What is logical OR assignment?

Written by Seth Corker on Benevolent Bytes

What is it?

The logical OR assignment operator assigns the right if the left is falsey. Something like `options.value ||= 10` — if value is falsey then we'll assign it the value `10` otherwise, we'll leave it unchanged. Let's have a look at a more in-depth example:

```
function makeCoffee(options = {}) {
  const updatedOptions = { ...options };
  updatedOptions.milk ||= "cow";
  updatedOptions.shots ||= 1;

  return `Coffee (${updatedOptions.shots}x shots) with ${updatedOptions.milk}`;
}

console.log("Order #1 - " + makeCoffee({ milk: "oat", shots: 2 }));
console.log("Order #2 - " + makeCoffee({ milk: "coconut" }));
console.log("Order #3 - " + makeCoffee({ shots: 3 }));
console.log("Order #4 - " + makeCoffee());

// Output
// Order #1 - Coffee (2x shots) with oat
// Order #2 - Coffee (1x shots) with coconut
// Order #3 - Coffee (3x shots) with cow
// Order #4 - Coffee (1x shots) with cow
```

We've got a function `makeCoffee` which takes some options. We're using `||=` to set defaults if nothing is passed in. Order #4 has nothing passed in so we default to cow's milk and a single shot.

Where is it useful?

The logical OR assignment operator is really useful for setting defaults in cases where you don't want a falsey value like `null`, `0`, `''`, `false`, etc. It can also be used in interesting ways because the right-hand side is only evaluated if the left-hand side is falsey. Let's have a look at another example:

```
function getGrindSetting() {
  console.log("No grind setting specified, defaulting to last used setting");
  return "medium";
}

function grindBeans(options = {}) {
  const updatedOptions = { ...options };
  updatedOptions.setting ||= getGrindSetting();
  return `${updatedOptions.setting} ground coffee`;
}

console.log(grindBeans());
console.log(grindBeans({ setting: "fine" }));

// Output
// No grind setting specified, defaulting to last used setting
// medium ground coffee
// fine ground coffee
```

This time we're calling a function which has a side effect if the setting is falsey, `getGrindSetting` will log to the console before returning a value. This type of validation is useful to let us know what's happening when we forget to provide a setting.

What’s the alternative?

The logical OR assignment operator is really just some syntax sugar. It’s nothing you can’t do already, but it does save on typing and simplifies readability. You can achieve the same thing with a conditional.

```
function grindBeans(options = {}) {
  const updatedOptions = { ...options };
  if (!updatedOptions.setting) {
    updatedOptions.setting = getGrindSetting();
  }
  return `${updatedOptions.setting} ground coffee`;
}
```

The logical OR assignment operator is a nice-to-have that can make your code a little less verbose. ## Resources - Logical OR assignment on MDN