

Tecnologias Web

HTML: Avançando na Linguagem

02 - Semântica, Imagens, Tabelas e Formulários

Roteiro

- Elementos Block e Inline
- HTML Semântico
- Imagens
- Tabelas
- Formulários

Relembrando:

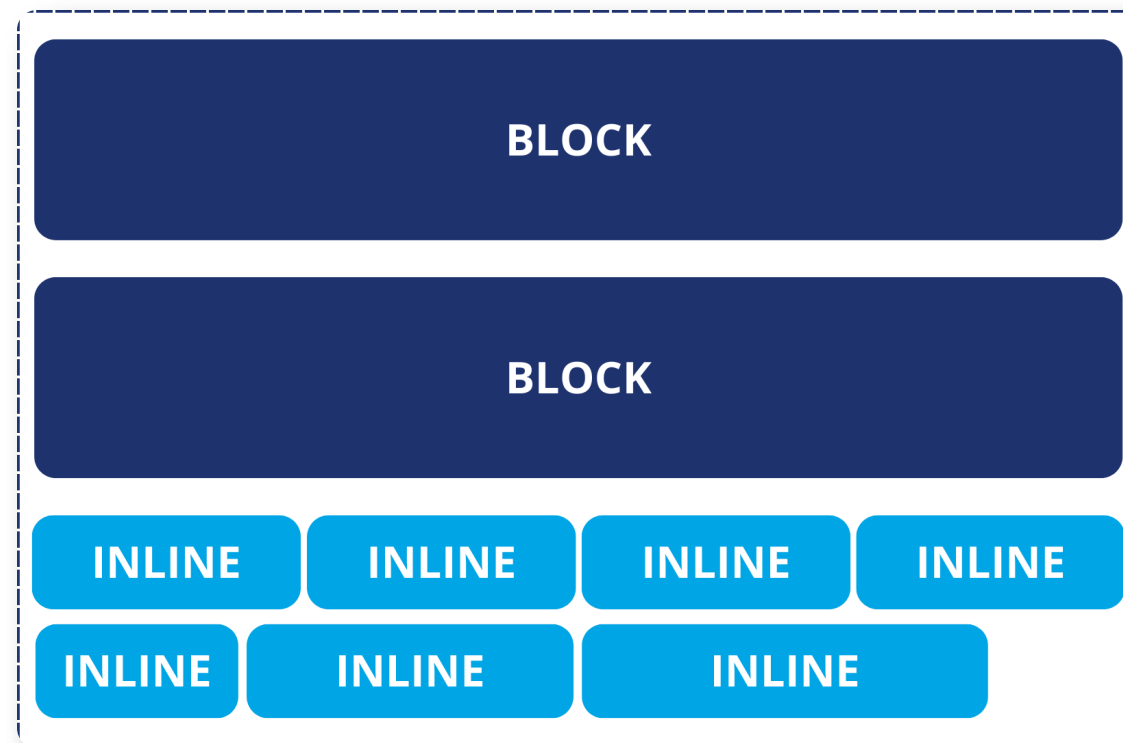
- Quais as tags básicas que estudamos até aqui?



Elementos Block e Inline

Todo elemento HTML tem um valor de exibição padrão

- O valor de exibição de um elemento HTML define a forma como ele é estruturado na página
- Os dois valores de exibição mais comuns são **block** (bloco) e **inline** (em linha)



Elementos Block e Inline

Todo elemento HTML tem um valor de exibição padrão

- **Elementos em nível de bloco:**

- sempre começam em uma nova linha e ocupam toda a largura disponível
- o navegador adiciona alguma margem antes e após o elemento
- a tag `<div>` é um elemento em nível de bloco bastante utilizado

- **Elementos em linha:**

- não iniciam em uma nova linha
- ocupam somente a largura necessária para o seu conteúdo
- a tag `` é um exemplo de elemento inline bastante utilizado

Elementos Block e Inline

Todo elemento HTML tem um valor de exibição padrão

- **Elementos em nível de bloco:**

- sempre começam em uma nova linha e ocupam toda a largura disponível
- o navegador adiciona alguma margem antes e após o elemento
- a tag `<div>` é um elemento em nível de bloco bastante utilizado

- **Elementos em linha:**

- não iniciam em uma nova linha
- ocupam somente a largura necessária
- a tag `` é um elemento em linha

<code><address></code>	<code><article></code>	<code><aside></code>	<code><blockquote></code>	<code><canvas></code>
<code><dd></code>	<code><div></code>	<code><dl></code>	<code><dt></code>	<code><fieldset></code>
<code><figcaption></code>	<code><figure></code>	<code><footer></code>	<code><form></code>	<code><h1>-<h6></code>
<code><header></code>	<code><hr></code>	<code></code>	<code><main></code>	<code><nav></code>
<code><noscript></code>	<code></code>	<code><p></code>	<code><pre></code>	<code><section></code>
<code><table></code>	<code><tfoot></code>	<code></code>	<code><video></code>	

Elementos Block e Inline

Todo elemento HTML tem um valor de exibição padrão

- Elementos em nível de bloco

- sempre começam em uma nova linha
- o navegador adiciona uma quebra de linha
- a tag `<div>` é um exemplo de elemento de bloco

<code><a></code>	<code><abbr></code>	<code><acronym></code>	<code></code>	<code><bdo></code>
<code><big></code>	<code>
</code>	<code><button></code>	<code><cite></code>	<code><code></code>
<code><dfn></code>	<code></code>	<code><i></code>	<code></code>	<code><input></code>
<code><kbd></code>	<code><label></code>	<code><map></code>	<code><object></code>	<code><output></code>
<code><q></code>	<code><samp></code>	<code><script></code>	<code><select></code>	<code><small></code>
<code></code>	<code></code>	<code><sub></code>	<code><sup></code>	<code><textarea></code>
<code><time></code>	<code><tt></code>	<code><var></code>		

- Elementos em linha:

- não iniciam em uma nova linha
- ocupam somente a largura necessária para o seu conteúdo
- a tag `` é um exemplo de elemento inline bastante utilizado

A Tag `<div>`

O elemento `<div>` é frequentemente usado como *container* para outros elementos

- A tag `<div>` representa um elemento *genérico* em nível de bloco, ou seja, não tem um significado específico
- Usamos `divs` principalmente como container para outros elementos
- Conseguimos agrupar um conjunto de outros elementos dentro de uma `div` e aplicar regras CSS para **estilizá-los e/ou posicioná-los** mais facilmente

A Tag <div>

O elemento `<div>` é frequentemente usado como *container* para outros elementos

- Exemplo de uso do elemento `<div>` como container:

```
1 <!-- HTML -->
2 <div>
3   <h2>IFCE</h2>
4   <p>Instituto Federal de Educação, Ciência e Tecnologia do Ceará.</p>
5   <p>Instituição de ensino que atua em cursos de nível técnico, graduação e pós-graduação.</p>
6 </div>
```

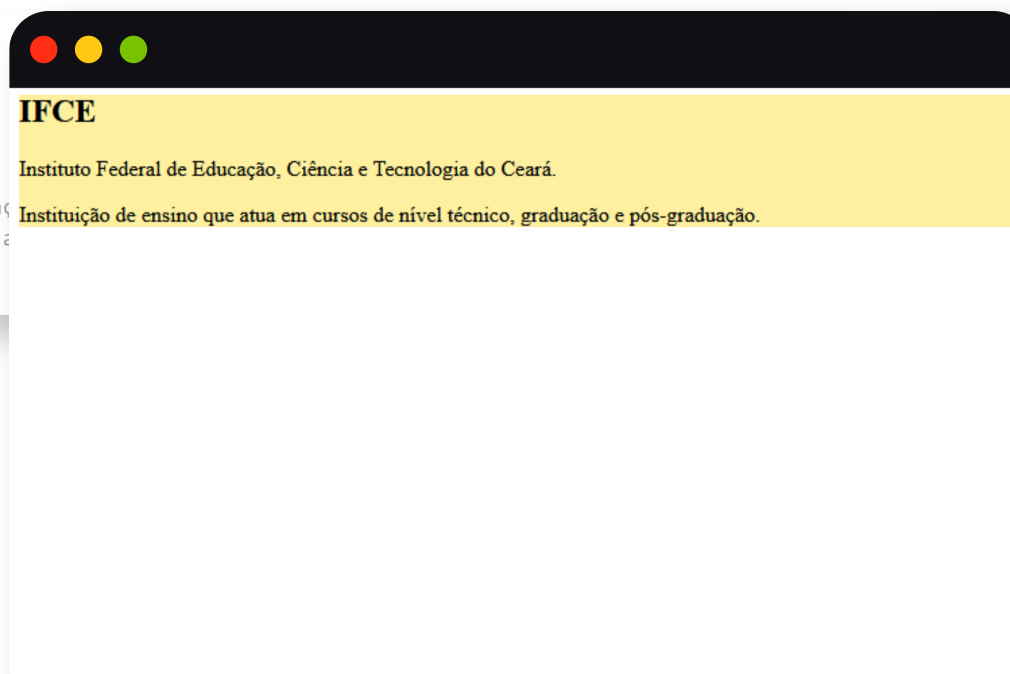
```
1 /* CSS */
2 div {
3   background-color: #FFF4A3;
4 }
```

A Tag <div>

O elemento `<div>` é frequentemente usado como *container* para outros elementos

- Exemplo de uso do elemento `<div>` como container:

```
1 <!-- HTML -->
2 <div>
3   <h2>IFCE</h2>
4   <p>Instituto Federal de Educação, Ciência e Tecnologia do Ceará.
5   <p>Instituição de ensino que atua em cursos de nível técnico, graduação e pós-graduação.
6 </div>
```



```
CSS */
{
background-color: #FFF4A3;
```

A Tag <div>

O elemento `<div>` é frequentemente usado como *container* para outros elementos

- Exemplo de uso do elemento `<div>` como container:

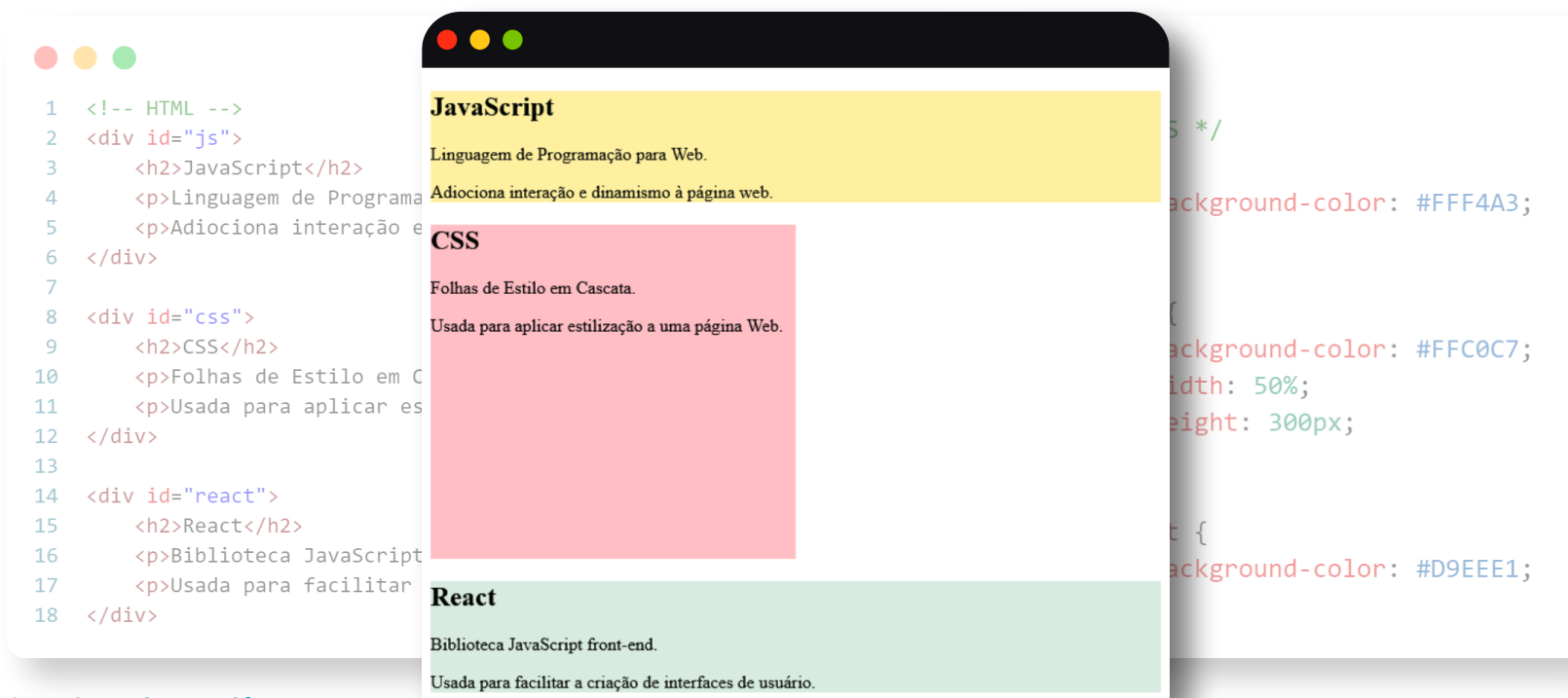
```
1  <!-- HTML -->
2  <div id="js">
3      <h2>JavaScript</h2>
4      <p>Linguagem de Programação para Web.</p>
5      <p>Adiciona interação e dinamismo à página web.</p>
6  </div>
7
8  <div id="css">
9      <h2>CSS</h2>
10     <p>Folhas de Estilo em Cascata.</p>
11     <p>Usada para aplicar estilização a uma página Web.</p>
12 </div>
13
14 <div id="react">
15     <h2>React</h2>
16     <p>Biblioteca JavaScript front-end.</p>
17     <p>Usada para facilitar a criação de interfaces de usuário.</p>
18 </div>
```

```
1  /* CSS */
2  #js {
3      background-color: #FFF4A3;
4  }
5
6  #css {
7      background-color: #FFC0C7;
8      width: 50%;
9      height: 300px;
10 }
11
12 #react {
13     background-color: #D9EEE1;
14 }
```

A Tag <div>

O elemento `<div>` é frequentemente usado como *container* para outros elementos

- Exemplo de uso do elemento `<div>` como container:



A Tag ``

O elemento `` é frequentemente usado para marcar uma parte de um texto/conteúdo

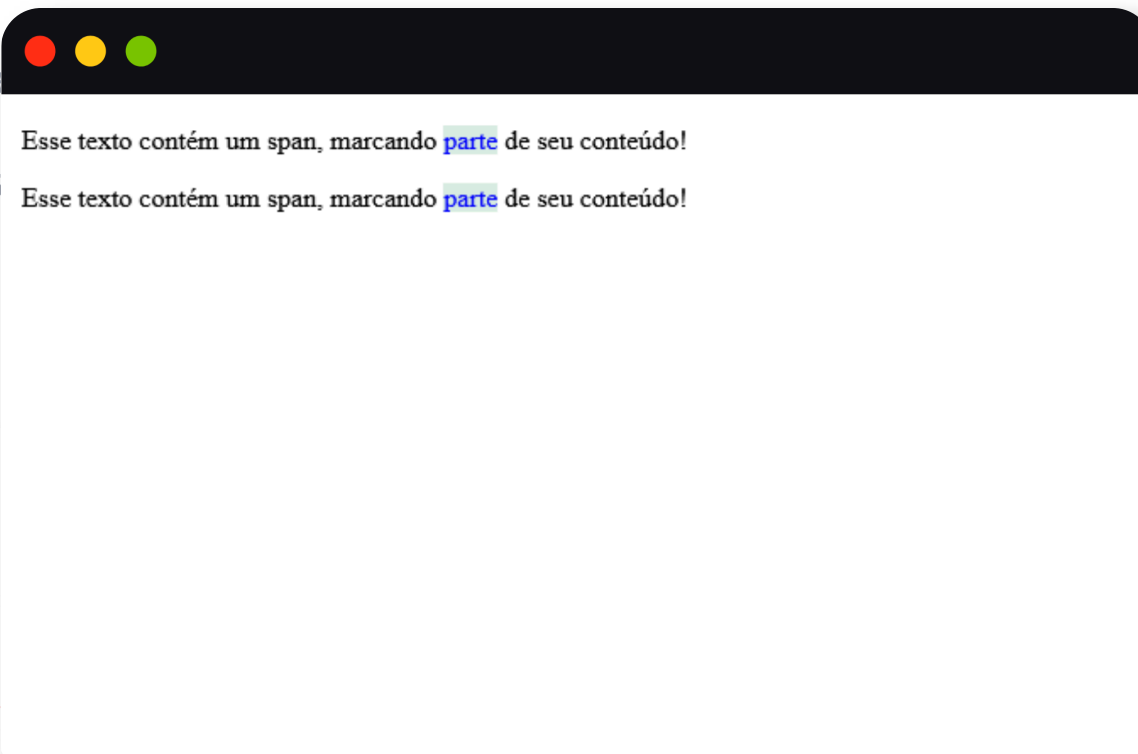
- A tag `` representa um elemento *genérico* em nível de linha (inline)
- Com ele podemos **marcar parte de um conteúdo** (texto, por exemplo) e, junto com o CSS, estilizá-lo de forma específica
- Por ser um elemento inline, **não é possível alterar suas dimensões (*width e height*)**

```
1 <!-- HTML -->
2 <p>Esse texto contém um span, marcando <span style="color: #0000FF; background-color: #D9EEE1;">parte</span> de seu
3   conteúdo!</p>
4
5 <p>Esse texto contém um span, marcando <span
6   style="color: #0000FF; background-color: #D9EEE1; width: 200px;">parte</span> de seu
7   conteúdo!</p>
```

A Tag

O elemento `` é frequentemente usado para marcar uma parte de um texto/conteúdo

- A tag `` representa
- Com ele podemos marcar uma parte de um texto/conteúdo para ser estilizada de forma específica
- Por ser um elemento de nível de bloco



```
1 <!-- HTML -->
2 <p>Esse texto
3   conteúdo!<
4
5 <p>Esse texto
6   styl
7   conteúdo!<
```

com o CSS, estilizá-lo de

(right)

te de seu

HTML Semântico

Elementos semânticos são elementos com um **significado claro**

- O que são elementos semânticos?
 - No HTML, elementos semânticos são tags que **descrevem claramente** seu significado, tanto para os navegadores como para os desenvolvedores
 - Assim, o **HTML Semântico** tem o objetivo de **descrever o significado** de conteúdo em um documento Web

HTML Semântico

Elementos semânticos são elementos com um **significado claro**

- Por exemplo, a tag `` define um texto importante, não somente deixa-o em negrito
- Outro exemplo é a tag `<address>` que define uma informação de contato para o autor de um documento ou artigo

```
1 <p>Endereço do IFCE: <i>Rua Antônio Teixeira Benevides, 01</i></p>
2
3 <p>Endereço do IFCE: <address>Rua Antônio Teixeira Benevides, 01</address></p>
4
5 <p>Este texto é <b> muito importante</b>, com a tag b</p>
6
7 <p>Este texto é <strong> muito importante</strong>, com a tag strong</p>
```

Endereço do IFCE: *Rua Antônio Teixeira Benevides, 01*

Endereço do IFCE:

Rua Antônio Teixeira Benevides, 01

Este texto é **muito importante**, com a tag b

Este texto é **muito importante**, com a tag strong

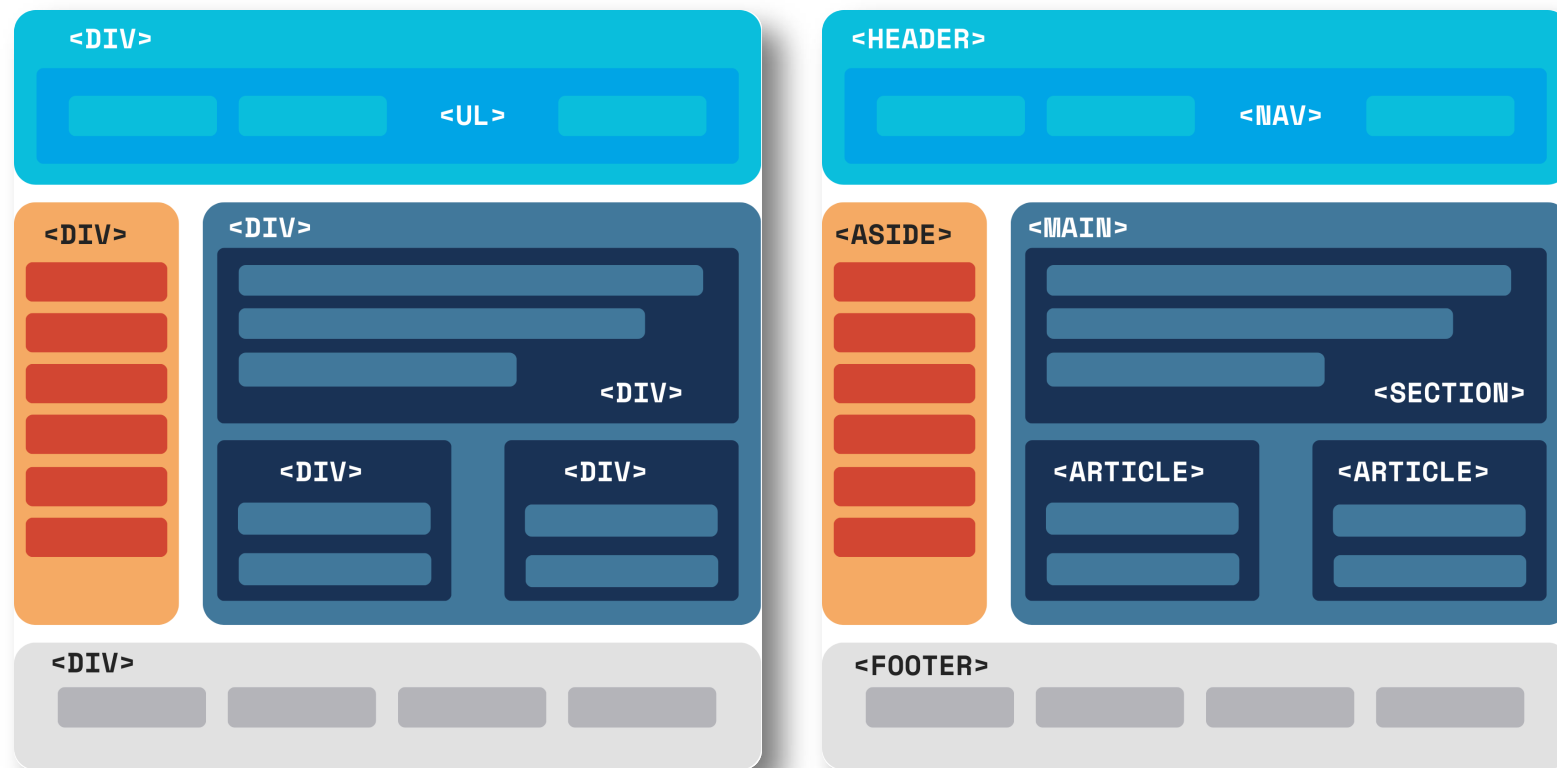
HTML Semântico

O HTML5 trouxe elementos semânticos para estruturar a página

- Antes do HTML5, era muito comum definir a estrutura e o posicionamento de elementos na página usando tags `<div>` (e ainda é...)
- Bem antes, até mesmo a tag `<table>` (para tabelas) era utilizada para esse propósito
- O HTML5 trouxe uma maior aderência à **Web Semântica**, adicionando mais tags semânticas, principalmente para definir o **layout** da página

HTML Semântico

O HTML5 trouxe elementos semânticos para estruturar a página



HTML Semântico

Layout sem semântica

```
1 <div id="header">
2   <h1>City Gallery</h1>
3 </div>
4
5 <div id="main">
6   <div id="aside">
7     <ul id="nav">
8       <li><a href="">London</a></li>
9       <li><a href="">Paris</a></li>
10      <li><a href="">Tokyo</a></li>
11    </ul>
12  </div>
13
14  <div id="section1">
15    <h2>London</h2>
16    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Voluptate dolore, atque natus quas omnis magni
17      tenetur in blanditiis ratione numquam odit quibusdam consequatur necessitatibus nobis laudantium illo
18      repellendus id! Nemo.</p>
19
20    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Voluptate dolore, atque natus quas omnis magni
21      tenetur in blanditiis ratione numquam odit quibusdam consequatur necessitatibus nobis laudantium illo
22      repellendus id! Nemo.</p>
23
24    
25  </div>
26 </div>
27
28 <div id="footer">
29   Copyright &copy; W3Schools.com
30 </div>
```

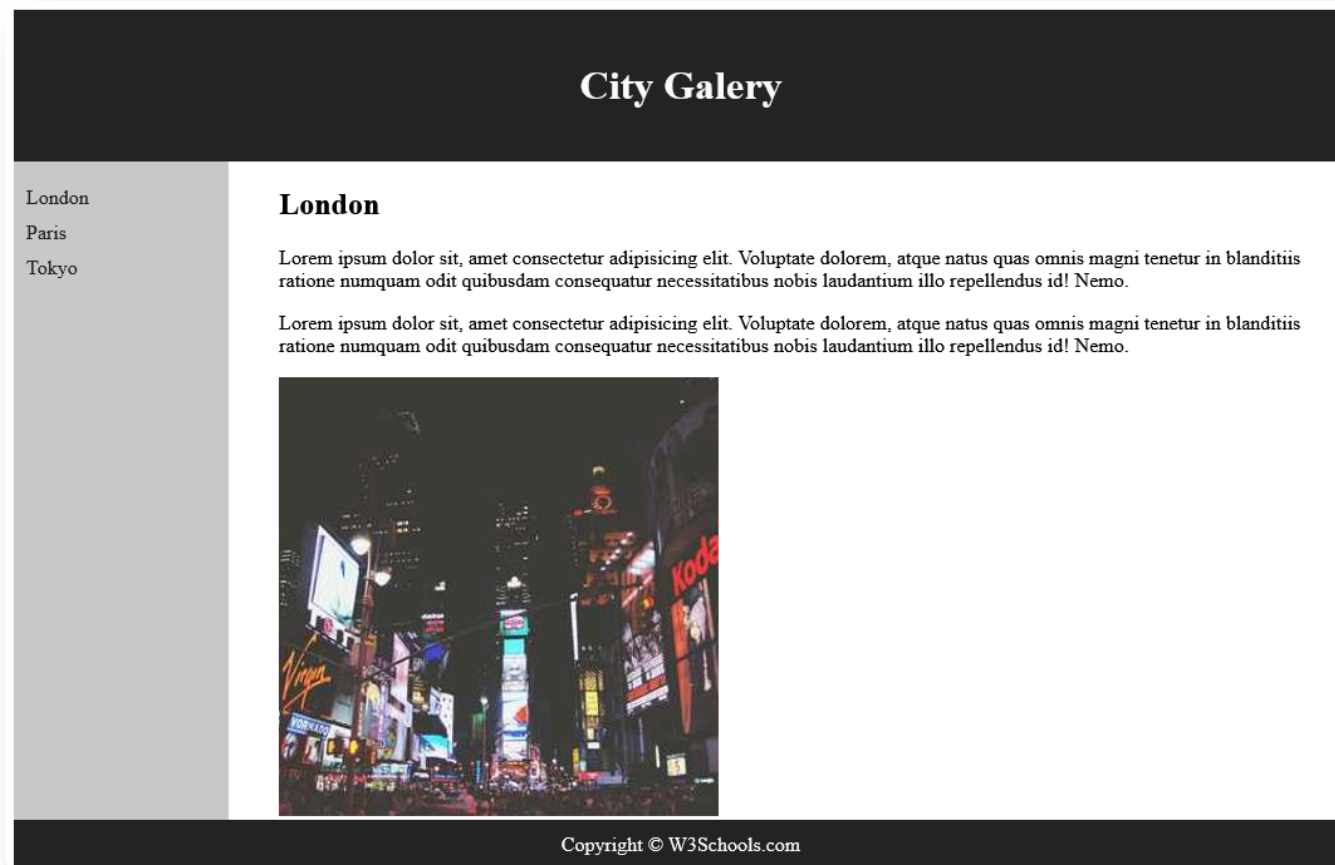
HTML Semântico

Layout com semântica

```
1 <header>
2   <h1>City Galery</h1>
3 </header>
4
5 <main>
6   <aside>
7     <nav>
8       <a href="#">London</a>
9       <a href="#">Paris</a>
10      <a href="#">Tokyo</a>
11    </nav>
12  </aside>
13
14  <section>
15    <h2>London</h2>
16    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Voluptate dolore, atque natus quas omnis magni
17      tenetur in blanditiis ratione numquam odit quibusdam consequatur necessitatibus nobis laudantium illo
18      repellendus id! Nemo.</p>
19
20    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Voluptate dolore, atque natus quas omnis magni
21      tenetur in blanditiis ratione numquam odit quibusdam consequatur necessitatibus nobis laudantium illo
22      repellendus id! Nemo.</p>
23
24    
25  </section>
26 </main>
27
28 <footer>
29   Copyright &copy; W3Schools.com
30 </footer>
```

HTML Semântico

Resultado em ambos os casos



Imagens

- Usamos a tag `img` para adicionar uma imagem em uma página Web
- Alguns formatos são comumente suportados por qualquer navegador, como o `jpg` e o `png`
- O formato `jpg` é bastante indicado por entregar um bom nível de compressão, tornando o arquivo da imagem menor (mais leve)
- O formato `png` é utilizado principalmente em logos ou outras imagens que precisam de algum nível de transparência
- Escolher o tamanho de imagem (do arquivo) adequado é essencial, visto que isso pode impactar no tempo de carregamento da página

Imagens

! Cuidado com direitos autorais

- Não só imagens, mas diversos conteúdos na Web podem estar protegidos por direitos autorais, então **cuidado**
- Evite pegar imagens diretamente da pesquisa do Google para usar em seus projetos (a menos que seja somente para estudo)
- Existem algumas opções de sites onde vocês poderão encontrar imagens de domínio público, como:
 - [Unsplash](#)
 - [Pexels](#)
 - [Freepik](#)

Imagens

Tá! Mas como coloco uma imagem na minha página?

- A sintaxe básica para inserir uma imagem no HTML é a seguinte:



```
1 
```

- O atributo `src` indica o local da imagem (pode ser um [URL](#) online)
- O atributo `alt` indica um texto alternativo, que aparece no lugar da imagem quando não é possível carregá-la

Imagens

Tá! Mas como coloco uma imagem na minha página?

- A sintaxe básica para inserir uma imagem no HTML é a seguinte:



```
1 
```

- O atributo `src` indica o local
- O atributo `alt` indica um texto para descrever a imagem carregada



Importante saber

O atributo `alt` é muito importante para acessibilidade na Web. Através dele, uma imagem pode ser descrita para um usuário com deficiência visual.

Imagens

Como posso pegar o caminho da imagem?

- Caso a imagem esteja na estrutura de pastas do seu projeto, você pode indicar o seu caminho relativo, como a seguir:



```
1 
```

- Nesse caso, a imagem está dentro de uma pasta chamada `img`, acessível a partir do arquivo HTML onde ela será inserida

Imagens

Como posso pegar o caminho da imagem?

- Você também pode passar a localização remota (URL) de uma imagem disponível na internet ou em um serviço de hospedagem de imagens, por exemplo:



```
1 
```

- Nesse caso, a imagem não está na estrutura de pastas do projeto do site, mas sim em um local (servidor) remoto na internet

Imagens

Outros atributos interessantes

- Podemos modificar as dimensões da imagem através dos atributos `width` e `height` (embora seja indicado fazer isso no CSS)
- O atributo global `title` pode ser usado para definir um título para a imagem, que deve aparecer ao passar o ponteiro do mouse por cima dela
- O atributo `loading` permite que o navegador retarde o carregamento das imagens que não são necessárias em determinado momento. Os valores podem ser `lazy` (adiar carregamento) ou `eager` (carregar imediatamente)

Tabelas

- Tabelas são úteis para estruturar dados
- São definidas com a tag `table` e suas linhas são definidas com a tag `tr`
- Para definir cada célula dentro de uma linha, usamos a tag `td` ou a tag `th` (se for uma célula de cabeçalho)
- Podemos definir áreas semânticas na tabela, através das tags `thead` (cabeçalho), `tbody` (corpo) e `tfoot` (rodapé)
- Se precisarmos mesclar linhas ou colunas, podemos usar o atributo `rowspan` (linha) ou `colspan` (coluna) na tag da célula `td`, passando a quantidade de linhas ou colunas a serem mescladas

Tabelas

- Qual código define a tabela abaixo? (estrutura)

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK

Tabelas

Estilizações básicas para tabelas

- Usamos a propriedade `border-collapse: collapse` para colapsar (juntar) as bordas da tabela e evitar bordas duplas como no exemplo abaixo:

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

Tabelas

Estilizações básicas para tabelas

- Link para código no CodePen:
<https://codepen.io/lucasifce/pen/gOyvLQZ>

Firstname	Lastname	Savings
Peter	Griffin	\$100
Lois	Griffin	\$150
Joe	Swanson	\$300
Cleveland	Brown	\$250

```
1  table {
2      width: 100%;
3      background-color: #f1f1c1;
4      border-collapse: collapse;
5  }
6
7  td,
8  th {
9      padding: 5px;
10 }
11
12 tr:nth-child(odd) {
13     background-color: #eee;
14 }
15
16 tr:nth-child(even) {
17     background-color: #fff;
18 }
19
20 th {
21     color: white;
22     background-color: black;
23 }
```


Formulários

Formulários permitem a **interação com o usuário** e o **envio de dados** ao servidor

- A tag `<form>` define um elemento que contém componentes de controle interativos na interface, como campos de texto e senha, elementos de seleção, caixas de seleção (*checkbox*), seleção de arquivos e botões
- As tags abaixo definem os principais componentes de controle interativo de um formulário:
 - `<input>` - define um campo de entrada de dados (texto, senha, email, etc)
 - `<select>...</select>` - define uma lista suspensa com opções
 - `<textarea>...</textarea>` - define um campo para entrada de texto longo (múltiplas linhas)
 - `<button>...</button>` - define um botão clicável

Formulários

 Exemplo básico de formulário HTML

Nome completo:

Idade:

Email:

Semestre:



```
1  <form>
2      <label for="nome">Nome completo:</label>
3      <input type="text" id="nome" name="nome">
4
5      <label for="idade">Idade:</label>
6      <input type="number" id="idade" name="idade">
7
8      <label for="email">Email:</label>
9      <input type="email" id="email" name="email">
10
11     <label for="semestres">Semestre:</label>
12     <select id="semestres" name="semestres">
13         <option value="s1">1º Semestre</option>
14         <option value="s2">2º Semestre</option>
15         <option value="s3">3º Semestre</option>
16     </select>
17 </form>
```

Formulários

⚠ A tag `label` é importante

- Você deve ter identificado o uso de uma tag `<label>` no exemplo anterior. Essa tag define um elemento de **rótulo** para um elemento de controle do formulário
- Esse elemento é bastante útil para usuários de **leitores de tela**, pois o leitor de tela irá ler o texto do **label** quando o usuário clicar no elemento de entrada correspondente
- Além disso, quando o usuário clicar no texto da tag **label**, o campo de entrada correspondente será selecionado, sendo particularmente útil quando existem elementos de entrada com tamanhos muito pequenos (facilita sua seleção)
- Para tanto, é necessário que o valor do atributo **for** na tag **label** corresponda exatamente com o valor do atributo **id** do elemento de entrada correspondente

Formulários

Elemento *input*

- Elemento mais utilizado em um formulário, existindo muitas variações (tipos) a depender do valor do atributo **type**
 - Texto - `<input type="text">`
 - Número - `<input type="number">`
 - Email - `<input type="email">`
 - Senha - `<input type="password">`

Nome completo:

Lucas Mendes

Idade:

25

Email:

lucas.mendes.wbds@gmail.com

Senha:

•••••

Formulários

Elemento *input*

- A **definição correta do tipo** do elemento facilita o processo de **validação dos dados**, pois os próprios navegadores podem realizar uma validação inicial

Idade:

Digite um número.

Email:


Digite um endereço de email.

Formulários

Elemento *input*: mais alguns tipos

- *Data* - `<input type="date">`
- *Arquivo* - `<input type="file">`
- *URL* - `<input type="url">`
- *Controle deslizante* - `<input type="range">`

Data de Nascimento:

Comprovante de Endereço:

Nenhum arquivo selecionado.

URL do Perfil no GitHub:


Nível de Fluência em Inglês:

Formulários

Elemento *input*: mais alguns tipos

- *Data* - `<input type="date">`
- *Arquivo* - `<input type="file">`
- *URL* - `<input type="url">`
- *Controle deslizante* - `<input type="range">`

Data de Nascimento:

dd / mm / aaaa 

<

maio de 2024 ▾

>

dom.	seg.	ter.	qua.	qui.	sex.	sáb.
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Limpar

Formulários

Elemento *input*: elementos de seleção

- Podemos definir dois tipos básicos de elementos de seleção na tag `input`, usando os tipos `radio` ou `checkbox`
- Os botões de opção (**radio buttons**) definidos com `<input type="radio">` possibilitam ao usuário **escolher uma única opção** em um conjunto limitado de opções possíveis

```
1 <form>
2   <p>Qual sua linguagem de programação favorita?</p>
3   <input type="radio" name="fav_language" id="java" value="Java">
4   <label for="java">Java</label><br>
5   <input type="radio" name="fav_language" id="js" value="JavaScript">
6   <label for="js">JavaScript</label><br>
7   <input type="radio" name="fav_language" id="python" value="Python">
8   <label for="python">Python</label><br>
9 </form>
```

Qual sua linguagem de programação favorita?

- ☒ Java
☐ JavaScript
☐ Python

Formulários

Elemento *input*: elementos de seleção

- Podemos definir dois tipos básicos de elementos de seleção na tag `input`, usando os tipos `radio` ou `checkbox`
- Os botões de opção (`radio buttons`) definidos com o mesmo valor no atributo `value` permitem escolher uma única opção em um conjunto limitado.

! Perceba que é necessário definir o mesmo valor no atributo `name` para todos os `radio buttons` pertencentes a um mesmo grupo.

```
1 <form>
2   <p>Qual sua linguagem de programação favorita?</p>
3   <input type="radio" name="fav_language" id="java" value="Java">
4   <label for="java">Java</label><br>
5   <input type="radio" name="fav_language" id="js" value="JavaScript">
6   <label for="js">JavaScript</label><br>
7   <input type="radio" name="fav_language" id="python" value="Python">
8   <label for="python">Python</label><br>
9 </form>
```

Qual sua linguagem de programação favorita?

- ☒ Java
☐ JavaScript
☐ Python

Formulários

Elemento *input*: elementos de seleção

- Podemos definir dois tipos básicos de elementos de seleção na tag `input`, usando os tipos `radio` ou `checkbox`
- As caixas de seleção (`checkboxes`) definidos com `<input type="checkbox">` possibilitam ao usuário **escolher várias opções** em um conjunto limitado de opções possíveis

```
1 <form>
2   <p>Você programa em quais linguagens?</p>
3   <input type="checkbox" name="java" id="java" value="Java">
4   <label for="java">Java</label><br>
5   <input type="checkbox" name="js" id="js" value="JavaScript">
6   <label for="js">JavaScript</label><br>
7   <input type="checkbox" name="python" id="python" value="Python">
8   <label for="python">Python</label><br>
9 </form>
```

Você programa em quais linguagens?

- ☒ Java
- ☒ JavaScript
- ☐ Python

Formulários

Elemento *input*: botões

- O elemento **input** também é utilizado para definir botões no formulário
 - Envio de dados - `<input type="submit" value="Enviar">`
 - O botão de submit é utilizado para enviar os dados do form para um script (geralmente no servidor) que manipula e processa esses dados
 - Geralmente o script manipulador do formulário é indicado pelo atributo *action* da tag `<form>`
 - Reset dos dados - `<input type="reset" value="Limpar">`
 - Botão clicável - `<input type="reset" value="Clique Aqui">`

Formulários

Elemento *input*: botões






```
1  <form>
2    <label for="nome">Nome completo:</label>
3    <input type="text" id="nome" name="nome">
4
5    <label for="email">Email:</label>
6    <input type="email" id="email" name="email">
7
8    <div>
9      <input type="submit" value="Enviar">
10     <input type="reset" value="Limpar">
11     <input type="button" value="Clique Aqui">
12   </div>
13 </form>
```

Nome completo:

Email:

Conteúdos Extras

Se aprofunde e vá além!

-  W3Schools:
 - https://www.w3schools.com/html/html_blocks.asp
 - https://www.w3schools.com/html/html5_semantic_elements.asp
-  MDN WebDocs:
 - https://developer.mozilla.org/pt-BR/docs/Glossary/Block-level_content
-  Site DPW (Desenvolvimento para Web):
 - <https://desenvolvimentoparaweb.com/html/web-semantic-7-alternativas-div/>

Isso é tudo, pessoal!

Sigam-me nas redes abaixo 📍 😊

 @lucas-lfm

 @prof_lucasmendes

 @prof-lucasmendes

Email: lucas.mendes@ifce.edu.br

Prof. Me. Lucas Mendes | GitHub: [@lucas-lfm](https://github.com/lucas-lfm)

Tecnologias Web 🧑💻

Referências

- <https://www.w3schools.com/html>
- <https://developer.mozilla.org/pt-BR/docs/Web/HTML>
- <https://desenvolvimentoparaweb.com/>
- <https://github.com/gustavoguanabara/html-css>

Email: lucas.mendes@ifce.edu.br

Prof. Me. Lucas Mendes | GitHub: [@lucas-lfm](https://github.com/lucas-lfm)

Tecnologias Web 🧑‍💻