

# LOGICAL ASSIGNMENT OPERATORS IN JAVASCRIPT

JavaScript provides three logical assignment operators, introduced in ES2021, which offer concise syntax for common logical operations combined with assignment:

## 1. Logical AND assignment (&&=):

- **Syntax:** `x &&= y`
- **Equivalent to:** `x && (x = y)`
- **Behavior:** Assigns the value of `y` to `x` only if `x` is a truthy value. If `x` is falsy, no assignment occurs.

JavaScript



```
let a = 5;
a &&= 10; // a is now 10 (since 5 is truthy)

let b = 0;
b &&= 20; // b is still 0 (since 0 is falsy)
```

## 2. Logical OR assignment (||=):

- **Syntax:** `x ||= y`
- **Equivalent to:** `x || (x = y)`
- **Behavior:** Assigns the value of `y` to `x` only if `x` is a falsy value. If `x` is truthy, no assignment occurs.

JavaScript



```
let c = null;
c ||= "default"; // c is now "default" (since null is falsy)

let d = "hello";
d ||= "world"; // d is still "hello" (since "hello" is truthy)
```

## Nullish coalescing assignment (??=):

- **Syntax:** `x ??= y`
- **Equivalent to:** `x ?? (x = y)`
- **Behavior:** Assigns the value of `y` to `x` only if `x` is `null` or `undefined`. If `x` has any other value (including falsy values like `0` or `""`), no assignment occurs.

JavaScript



```
let e = undefined;
e ??= "fallback"; // e is now "fallback" (since undefined is nullish)

let f = 0;
f ??= 100; // f is still 0 (since 0 is not nullish)
```