

Challenge 2: Compute an Expression Using Logical Operators

In this exercise, you need to compute an expression using logical operators.

We'll cover the following ^

- Problem statement
- Coding exercise

Problem statement

There's always some logic in whatever we do, whether it seems like that or not. This time, we are going to give you the logic, and all you need to do is implement it. The challenge is to create the expression explained below and find what result it gives!

Coding exercise

The first step is done for you, which is the method. The method takes in two variables and computes a logical expression using them. For explanation's sake, the parameters are called `x` and `y`.

Now you must do the following:

- Find the **Boolean NOT** of `x`
- **Boolean XOR** the result of above with `x` itself
- Find the **Boolean AND** of the above answer with `y`
- Return the **Boolean NOT** of the entire expression

Only write the code where instructed in the snippet below. You need to calculate the value of the expression, and then store your final result in the variable `answer`. The **return** statement and the **variable** to be returned are already mentioned for you. Don't worry too much about the return statement for the moment and just set the value of the `answer` correctly.

Test your code against our cases and see if you can pass them.

The solution is given in case you get stuck, and the next lesson will include a review of the solution, but it is **highly recommended** that you try it yourself first!

Good Luck!

Your solution will be considered correct only when all the **four** test cases are passed in the below widget.

```
class exercise {  
    public static boolean exercise_two(boolean x, boolean y) {  
        boolean answer = false;  
  
        // Enter your code here  
        // Calculate the value of an expression and store the final value in the answer  
  
        /* You do not need to worry too much about the return statement for the  
        moment and just set the value of "answer" correctly*/  
        return answer;  
    }  
}
```



In the next lesson, we will review the solution to the above challenge.