

Desenvolvimento Web

Front-End

História do HTML

O HTML foi proposto no início de 1990 e recebeu novas funcionalidades ao longo do tempo visando atender às novas exigências de comunicação e representação da informação. Atualmente o padrão HTML 5 é amplamente utilizado, no momento já temos o padrão HTML 5.3, lançado em janeiro de 2021.

Conheça mais sobre o HTML, visitando o site da Organização Mozilla, neste site você vai encontrar um manual completo e detalhado com exemplos: [Mozilla.org - HTML](https://developer.mozilla.org/pt-BR/docs/Web/HTML)

Se desejar praticar e avançar mais em seus estudos o site W3Schools oferece uma ferramenta para codificar e visualizar os resultados: [W3Schools.com - HTML](https://www.w3schools.com/html/onlinehtmleditor.asp)

O objetivo do HTML é oferecer um padrão compreensível por nós humanos e que pode ser interpretado pelos navegadores. Um dos fatores responsáveis pelo sucesso do HTML é sua simplicidade de desenvolvimento, pois exige apenas um editor de texto e um navegador web. Além disso, ao navegar por um site qualquer na internet podemos facilmente explorar o código HTML que está sendo apresentado. Para explorar o código de uma página qualquer vamos utilizar o **DevTools**.

O HTML é uma linguagem interpretada pelos navegadores. Mas o que exatamente significa isso?

Significa que todas as tags que desenvolvemos são executadas e geram algum resultado para ser exibido ao usuário. Isso significa que as tags não são “validadas”, nem mesmo são reportados erros caso ocorram. Este é um fator que define o HTML como linguagem de marcação e não como linguagem de programação. Uma linguagem de programação passa por um processo de validação dos comandos, onde cada um deles é executado podendo ou não gerar uma saída.

De forma resumida isso significa que quando nosso código HTML estiver incorreto, não vai apresentar erro, simplesmente não vai funcionar.

Existem diversos padrões que os programadores podem adotar, então é fundamental você conheça bastante a linguagem HTML pois vai se deparar com códigos desenvolvidos por outros programadores ao longo de sua jornada profissional e antes mesmo de executar o código é importante que você faça a validação do mesmo e entenda o que o código deve realizar.

Introdução à HTML

HTML é uma sigla para representar o termo **Linguagem de Marcação de HiperTexto**, originado do inglês *HyperText Markup Language*. Portanto a HTML é uma linguagem para criar documentos com textos e outros elementos do meio digital, tais como imagens, sons e vídeos. Todos estes elementos serão ligados por meio de tags ou elementos, que vamos estudar nesta unidade/disciplina. As *tags* são recursos fundamentais da HTML. Abaixo segue exemplo de uma *tag*.

```

```

A *tag* apresentada acima é a *tag* `img`, que indica uma abreviação de *image*, que traduzindo é imagem. Note que a *tag* `img` tem o atributo `src`, que indica uma abreviação de *source*, que traduzindo é origem. Os atributos são recursos presentes na maioria das *tags*.

Resumindo o conceito de tag com relação ao HTML:



→ **tag**: etiqueta, rótulo ou marcação, exemplo `` é um exemplo de tag de imagem

→ **atributo**: característica que desejamos especificar, geralmente um atributo é seguido por sinal de igualdade, exemplo `src=`

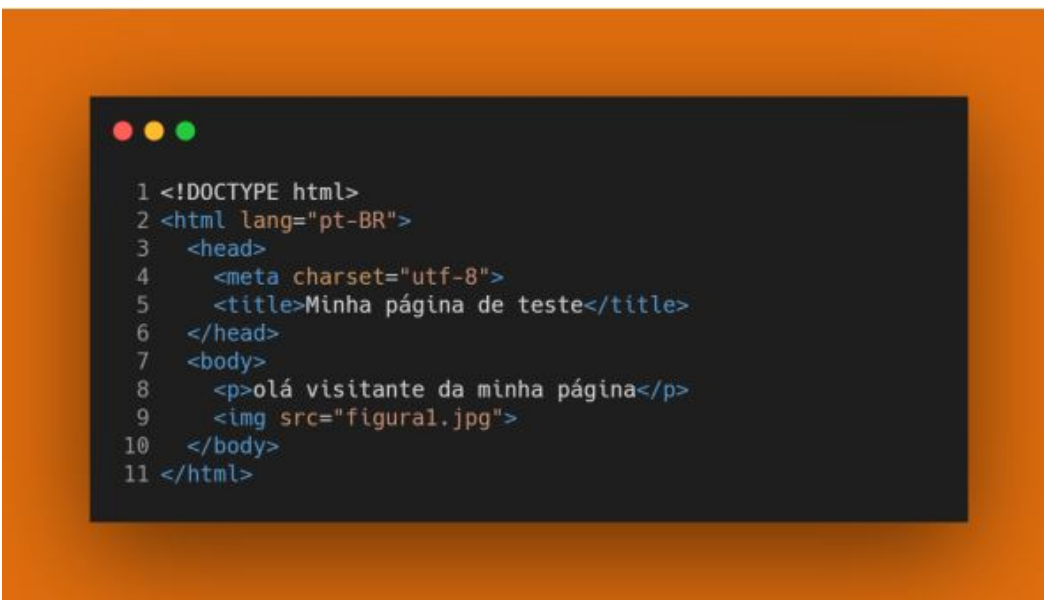
→ **valor**: geralmente fica entre aspas duplas. Em alguns casos, o valor pode ficar entre aspas simples, dependendo do padrão adotado. Vamos manter o padrão de aspas duplas.

Notas importantes

- ◆ Perceba que a *tag* inicia com o símbolo de menor, ou seja `<`. Observe também um sinal de maior no final, ou seja `>`.
- ◆ Todas as *tags* tem uma representação de finalização opcionais. Veremos ao logo desta disciplina/unidade tais particularidades.
- ◆ Conheça mais tags HTML visitando este site: [W3Schools.com](https://www.w3schools.com)

Estrutura básica

Agora que já sabemos sobre tags, vamos explorar a estrutura básica de um documento HTML. Um documento HTML deve ter a declaração da tag inicial `<!DOCTYPE html>`, embora um documento HTML possa ser interpretado sem a declaração, ela é altamente recomendada. Abaixo segue *tag* DOCTYPE:



A tag DOCTYPE pode ser escrita em minúscula, ou seja, não é *case sensitive*. Algumas linguagens exigem a escrita usando letras maiúsculas em certas situações e não funcionam caso escrevemos uma letra minúscula no lugar de uma letra maiúscula, por exemplo em JavaScript uma variável definida como `pi` é diferente de uma variável definida como `PI`. Podemos afirmar que HTML não é *case sensitive*, ou seja, tanto faz se escrevermos com letra maiúscula ou minúscula, já a linguagem JavaScript é *case sensitive*. Sempre que possível vamos obedecer ao padrão e digitar corretamente as letras.

A tag `<html>` é a primeira a ser aberta, antes dela sempre teremos apenas a tag `!DOCTYPE`. A tag `<html>` é fechada na linha 11, da seguinte forma `</html>`. A tag `img` não fechamos, apenas abrimos. Mais adiante veremos exemplos de tag `img` com fechamento.

A tag `<head>` é utilizada para definir configurações do cabeçalho de nossa página tais como seu conteúdo, título, palavras-chave e outras configurações que são processadas logo que a página é carregada.

Na tag `<head>` temos também inclusão de arquivos CSS e JS.

Os **títulos** são representados dentro da tag `<title>` que por sua vez estão dentro da tag `<head>` e são considerados pelos motores de busca, isso significa que se você deve sempre criar títulos relevantes e criativos para suas páginas.

Um documento HTML deve ter apenas uma tag `<html>`. Tanto a tag `<head>` quanto a tag `<body>` são únicas por padrão.

A próxima tag que vamos explorar é `<body>`. Nela podemos colocar tudo que desejamos exibir para nosso usuário, tais como textos, figuras e links. No exemplo colocamos dentro da tag `<body>` a tag `<p>` que representa um parágrafo no HTML e tag `` que utilizamos para adicionar uma imagem ao HTML.

Semântica HTML

As tags HTML devem ser utilizadas de forma que seja compreensível para nós seres humanos e por computadores. Dar semântica a um documento significa simplificar sua codificação e fazer com o documento seja compreensível sem muito esforço. As páginas HTML são interpretadas por outras aplicações, pois uma vez estando “hospedadas” e acessíveis, qualquer aplicação pode abrir nossa página e obter as informações que precisa. Para organizar as informações de uma página existe um grupo de elementos, listados abaixo:

`<header>`
`<section>`
`<article>`
`<nav>`
`<aside>`
`<main>`
`<figure>`
`<footer>`
`<a>`
``
``
`<cite>`
`<q>`
`<time>`

A descrição e utilização de cada uma dessas tags estará disponível em nosso [Glossário HTML](#) no Class. Veja Exemplos para cada uma destas tags semânticas:

```
1 <main>
2   <h2>O que é o Programa Starter?</h2>
3   <p>O Programa Starter é um curso? Não! Te decepcionei? Calma que vou te explicar e
   te mostrar que é muito mais do que isso!</p>
4   <section>
5     <p>O Programa Starter, além de uma formação completa em Desenvolvimento Web
   Full Stack, com metodologia de ensino e tecnologias direcionadas para o que as
   empresas demandam, tem o principal objetivo de preparar e apoiar a sua inserção no
   mercado de trabalho.</p>
6   </section>
7 </main>
8
```

```
1 <article>
2   <header>Nesta página estamos criando exemplos de tags semânticas</header>
3   <p>Aqui é apenas um parágrafo.</p>
4   <footer>Por fim temos um rodapé do artigo.</footer>
5 </article>
```

```
1 <aside>
2   <nav>
3     <ul>
4       <li><a href="https://www.javascript.com/" target="_blank">JS</a></li>
5       <li><a href="https://www.w3schools.com/js/DEFAULT.asp">JS W3S</a></li>
6     </ul>
7   </nav>
8 </aside>
```

```
1 <p>
2   <q>Confie naqueles que buscam a verdade, mas duvide daqueles que dizem que a encontraram.</q>
3   <cite>Andre Gide</cite>
4 </p>
```

```
1 <figure>
2   
3   <figcaption>Figura 1. Equipe do Programa Starter</figcaption>
4 </figure>
```

```
1 <header>
2   <h1>Formação FullStack</h1>
3   <h2>HTML CSS JavaScript</h2>
4   <h3>Tags</h3>
5 </header>
```

```
1 <nav>
2   <ul>
3     <li><a href="https://www.google.com/chrome/">Chrome</a></li>
4     <li><a href="https://www.microsoft.com/pt-br/download/internet-explorer.aspx">Internet Explorer</a></li>
5     <li><a href="https://www.mozilla.org/pt-BR/firefox/">FireFox</a></li>
6     <li><a href="https://support.apple.com/downloads/safari">Safari</a></li>
7     <li><a href="https://www.opera.com/pt-br/download">Opera</a></li>
8   </ul>
9 </nav>
```

```
1 <section>
2   <h4>Formação FullStack</h4>
3   <p>Nesta formação você vai aprender fundamentos Front-end e Back-end</p>
4 </section>
```

```
1 <p>Você deseja participar de uma <em>formação completa em Desenvolvimento Web Full Stack</em>?</p>
2 <p>O principal objetivo é <strong>preparar</strong> e <strong>apoiar</strong> sua inserção no mercado de trabalho</p>
3
```

```
1 <!DOCTYPE html>
2 <html lang="pt-BR">
3   <head>
4     <meta charset="utf-8">
5     <title>Página Exemplo Semântico</title>
6   </head>
7   <body>
8     <header>
9       <h1>Programa Starter</h1>
10    </header>
11    <main>
12      <section>
13        <h2>HTML</h2>
14        <article>
15          <h3>Desenvolvimento de páginas simples</h3>
16          <p>Vamos explorar as tags e seus exemplos.</p>
17        </article>
18      </section>
19      <section>
20        <h2>CSS</h2>
21        <article>
22          <h3>Criação de estilos</h3>
23          <p>Personalização de cores tamanhos e demais elementos visuais.</p>
24        </article>
25        <article>
26          <h3>Arquivos css</h3>
27          <p>Separação de estilos em arquivos específicos.</p>
28        </article>
29      </section>
30      <section>
31        <h2>JavaScript</h2>
32        <article>
33          <h3>Lógica de programação</h3>
34          <p>Vamos explorar conceitos básicos de lógica de programação.</p>
35        </article>
36        <article>
37          <h3>Eventos e Funções</h3>
38          <p>Vamos explorar eventos e funções.</p>
39        </article>
40      </section>
41    </main>
42    <footer>
43      <p>GrowDev - Programa Starter.</p>
44    </footer>
45  </body>
46 </html>
```

Atividade Prática Aula 02 - 7a Edição do Programa Starter 2022 - [@Growdev](#)