

Relational operator:

Operator	Explain
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equality
!=	Not equal

Example 1:

```
int x = 10, y = 3;  
boolean b;
```

```
b = x > y;  
b = true
```

```
b = x == y;  
b = false
```

```
b = x != y;  
b = true
```

```
b = x + y;  
Error x+y is arithmetic expression, can not store in boolean variable.
```

Example 2:

```
boolean a = true, b = false, c;
```

```
c = a == b;  
c = false
```

```
c = !b;  
c = true
```

```
c = a > b;  
Error: can not use > with boolean variables.
```

Logical Operators:

Operator	Explain
&&	and
	or
!	not
&	Bitwise AND
	Bitwise or

true && true	true
true && false	false
false && true	false
false && false	false

true true	true
true false	true
false true	true
false false	false

! true	false
! false	true

Example:

```
int x = 10;  
int y = 2;  
int z = 0;
```

```
boolean b;
```

```
b = x > y && z != 0;  
b = true && false  
b = false
```

using &&:

when the first condition is true the second will not be evaluate it.

```
b = x < y && z++ != 0;  
b = false
```

it will not check `z++ != 0` and `z` will stay 0

if you use `or (||)`, if first condition is true then the second condition will not be checked.

Both `&` or `|` will check the 2 operands.