

Sumário

1	COMEÇANDO COM JAVASCRIPT	2
1.1	O JavaScript e as páginas Web	2
1.2	A estrutura de uma página HTML	2
1.3	Adicionando JavaScript a uma página HTML	3
2	VARIÁVEIS E TIPOS DE DADOS	5
2.1	Declaração de variáveis	5
2.1.1	Declarações múltiplas	5
2.1.2	Nomeando variáveis	5
2.1.3	Convenções de nomeação de variáveis	6

Capítulo 1

COMEÇANDO COM JAVASCRIPT

1.1 O JavaScript e as páginas Web

As páginas Web são formadas pela combinação de três tecnologias:

- **HTML** (*Hypertext Markup Language*, isto é, linguagem de marcação de hipertexto): é responsável por estruturar o conteúdo da página, ou seja, **o que** você vê nela.
- **CSS** (*Cascading Style Sheets*, folhas de estilo em cascata): cuida da aparência da página, como cores, fontes e alinhamento dos elementos. Controla **como** o conteúdo é exibido na página.
- **JavaScript**: é uma linguagem de **programação** que pode ser adicionada às páginas Web, conferindo-lhes interatividade para com o usuário. O JavaScript é capaz, por exemplo, de verificar em um formulário se o usuário digitou algo diferente do esperado ou de fazer coisas se movimentarem pela página.

O JavaScript será o objeto do nosso estudo. No entanto, como iremos utilizá-lo **dentro** de uma página Web, precisamos saber e entender o básico de HTML.

1.2 A estrutura de uma página HTML

Uma página Web (também chamada de página HTML) é um arquivo de texto simples com extensão `.html` ou `.htm`. A listagem a seguir apresenta a estrutura básica de um arquivo HTML.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="UTF-8">
5     <title>Minha primeira página HTML</title>
6 </head>
7 <body>
8     <p>Olá, mundo!</p>
9 </body>
10 </html>
```

IMPORTANTE

Os números que aparecem à esquerda do código **não** fazem parte dele. Servem apenas para que possamos nos referir a diferentes partes do código usando o número da linha.

Como podemos ver, o HTML é formado por elementos delimitados pelos caracteres `<` e `>`, os quais são chamados **tags**.

Muitas tags vêm em pares, formando **seções**. Podemos observar, por exemplo, que a tag `<body>`, chamada tag de abertura, tem a correspondente tag de fechamento `</body>` (note a presença da `/` antes do nome da tag).

Agora, vamos analisar cada uma das partes desse código.

- `<!DOCTYPE html>` (linha 1): essa *tag* serve para indicar ao navegador Web que irá exibir a página qual a versão da linguagem HTML está sendo usada. No caso, esse *doctype* indica que se trata da versão 5 do HTML, a mais recente.
- Seção **html** (linhas 2 a 10): a maior parte do código da página fica nessa grande seção. Dentro dessa grande seção, temos as seções **head** e **body**.
- Seção **head** (linhas 3 a 6): aqui colocadas *tags* de configuração da página, como a `<meta charset="UTF-8">` (linha 4), para garantir que os caracteres acentuados sejam exibidos corretamente. Os elementos dessa seção, normalmente, não têm um efeito visível para o usuário. Uma exceção é a *tag* `<title>` (linha 5), cujo conteúdo aparece na aba do navegador onde a página estiver sendo exibida.
- Seção **body** (linhas 7 a 9): todo o conteúdo da página que será visível para o usuário é colocado nessa seção. No código de exemplo, temos um parágrafo (`<p>`, linha 8) contendo um texto a ser exibido.

Quando o arquivo HTML contendo este código for exibido em um navegador Web, veremos um resultado semelhante a este:

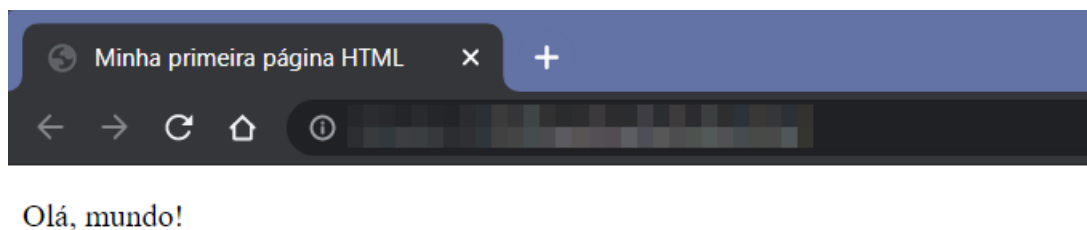


Figura 1.1: Exibição de uma página HTML

1.3 Adicionando JavaScript a uma página HTML

Para utilizar JavaScript em uma página HTML, precisamos criar uma seção `<script></script>`, normalmente dentro da seção **head**, e adicionar o código JavaScript dentro dela.

O código listado a seguir irá apresentar uma nova versão da mensagem “Olá, mundo!” mas, desta vez, quem dará o recado é a linguagem JavaScript.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Minha primeira página HTML</title>
6   <script>
7     alert('Olá, mundo!')
8   </script>
9 </head>
```

```
10 <body>
11
12 </body>
13 </html>
```

Observe com atenção as linhas 6 a 8. Aberto no navegador, um arquivo HTML com este código produz o seguinte resultado:

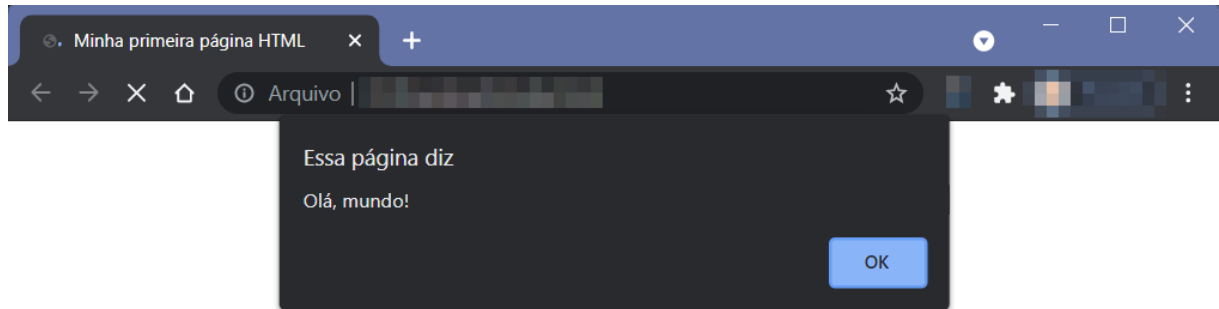


Figura 1.2: Mensagem exibida usando JavaScript

Portanto, agora você já sabe. Toda vez que formos usar JavaScript em uma página Web, devemos:

1. Criar um arquivo com extensão `.html` ou `.htm`.
2. Colocar dentro desse arquivo a estrutura básica de um arquivo HTML. Editores de código, como o Visual Studio Code ou o Gitpod possuem recursos que geram automaticamente este código.
3. Adicionar uma seção `<script></script>` na seção **head** e colocar as instruções JavaScript dentro dela.

Capítulo 2

VARIÁVEIS E TIPOS DE DADOS

2.1 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 devem 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.

2.1.1 Declarações múltiplas

Várias variáveis podem ser declaradas simultaneamente:

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

2.1.2 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
```

2.1.3 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.