

Para saber mais: Trabalhando com strings

O JavaScript traz em sua biblioteca-base vários métodos que usamos para manipular strings de texto: alterar de maiúsculas para minúsculas, contar quantas letras tem uma palavra, retirar espaços, juntar duas strings, etc.

Vamos pensar em alguns exemplos práticos para fazer esse tipo de alteração. Por exemplo, para padronizar uma comparação entre strings:

```
const cidade = "belo horizonte";  
const input = "Belo Horizonte";  
  
console.log(cidade === input); // false
```

[COPIAR CÓDIGO](#)

Nós, como pessoas, conseguimos perceber o valor das variáveis `cidade` e `input` como sendo da mesma cidade, Belo Horizonte. Porém, para o JavaScript, ambos os dados são apenas sequências de caracteres, e a comparação vai falhar, pois como já vimos, **o JavaScript diferencia minúsculas e maiúsculas, tanto nos valores dos dados quanto no código que escrevemos.**

Uma das formas de tratar isso é padronizando todos os inputs para o formato de texto que será comparado antes mesmo de fazer a comparação. Nesse caso, transformando todos os caracteres em letras minúsculas.

```
const cidade = "belo horizonte";  
const input = "Belo Horizonte";  
  
const inputMinusculo = input.toLowerCase();
```



```
console.log(cidade === inputMinusculo); // true
```

[COPIAR CÓDIGO](#)

Acima, vemos um dos **métodos de string nativos do JavaScript** em ação, o `toLowerCase()` que converte todos os caracteres da string informada (no caso, `input`) para letras minúsculas (se forem algarismos, nada é convertido). Você pode conferir mais sobre este método no [MDN \(https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String/toLowerCase\)](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String/toLowerCase).

Outro exemplo: qualquer inserção de texto que exija uma quantidade mínima de caracteres, como uma senha ou um nome. A propriedade `length` pode ser utilizada para sabermos quantos caracteres uma string contém:

```
const senha = "minhaSenha123"
console.log(senha.length) // 13 caracteres
```

[COPIAR CÓDIGO](#)

A propriedade `length` é muito usada no dia a dia do desenvolvimento web. Você pode descobrir mais sobre ela [aqui \(https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String/length\)](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String/length).

Percebeu que `length` não leva parênteses no final da palavra? Há uma diferença entre **métodos** e **propriedades** que não vamos abordar durante este curso, mas vamos deixar aqui a dica caso tenha curiosidade! ;)

Você pode conferir a lista completa de [métodos de string do MDN \(https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String\)](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String)

[BR/docs/Web/JavaScript/Reference/Global_Objects/String#m%C3%A9todos](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String#m%C3%A9todos)).

(são vários), com a descrição de cada um, e praticar com os exemplos.