

Logical Operators - A Quick Summary

As a **reference** which you can come back to (or print out), here's a **quick summary** of how **logical operators** and **comparison operators** behave in JavaScript:

```
1. const userName = 'Max';
2. const altName = '';
3. console.log(userName === 'Max'); // generates and prints a boolean => true
4. console.log(userName); // wasn't touched, still is a string => 'Max'
5.
6. console.log(userName || null); // userName is truthy and therefore returned
   by || => 'Max'
7. console.log(altName || 'Max'); // altName is falsy (empty string), hence
   'Max' is returned => 'Max'
8. console.log(altName || ''); // both altName and '' are falsy but if the
   first operand is falsy, the second one is always returned => ''
9. console.log(altName || null || 'Anna'); // altName and null are falsy,
   'Anna' is returned => 'Anna'
10.
11. console.log(userName && 'Anna'); // userName is truthy, hence second (!)
   value is returned => 'Anna'
12. console.log(altName && 'Anna'); // altName is falsy, hence first value is
   returned => ''
13. console.log(userName && ''); // userName is truthy, hence second value is
   returned => ''
```

Always keep in mind: **NO** operator (neither `===`, `>` etc. nor `&&` or `||`) changes the variable you might be using in the comparison. In the above examples, the values stored in `userName` and `altName` are **NEVER** changed.

`===`, `>` etc. just **generate new boolean values** which are used in the comparison. `||` and `&&` **generate NO booleans**, they just treat the **values before and after them as conditions** (which therefore need to yield boolean values and are coerced to booleans if required).

Because of the above-described behaviors, you often use `||` in JavaScript to assign default/ fallback values to variables/ constants:

```
1. const enteredValue = ''; // let's assume this is set based on some input
   provided by the user, therefore it might be an empty string
2.
3. const userName = enteredValue || 'PLACEHOLDER'; // will assign 'PLACEHOLDER'
   if enteredValue is an empty string
```