Difference Between the Assignment and Equal to Operators

```
int newValue = 50;
if (newValue = 50) {
    System.out.println("This is an error");
}
```

As you can see, we've used the assignment operator (one equal sign) in the if statement.

What we need to do, is to use the "equals to" operator (two equal signs).

Difference Between the Assignment and Equal to Operators

```
int newValue = 50;
if (newValue = 50) {
    System.out.println("This is an error");
}
```

This is what the code should look like:

```
int newValue = 50;
if (newValue == 50) {
    System.out.println("This is an error");
}
```

We're not assigning a value here, instead we want to test if the values are equal to each other.

The NOT Operator

The exclamation mark (!), or **NOT** operator, is also known as the Logical Complement Operator.

It can be used with a boolean variable, to test for the opposite value.

```
boolean isCar = false;
if (isCar) {
```

In the code above, we are simply testing the value in the isCar variable, false in this case.

```
boolean isCar = false;
if (!isCar) {
```

But if we use the **NOT** operator, we are testing for the opposite value of the isCar variable, true in this case.

The NOT Operator

I'd generally recommend using the abbreviated form, if your variables are booleans, for two reasons.

One, It's much harder to identify the error, if you accidentally use an assignment operator.

Remember, IntelliJ won't flag this as an error when you're testing a boolean variable, so the only way you'll know you made this common mistake, is by discovering the output from your code isn't what you expected.

Secondly, the code is more concise, and more concise code can often be more readable code.