

Model 1 Conditional Operators

Boolean expressions, like `written > problem` and `teamwork < 75.0`, can be combined using the *conditional operators*:

Operator	Meaning
!	not
&&	and
	or

For example, `written > problem && teamwork < 75.0` is false, because teamwork is not less than 75.0. (Both conditions need to be true in order for `&&` to be true.)

The following table summarizes the result of `&&`, `||`, and `!` for all possible inputs. The variables `p` and `q` represent conditions like `written > problem` and `teamwork < 75.0`.

p	q	p && q	p q	!p
false	false	false	false	true
false	true	false	true	true
true	false	false	true	false
true	true	true	true	false

Questions (20 min)

Start time:

1. Consider the following variables:

```
double initiative = 74.2;
double analytical = 71.9;
double workEthic = 70.8;
boolean hired = true;
boolean fired = false;
```

What are the results (true or false) of the following expressions?

Expression	Result
<code>!fired</code>	
<code>!(workEthic < initiative)</code>	
<code>workEthic < 71.0 && 71.0 < initiative</code>	
<code>initiative < 70.0 workEthic > 70.0</code>	
<code>fired workEthic < 50.0</code>	
<code>analytical < initiative && fired</code>	
<code>hired && !fired</code>	

2. Write a boolean expression that ...

- a) uses initiative, analytical, and !, and evaluates to false.
- b) uses analytical, workEthic, and !, and evaluates to true.
- c) uses any variable(s), and evaluates to false.
- d) uses any variable(s), and evaluates to true.

3. Using your answers to the previous question, write a boolean expression “p && q” where p is your answer to part a) and q is your answer to part b).

a) Your expression:

b) Result of p && q:

Relational operators (<, >, and ==) are evaluated before conditional operators (!, &&, and ||). When multiple conditional operators are used, Java evaluates ! first, then &&, and finally ||.

4. Show the intermediate result of each operator below. In other words, show your work as you evaluate the code in the same order that Java would.

`!(initiative < analytical) && workEthic > analytical`

	Operator	Expression	Result
1st	<	initiative < analytical	false
2nd			
3rd			
4th			

5. Change the parentheses in the original expression (from the previous question) so that the && is evaluated before the !. Then remove any unnecessary parentheses.

a) Expression:

b) New result:

6. Review the table from Model 1 for evaluating && and ||. Looking only at the p and && columns, when is it necessary to examine q to determine how p && q should be evaluated?

7. Review the table from Model 1 for evaluating `&&` and `||`. Looking only at the p and || columns, when is it necessary to examine q to determine how p || q should be evaluated?

8. In Java, `&&` and `||` are *short circuit* operators, meaning they evaluate only what is necessary. If the expression p is more likely to be true than the expression q, which one should you place on the left of each operator to avoid doing extra work?

a) left of the `&&` expression:

b) left of the `||` expression:

9. What is the result of the following expressions?

a) `1 + 0 > 0 && 1 / 0 > 0`

b) `1 + 0 > 0 || 1 / 0 > 0`