South Louisiana Community College
ASDV 1220, Programming Fundamentals
Lab Conditional If Statement

Learning Objectives

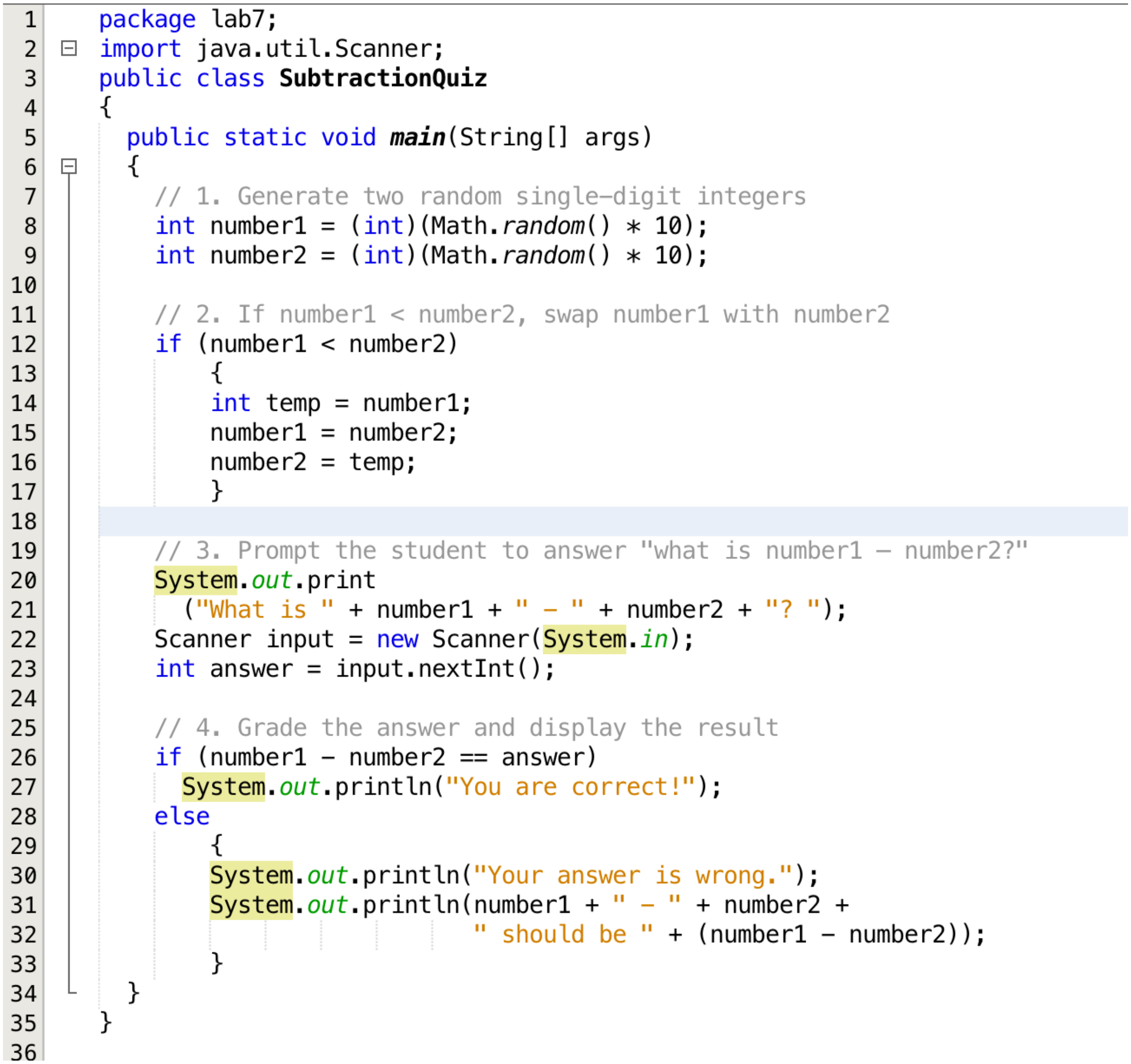
After completion of this lab, you should be able to

1. Understand the **if** construction and usage
2. Understand the **if-else** construction and usage
3. Understand the logical operators && AND, || OR and ! NOT
4. Understand DeMorgan's Law
5. Practice with the String class to read text from the console

Create project Lab7

Problem 1

Create a class **SubtractionQuiz**, write the code as shown below and run it multiple times. The steps are explained in the comments. Read them please. Understand how the program works and how you can swap 2 variables using a temporary variable to do it. Compile and run the class many times.

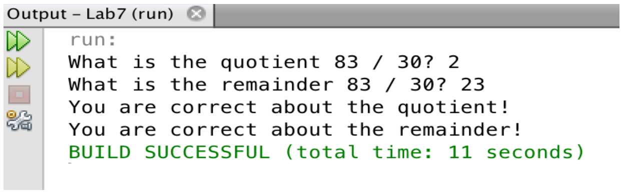


```
package lab7;
import java.util.Scanner;
public class SubtractionQuiz
{
  public static void main(String[] args)
  {
    // 1. Generate two random single-digit integers
    int number1 = (int)(Math.random() * 10);
    int number2 = (int)(Math.random() * 10);

    // 2. If number1 < number2, swap number1 with number2
    if (number1 < number2)
        {
        int temp = number1;
        number1 = number2;
        number2 = temp;
        }

    // 3. Prompt the student to answer "what is number1 – number2?"
    System.out.print
      ("What is " + number1 + " - " + number2 + "? ");
    Scanner input = new Scanner(System.in);
    int answer = input.nextInt();

    // 4. Grade the answer and display the result
    if (number1 - number2 == answer)
      System.out.println("You are correct!");
    else
        {
        System.out.println("Your answer is wrong.");
        System.out.println(number1 + " - " + number2 +
                           " should be " + (number1 - number2));
        }
  }
}
```

Problem 2

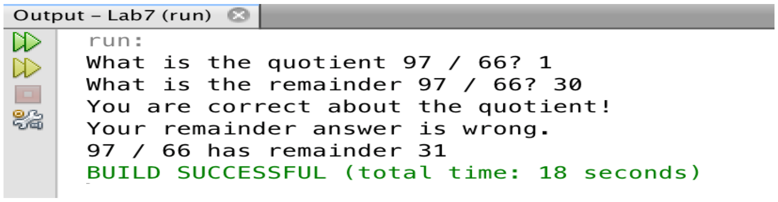
Create a class called **DivisionQuiz**. Similarly to SubtractionQuiz, the DivisionQuiz teaches the pupil how to calculate the quotient and the remainder of a division of 2 numbers in the range 1 to 100 inclusively. Generate the numbers using the Math.random.

Sample run 1



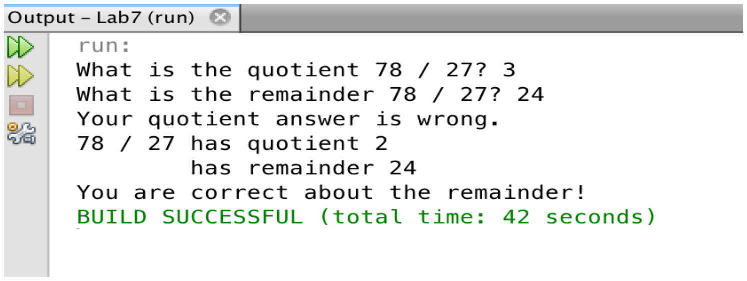
```
run:
What is the quotient 83 / 30? 2
What is the remainder 83 / 30? 23
You are correct about the quotient!
You are correct about the remainder!
BUILD SUCCESSFUL (total time: 11 seconds)
```

Sample run 2



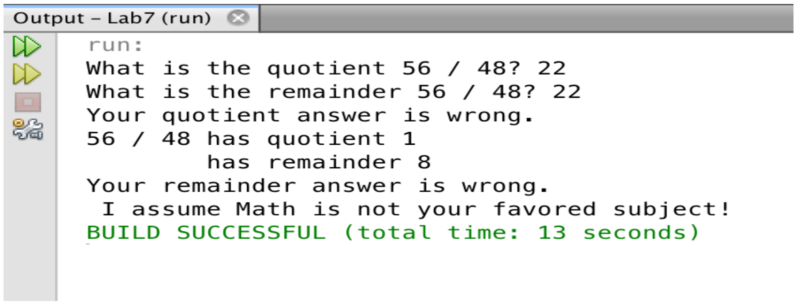
```
run:
What is the quotient 97 / 66? 1
What is the remainder 97 / 66? 30
You are correct about the quotient!
Your remainder answer is wrong.
97 / 66 has remainder 31
BUILD SUCCESSFUL (total time: 18 seconds)
```

Sample run 3



```
run:
What is the quotient 78 / 27? 3
What is the remainder 78 / 27? 24
Your quotient answer is wrong.
78 / 27 has quotient 2
        has remainder 24
You are correct about the remainder!
BUILD SUCCESSFUL (total time: 42 seconds)
```

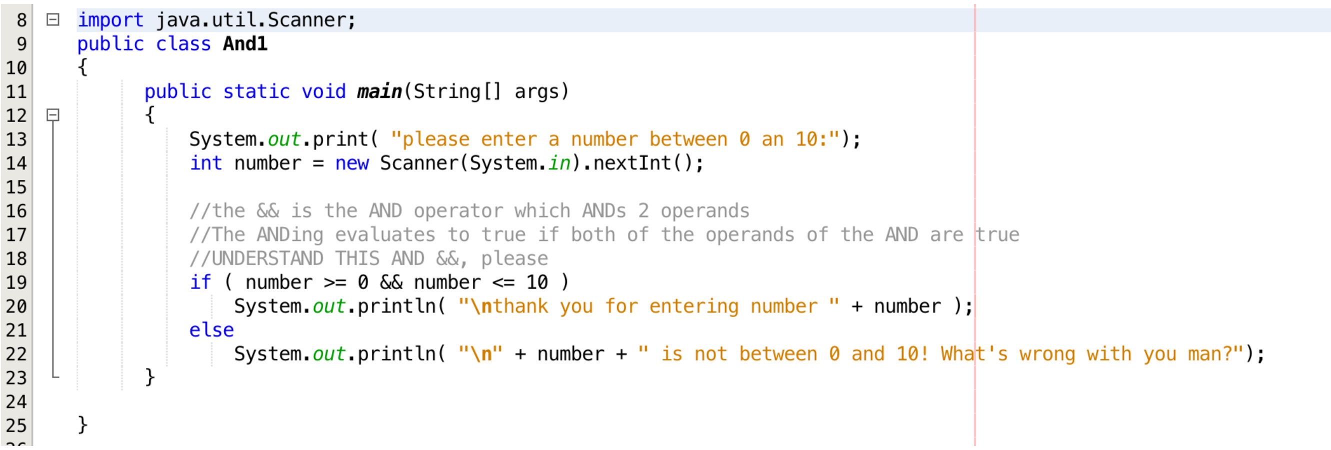
Sample run 4



```
run:
What is the quotient 56 / 48? 22
What is the remainder 56 / 48? 22
Your quotient answer is wrong.
56 / 48 has quotient 1
        has remainder 8
Your remainder answer is wrong.
 I assume Math is not your favored subject!
BUILD SUCCESSFUL (total time: 13 seconds)
```

Problem 3

Create a class called **And1**. Type in the code as given below. Compile and run the class. Observe line 19. Inside the **if-condition**, there is the **&& operator** which ANDs 2 operands. The evaluation of the ANDing is **true** if both of the operands evaluate to true. Otherwise the ANDing evaluates to **false**.

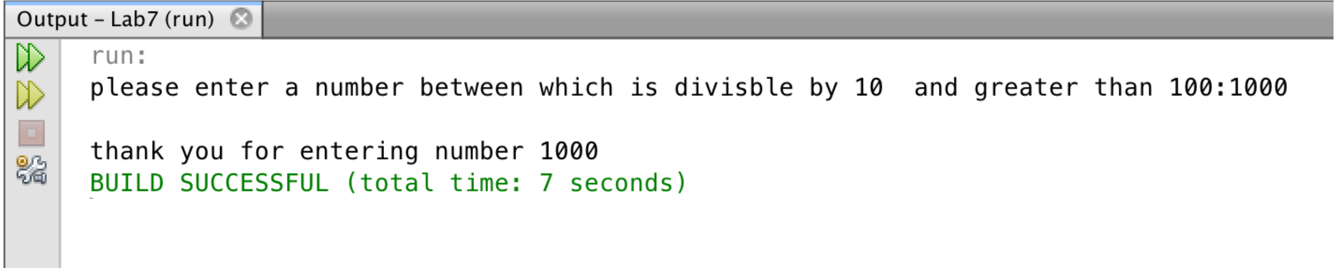


```
 8  import java.util.Scanner;
 9  public class And1
10  {
11      public static void main(String[] args)
12      {
13          System.out.print( "please enter a number between 0 an 10:");
14          int number = new Scanner(System.in).nextInt();
15
16          //the && is the AND operator which ANDs 2 operands
17          //The ANDing evaluates to true if both of the operands of the AND are true
18          //UNDERSTAND THIS AND &&, please
19          if ( number >= 0 && number <= 10 )
20              System.out.println( "\nthank you for entering number " + number );
21          else
22              System.out.println( "\n" + number + " is not between 0 and 10! What's wrong with you man?");
23      }
24
25  }
```

Problem 4

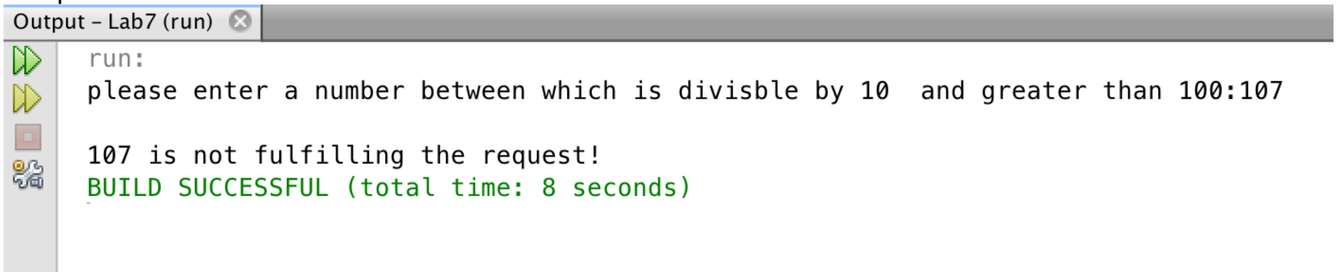
Create a class called **And2**. Ask the user to enter a number greater than 100 and divisible by 10. Display the proper message depending on the input. Compile and run the class multiple times.

Sample run 1



```
run:
please enter a number between which is divisble by 10  and greater than 100:1000

thank you for entering number 1000
BUILD SUCCESSFUL (total time: 7 seconds)
```

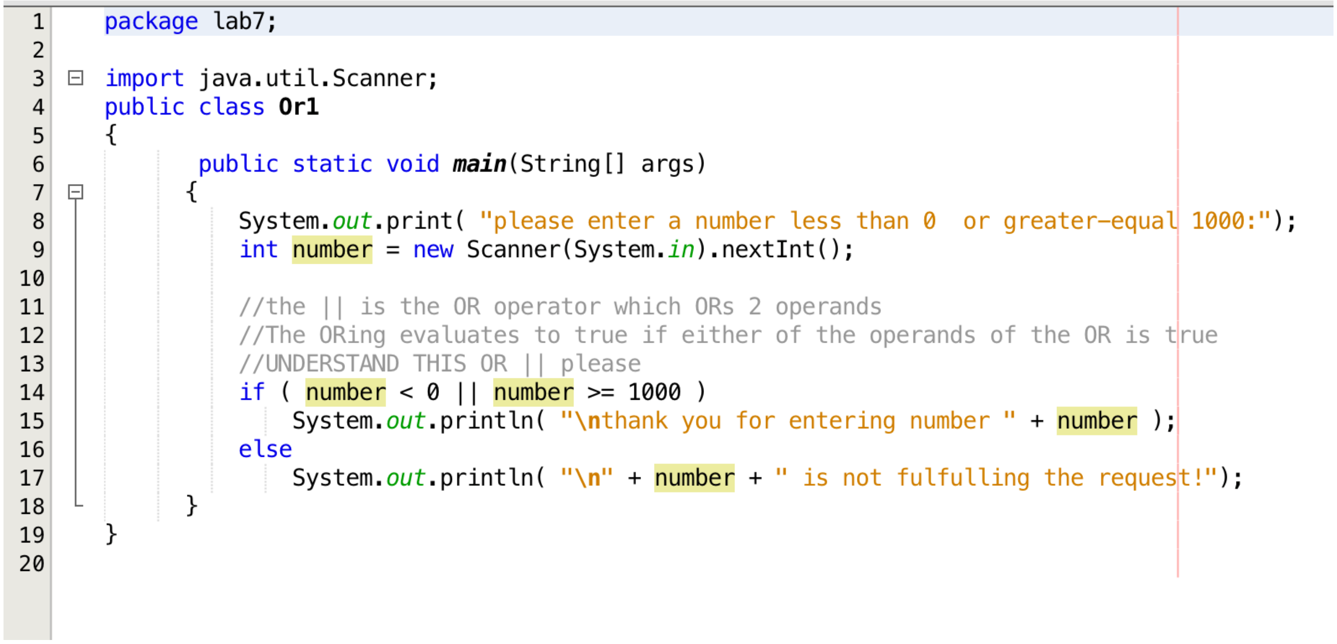
Sample run 2



```
run:
please enter a number between which is divisble by 10  and greater than 100:107

107 is not fulfilling the request!
BUILD SUCCESSFUL (total time: 8 seconds)
```

Problem 5

Create a class called **Or1**. Type in the code as given below. The **OR ||** operator evaluates to true if at least one of its two operands evaluates to true . Compile and run the class multiple times for different typed numbers.

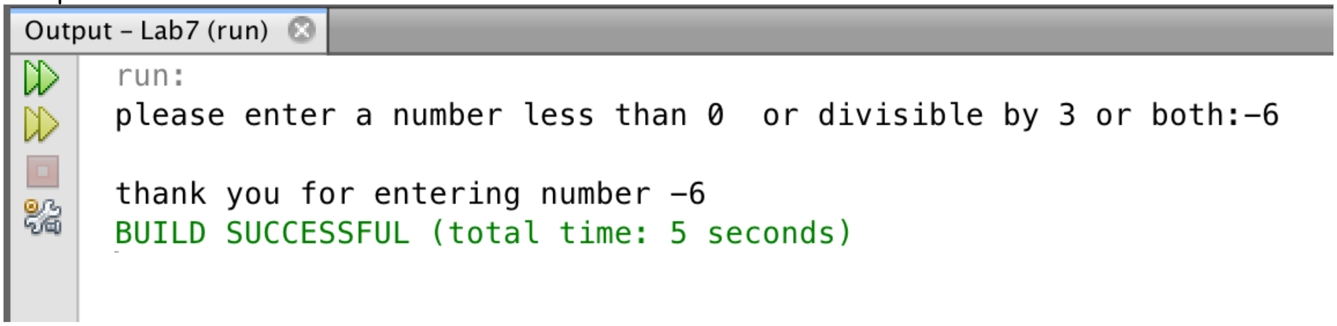


```
package lab7;

import java.util.Scanner;
public class Or1
{
       public static void main(String[] args)
      {
          System.out.print( "please enter a number less than 0  or greater-equal 1000:");
          int number = new Scanner(System.in).nextInt();

          //the || is the OR operator which ORs 2 operands
          //The ORing evaluates to true if either of the operands of the OR is true
          //UNDERSTAND THIS OR || please
          if ( number < 0 || number >= 1000 )
              System.out.println( "\nthank you for entering number " + number );
          else
              System.out.println( "\n" + number + " is not fulfulling the request!");
      }
}
```

Problem 6

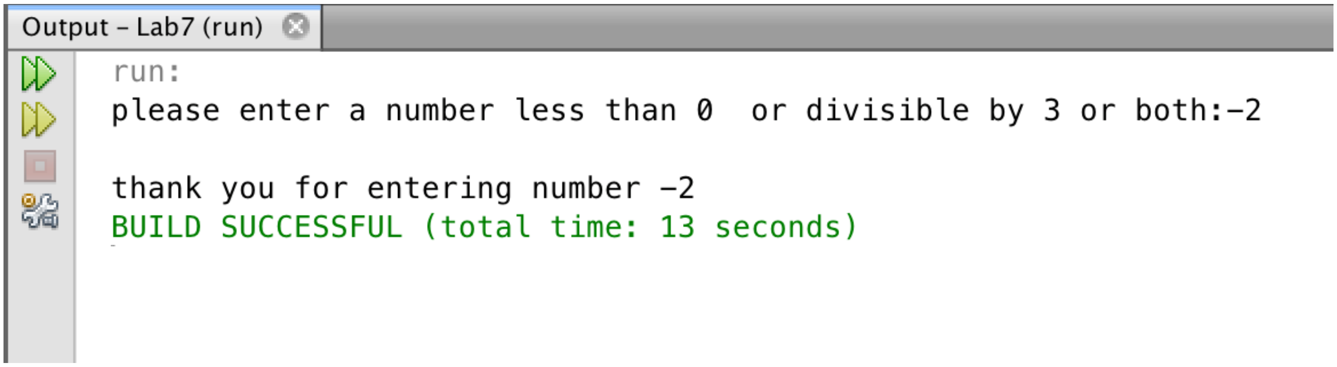
Create a class called **Or2**. Inside its main write code that prompts the user to type a number which is divisible by 3 or negative or both. Display the proper message.

Sample run 1:



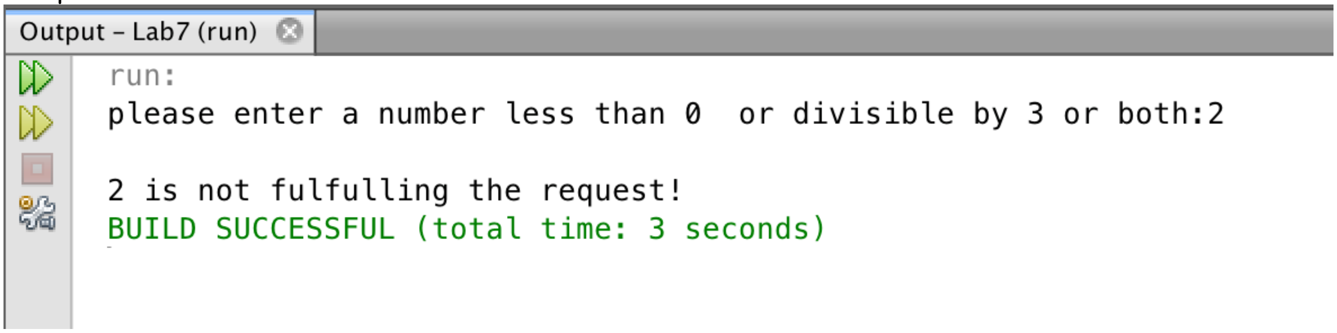
```
run:
please enter a number less than 0  or divisible by 3 or both:-6

thank you for entering number -6
BUILD SUCCESSFUL (total time: 5 seconds)
```

Sample run 2:



```
run:
please enter a number less than 0  or divisible by 3 or both:-2

thank you for entering number -2
BUILD SUCCESSFUL (total time: 13 seconds)
```

Sample run 3:



```
run:
please enter a number less than 0  or divisible by 3 or both:2

2 is not fulfulling the request!
BUILD SUCCESSFUL (total time: 3 seconds)
```

Problem 7

Create a class called Not1 Type in, the code as given below. Compile and run the class.

Run it multiple times and observe the different output depending on the input value. Understand how it works by running for different inputs.

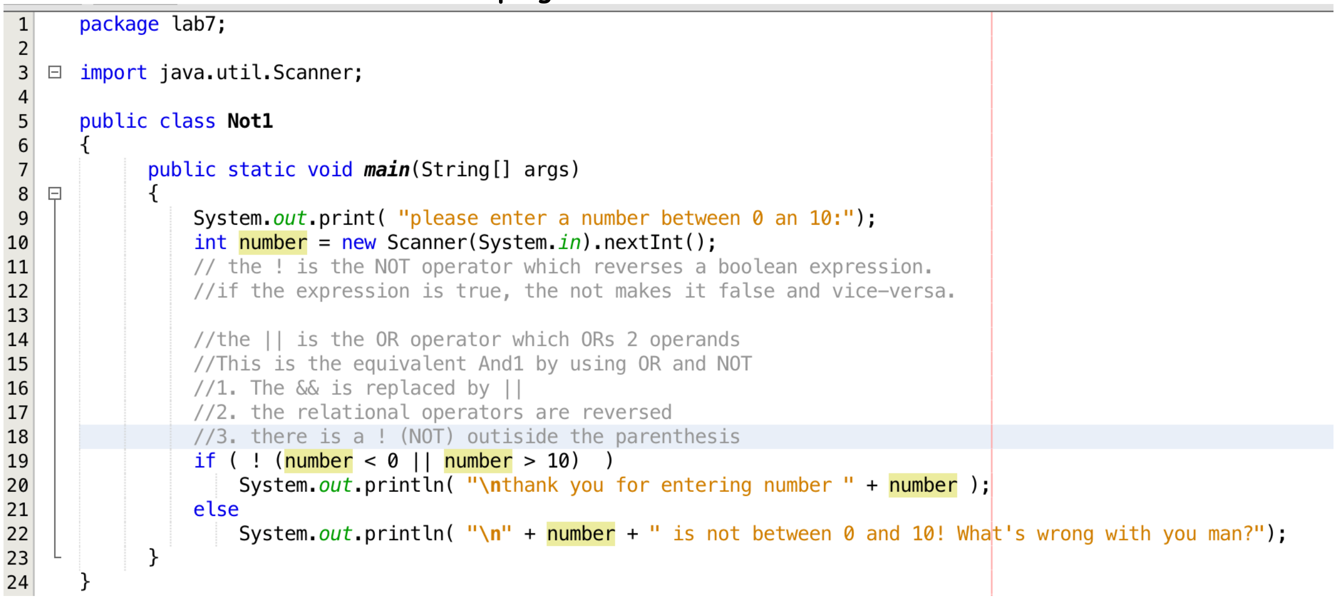
The code uses the **NOT !** operator. This is a **unary operator** meaning it takes one operand which is a boolean expression. The NOT ! operator reverses the boolean expression from true to false and vice versa.

The if-expression below in the code uses the **DeMorgan's Law** .OR || instead of AND && to have the same results.

Steps of DeMorgan's Law to convert AND to OR

1. the && is replaced by ||
2. the relational operators are reversed
3. there is a ! (NOT) outside the parenthesis

The results should be the same as in program And1



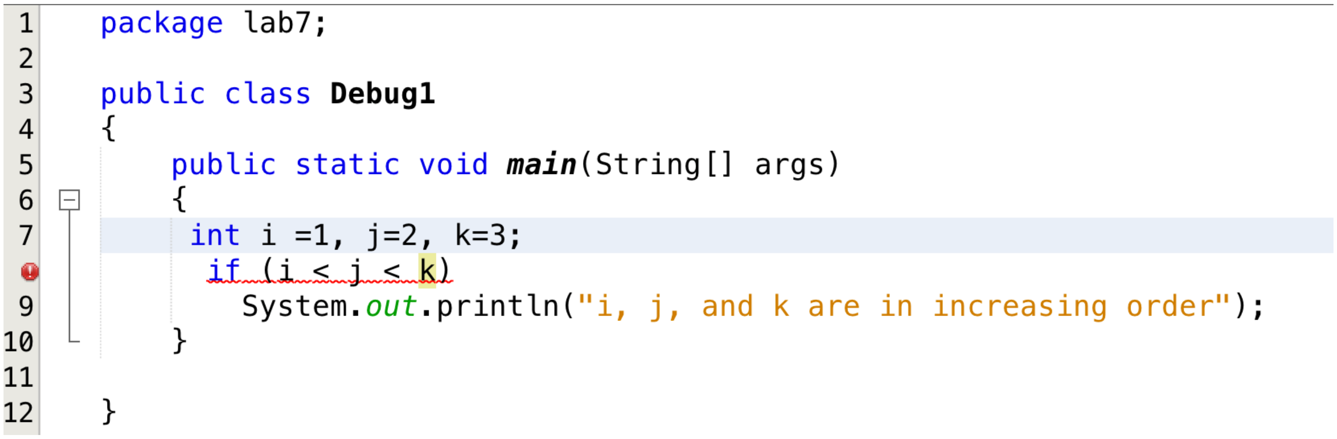
```java
package lab7;

import java.util.Scanner;

public class Not1
{
    public static void main(String[] args)
    {
        System.out.print( "please enter a number between 0 an 10:");
        int number = new Scanner(System.in).nextInt();
        // the ! is the NOT operator which reverses a boolean expression.
        //if the expression is true, the not makes it false and vice-versa.

        //the || is the OR operator which ORs 2 operands
        //This is the equivalent And1 by using OR and NOT
        //1. The && is replaced by ||
        //2. the relational operators are reversed
        //3. there is a ! (NOT) outiside the parenthesis
        if ( ! (number < 0 || number > 10)   )
            System.out.println( "\nthank you for entering number " + number );
        else
            System.out.println( "\n" + number + " is not between 0 and 10! What's wrong with you man?");
    }
}
```

Problem 8

Create a class called **Not2**. Inside the main of Not2, modify your existing program called And2 by use DeMorgan's Law as in problem 7. In other words use OR not AND. Your output should be the same is in And2.

Problem 9

Create a class called Debug1 and type the code below. Fix it that it works and prints the message.



```
package lab7;

public class Debug1
{
    public static void main(String[] args)
    {
     int i =1, j=2, k=3;
      if (i < j < k)
        System.out.println("i, j, and k are in increasing order");
    }

}
```

Problem 10
Create a class Bonus
a) Implement the class according to the description that follows.
b) Provide and upload a Word document with Test Plan for all the variables used in your program.

Diana Lee, a supervisor in a manufacturing company, wants to know which employees have increased their production this year over last year so that she can issue them certificates of commendation and bonuses.

The program accepts each worker's first and last names, this year's number of units produced, and last year's number of units produced. Displays the name and a bonus amount. The bonuses will be distributed as follows:

If this year's production is greater than last year's production and this year's production is:

- 1,000 units or fewer, the bonus is $25
- 1,001 to 3,000 units, the bonus is $50
- 3,001 to 6,000 units, the bonus is $100
- 6,001 units and up, the bonus is $200

Sample Runs

<u>Output #1</u>

```
Enter last name:
Nguyen
Enter first name:
John
Enter this year's units:
3000
Enter last year's units:
4000
Nguyen, John
Bonus is $0.0
```

<u>Output #2</u>

```
Enter last name:
Nguyen
Enter first name:
John
Enter this year's units:
900
Enter last year's units:
300
Nguyen, John
Bonus is $25.0
```

Output #3

```
Enter last name:
Nguyen
Enter first name:
John
Enter this year's units:
2500
Enter last year's units:
2000
Nguyen, John
Bonus is $50.0
```

Output #4

```
Enter last name:
Nguyen
Enter first name:
John
Enter this year's units:
4500
Enter last year's units:
4000
Nguyen, John
Bonus is $100.0
```

Output #5

```
Enter last name:
Nguyen
Enter first name:
John
Enter this year's units:
7000
Enter last year's units:
6000
Nguyen, John
Bonus is $200.0
```