Because Svelte's reactivity is triggered by assignments, using array methods like `push` and `splice` won't automatically cause updates. For example, clicking the button doesn't do anything.

One way to fix that is to add an assignment that would otherwise be redundant:

```
function addNumber() {
    numbers.push(numbers.length + 1);
    numbers = numbers;
}
```

But there's a more idiomatic solution:

```
function addNumber() {
    numbers = [...numbers, numbers.length + 1];
}
```

You can use similar patterns to replace `pop`, `shift`, `unshift` and `splice`.

Assignments to *properties* of arrays and objects — e.g. `obj.foo += 1` or `array[i] = x` — work the same way as assignments to the values themselves.

```
function addNumber() {
    numbers[numbers.length] = numbers.length + 1;
}
```

A simple rule of thumb: the name of the updated variable must appear on the left hand side of the assignment. For example this...

```
const foo = obj.foo;
foo.bar = 'baz';
```

...won't trigger reactivity on `obj.foo.bar`, unless you follow it up with `obj = obj`.