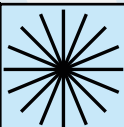


Ciclo formativo básico em tecnologia

Módulo1 :



HTML5

CSS3

ciclo
forma-
tivo



Parte I: introdução à HTML

Introdução



O grande pai da web foi o físico Tim Berners Lee. Em 1989, Lee criou uma linguagem de marcação de hipertexto para construir páginas no ambiente virtual. Em 1990, ele criou com sucesso a primeira comunicação entre cliente HTTP e servidor usando essa linguagem. Em 1991, Tim criou a primeira versão da HTML, inicialmente projetada para conectar e compartilhar documentos entre instituições de pesquisa próximas. Em 1992, a HTML foi liberada para uma biblioteca de desenvolvimento de alcance mundial e passou por várias versões até atingir a quinta, que iremos usar. Em 1995, a W3C (World Wide Web Consortium), entidade que regula os padrões da web, tornou-se responsável por sua supervisão e controle.

O termo “linguagem de marcação de texto” (“*Hyper Text Markup Language*”, em inglês) se refere a um texto contendo diversos formatos de informações, como imagens, vídeos, áudios, textos etc. O acesso às informações se dá através dessa ligação, por meio de links.

Escrevendo em HTML5



O primeiro passo para escrever códigos em HTML é definir a ferramenta a ser usada. Podemos usar até um bloco de notas, embora a prática não seja ideal ou tão efetiva. As ferramentas mais usadas são as IDEs (sigla para *Integrated Development Environment* em inglês, ou Ambiente de Desenvolvimento Integrado), uma vez que trazem vários atalhos úteis para escrever código. Notepad++, Sublime Text e Visual Studio Code são alguns dos exemplos de ferramentas.



*Nos exemplos da apostila, usaremos Visual Studio Code.

Tags

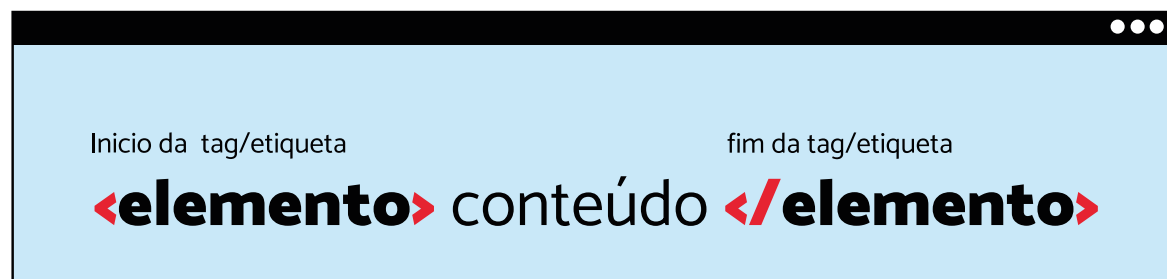


Antes de começar nosso primeiro HTML, é importante conhecermos *tags*.

Suponha que possuímos algumas caixas de papelão fechadas contendo diferentes itens. Como podemos saber o conteúdo de cada caixa sem precisar abri-las?

A melhor maneira seria etiquetar cada caixa com as respectivas informações. O mesmo raciocínio se aplica à HTML: para cada informação incluída no código, é preciso declarar “caixas”; dentro delas, as respectivas informações.

Essa forma de despachar caixas marcando elementos e determinando onde cada uma começa e termina é chamada de *tag* (também vem daí o termo “linguagem de marcação de texto”). Toda *tag* é declarada pelos sinais de maior e menor (< >) e fechada pelos mesmos sinais precedidos por uma barra (< / >). É importante observar que a grande maioria das *tags* sempre precisa ser aberta e fechada, conforme exemplo abaixo.



Iniciando um arquivo HTML



O primeiro passo é criar o arquivo e salvá-lo usando a extensão .html. Dentro desse arquivo, que pode ser visualizado em qualquer navegador de sua preferência, precisamos declarar todas as *tags*.

A primeira *tag* de todo arquivo html é **<!DOCTYPE html>**: Ela informa ao navegador escolhido que é um documento tipo HTML e, portanto, precisa vir antes de todas as outras, na primeira linha do seu código.

Hierarquia de estrutura



O HTML possui uma estrutura que sempre precisa ser declarada e respeitada para o bom funcionamento das páginas. A partir dessa estrutura, determinamos a forma correta de inserir marcações em nossa página. Um bom exemplo é uma redação: quando a redigimos, ela precisa ter um começo, um meio e um fim. Uma página em HTML também precisa sempre ter começo, meio e fim.

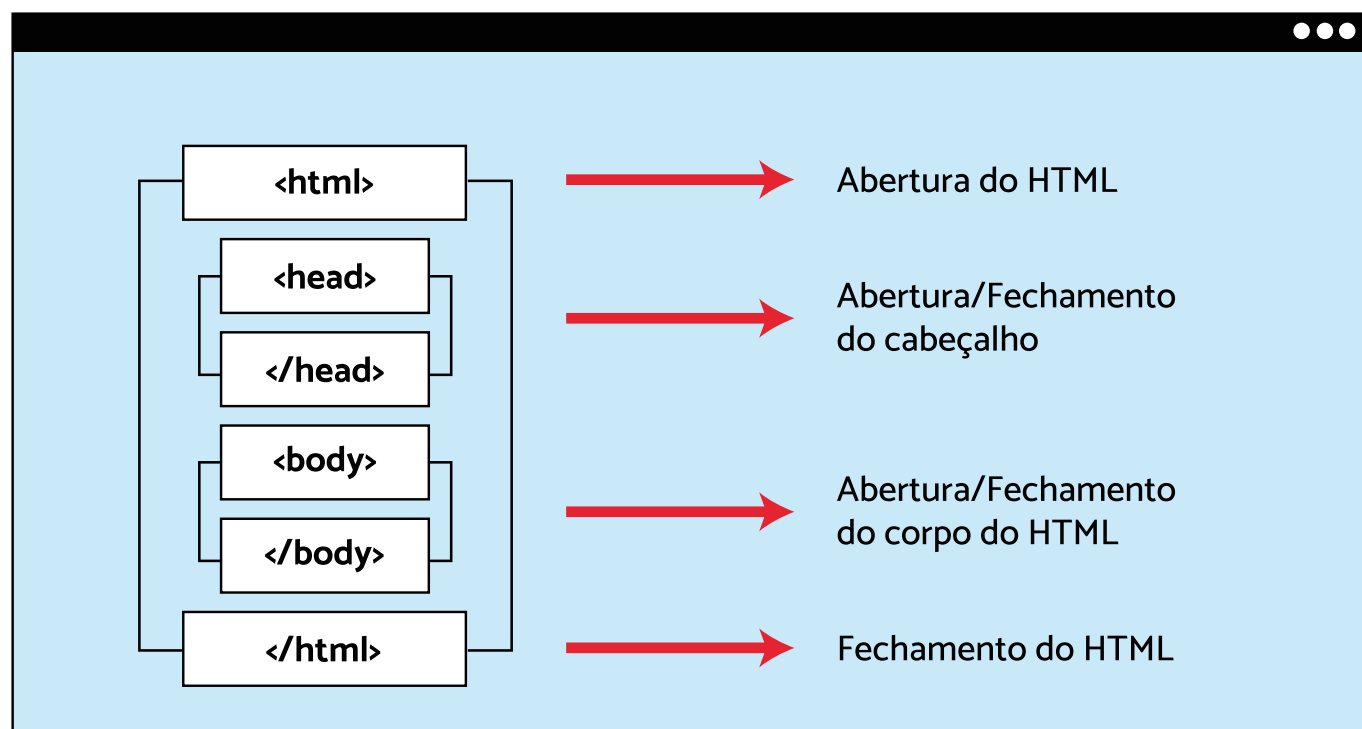
Os significados das *tags* da estrutura de um arquivo HTML são:

`<html></html>` : início e fim de estrutura HTML;

`<head></head>` : início e fim de cabeçalho em um arquivo html;

`<body></body>` : início e fim do corpo de arquivo html, ou seja, tudo aquilo que será exibido no navegador.

Abaixo, exemplo de estrutura hierárquica de uma página.



No exemplo abaixo, criamos um arquivo, o nomeamos "**index.html**", o identificamos como "**DOCTYPE html**" e criamos uma hierarquia mínima para nossa página funcionar.



```
<> index.html M X
<> index.html > ...
You, há 1 segundo | 2 authors (You and others)
1  <!DOCTYPE html>
2  <html>
3    <head>
4
5    </head>
6
7    <body>
8
9    </body>
10 </html>
11
```

Cabeçalho



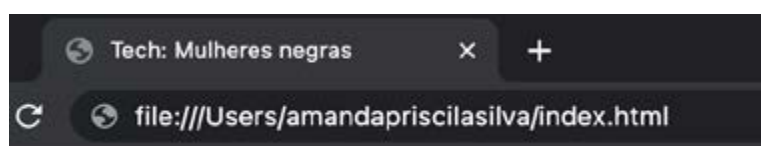
Tudo aquilo que o navegador consegue ler é definido nessa etapa, como nome do site na aba do navegador, comunicação com outros arquivos externos e configuração para mecanismos de pesquisa. O cabeçalho (*head*, em inglês) do HTML é representado pela *tag* **<head></head>**. Algumas *tags* liberadas para uso dentro da **<head>** incluem:

- <title></title>** : define o que aparece na aba do navegador;
- <meta>** : auxilia ferramentas de busca como Google a encontrarem seu site na web. Aqui, podemos incluir um pequeno resumo sobre a página como forma de indexar o conteúdo web disponível. Uma ação importante nessa *tag* é a passagem do atributo **charset="utf-8"**, uma vez que essa informação diz ao nosso navegador que ele precisa comunicar aos demais o formato de codificação de caracteres utilizado no documento e falado na língua portuguesa. Essa informação determina até o reconhecimento dos acentos, etc.;
- <script>** : traz instruções de *script* ou aponta para um arquivo externo de *script*, por meio do atributo **src**;
- <link>**: passa o caminho para outro arquivo.

No código, nosso cabeçalho fica da seguinte forma:

```
<head>  
  <title>Tech: Mulheres negras</title>  
  <meta charset="utf-8">  
  <meta name="description" content="Portfólio">  
  <link rel="stylesheet" href="index.css">  
</head>
```

Se abrirmos o arquivo em um navegador, veremos uma página em branco, mas o título definido previamente no cabeçalho aparecerá conforme abaixo.



Indentação



Nossas *tags* estão contidas umas nas outras; algumas, no canto, outras, mais para dentro. Chamamos esse processo de “indentação do código”. Uma boa prática que pode ser observada não apenas em HTML como também em tudo que envolve programação. Ao Indentar, você organiza seu código, ajuda no entendimento, busca por uma forma mais rápida para elementos que precisem de ajustes ou estejam compreendidos e até insere novas linhas de códigos a partir de um determinado trecho. É importante que o código seja lido de forma limpa e com facilidade por você ou qualquer outra pessoa que pegue-o na sequência.

No exemplo de cabeçalho mais acima, declaramos a abertura e fechamento da *tag* **<head>** **</head>**. Seu conteúdo (*title*, *meta*, *link* e *script*) não pode estar na mesma linha hierárquica, por isso as *tags* aparecem mais para dentro. Neste processo de indentação, tudo dentro das *tags* declaradas precisa respeitar hierarquia e ficar alinhado.

Body



O corpo (*body*, em inglês) do HTML, representado pela *tag* `<body></body>`, diz respeito ao conteúdo de uma página HTML, ou seja, tudo que é exibido na tela e que outras pessoas veem em sua página. Dentro da *tag* `body`, declaramos textos, imagens, formulários, listas, tabelas e tudo aquilo que aprenderemos durante o curso.



Parte II: aprofundando em HTML

Níveis de título



Usamos as *tags* `<h1></h1>` e `<h6></h6>` para definir cabeçalhos e títulos em nosso arquivo HTML atentando aos seguintes pontos:

- `<h1>` : define o título mais importante;
- `<h1>` : define o título menos importante.

Abaixo, exemplos no código e no navegador:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Tech: Mulheres negras</title>
    <meta charset="utf-8">
    <meta name="description" content="Portfólio">
    <link rel="stylesheet" href="index.css">
  </head>

  <body>
    <h1>Introdução a front-end</h1>
    <h2>Introdução a front-end</h2>
    <h3>Introdução a front-end</h3>
    <h4>Introdução a front-end</h4>
    <h5>Introdução a front-end</h5>
    <h6>Introdução a front-end</h6>
  </body>
</html>
```

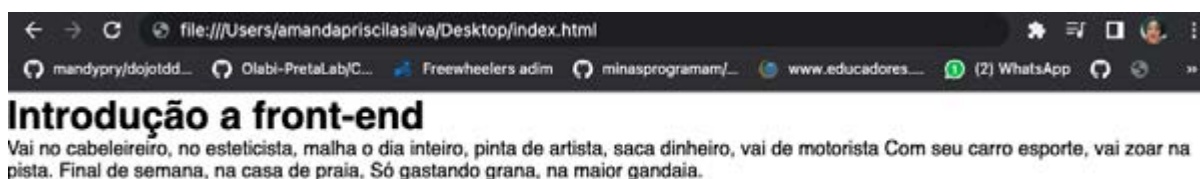


Parágrafos



Para inserir textos em nosso HTML, podemos declarar a *tag* `<p></p>`, conforme abaixo.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Tech: Mulheres negras</title>
    <meta charset="utf-8">
    <meta name="description" content="Portfólio">
    <link rel="stylesheet" href="index.css">
  </head>
  <body>
    <h1>Introdução a front-end</h1>
    <p>
      Vai no cabeleireiro, no esteticista, malha o dia inteiro, pinta de artista, saca dinheiro, vai de motorista
      Com seu carro esporte, vai zoar na pista. Final de semana, na casa de praia, Só gastando grana, na maior gandaia.
    </p>
  </body>
</html>
```



Tag para quebra de linha



Podemos pular uma linha usando a *tag* `
`, que pode ser fechada dentro de sua própria declaração pela barra no final dispensando necessidade de duas *tags* (uma para abertura e outra para fechamento).

Tags de formatação de texto



Podemos definir o formato de texto usando *tags* como:

`` : negrito

`<i></i>` : itálico

`` : forte

`<small></small>` : pequeno


```
<body>
  <b>Amanda Silva</b>
  <br/>
  <i>Amanda Silva</i>
  <br/>
  <strong>Amanda Silva</strong>
  <br/>
  <small>Amanda Silva</small>
</body>
```

Exemplo em código

Amanda Silva
Amanda Silva
Amanda Silva
Amanda Silva

Exemplo em navegador

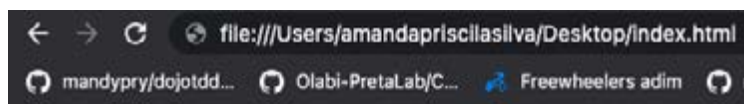
Listas



É possível listar um conjunto de itens de determinado grupo de diversas formas, dentre elas, **ordenadas** e **desordenadas**. Para declarar uma lista dentro do *body* `<body></body>`, usamos a *tag* `` para listas ordenadas e `` para não ordenadas. Também podemos inserir itens em nossa lista individualmente dentro da *tag* ``, conforme exemplo abaixo.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Tech: Mulheres negras</title>
    <meta charset="utf-8">
    <meta name="description" content="Portfólio">
    <link rel="stylesheet" href="index.css">
  </head>

  <body>
    <h1>Exemplo de Lista Ordenada</h1>
    <ol>
      <li>Encontrar um trabalho</li>
      <li>Fazer muito dinheiro</li>
      <li>Mudar para outra cidade</li>
    </ol>
  </body>
</html>
```



Exemplo de Lista Ordenada

1. Encontrar um trabalho
2. Fazer muito dinheiro
3. Mudar para outra cidade

Ao declarar listas, podemos passar atributos dentro da *tag* ``. Eles definem o número com o qual iniciaremos nossa lista e podem até ordená-los alfabeticamente, conforme abaixo (**type**, ou tipo, em inglês, indica o tipo de ordenação):

‘a’ indica letras minúsculas;

‘A’ indica letras maiúsculas;

‘i’ indica algarismos romanos minúsculos;

‘I’ indica algarismos romanos maiúsculos;

‘1’ indica números (padrão).

Abaixo, exemplo no código e navegador.

```
<body>
  <h1>Lista ordenada alfabeticamente com letras maiusculas</h2>
  <ol type="A">
    <li>Nome</li>
    <li>Telefone</li>
    <li>Endereço</li>
    <li>País</li>
  </ol>

  <h1>Lista ordenada por números romanos</h2>
  <ol type="I">
    <li>Nome</li>
    <li>Telefone</li>
    <li>Endereço</li>
    <li>País</li>
  </ol>
</body>
```



Lista ordenada alfabeticamente com letras maiusculas

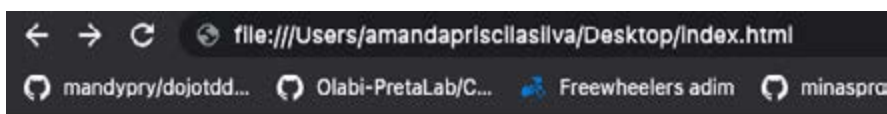
- A. Nome
- B. Telefone
- C. Endereço
- D. País

Lista ordenada por números romanos

- I. Nome
- II. Telefone
- III. Endereço
- IV. País

Start (início, em inglês) define a primeira posição da lista ordenada. Abaixo, exemplo no código e navegador.

```
<body>
  <h2>Exemplo de Lista Ordenada iniciando no número 4</h2>
  <ol start="4" >
    <li>Encontrar um trabalho</li>
    <li>Fazer muito dinheiro</li>
    <li>Mudar para outra cidade</li>
  </ol>
</body>
```



Exemplo de Lista Ordenada iniciando no número 4

4. Encontrar um trabalho
5. Fazer muito dinheiro
6. Mudar para outra cidade

Com listas não ordenadas tipo ****, é possível definir o marcador através do atributo **type**, como círculo, disco e quadrado.

```
<body>
  <h2>Lista desordenada quadrado</h2>
  <ul type="square">
    <li>Farinha</li>
    <li>Milho</li>
    <li>Molho</li>
    <li>Macarrão</li>
  </ul>

  <h2>Lista desordenada círculo vazio</h2>
  <ul type="circle">
    <li>Farinha</li>
    <li>Milho</li>
    <li>Molho</li>
    <li>Macarrão</li>
  </ul>

  <h2>Lista desordenada círculo cheio</h2>
  <ul type="disc">
    <li>Farinha</li>
    <li>Milho</li>
    <li>Molho</li>
    <li>Macarrão</li>
  </ul>
</body>
```

Código



Lista desordenada quadrado

- Farinha
- Milho
- Molho
- Macarrão

Lista desordenada círculo vazio

- Farinha
- Milho
- Molho
- Macarrão

Lista desordenada círculo cheio

- Farinha
- Milho
- Molho
- Macarrão

Navegador

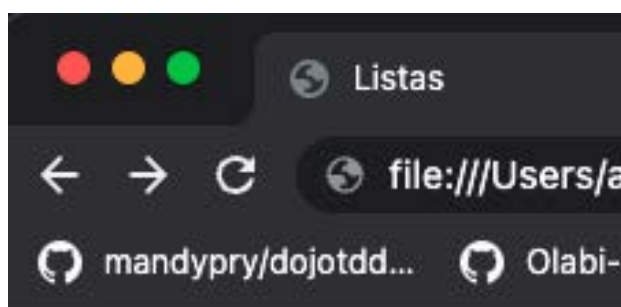
Tabelas



Podemos criar tabelas usando a tag `<table></table>` e determinar a quantidade de linhas com a tag `<tr></tr>`, desde que declarando cada uma delas. Em todas estas linhas, é possível definir a coluna de título pela tag `<th></th>` e cada uma delas individualmente incluindo suas informações por meio da tag `<td></td>`.

```
<body>
  <h2>Estrutura de Tabela</h2>
  <table >
    <tr>
      <th>Nome</th>
      <th>Idade</th>
      <th>Profissão</th>
    </tr>
    <tr>
      <td>Nome</td>
      <td>Idade</td>
      <td>Profissão</td>
    </tr>
    <tr>
      <td>Ted</td>
      <td>22</td>
      <td>Estudante</td>
    </tr>
    <tr>
      <td>Ralf</td>
      <td>26</td>
      <td>Designer</td>
    </tr>
  </table>
</body>
```

Código



Estrutura de Tabela

Nome	Idade	Profissão
Nome	Idade	Profissão
Ted	22	Estudante
Ralf	26	Designer

Navegador

Na declaração da tabela, usamos o atributo **border** (borda, em inglês) para definir uma borda e escolher a espessura da tabela.

```
<body>
  <h2>Estrutura de Tabela</h2>
  <table border="2">
    <tr>
      <th>Nome</th>
      <th>Idade</th>
      <th>Profissão</th>
    </tr>
    <tr>
      <td>Amanda</td>
      <td>34</td>
      <td>Desenvolvedora</td>
    </tr>
    <tr>
      <td>Lívia</td>
      <td>14</td>
      <td>Estudante</td>
    </tr>
    <tr>
      <td>Pandora</td>
      <td>05</td>
      <td>Pet do amor</td>
    </tr>
  </table>
</body>
```

Código

Estrutura de Tabela

Nome	Idade	Profissão
Amanda	34	Desenvolvedora
Lívia	14	Estudante
Pandora	05	Pet do amor

Navegador

Também podemos mesclar linhas com o atributo **rowspan** (expansão por linha, em inglês); colunas, com **colspan** (expansão por coluna, em inglês).

```
<body>
  <h2>Mesclando colunas (colspan)</h2>
  <table border="1">
    <tr>
      <th>Nome</th>
      <th colspan="2">Telefones</th>
    </tr>
    <tr>
      <td>Ted</td>
      <td>8888-8888</td>
      <td>9999-9999</td>
    </tr>
    <tr>
      <td>Junior</td>
      <td>1111-1111</td>
      <td>2222-2222</td>
    </tr>
  </table>

  <h2>Mesclando linhas (rowspan)</h2>
  <table border="1">
    <tr>
      <th>Primeiro Nome:</th>
      <td>Ted</td>
    </tr>
    <tr>
      <th rowspan="2">Telefone:</th>
      <td>8888-8888</td>
    </tr>
    <tr>
      <td>9999-9999</td>
    </tr>
  </table>
</body>
```

Código

Mesclando células (colspan)

Nome	Telefones	
Ted	8888-8888	9999-9999
Junior	1111-1111	2222-2222

Mesclando células (rowspan)

Primeiro Nome:	Ted
Telefone:	8888-8888
	9999-9999

Navegador

Uma boa prática consiste em definir o cabeçalho da nossa tabela com a tag **<thead></thead>** e o corpo com a tag **<tbody></tbody>**. Dentro do cabeçalho, colocamos a linha contendo a tag de título e no corpo, as linhas restantes. Visualmente não há mudança navegador, mas o código fica mais limpo e legível.

```
<body>
  <table border="1">
    <thead>
      <tr>
        <th>titulo 1</th>
        <th>titulo 2</th>
        <th>titulo 3</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Body1 linha1</td>
        <td>Body2 linha1</td>
        <td>Body3 linha1</td>
      </tr>
      <tr>
        <td>Body1 linha2</td>
        <td>Body2 linha2</td>
        <td>Body3 linha2</td>
      </tr>
    </tbody>
  </table>
</body>
```

titulo 1	titulo 2	titulo 3
Body1 linha1	Body2 linha1	Body3 linha1
Body1 linha2	Body2 linha2	Body3 linha2
Foot 1	Foot 2	Foot 3

Código

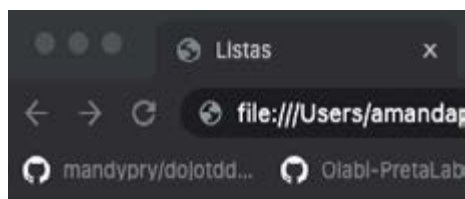
Navegador

Imagens



A tag **** insere uma imagem em uma página HTML. Tecnicamente, as imagens não são inseridas em uma página web, mas sim vinculadas. Na tag ****, é preciso incluir o atributo que determina o caminho da imagem **src** a ser exibida no navegador.

```
<body>
  <h2>inserindo foto de perfil</h2>
  
</body>
```



inserindo foto de perfil



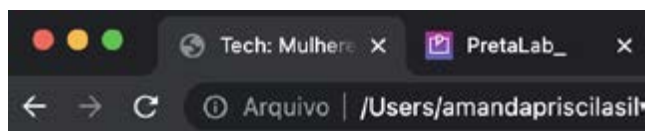
Âncoras e Link



A tag `<a>` insere um link na página HTML que direciona o usuário para outra em uma página HTML. Tecnicamente, as imagens não são inseridas em uma página web, mas sim vinculadas.

O atributo `href` define o alvo (ou caminho) do link para o onde a página deve direcionar o usuário quando a âncora for clicada. O atributo `target` (alvo, em inglês) após recebimento do valor `_blank` é fundamental para que outra aba possa ser aberta no navegador, não na mesma na qual o usuário já se encontra.

```
<body>
  <a href="https://www.pretalab.com/" target="_blank">Pretalab</a>
</body>
```



[Pretalab](#)

Ao clicar no link do navegador, outra aba é aberta no site.



Ícones



Para inserir ícones em HTML, a *tag* mais utilizada é **<i>**. Lembram quando falamos no início sobre **<head>**, o cabeçalho do arquivo HTML que guarda algumas informações/ *tags* importantes não exibidas para usuários no navegador?

Existem bibliotecas com diversos ícones para chamarmos em nosso código pelo nome, dentro da *tag* de ícone. A *Font Awesome* é bastante utilizada; para inseri-la, é necessário declarar o link externo da biblioteca dentro da *tag* **<head>**.

```
<head>
  <link href="img/foto-perfil.png">
  <title>Tech: Mulheres negras</title>
  <meta charset="utf-8">
  <meta name="description" content="Portfólio">
  <script src="https://use.fontawesome.com/releases/v5.15.3/js/all.js" data-auto-replace-svg="nest"></script>
</head>
```

Para ícones da biblioteca externa, é necessário buscar no [site](#) o nome do ícone que desejamos usar. No site, é possível ver como a *tag* **<i>** precisa ficar para que o ícone seja exibido na tela.

linkedin

f08c in </> ↓



HTML React Vue.js




```
<i class="fa-brands fa-linkedin"></i>
```

Start Using This Icon

No caso da *tag* `<i>`, o atributo **class** (classe, em inglês) chama o ícone pelo nome.

No exemplo a seguir, três ícones no navegador pesquisados na biblioteca *Font Awesome*: envelope de e-mail; o ícone do LinkedIn; e do github. Na sequência, podemos inseri-los em nossas *tags*.

```
<body>
  <ul>
    <li><i class="fas fa-envelope-square"></i> &nbsp; Isso é um ícone de e-mail</li>
    <li><i class="fab fa-linkedin"></i> &nbsp; Isso é um ícone do LinkedIn</li>
    <li><i class="fab fa-github-square"></i> &nbsp; Isso é um ícone do github</li>
  </ul>
</body>
```

-  Isso é um ícone de e-mail
-  Isso é um ícone do LinkedIn
-  Isso é um ícone do github

Semântica



O HTML semântico ajuda na explicação e compreensão das informações da página da web; nas buscas em ferramentas de busca como o Google; e no entendimento de leitores que dependem de acessibilidade. Para deixar a página semântica, é necessário separar todo o conteúdo do nosso `<body></body>` em blocos separados, por *tags* que, por sua vez, separam as partes do código.

Algumas *tags* semânticas:

- `<header></header>`: introdução, o cabeçalho do *body*;
- `<main></main>`: conteúdo principal da página;
- `<section></section>`: seção em documento HTML;
- `<nav></nav>`: seção de página que aponta para outras;
- `<footer></footer>`: rodapé;
- `<article></article>`: parte independente em um HTML.

Abaixo, exemplo de cabeçalho dentro de **<body>** declarado com a tag **<header>**.

```
<header class="header-menu background-lilas">
  <nav>
    <ul class="lista-menu">
      <li>
        <a href="#formacao">Formação</a>
      </li>
      <li>
        <a href="#portfolio">Portfólio</a>
      </li>
      <li>
        <a href="#contato">Contato</a>
      </li>
    </ul>
  </nav>
</header>
```

Exemplo de tag **<main>** dentro do **<body>** declarando a parte principal em nosso site.

```
<main>
  <section>
    <article>
      
      <h3>Desenvolvedora Web</h3>
      <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce a nisl in sem tempus tincidunt. Integer magna ante, vulputate sit amet sodales sed, pharetra ac velit. In hac habitasse, donec vulputate tellus sed mollis nulla porta. Aliquam turpis leo, pellentesque et velit a, sagittis scelerisque quam. Suspendisse imperdiet molestie leo. Praesent sed
      </p>
    </article>
  </section>
  <section>
    <article>
      <h3>Portfólio</h3>
      <ul>
        <li>
          <p>Projetos que fazem história</p>
          <p>
            <a href="https://google.com/" target="_blank">Curso Front end 2022</a>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce a nisl in sem tempus tincidunt. Integer magna ante, vulputate sit amet sodales sed, pharetra ac velit.
          </p>
          </p>
        </li>
        <li>
          <p>Desenvolvedora Web</p>
          <p>
            <a href="https://google.com/" target="_blank">Projetos 2022</a>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce a nisl in sem tempus tincidunt. Integer magna ante, vulputate sit amet sodales sed, pharetra ac velit.
          </p>
          </p>
        </li>
      </ul>
    </article>
  </section>
</main>
```

E exemplo de tag **<footer>** dentro do **<body>** declarando o rodapé em nosso site.

```
<footer>
  <ul>
    <li><i class="fas fa-envelope-square estilo-icon"></i><a href="mailto:amanda.adgt@gmail.com">email@email.com</a></li>
    <li><i class="fab fa-linkedin estilo-icon"></i><a href="https://linkedin.com/">marina-silva</a></li>
    <li><i class="fab fa-github-square estilo-icon"></i><a href="https://github.com/">maridesenvolvedora</a></li>
  </ul>
</footer>
```

Entidade



Alguns caracteres interpretados pelo HTML talvez não apareçam na tela quando digitados, como sinais de maior e menor (< >). Se precisarmos exibi-los na tela, usamos as entidades do HTML conforme abaixo.

Resultado	Descrição	Nome da entidade
	espaço	
<	menor que	<
>	maior que	>
&	ampersand	&
"	aspas duplas	"
'	aspas unica	'
¢	centavo (dólar)	¢
£	peso	£
¥	yen	¥
€	euro	€
©	copyright	©
®	marca registrada	®

fonte: [devmedia](#)

Ao final de cada frase referente aos itens da lista acima, inserimos ** ** no código e um espaço no texto entre o próximo item (que seria um ícone).

```
<ul>  
  <li>ícone e-mail &nbsp; &nbsp;<i class="fas fa-envelope-square estilo-icon"></i></li>  
  <li>ícone linkedin &nbsp; <i class="fab fa-linkedin estilo-icon"></i></li>  
  <li>ícone github &nbsp; &nbsp;<i class="fab fa-github-square estilo-icon"></i></li>  
</ul>
```

ícone e-mail



ícone linkedin



ícone github



Acessibilidade na web



Existe um conjunto de boas práticas para escrever códigos em HTML que facilita acesso à página e concede autonomia a pessoas com deficiência, que dependem de tecnologias assistivas como leitores de tela, ampliadores de tela, *mouses* e teclados especiais:

- Tamanho das fontes;

- Alinhamento e espaçamento entre os textos;

- Regras de contraste e posicionamento das *tags* em nosso código;

- Na declaração inicial da *tag* `<html>`, definir qual linguagem está sendo usada (`<html lang="pt-br">`);

- Utilizar estrutura semântica;

- Em *tags* de imagens, usar o atributo `alt` com texto alternativo descrevendo a imagem. Visualmente, não há alteração na página, mas leitores de tela leem exibem descrição, como no exemplo abaixo.

```
<body>  
    
</body>
```

Forms



Para criar *forms* (formulários, em inglês) em nossa página web, usamos a *tag* `<form></form>`. O título de cada campo do formulário é declarado com a *tag* `<label></label>`, e o campo no qual o usuário pode digitar no formulário, com a *tag* `<input></input>`.

A *tag* `<input>` possui o atributo `type`, que define o tipo de informação incluída pelo usuário, como texto, telefone, e-mail. Também possui botão que submete o formulário após preenchimento (`required`, ou obrigatório, em inglês, que define que quando campo é de preenchimento obrigatório) e `placeholder` (nome do espaço, em inglês, que deixa uma mensagem de aviso no campo que some quando o usuário começa a digitar). Abaixo, um exemplo de formulário completo em nosso código.

Entre em contato

```
<body>
  <form>
    <h1>Entre em contato</h1>
    <label>Nome</label><br/>
    <input type="text" placeholder="Digite seu nome" required="required"/>
    <br/><br/>
    <label>Fone</label><br/>
    <input type="tel" placeholder="(xx)xx-xx-xx-xx"/>
    <br/><br/>
    <label>Email</label><br/>
    <input type="email" placeholder="fulano@mail.com"/>
    <br/><br/>
    <label>Opinião</label><br/>
    <textarea placeholder="Deixe sua opinião"></textarea>
    <br/><br/>
    <input type="submit" value="Enviar" />
  </form>
</body>
```

Código



Navegador

Botões



Com a tag **<button></button>**, é possível inserir botões clicáveis na página da web. Com o atributo **type**, definimos o tipo (**reset**, que restaura, ou **submit**, que envia).

```
<body>
  <h1>Criando botão</h1>
  <br/>
  <button>Eu sou um botão</button>
</body>
```

Código

Criando botão



Navegador

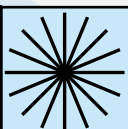
Referências



W3School: [w3school HTML](#)

Developer Mozilla: [Developer mozilla](#)

Módulo1:



HTML5

CSS3

