

VARIÁVEIS E TIPOS DE DADOS

Declaração de variáveis

JavaScript é uma linguagem de tipagem dinâmica. Isso significa que os tipos de dados de suas variáveis não são determinados no momento em que são declaradas. Em vez disso, os tipos de dados são deduzidos a partir dos **valores** atribuídos a elas.

Veja alguns exemplos de declarações de variáveis em JS:

```
1 let x
2 var preco
3 const meuNome = 'Fausto'
```

NOTE BEM: para declarar uma variável em JS, basta uma das palavras-chave reservadas (**let** ou **var**) para esse fim e o nome da variável, nada mais. **const** exige também que um valor seja atribuído à variável no momento da declaração.

Há três palavras-chave utilizadas para declarar variáveis em JavaScript:

- **let**: atualmente, é o método recomendado de criação de variáveis. Uma das vantagens de sua utilização é a impossibilidade de se declarar mais de uma variável com o mesmo nome, o que ajuda a evitar erros de lógica no código. Tem também outras vantagens e características que serão explicadas ao longo desta apostila. É uma adição relativamente recente à linguagem (foi introduzida na versão ES6, de 2015).
- **var**: é a palavra-chave originalmente disponível para a declaração de variáveis, desde a primeira versão do JS. Seu uso apresenta alguns problemas, como a possibilidade de redeclarar uma variável já declarada, o que pode induzir a erros de lógica. Evite utilizá-la, até compreender completamente as consequências de seu emprego.
- **const**: em algumas situações, é necessário representar valores que não deve mais ser alterados posteriormente. São as chamadas **constantes**. Variáveis declaradas com **const** devem receber um valor quando declaradas e não aceitam que este valor seja modificado depois.

Declarações múltiplas

Várias variáveis podem ser declaradas simultaneamente:

```
1 let quantidade, precoUnitario, precoTotal
2 let nome, email, telefone, celular
```

Nomeando variáveis

Em JS, nomes de variáveis devem começar com uma letra; os caracteres \$ e _ também são aceitos na posição inicial. Dígitos (0 a 9) podem ser utilizados, mas

não no início.

Você também não pode usar para nomear variáveis nenhuma das palavras reservadas pelo JS para o seu próprio uso.

É importante frisar que JS é uma **linguagem sensível a caixa alta e caixa baixa** (*case sensitive*, em inglês), isto é, ela trata letras maiúsculas e minúsculas como coisas diferentes. Assim, uma variável de nome `num` é diferente de outra variável de nome `NUM`, e ambas são diferentes de uma terceira variável nomeada como `Num`.

Apesar de as especificações da linguagem JS assim permitirem, não é recomendável declarar variáveis que contenham caracteres acentuados.

IMPORTANTE

O exemplo abaixo contém comentários iniciados com `//`, chamados **comentários de linha**. Esta é uma das formas de se fazer comentários em JS.

```
1 let x           // OK!
2 let primeiroNome // OK!
3 let lnome       // INVÁLIDO: começa com um dígito
4 let $valor      // OK, mas pouco usual
5 let _num        // OK, mas pouco usual
6 let %resultado  // Caractere inicial INVÁLIDO
7 let área        // OK, mas acentos não são recomendados
```

Convenções de nomeação de variáveis

Quando muitos desenvolvedores trabalham num mesmo projeto, é comum que surja, mais cedo ou mais tarde, alguma discórdia sobre a forma de nomear as variáveis.

Por isso, as comunidades de cada linguagem acabam adotando convenções, que não são regras definidas na própria linguagem, mas sim uma espécie de “combinado” entre os membros.

A convenção mais comum entre os desenvolvedores JS é a seguinte:

1. **Sempre** iniciar o nome das variáveis com uma letra **minúscula**;
2. Se o nome da variável for composto por mais de uma palavra, é utilizada **inicial maiúscula a partir da segunda** palavra.

Observe os exemplos a seguir:

```
1 // Uma palavra, inicial minúscula
2 let area
3 // Duas palavras, a segunda com inicial maiúscula
4 let areaTerreno
```

```
5 // Três palavras, iniciais maiúsculas a partir da segunda  
6 let areaTerrenoPadrao
```

CURIOSIDADE: esse tipo de convenção é chamado, em inglês, de *camel case* (*camel* significa “camelo”). O motivo é que as letras maiúsculas no meio do nome das variáveis acabam se parecendo com as corcovas de um camelo.