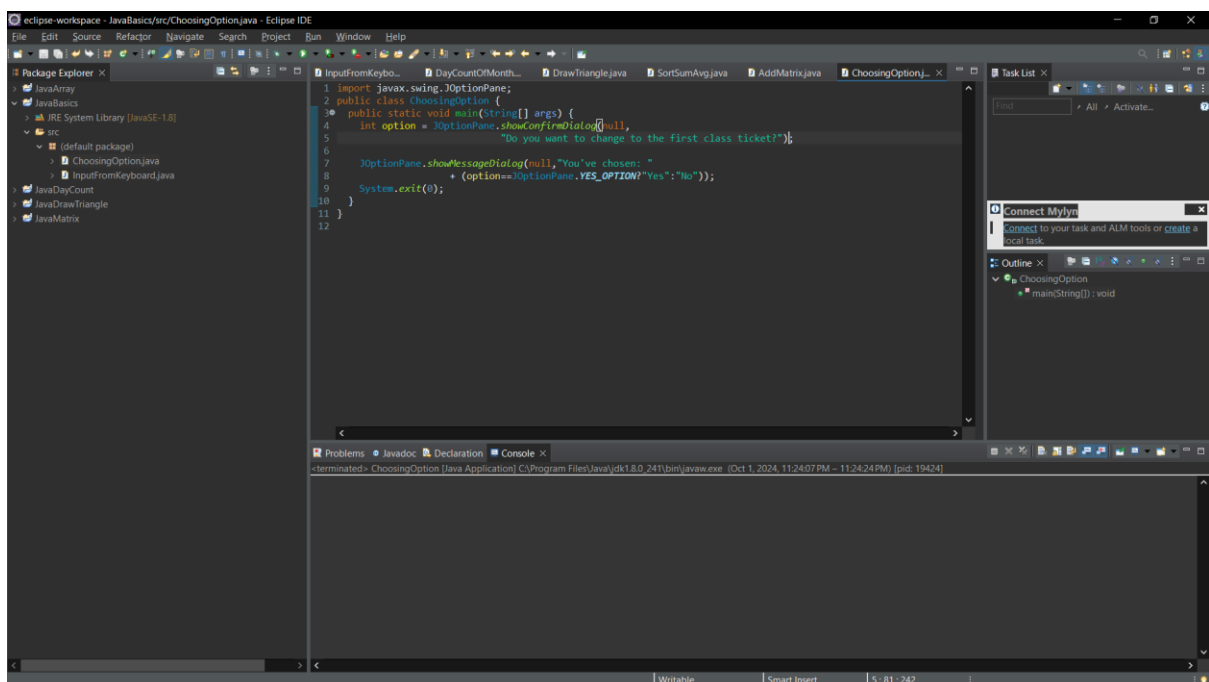
6.1 Write, compile and run the ChoosingOption program



```
1 import javax.swing.JOptionPane;
2 public class ChoosingOption {
3     public static void main(String[] args) {
4         int option = JOptionPane.showConfirmDialog(null,
5             "Do you want to change to the first class ticket?");
6
7         JOptionPane.showMessageDialog(null, "You've chosen: "
8             + (option==JOptionPane.YES_OPTION?"Yes":"No"));
9         System.exit(0);
10    }
11 }
12
```

Console Output:

```
<terminated> ChoosingOption [Java Application] C:\Program Files\Java\jdk1.8.0_241\bin\javaw.exe (Oct 1, 2024, 11:24:07 PM -- 11:24:24 PM) [pid: 19424]
```



```
1 import javax.swing.JOptionPane;
2 public class ChoosingOption {
3     public static void main(String[] args) {
4         int option = JOptionPane.showConfirmDialog(null,
5             "Do you want to change to the first class ticket?");
6
7         JOptionPane.showMessageDialog(null, "You've chosen: "
8             + (option==JOptionPane.YES_OPTION?"Yes":"No"));
9         System.exit(0);
10    }
11 }
12
```

Console Output:

```
<terminated> ChoosingOption [Java Application] C:\Program Files\Java\jdk1.8.0_241\bin\javaw.exe (Oct 1, 2024, 11:24:07 PM -- 11:24:24 PM) [pid: 19424]
```

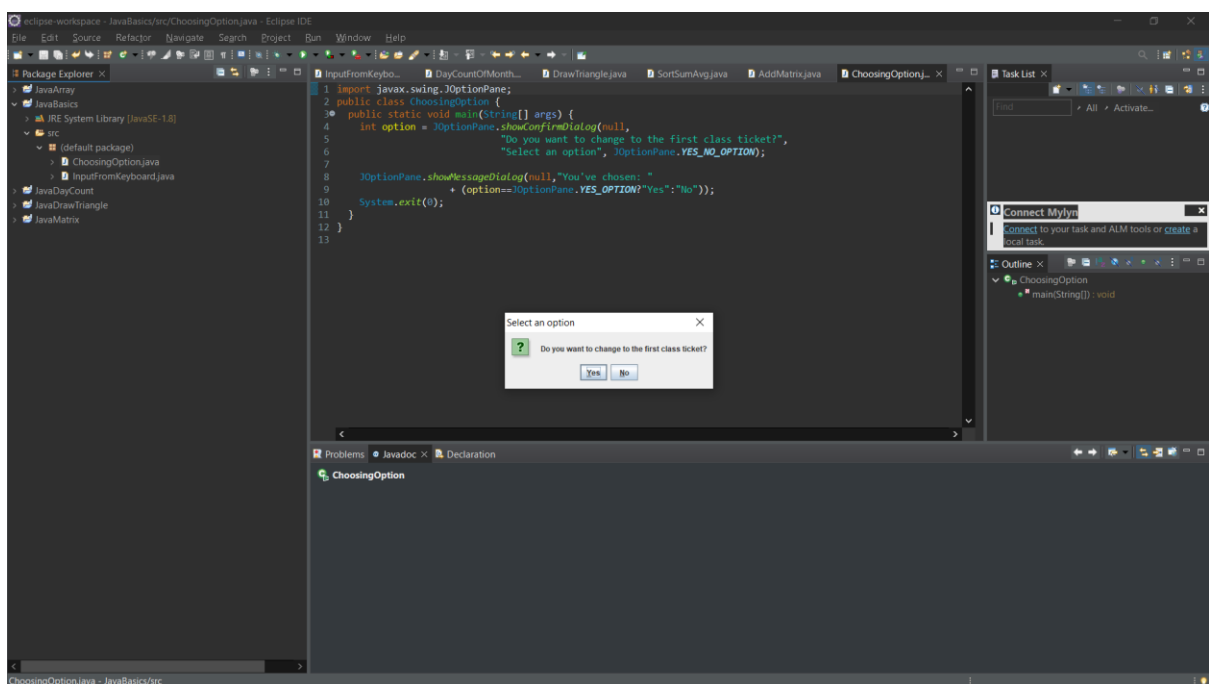
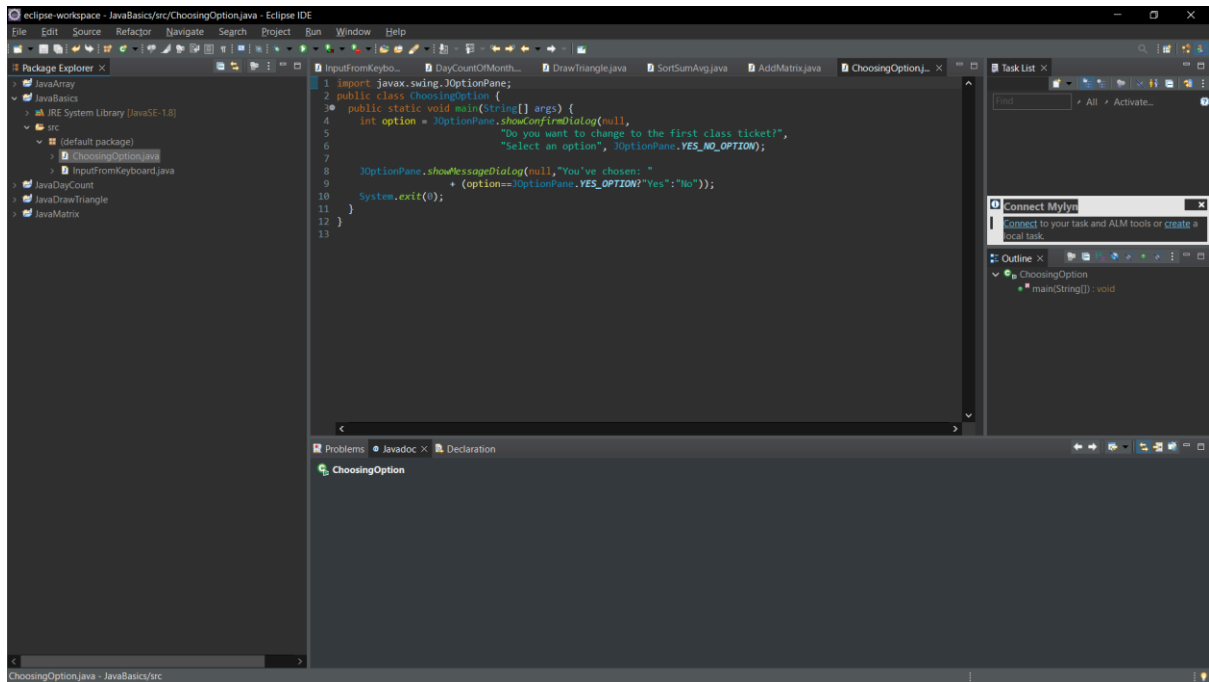
Questions:

- What happens if users choose “Cancel”?

-> Same as choosing “No”

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

-> Use another *showConfirmMessage* method that has *optionType* parameter set to ‘JOptionPane.YES_NO_OPTION’



6.2 Write a program for input/output from keyboard

The screenshot shows the Eclipse IDE with the `InputFromKeyboard.java` file open. The code uses `Scanner` to read user input for name, age, and height, and prints the results. The Package Explorer on the left shows the project structure. The Console at the bottom is empty, indicating the program has not yet been executed.

```
1 import java.util.Scanner;
2 public class InputFromKeyboard {
3
4     public static void main(String args[]) {
5         Scanner keyboard = new Scanner(System.in);
6
7         System.out.println("What's your name?");
8         String strName = keyboard.nextLine();
9         System.out.println("How old are you?");
10        int iAge = keyboard.nextInt();
11        System.out.println("How tall are you (m)?");
12        double dHeight = keyboard.nextDouble();
13
14        //similar to other data types
15        //nextByte(), nextShort(), nextLong()
16        //nextFloat(), nextBoolean()
17
18        System.out.println("Mrs/Ms. " + strName + ", " + iAge + " years old. "
19                            + "Your height is " + dHeight + "m.");
20
21        keyboard.close();
22    }
23 }
24 }
25 }
```

The screenshot shows the Eclipse IDE with the `InputFromKeyboard.java` file open. The code is the same as in the previous screenshot. The Console at the bottom now shows the output of the program, indicating it has been executed successfully.

```
<terminated> InputFromKeyboard [Java Application] C:\Program Files\Java\jdk1.8.0_241\bin\java.exe (Oct 1, 2024, 11:25:07 PM - 11:25:38 PM) [pid: 9580]
What's your name?
Kien
How old are you?
33
How tall are you (m)?
1.6
Mrs/Ms. Kien, 33 years old. Your height is 1.6.
```

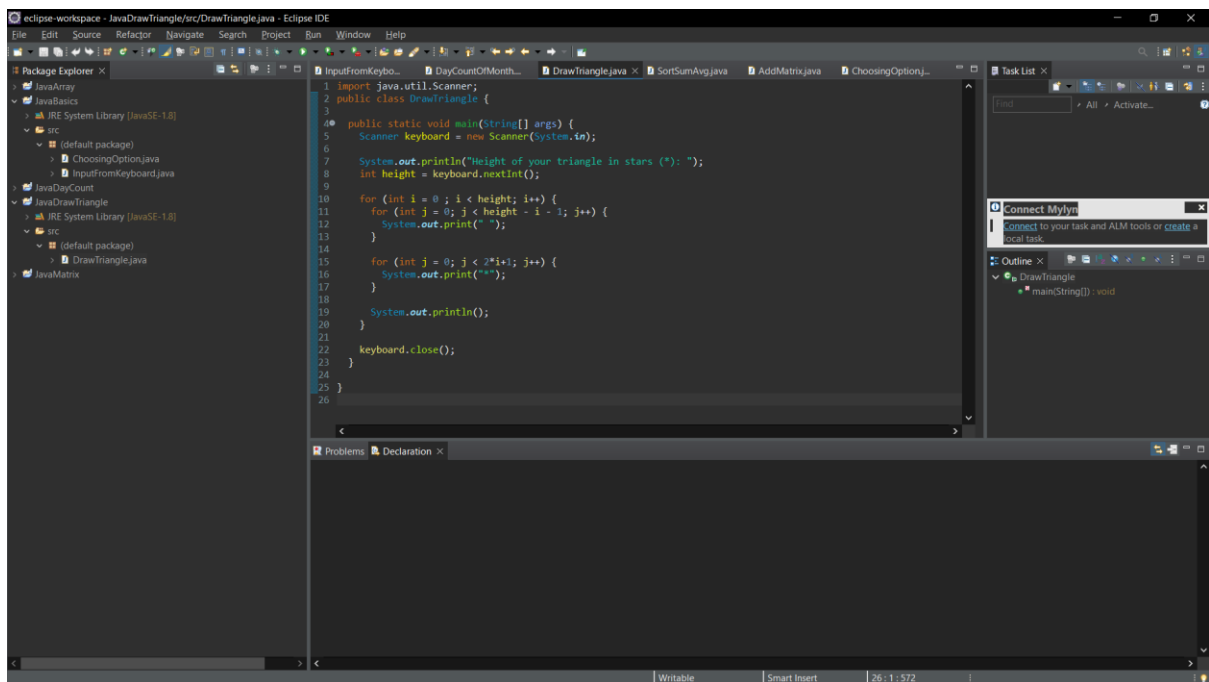
6.3 Write a program to display a triangle with a height of n stars (*), n is entered by users.

E.g. n=5:

```

      *
     ***
    *****
   *****
  *****
 *****

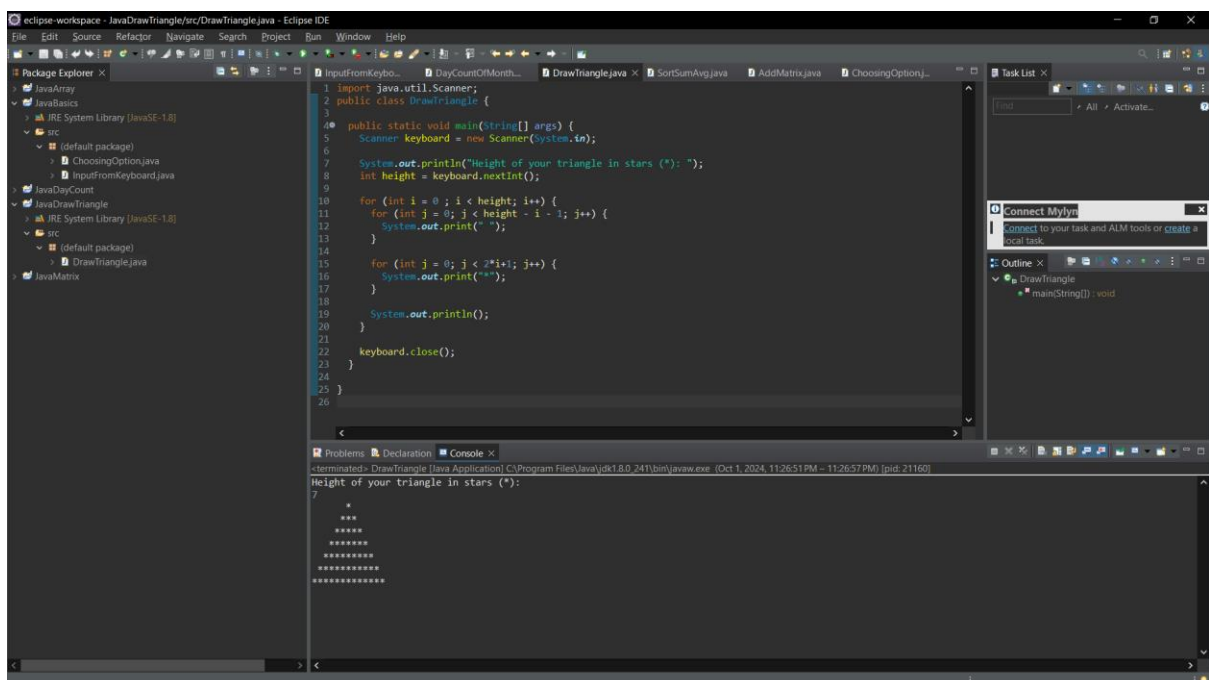
```



The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like `java.util`, `java.io`, and `java.math`.
- Editor:** Displays the `DrawTriangle.java` file with the following code:

```
1 import java.util.Scanner;
2 public class DrawTriangle {
3
4     public static void main(String[] args) {
5         Scanner keyboard = new Scanner(System.in);
6
7         System.out.println("Height of your triangle in stars (*): ");
8         int height = keyboard.nextInt();
9
10        for (int i = 0; i < height; i++) {
11            for (int j = 0; j < height - i - 1; j++) {
12                System.out.print(" ");
13            }
14
15            for (int j = 0; j < 2*i+1; j++) {
16                System.out.print("*");
17            }
18
19            System.out.println();
20        }
21
22        keyboard.close();
23    }
24 }
25
```
- Task List:** Shows a task titled "Connect Mylyn" with a description "Connect to your task and ALM tools or create a local task."
- Outline:** Shows the class structure with a method `main(String[] args) : void`.
- Problems:** Shows no problems.
- Declaration:** Shows no declarations.



The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Same as the previous screenshot.
- Editor:** Same as the previous screenshot.
- Task List:** Same as the previous screenshot.
- Outline:** Same as the previous screenshot.
- Problems:** Same as the previous screenshot.
- Declaration:** Same as the previous screenshot.
- Console:** Shows the output of the program:

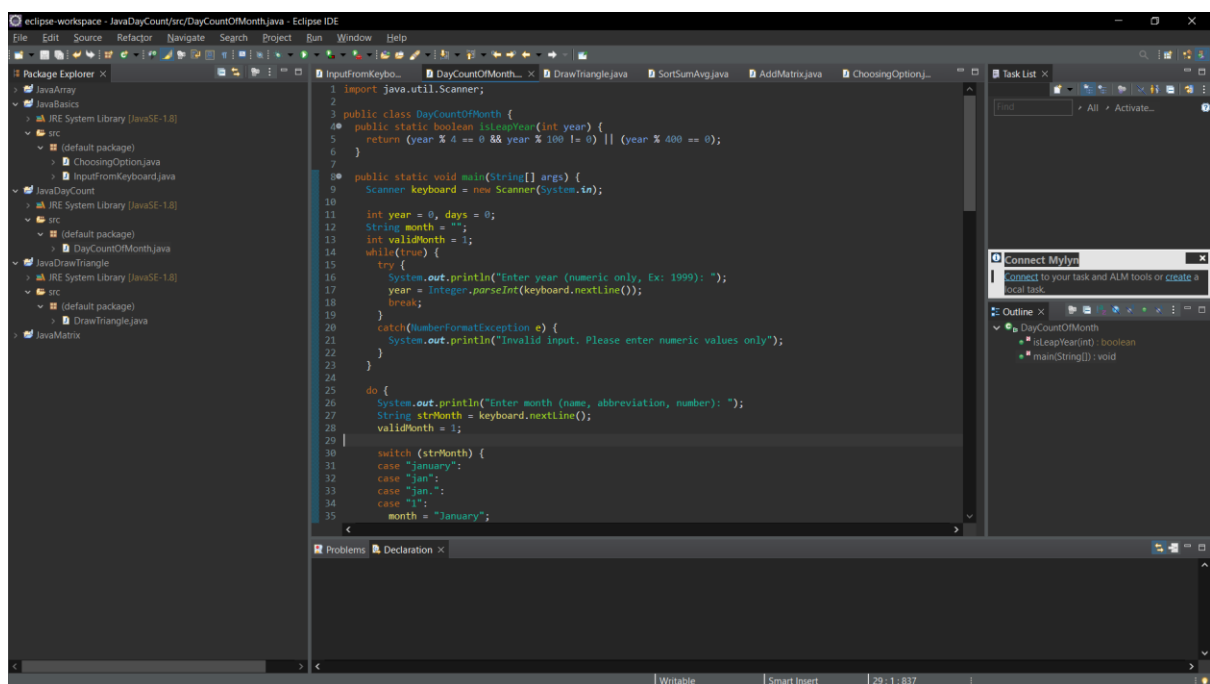
```
<terminated> DrawTriangle [Java Application] C:\Program Files\Java\jdk1.8.0_241\bin\java.exe (Oct 1, 2024, 11:26:51 PM - 11:26:57 PM) [pid: 21160]
Height of your triangle in stars (*):
7
 *
 ***
 *****
 *****
 *****
 *****

```

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.

Note: You must create a new Java project for this exercise.

- The user can either enter a month in its full name, abbreviation, in 3 letters, or in number. To illustrate, the valid inputs of January are January, Jan., Jan, and 1.
- The user must enter a year in a non-negative number and enter all the digits. For instance, the valid inputs of year 1999 is only 1999, but not 99, “one thousand nine hundred ninety-nine”, or anything else.
- A year is either a common year of 365 days or a leap year of 366 days. Every year that is divisible by 4 is a leap year, except for years that are divisible by 100, but not by 400. For instance, year 1800 is not a leap year, yet year 2000 is a leap year.



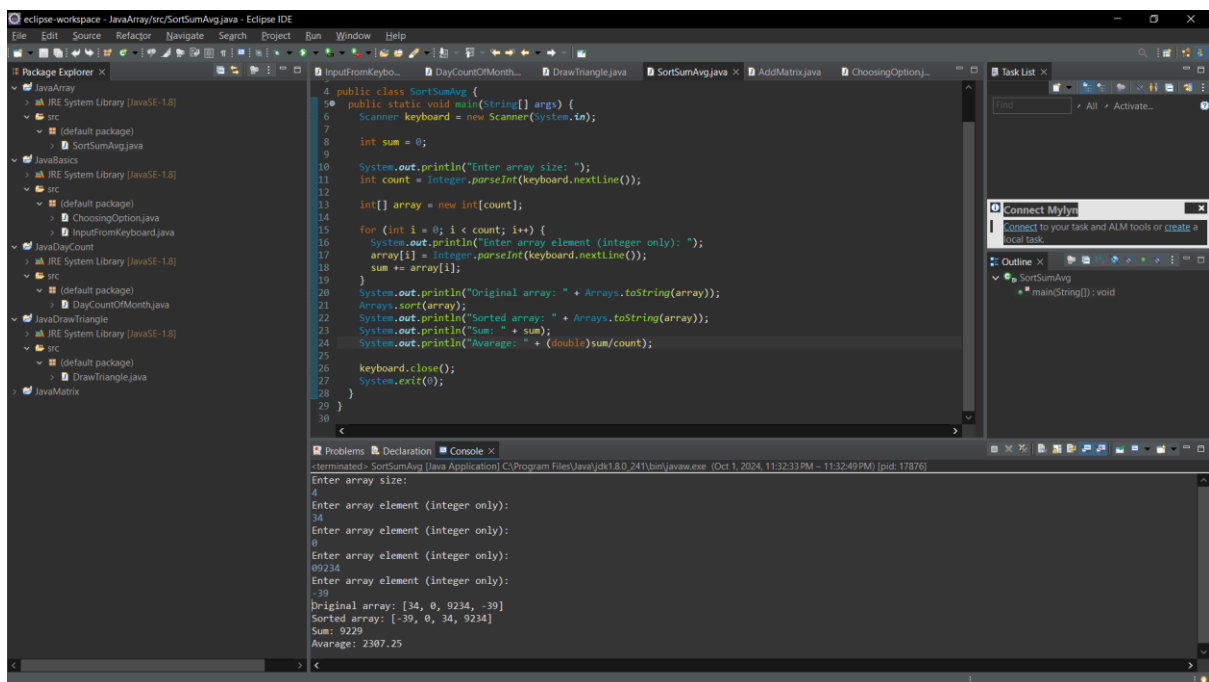
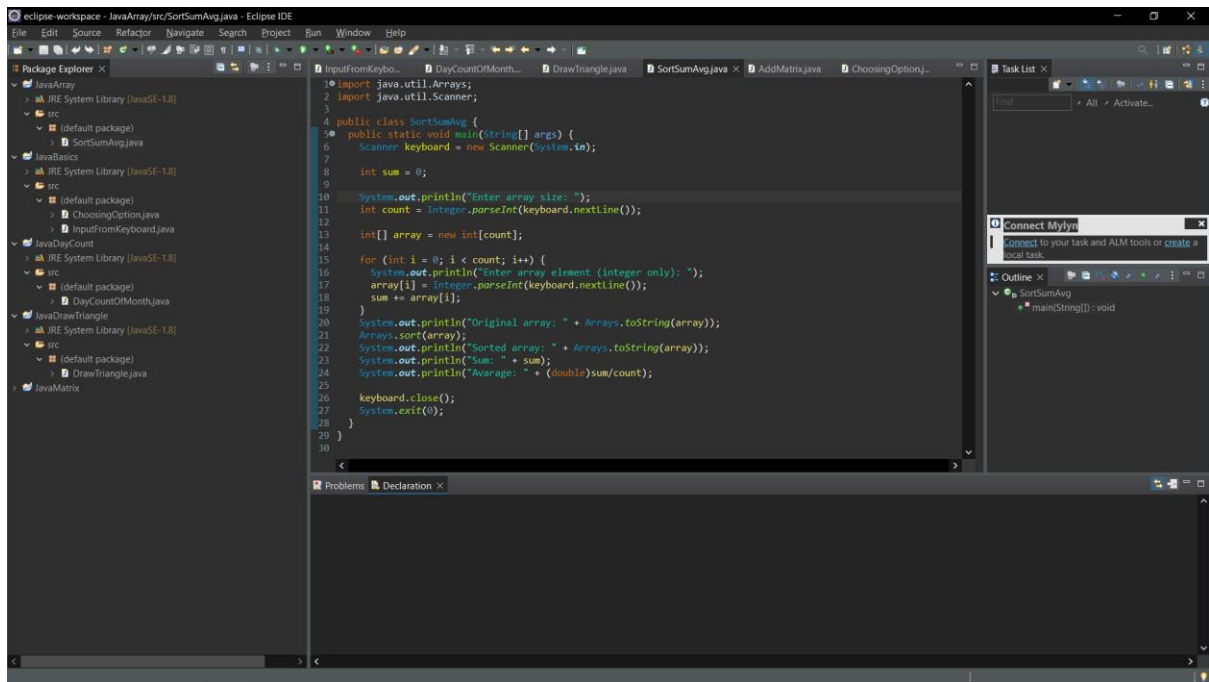
```
99      break;
100     case "november":
101     case "nov":
102     case "11":
103     case "11":
104     month = "November";
105     days = 30;
106     break;
107     case "december":
108     case "dec":
109     case "dec.":
110     case "12":
111     month = "December";
112     days = 31;
113     break;
114     default:
115     System.out.println("Invalid input. Please enter a valid month.");
116     validMonth = 0;
117     break;
118     } while(validMonth == 0);
119 }
120
121 System.out.println("Month: " + month);
122 System.out.println("Number of days: " + days);
123 keyboard.close();
124 }
125 }
126 }
```

Enter year (numeric only, Ex: 1999):
1700
Enter month (name, abbreviation, number):
Feb.
Month: February
Number of days: 28

```
99      break;
100     case "november":
101     case "nov":
102     case "11":
103     case "11":
104     month = "November";
105     days = 30;
106     break;
107     case "december":
108     case "dec":
109     case "dec.":
110     case "12":
111     month = "December";
112     days = 31;
113     break;
114     default:
115     System.out.println("Invalid input. Please enter a valid month.");
116     validMonth = 0;
117     break;
118     } while(validMonth == 0);
119 }
120
121 System.out.println("Month: " + month);
122 System.out.println("Number of days: " + days);
123 keyboard.close();
124 }
125 }
126 }
```

Enter year (numeric only, Ex: 1999):
2024
Enter month (name, abbreviation, number):
2
Month: February
Number of days: 29

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



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

