



Um pouco mais sobre o reduce

Vejamos um exemplo com `reduce` que soma todos os números de um array:

```
let numeros = [1,2,3,4];

let resultado = numeros.reduce((anterior, atual) => anterior + atual);
alert(resultado);
```

O resultado é 10. Contudo, muitas vezes queremos começar a operação considerando um valor de inicialização. Por exemplo, queremos realizar a mesma operação, só que dessa vez, queremos começar com o valor 5.

```
let resultado2 = numeros.reduce((anterior, atual) => anterior + atual, 5);
alert(resultado2);
```

Desta vez, o resultado final é 15!

Sendo assim, nada nos impede de fazer o primeiro `reduce` passando 0:

```
let numeros = [1,2,3,4];

let resultado = numeros.reduce((anterior, atual) => anterior + atual, 0);
alert(resultado);
```

O resultado será 10 também. Mas vale a pena passar o segundo parâmetro da função `reduce` já que neste caso quando o array for vazio, será lançada a exceção `Reduce of empty array with no initial value`.