Java programming

Group members

- **Miriam Njeri 23/03696**
- **Sheldon Wambugi 23/03062**
- **Gatobu Joy kananu 23/03707**

Instructions

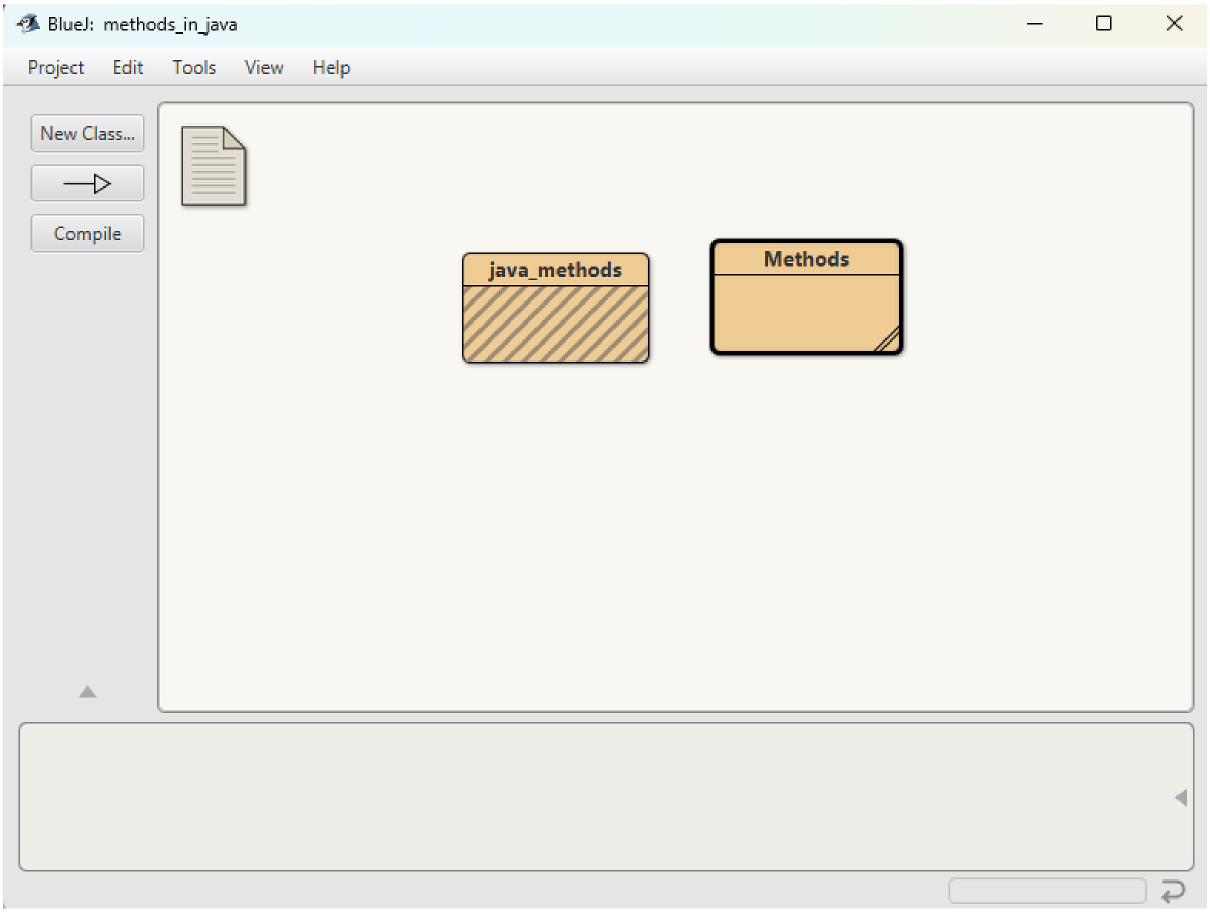
Answer all the questions.
Each question is five marks.
To be done on groups of at most three

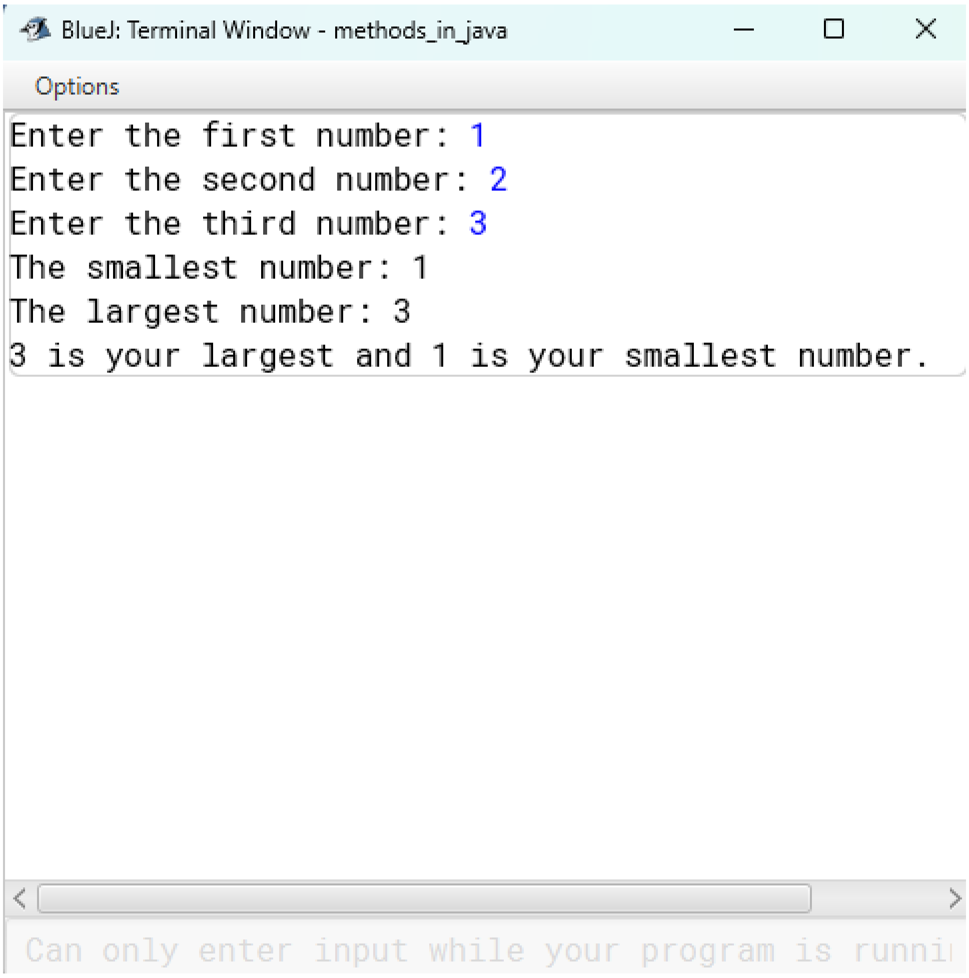
QUESTION ONE:

Create a java project, name it methods_in_java, in the project create a package named java_methods and in the package, create a class and named methods.
b. in the classmethods, write a method that asks user to enter three numbers, using if statement determine the largest number, the smallest number and display the results in the following format.
The smallest number: ?
The largest number number: ?
? is your largest and ? smallest number.

Answer part

```
import java.util.Scanner;

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

        System.out.print("Enter the first number: ");
        int num1 = scanner.nextInt();

        System.out.print("Enter the second number: ");
        int num2 = scanner.nextInt();

        System.out.print("Enter the third number: ");
        int num3 = scanner.nextInt();

        int largest, smallest;

        // Determine the largest number
        if (num1 >= num2 && num1 >= num3) {
            largest = num1;
        } else if (num2 >= num1 && num2 >= num3) {
            largest = num2;
        } else {
            largest = num3;
        }

        // Determine the smallest number
        if (num1 <= num2 && num1 <= num3) {
            smallest = num1;
        } else if (num2 <= num1 && num2 <= num3) {
            smallest = num2;
        } else {
            smallest = num3;
        }

        // Display the results
        System.out.println("The smallest number: " + smallest);
        System.out.println("The largest number: " + largest);
        System.out.println(largest + " is your largest and " + smallest + " is your smallest
number.");

        scanner.close();
    }
}
```


BlueJ: methods_in_java
Project
Edit
Tools
View
Help
New Class...
Compile
java_methods
Methods



BlueJ: Terminal Window - methods_in_java
Options

```
Enter the first number: 1
Enter the second number: 2
Enter the third number: 3
The smallest number: 1
The largest number: 3
3 is your largest and 1 is your smallest number.
```

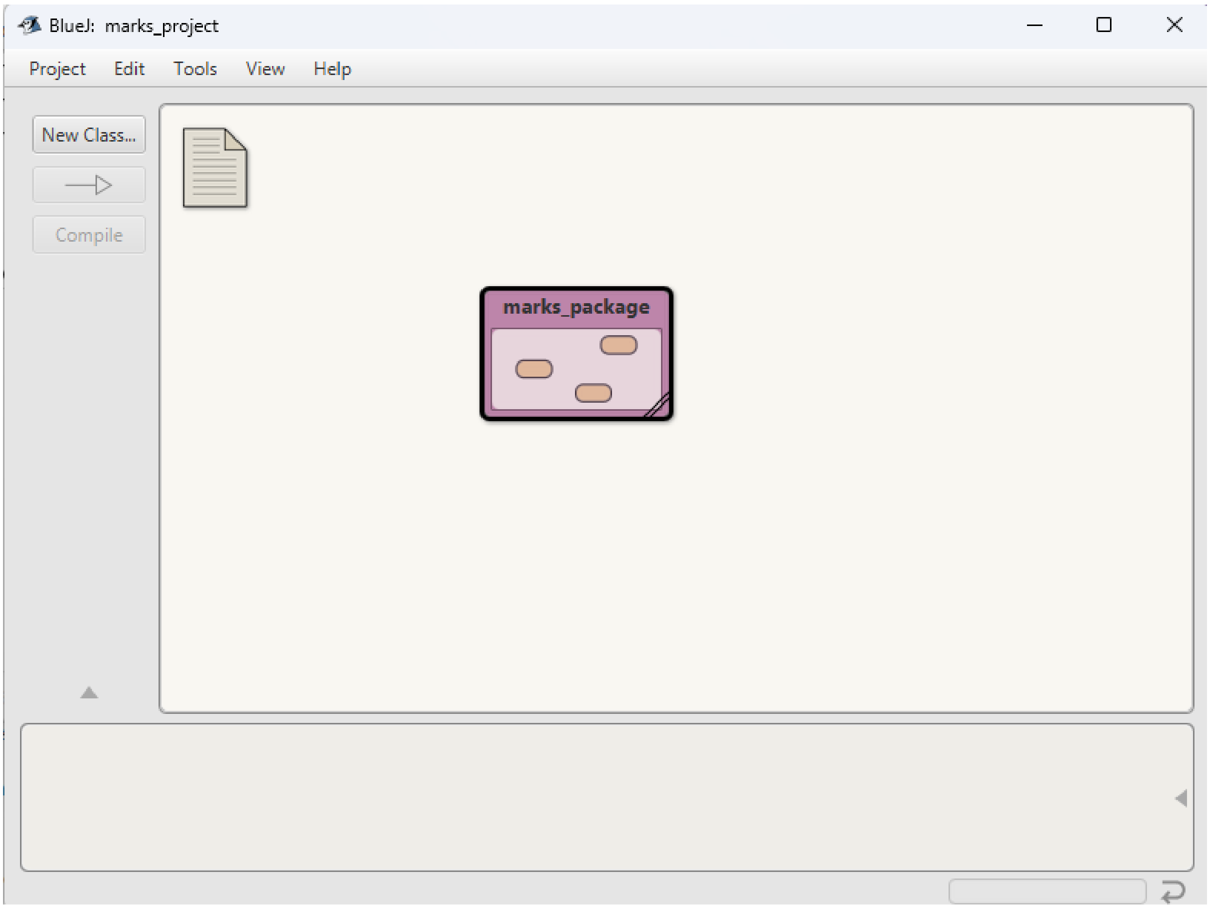
Can only enter input while your program is runni

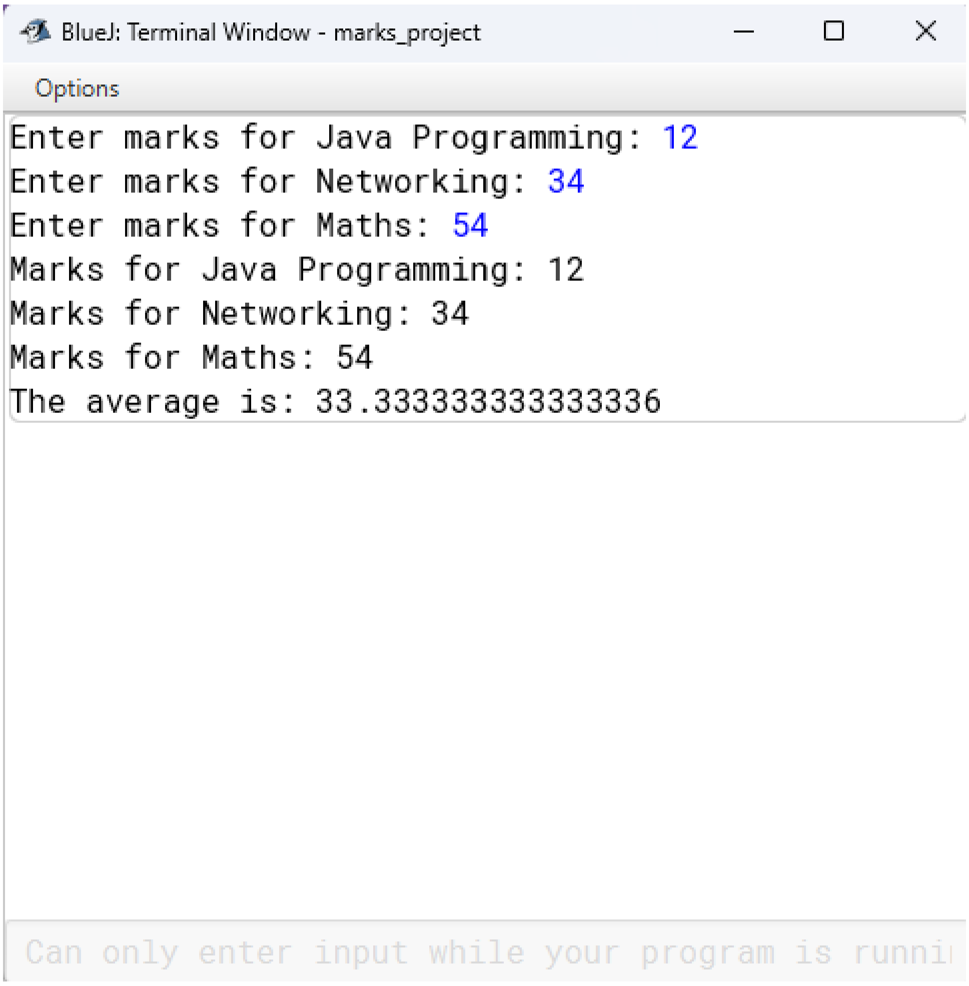
QUESTION 2:
Create a java project, package and class. In the class, write a method that asks a lecturer to enter marks for three units, java programming, networking and maths. The method should compute the average marks for three units and output the data in the following format.

marks for java programming is: ?
marks for networking is: ?
marks for maths is: ?
the average is: ?

Answer part

```
package marks_package;

import java.util.Scanner;

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

      System.out.print("Enter marks for Java Programming: ");
      int javaMarks = scanner.nextInt();

      System.out.print("Enter marks for Networking: ");
      int networkingMarks = scanner.nextInt();

      System.out.print("Enter marks for Maths: ");
      int mathsMarks = scanner.nextInt();

      // Compute the average
      double average = (javaMarks + networkingMarks + mathsMarks) / 3.0;

      // Display the results
      System.out.println("Marks for Java Programming: " + javaMarks);
      System.out.println("Marks for Networking: " + networkingMarks);
      System.out.println("Marks for Maths: " + mathsMarks);
      System.out.println("The average is: " + average);

      scanner.close();
   }
}
```


BlueJ: marks_project
Project
Edit
Tools
View
Help
New Class...
Compile
marks_package



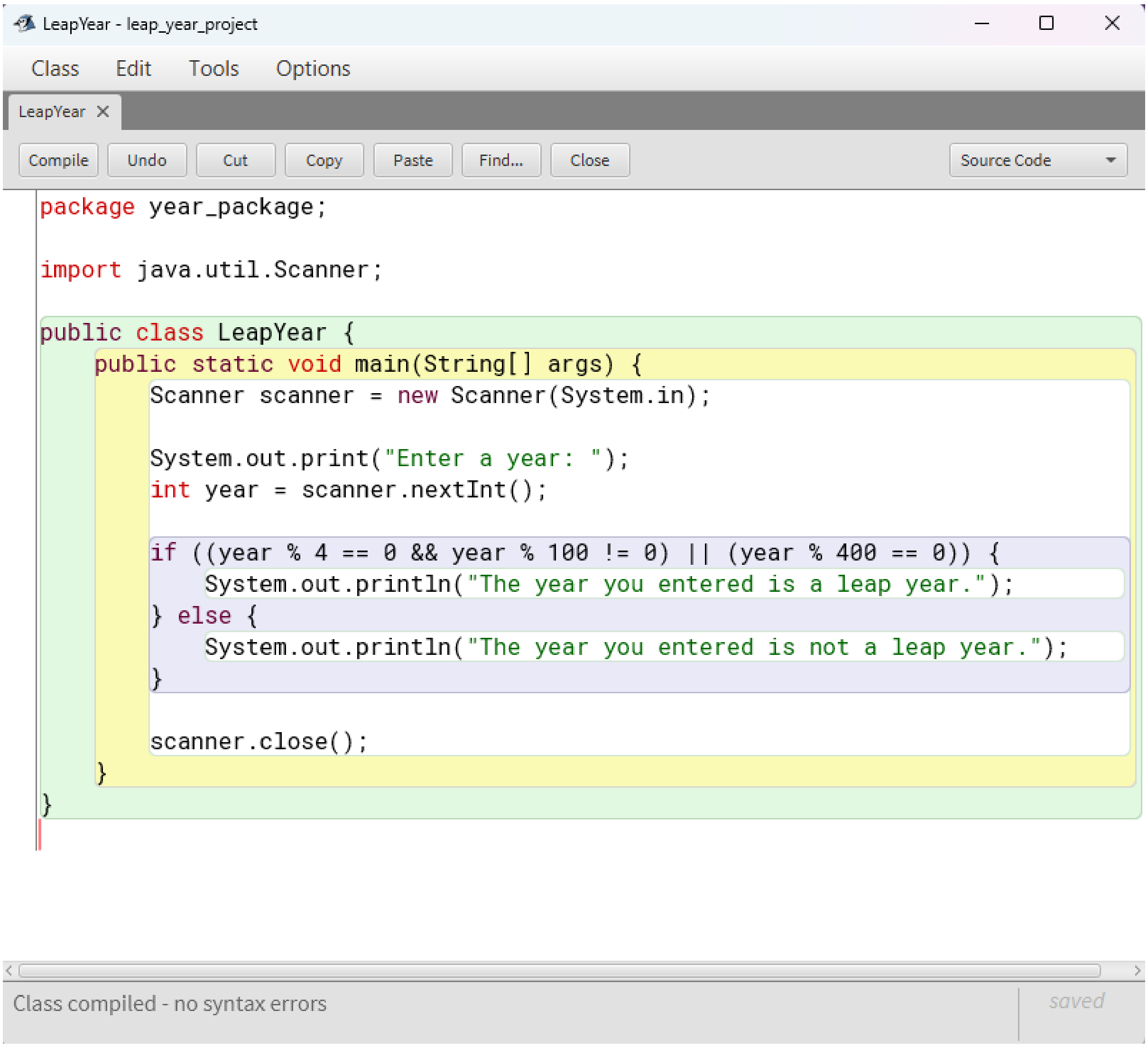
BlueJ: Terminal Window - marks_project
Options

```
Enter marks for Java Programming: 12
Enter marks for Networking: 34
Enter marks for Maths: 54
Marks for Java Programming: 12
Marks for Networking: 34
Marks for Maths: 54
The average is: 33.333333333333336
```

Can only enter input while your program is runni

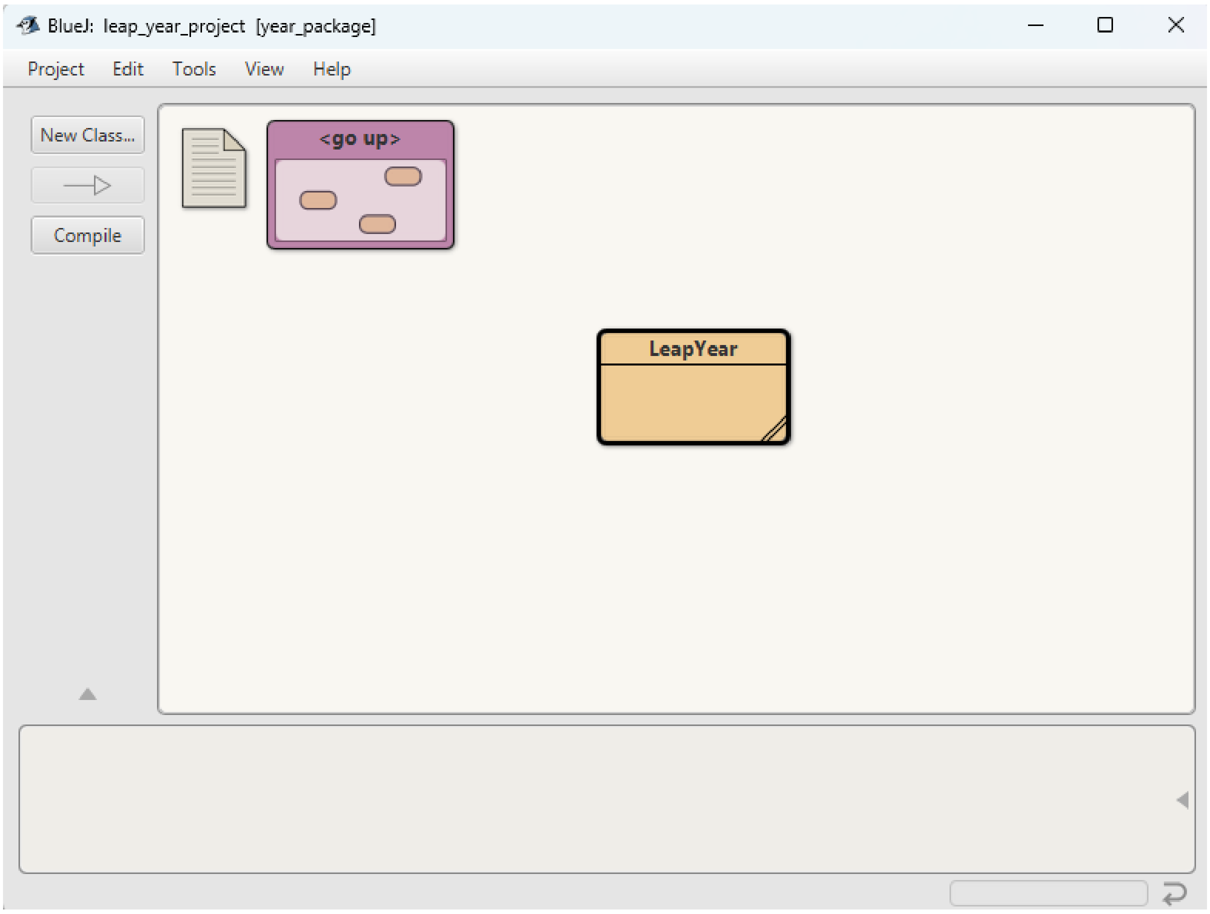
QUESTION 3:

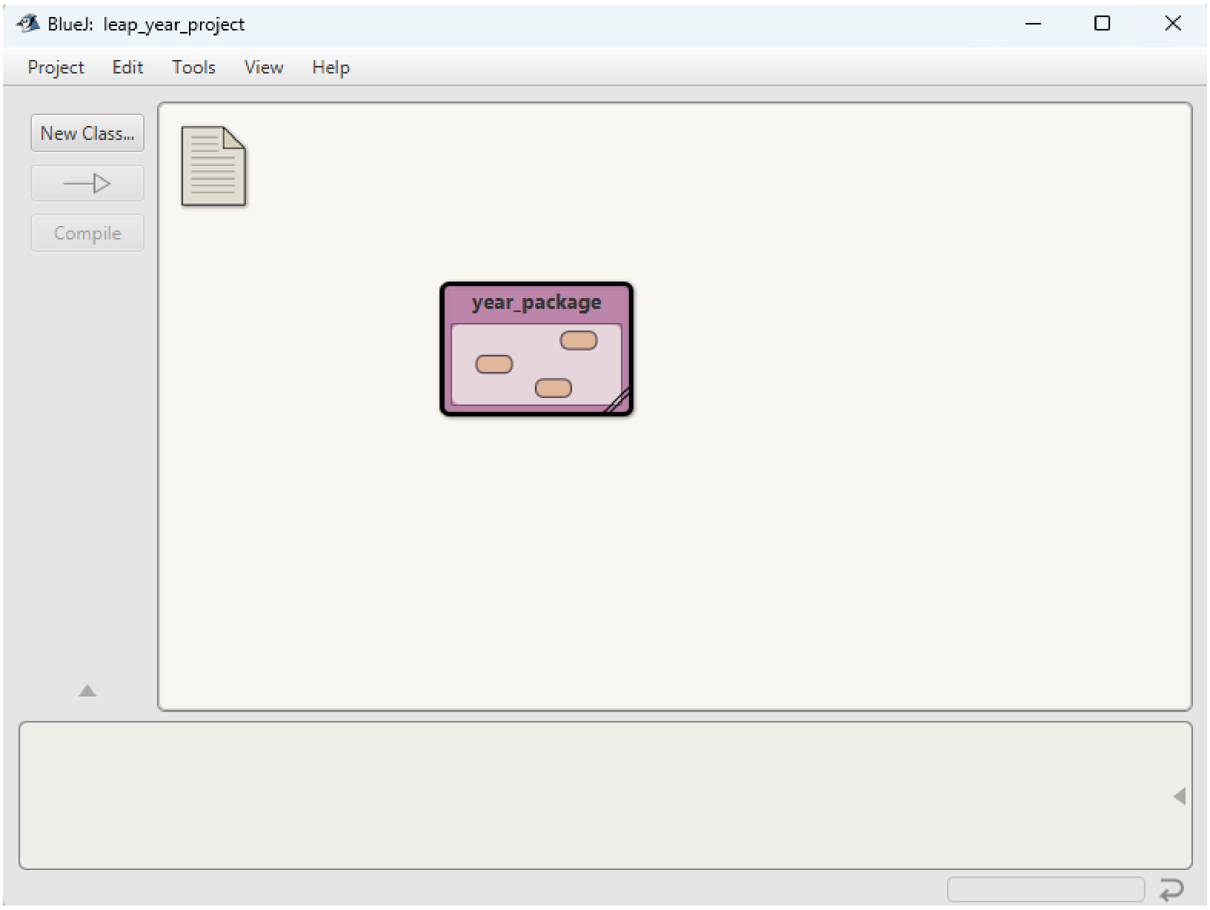
Write a method that asks user to enter the year, the program should check if the year is a leap year, and output the text the year you entered is a leap year.



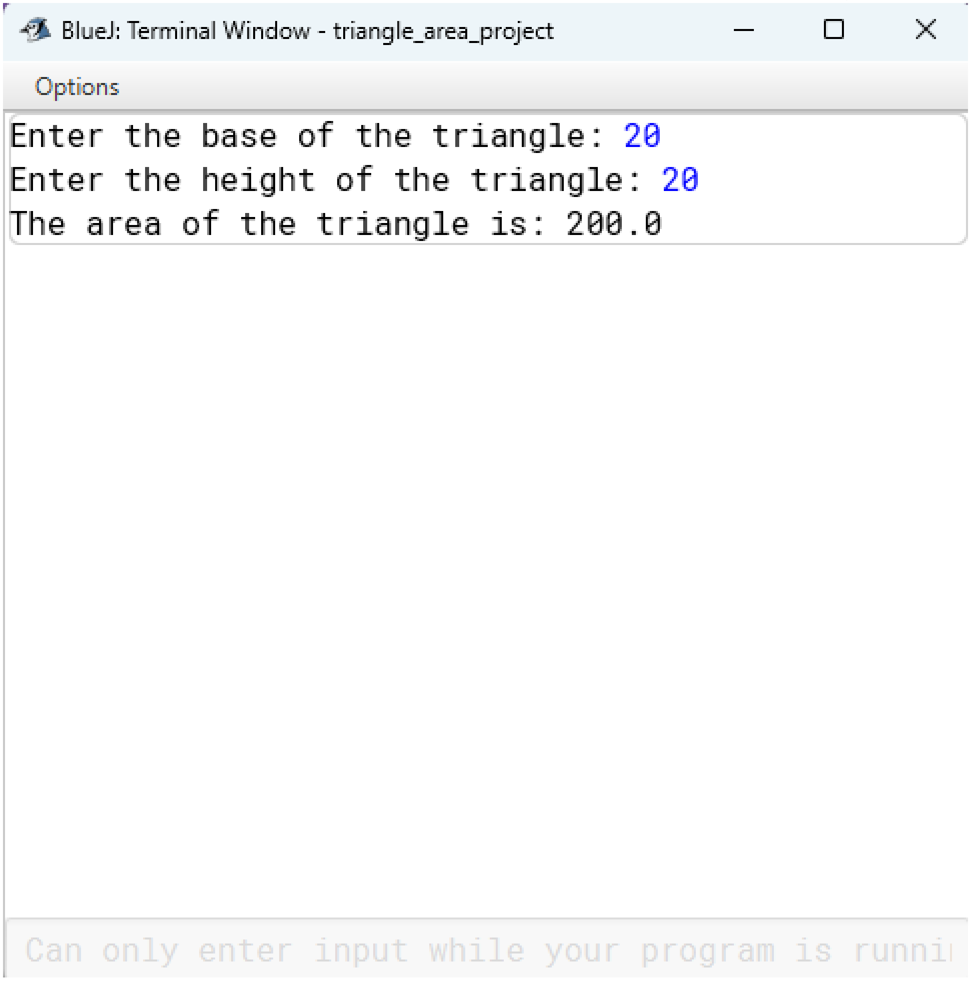
```
package year_package;

import java.util.Scanner;

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

        System.out.print("Enter a year: ");
        int year = scanner.nextInt();

        if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
            System.out.println("The year you entered is a leap year.");
        } else {
            System.out.println("The year you entered is not a leap year.");
        }

        scanner.close();
    }
}
```

Class compiled - no syntax errors


BlueJ: leap_year_project [year_package]
Project
Edit
Tools
View
Help
New Class...
Compile
<go up>
LeapYear


BlueJ: leap_year_project
Project
Edit
Tools
View
Help
New Class...
Compile
year_package

Create a java project, a package and a class, in the class, write a program to calculate the area of a triangle. The program should have thee non-static methods:
One of the methods should ask the user to enter the base and the height of a triangle.
The other method should compute the area of the rectangle.
The other method should output the calculated area of the triangle an display it to the user.


BlueJ: Terminal Window - triangle_area_project
Options

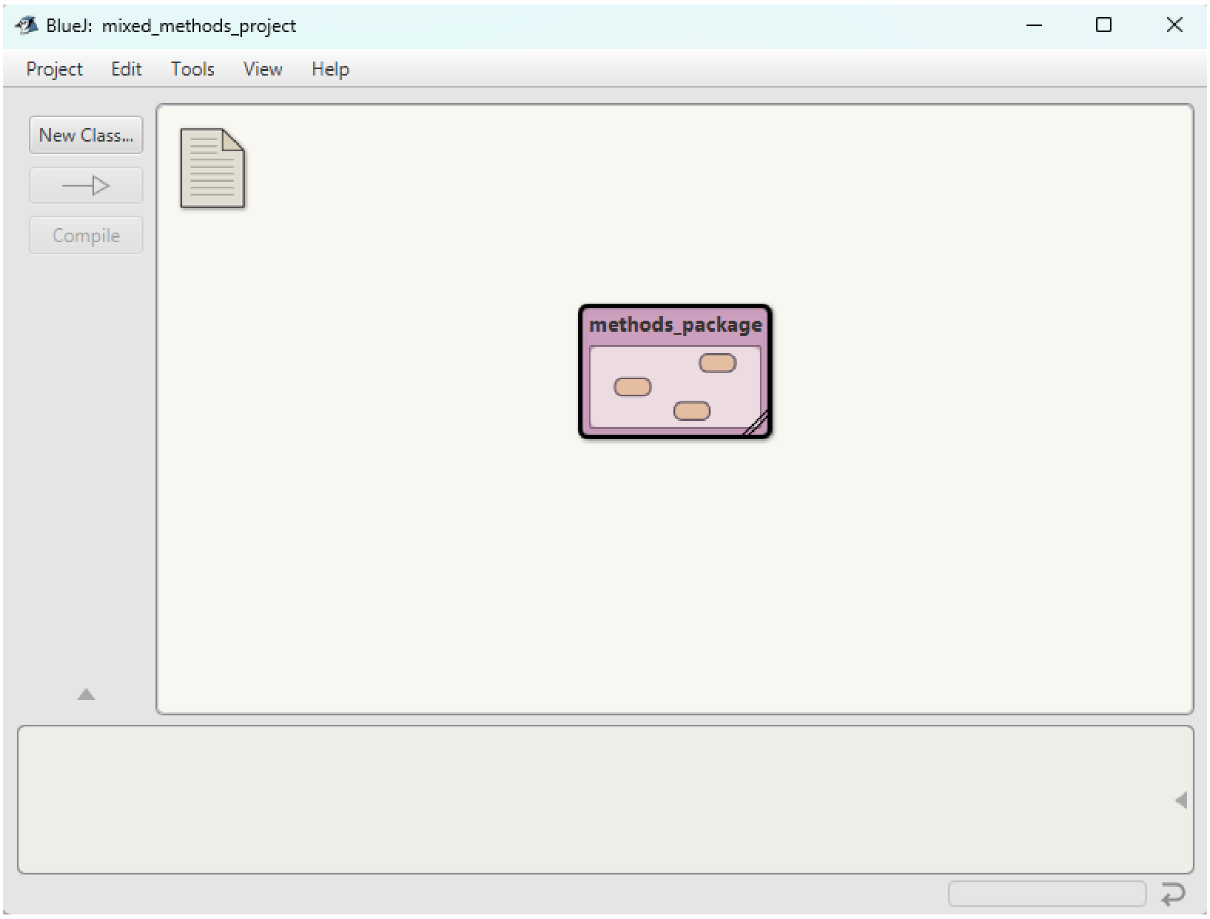
```
Enter the base of the triangle: 20
Enter the height of the triangle: 20
The area of the triangle is: 200.0
```

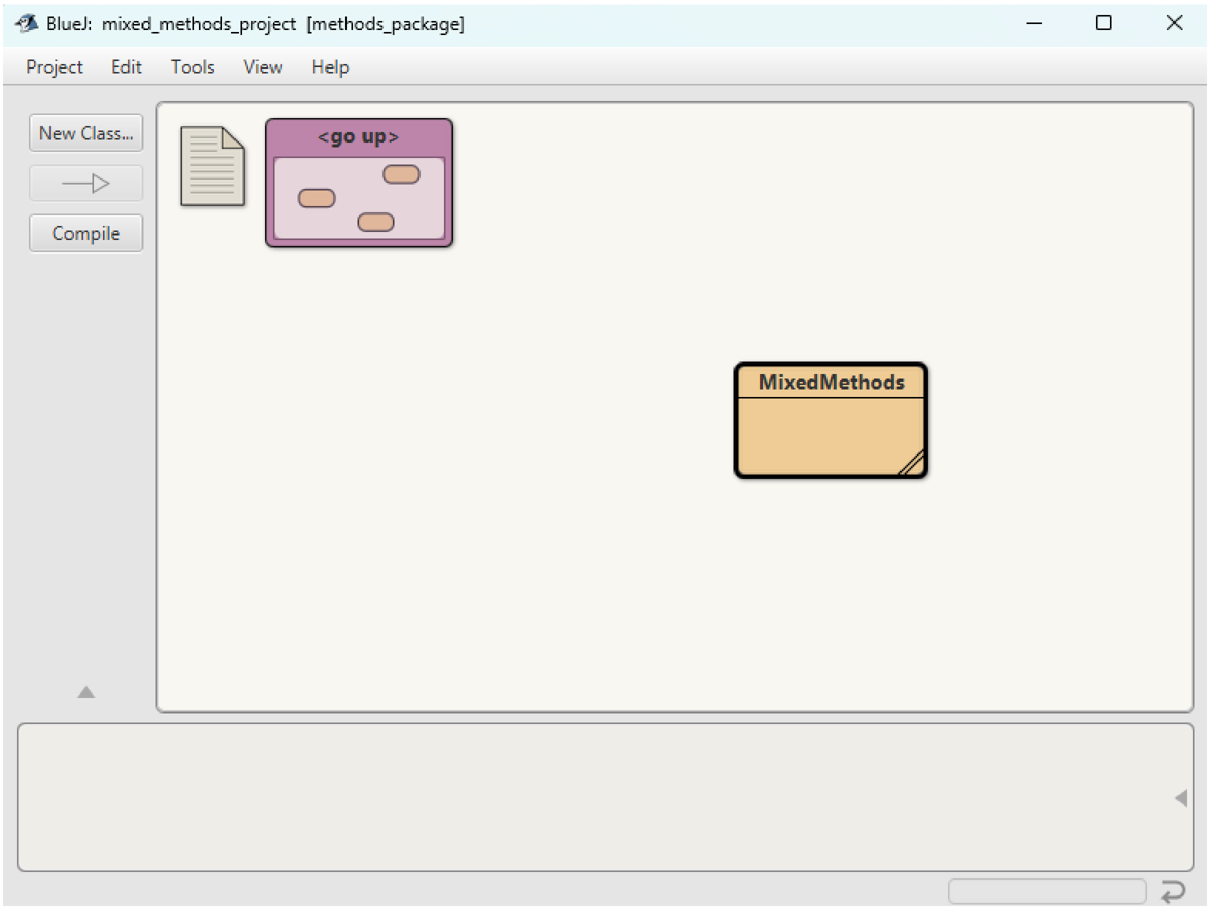
Can only enter input while your program is runni

```java
package triangle_package;

import java.util.Scanner;

public class Triangle {
    double base;
    double height;
    double area;

    // Method to ask the user to enter the base and height of a triangle
    public void inputDimensions() {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the base of the triangle: ");
        base = scanner.nextDouble();

        System.out.print("Enter the height of the triangle: ");
        height = scanner.nextDouble();
    }

    // Method to compute the area of the triangle
    public void computeArea() {
        area = 0.5 * base * height;
    }

    // Method to output the calculated area of the triangle
    public void displayArea() {
        System.out.println("The area of the triangle is: " + area);
    }

    // Main method to run the program
    public static void main(String[] args) {
        Triangle triangle = new Triangle();
        triangle.inputDimensions();
        triangle.computeArea();
        triangle.displayArea();
    }
}
```

QUESTION 4

Create a java program that has one non-static method, two static methods and a constructor.





```
package methods_package;


public class MixedMethods {
  // Instance variable
  private int number;


  // Constructor
  public MixedMethods(int number) {
    this.number = number;
  }
```

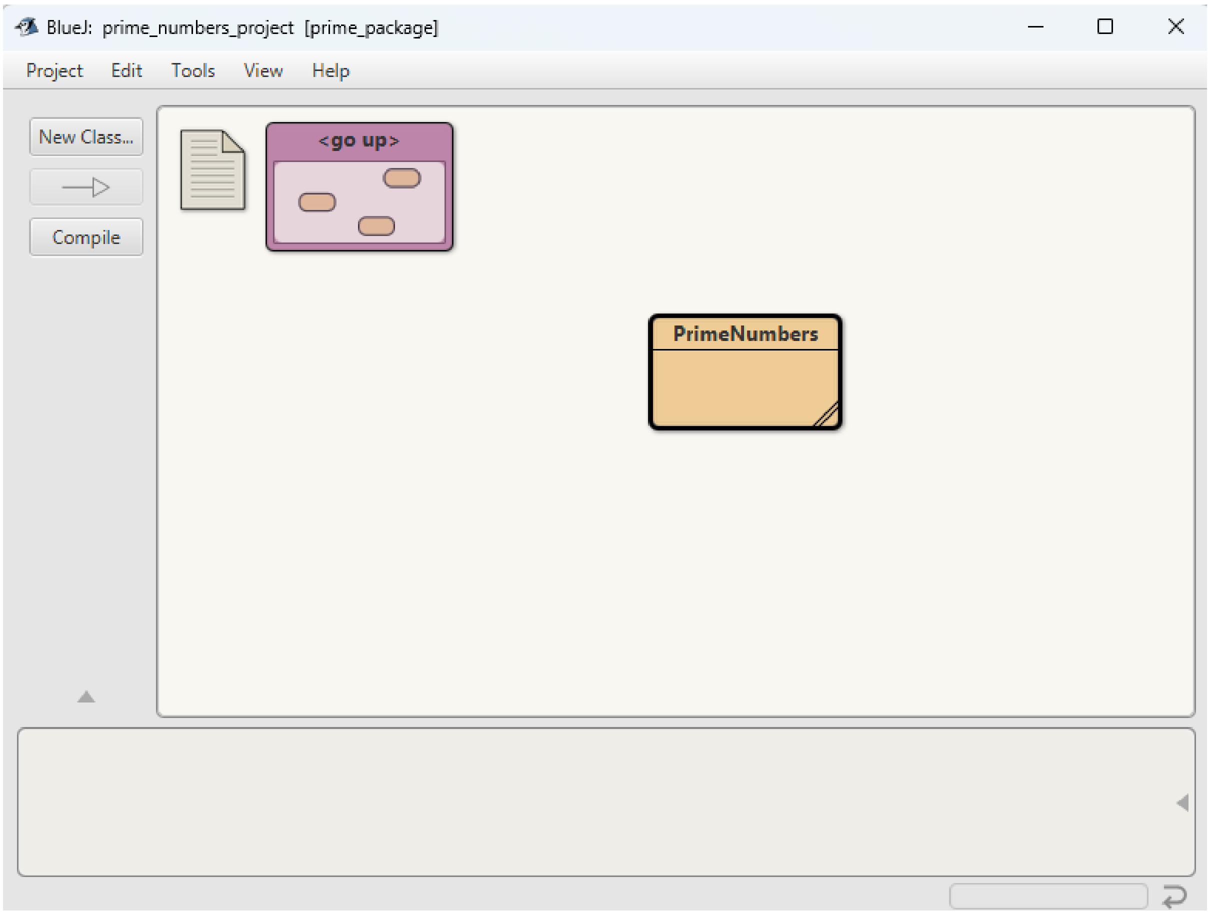
```
// Non-static method
public void displayNumber() {
    System.out.println("The number is: " + number);
}

// Static method to calculate the square of a number
public static int square(int num) {
    return num * num;
}

// Static method to calculate the cube of a number
public static int cube(int num) {
    return num * num * num;
}

// Main method
public static void main(String[] args) {
    // Create an instance of MixedMethods
    MixedMethods mixedMethods = new MixedMethods(5);

    // Call the non-static method
    mixedMethods.displayNumber();
```

```
        // Call the static methods
        int squareOfNumber = MixedMethods.square(5);
        int cubeOfNumber = MixedMethods.cube(5);

        // Display the results of static methods
        System.out.println("The square of 5 is: " + squareOfNumber);
        System.out.println("The cube of 5 is: " + cubeOfNumber);
    }
}
```

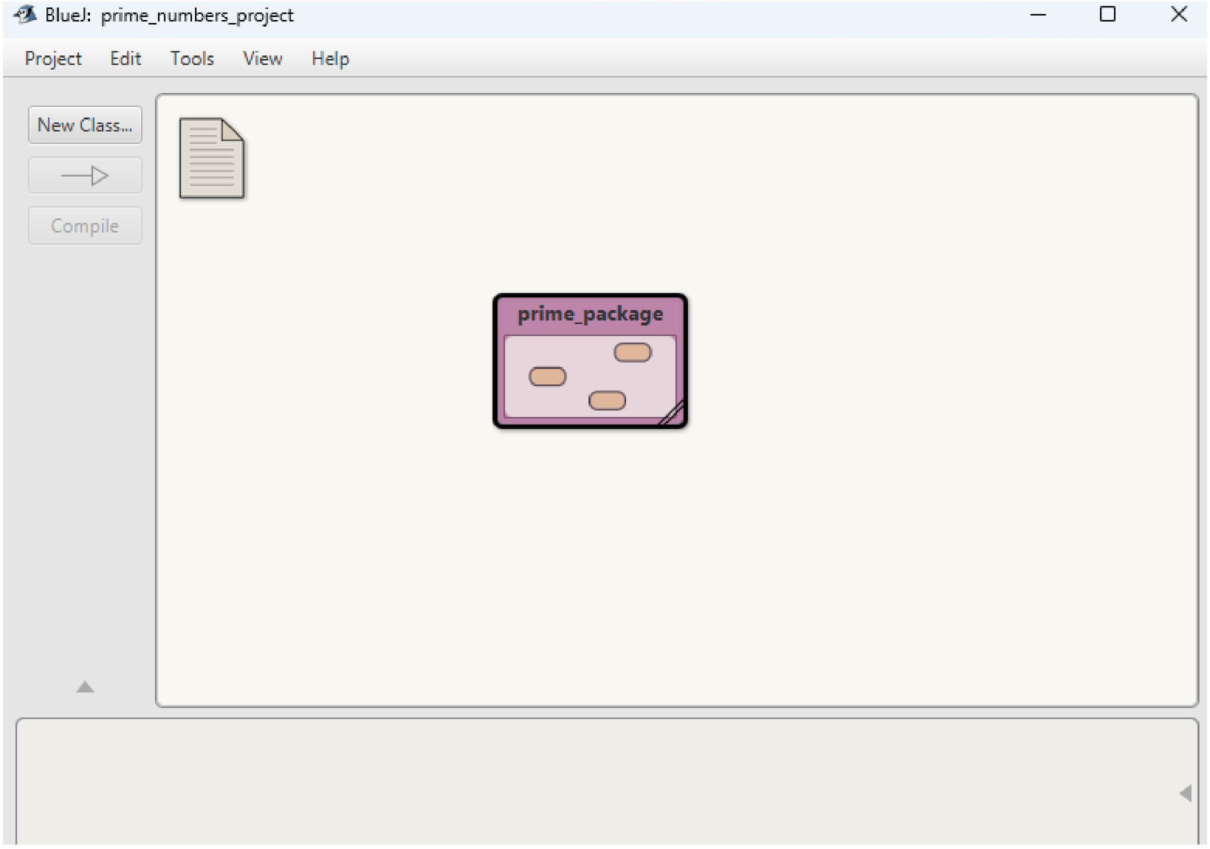
Question one: [15 marks]

a. A prime number is a number that is evenly divisible only by itself and 1. For example, the number 5 is prime because it can be evenly divided only by 1 and 5. The number 6, however, is not prime because it can be divided evenly by 1, 2, 4, and 6.
   Write a method named isPrime, which takes an integer as an argument and returns true if the argument is a prime number, or false otherwise. Also write main method that displays prime numbers between 1 to 500.

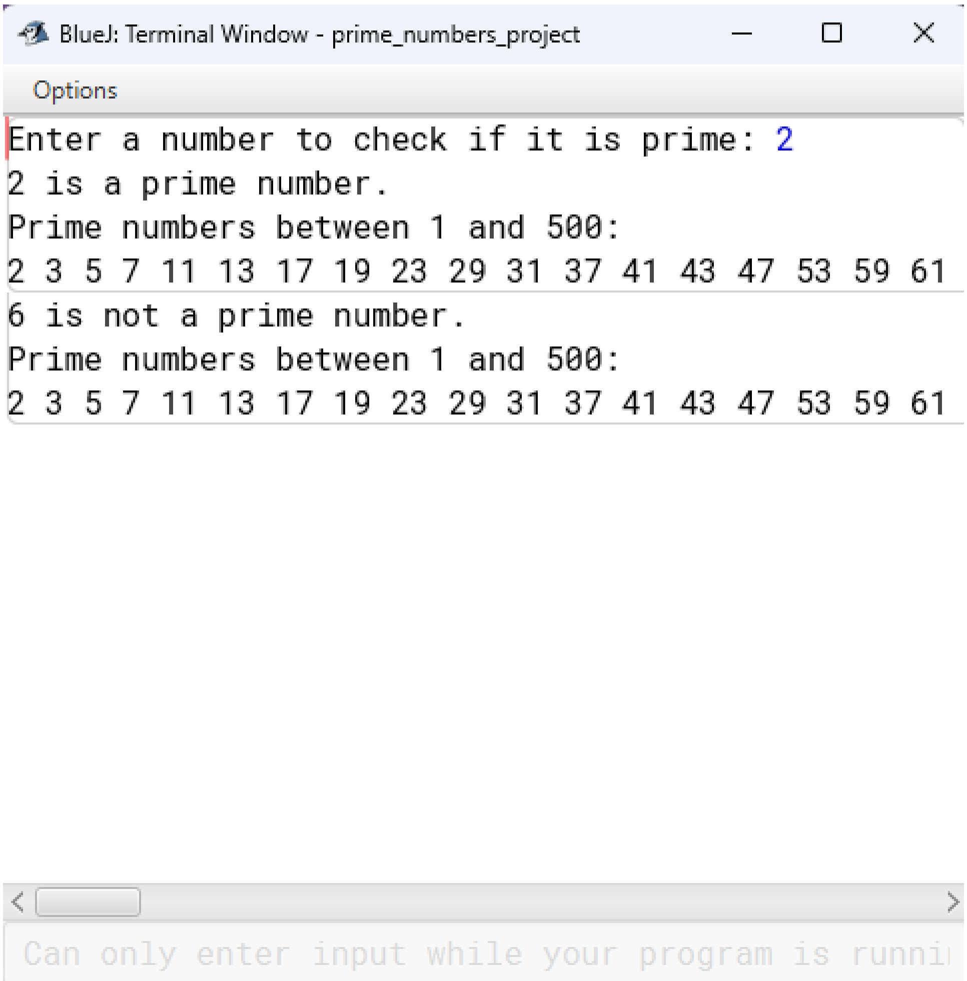
```
package prime_package;

import java.util.Scanner;

public class PrimeNumbers {
    // Method to check if a number is prime
    public static boolean isPrime(int number) {
        if (number <= 1) {
            return false;
        }
        for (int i = 2; i <= Math.sqrt(number); i++) {
            if (number % i == 0) {
                return false;
            }
        }
        return true;
    }

    // Main method
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Prompt user to enter a number and check if it's prime
        System.out.print("Enter a number to check if it is prime: ");
        int userNumber = scanner.nextInt();
        if (isPrime(userNumber)) {
            System.out.println(userNumber + " is a prime number.");
        } else {
            System.out.println(userNumber + " is not a prime number.");
        }

        // Display prime numbers between 1 and 500
        System.out.println("Prime numbers between 1 and 500:");
        for (int i = 1; i <= 500; i++) {
            if (isPrime(i)) {
                System.out.print(i + " ");
            }
        }

        scanner.close();
    }
}
```



BlueJ: prime_numbers_project [prime_package]



BlueJ: prime_numbers_project
Project
Edit
Tools
View
Help
New Class...
Compile
prime_package

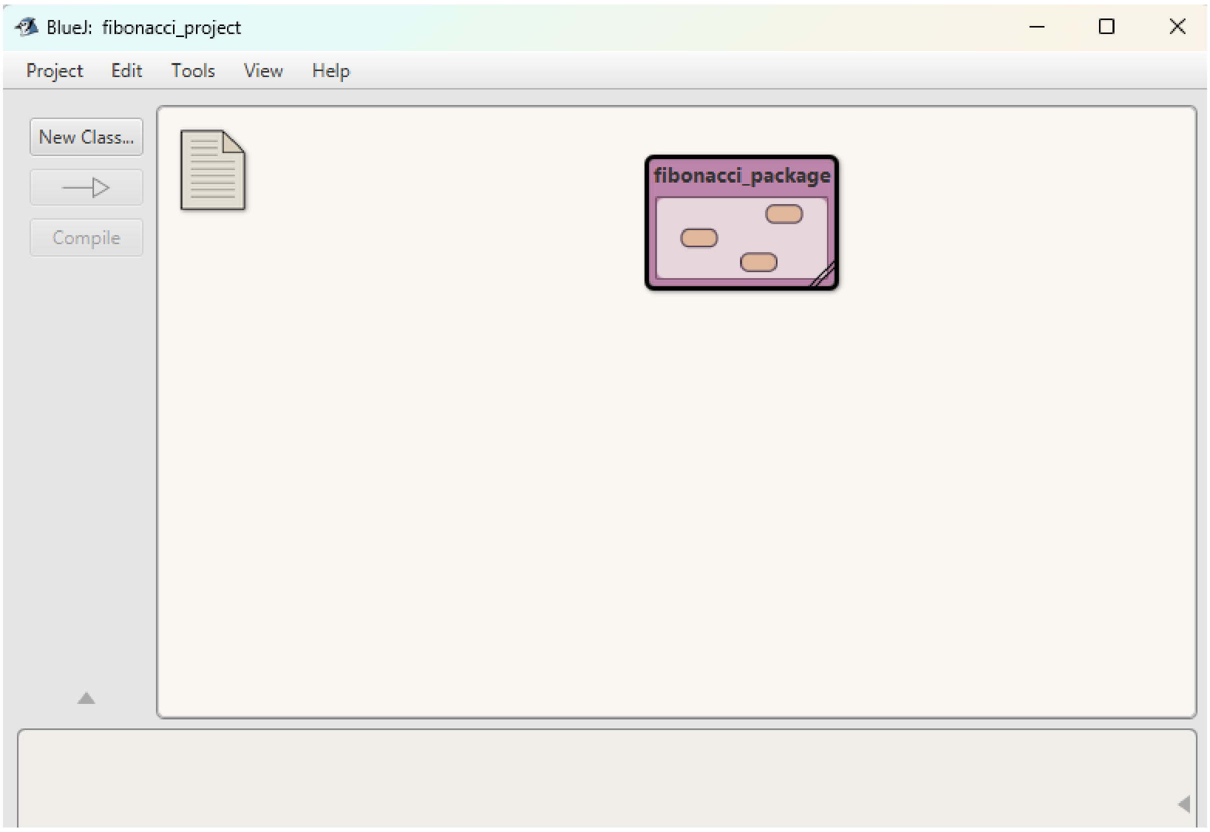

BlueJ: Terminal Window - prime_numbers_project
Options

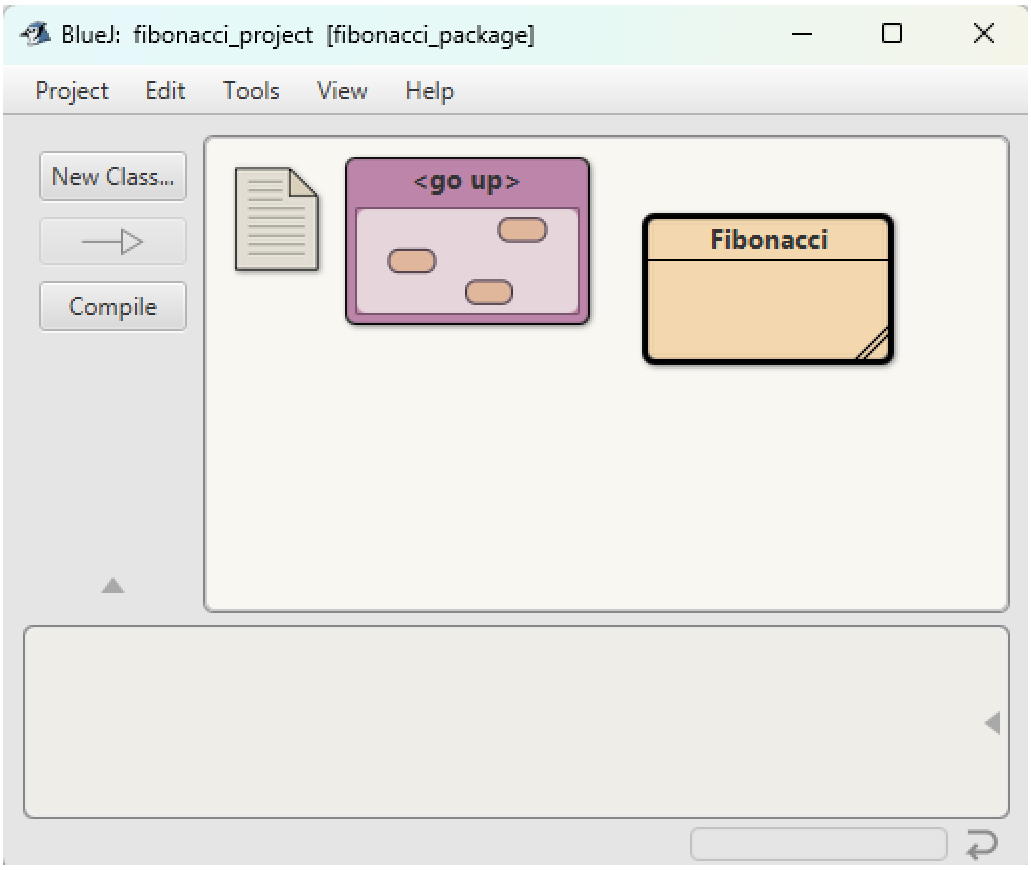
```
Enter a number to check if it is prime: 2
2 is a prime number.
Prime numbers between 1 and 500:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61
6 is not a prime number.
Prime numbers between 1 and 500:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61
```

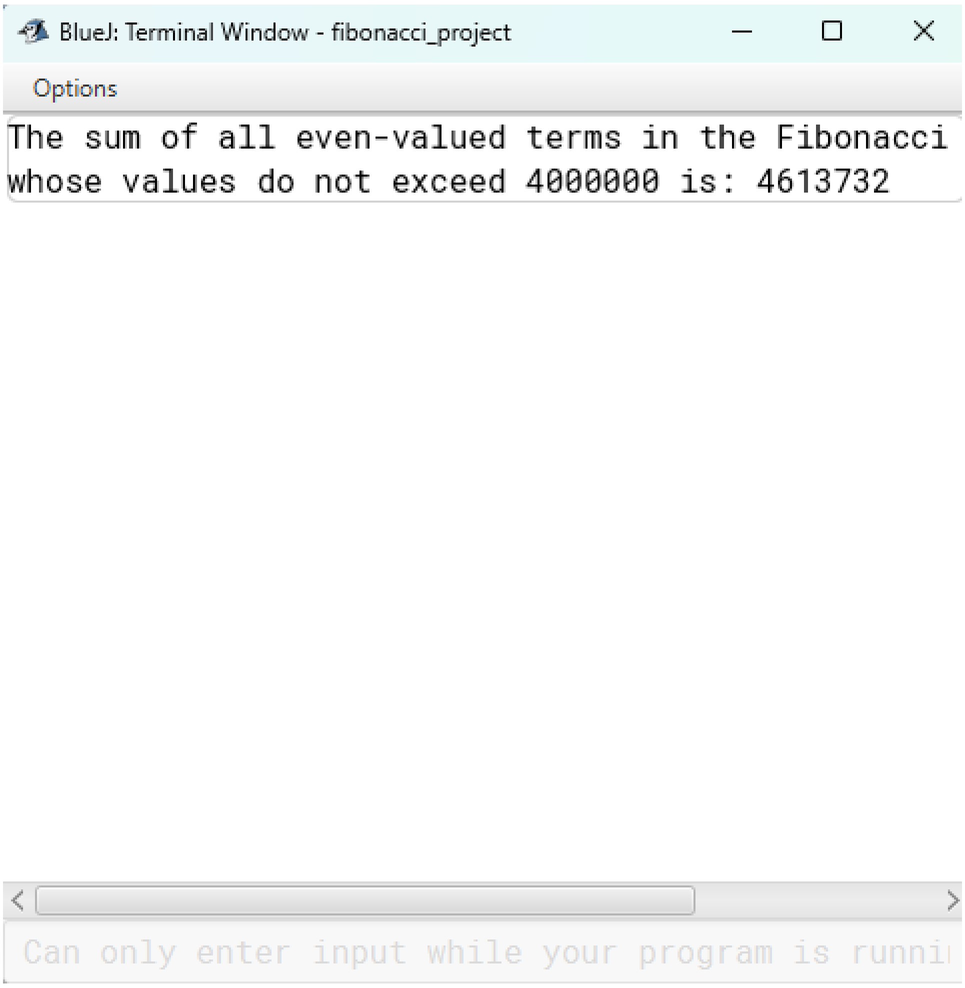
Can only enter input while your program is runni

b. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

c. By considering the terms in the Fibonacci sequence whose values do not exceed four million, write a Java method to find the sum of all the even- valued terms.




BlueJ: fibonacci_project [fibonacci_package]
Project
Edit
Tools
View
Help
New Class...
Compile
<go up>
Fibonacci


BlueJ: Terminal Window - fibonacci_project
Options

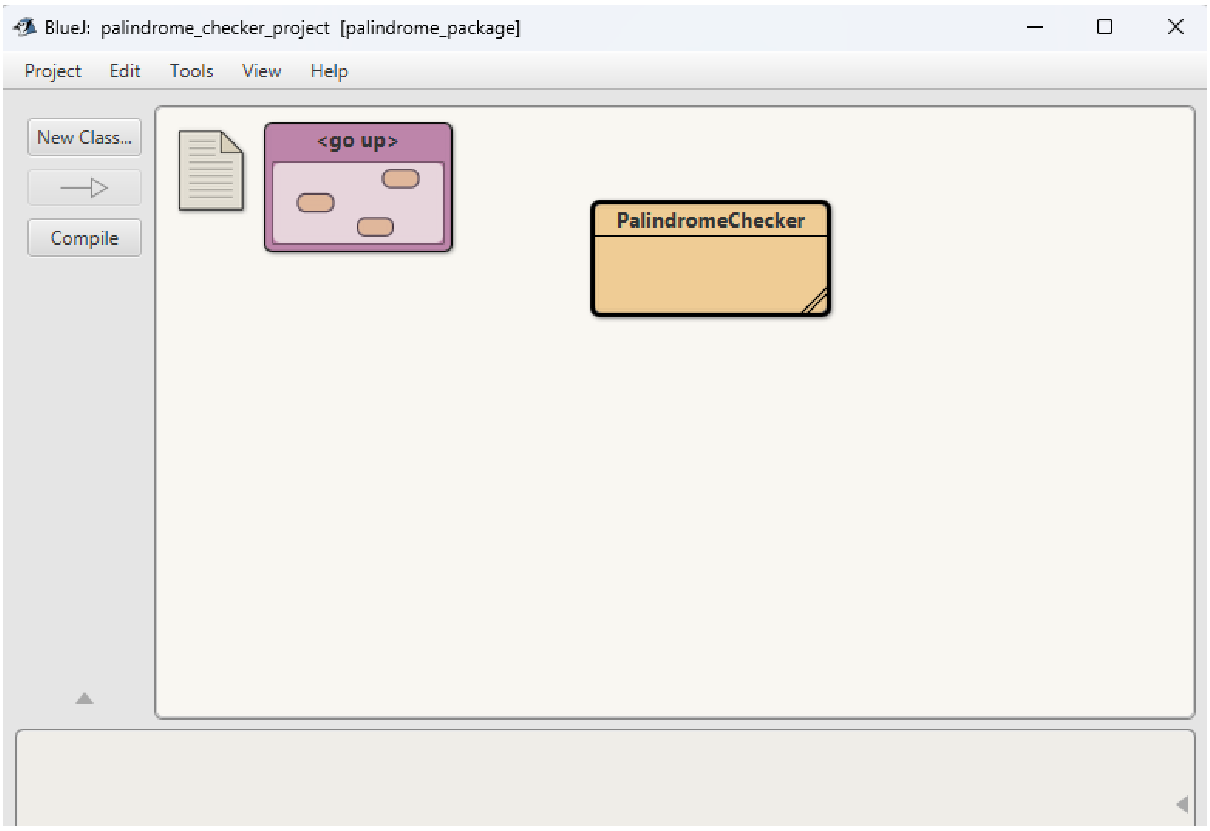
```
The sum of all even-valued terms in the Fibonacci
whose values do not exceed 4000000 is: 4613732
```

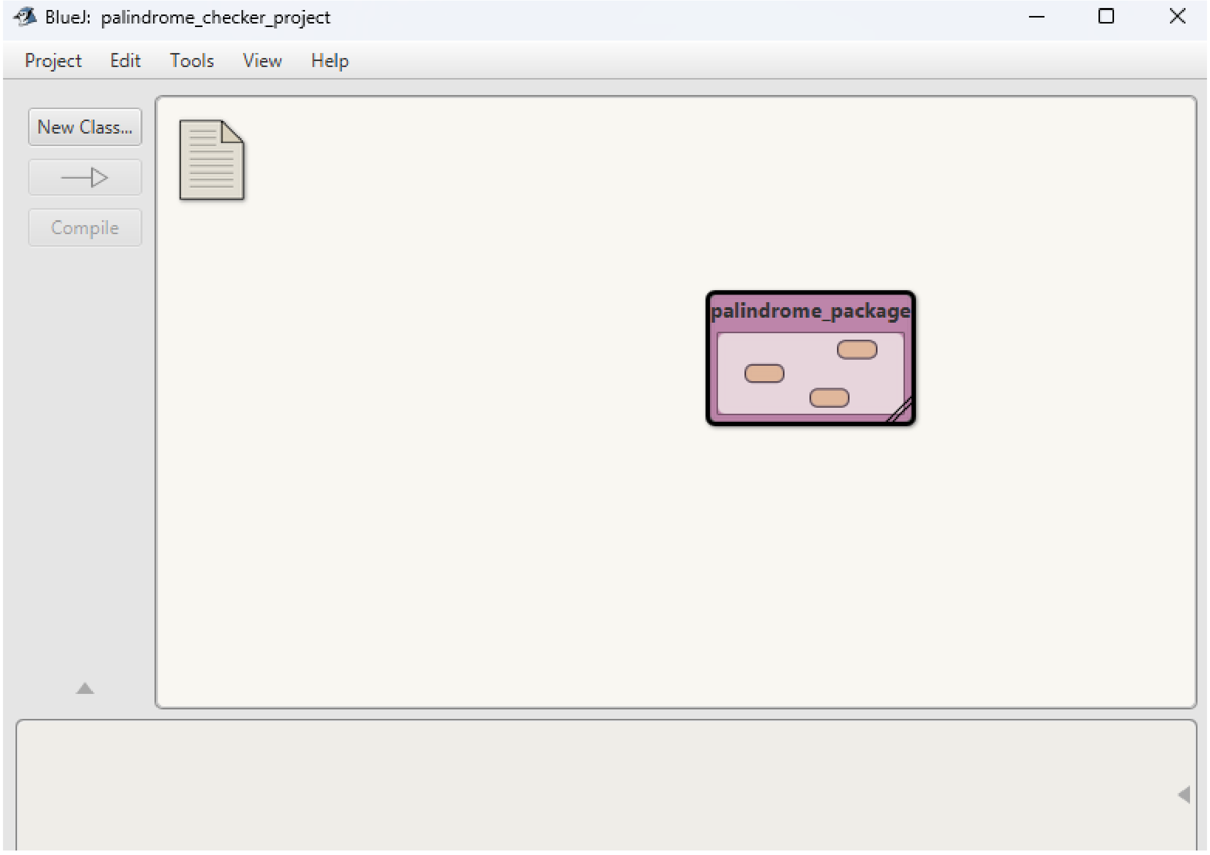
Can only enter input while your program is runni

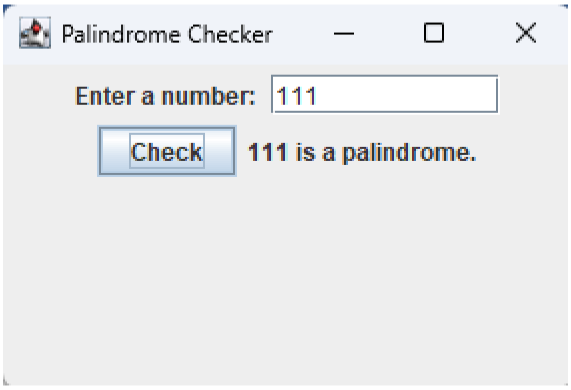
```
package fibonacci_package;


public class Fibonacci {

    // Method to calculate the sum of all even-valued terms in the Fibonacci sequence

    // whose values do not exceed four million

    public static int sumOfEvenFibonacci(int limit) {

        int sum = 0;

        int a = 1; // First term

        int b = 2; // Second term
}

        while (b <= limit) {
            if (b % 2 == 0) {
                sum += b;
            }

            // Generate the next term in the Fibonacci sequence
            int nextTerm = a + b;
            a = b;
            b = nextTerm;
        }

        return sum;
    }

    // Main method
    public static void main(String[] args) {
        int limit = 4000000;
        int sum = sumOfEvenFibonacci(limit);

        System.out.println("The sum of all even-valued terms in the Fibonacci sequence");
        System.out.println("whose values do not exceed " + limit + " is: " + sum);
    }
}
```

Question two: [15 marks]

A palindrome number is a number that remain the same when read from behind or front (a number that is equal to reverse of number) for example, 353 is palindrome because reverse of 353 is 353 (you see the number remains the same). But a number like 591 is not palindrome because reverse of 591 is 195 which is not equal to 591. Write Java program to check if a number entered by the user is palindrome or not. You should provide the user with a GUI interface to enter the number and display the results on the same interface.

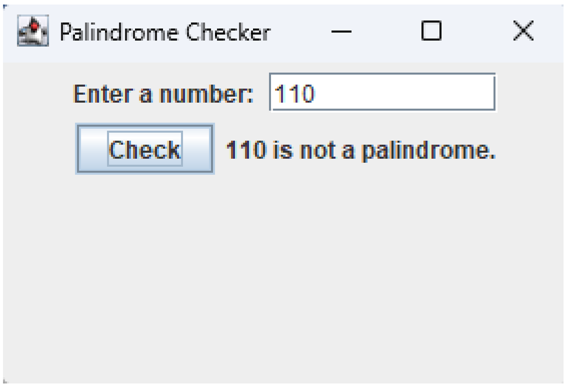






Enter a number: 111

111 is a palindrome.



Enter a number: 110

110 is not a palindrome.

```java
package palindrome_package;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class PalindromeChecker extends JFrame implements ActionListener {
    private JTextField numberField;
    private JLabel resultLabel;

    // Constructor to set up the GUI
    public PalindromeChecker() {
        setTitle("Palindrome Checker");
        setSize(300, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());

        JLabel promptLabel = new JLabel("Enter a number: ");
        add(promptLabel);

        numberField = new JTextField(10);
        add(numberField);
        JButton checkButton = new JButton("Check");
        checkButton.addActionListener(this);
        add(checkButton);

        resultLabel = new JLabel("");
        add(resultLabel);
    }

    // Method to check if a number is a palindrome
    public boolean isPalindrome(int number) {
        int originalNumber = number;
        int reverse = 0;
        while (number != 0) {
            int digit = number % 10;
            reverse = reverse * 10 + digit;
            number /= 10;
        }
        return originalNumber == reverse;
    }

    // Action event handler
    public void actionPerformed(ActionEvent e) {
        try {
            int number = Integer.parseInt(numberField.getText());

            if (isPalindrome(number)) {

                resultLabel.setText(number + " is a palindrome.");

            } else {

                resultLabel.setText(number + " is not a palindrome.");

            }

        } catch (NumberFormatException ex) {

            resultLabel.setText("Please enter a valid number.");

        }

    }


    // Main method to run the program

    public static void main(String[] args) {

        SwingUtilities.invokeLater(new Runnable() {

            public void run() {

                new PalindromeChecker().setVisible(true);

            }

        });

    }

}
```

Question three: [15 marks]

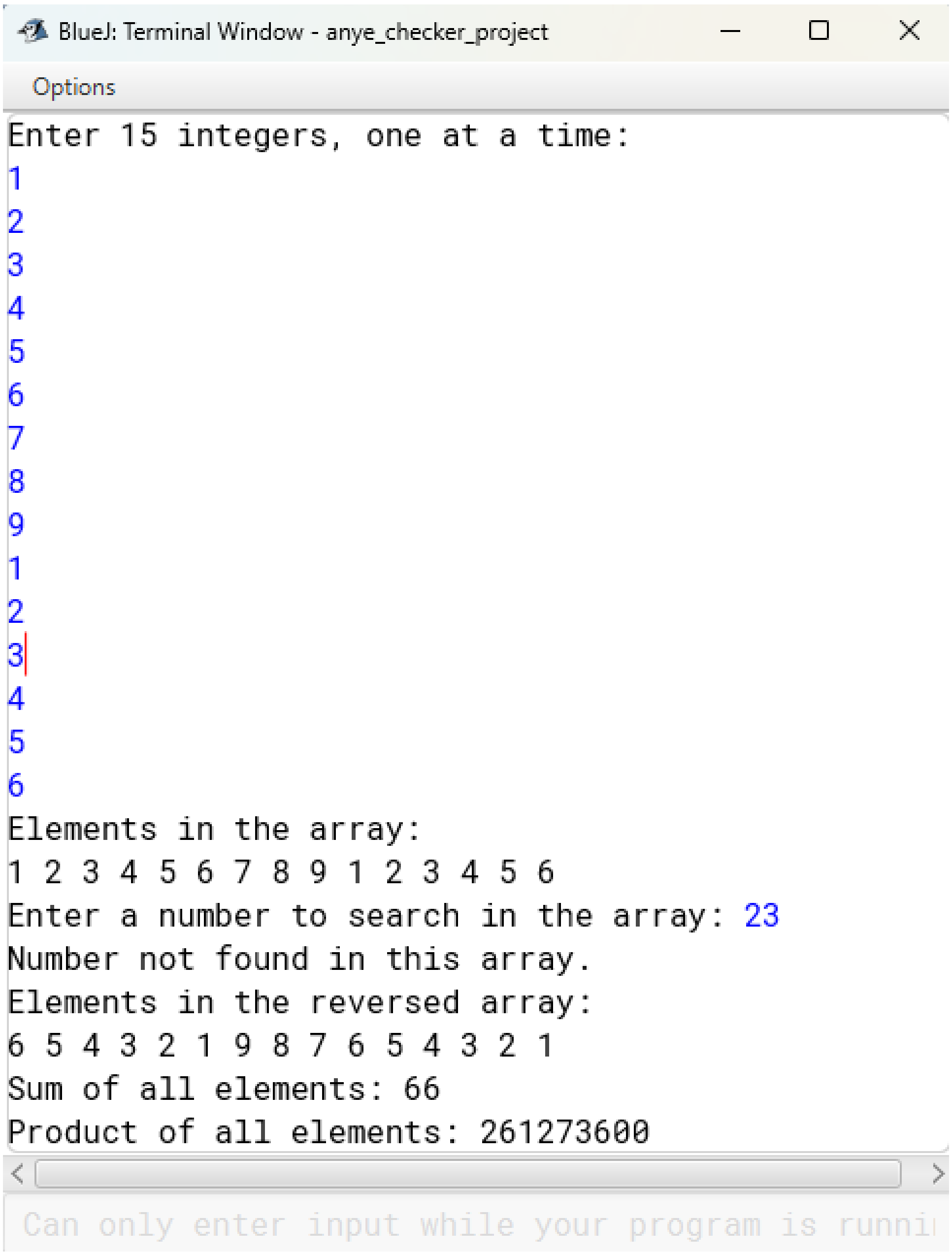
Write a Java program that takes 15 values of type integer as inputs from user, store the values in an array.

a) Print the values stored in the array on screen.
b) Ask user to enter a number, check if that number (entered by user) is present in array

or not. If it is present print, “the number found at index (index of the number) ” and the text “number not found in this array”

c) Create another array, copy all the elements from the existing array to the new array but in reverse order. Now print the elements of the new array on the screen

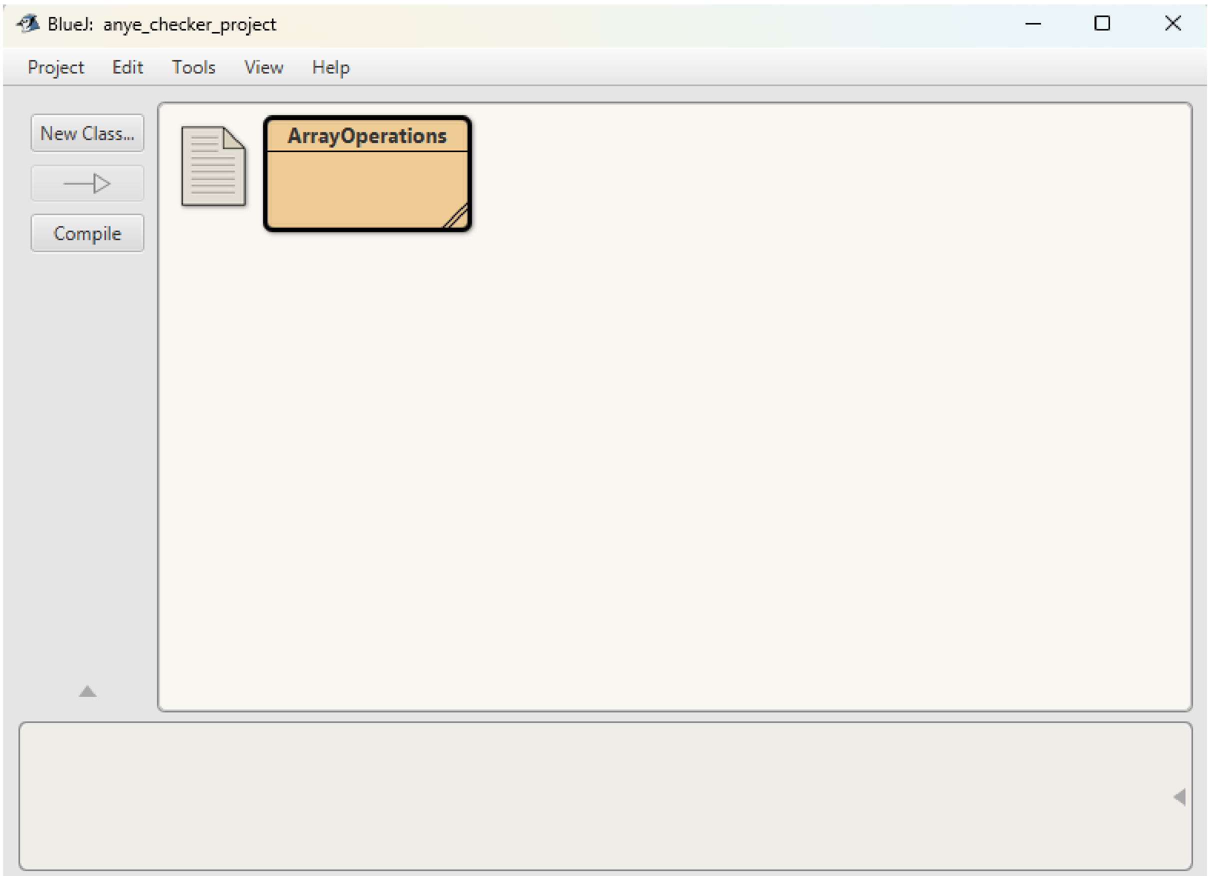
d) Get the sum and product of all elements of your array. Print product and the sum each on its own line.



BlueJ: Terminal Window - anye_checker_project
Options

```
Enter 15 integers, one at a time:
1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
Elements in the array:
1 2 3 4 5 6 7 8 9 1 2 3 4 5 6
Enter a number to search in the array: 23
Number not found in this array.
Elements in the reversed array:
6 5 4 3 2 1 9 8 7 6 5 4 3 2 1
Sum of all elements: 66
Product of all elements: 261273600
```

Can only enter input while your program is runni



```
import java.util.Scanner;

public class ArrayOperations {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] numbers = new int[15]; // Array to store 15 integers

        // Prompt user to enter 15 integers
        System.out.println("Enter 15 integers, one at a time:");

        // Input loop to populate the array
        for (int i = 0; i < 15; i++) {
            numbers[i] = scanner.nextInt();
        }

        // Print elements stored in the array
        System.out.println("Elements in the array:");
        for (int i = 0; i < 15; i++) {
            System.out.print(numbers[i] + " ");
        }
        System.out.println();

        // Ask user to enter a number to search
        System.out.print("Enter a number to search in the array: ");
        int searchNumber = scanner.nextInt();
    boolean found = false;
    int index = -1;

    // Search for the number in the array
    for (int i = 0; i < 15; i++) {
        if (numbers[i] == searchNumber) {
            found = true;
            index = i;
            break;
        }
    }

    // Print result based on search
    if (found) {
        System.out.println("The number found at index " + index);
    } else {
        System.out.println("Number not found in this array.");
    }

    // Create a new array to store elements in reverse order
    int[] reversedArray = new int[15];

    // Copy elements from original array to reversed array in reverse order
    for (int i = 0; i < 15; i++) {
        reversedArray[i] = numbers[14 - i];
    }

    // Print elements of the reversed array
    System.out.println("Elements in the reversed array:");
    for (int i = 0; i < 15; i++) {
        System.out.print(reversedArray[i] + " ");
    }
    System.out.println();

    // Calculate sum and product of elements in the original array
    int sum = 0;
    int product = 1;

    for (int i = 0; i < 15; i++) {
        sum += numbers[i];
        product *= numbers[i];
    }

    // Print sum and product
    System.out.println("Sum of all elements: " + sum);
    System.out.println("Product of all elements: " + product);
}
```

```
}
```