

CURSO COMPLETO

HTML & CSS para iniciantes



INSTRUTOR
LUIZ AUGUSTO VIEIRA



Pré-requisitos

Para iniciar este Curso, você não precisa de nenhum conhecimento prévio em **HTML** e **CSS**. Porém, você deve ter ao menos uma familiaridade básica em utilizar computadores e utilizar a **Web** passivamente (apenas navegando, consumindo conteúdo).

COMO FUNCIONA A WEB?

Introdução a Web

A **Web**, abreviação de " **World Wide Web** " (que nos dá a sigla *www*), é o nome de uma das maneiras que a Internet permite que as pessoas naveguem em documentos conectados por links de hipertexto.

O conceito da Web foi aperfeiçoado no CERN (*Conseil Européen pour la Recherche Nucléaire* , ou a *Organização Europeia para Pesquisa Nuclear*) em 1991 por um grupo de pesquisadores que incluía Tim-Berners Lee, o criador do hiperlink, que é hoje considerado o pai da web.

Introdução a Internet

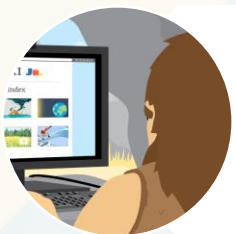
O princípio da Web é baseado no uso de hiperlinks para navegar entre documentos (chamados páginas da Web) com um programa chamado navegador.

Uma página da Web é um arquivo de texto simples escrito em uma linguagem de marcação chamada (HTML) que codifica o layout do documento, elementos gráficos e links para outros documentos, tudo com a ajuda de Tags.

Além dos links que conectam documentos formatados uns aos outros, a Web usa o protocolo HTTP para vincular documentos hospedados em computadores distantes chamados (Servidores Web) em oposição ao cliente representado pelo servidor.

Na internet, documentos são identificados com um endereço único, chamado de URL, que pode ser usado para localizar qualquer recurso na internet, não importando qual servidor possa estar hospedando-o.

O que é um site?



Saiba mais sobre sites:
Estáticos e Dinâmicos.

Um site é um grupo de arquivos HTML que são armazenados em um computador Host que está permanentemente conectado à Internet (um servidor web).

Um site normalmente é construído em torno de uma página central, chamada de página de boas - vindas, que oferece links para um grupo de outras páginas hospedadas no mesmo servidor e também links externos, que levam as páginas hospedadas para outro servidor.

Uma URL é algo assim: `http://www.htmlecsspro.com`

O “`http://`” indica que queremos navegar na Web usando o protocolo HTTP, o protocolo padrão para navegar na Web. Existem outros protocolos para outros usos da internet.

O `www.htmlecsspro.com`, corresponde ao endereço do servidor que hospeda as páginas da Web. Por convenção, os servidores da Web têm um nome que começa com `www`, para deixar claro que são servidores Web dedicados e facilitar a memorização do endereço.

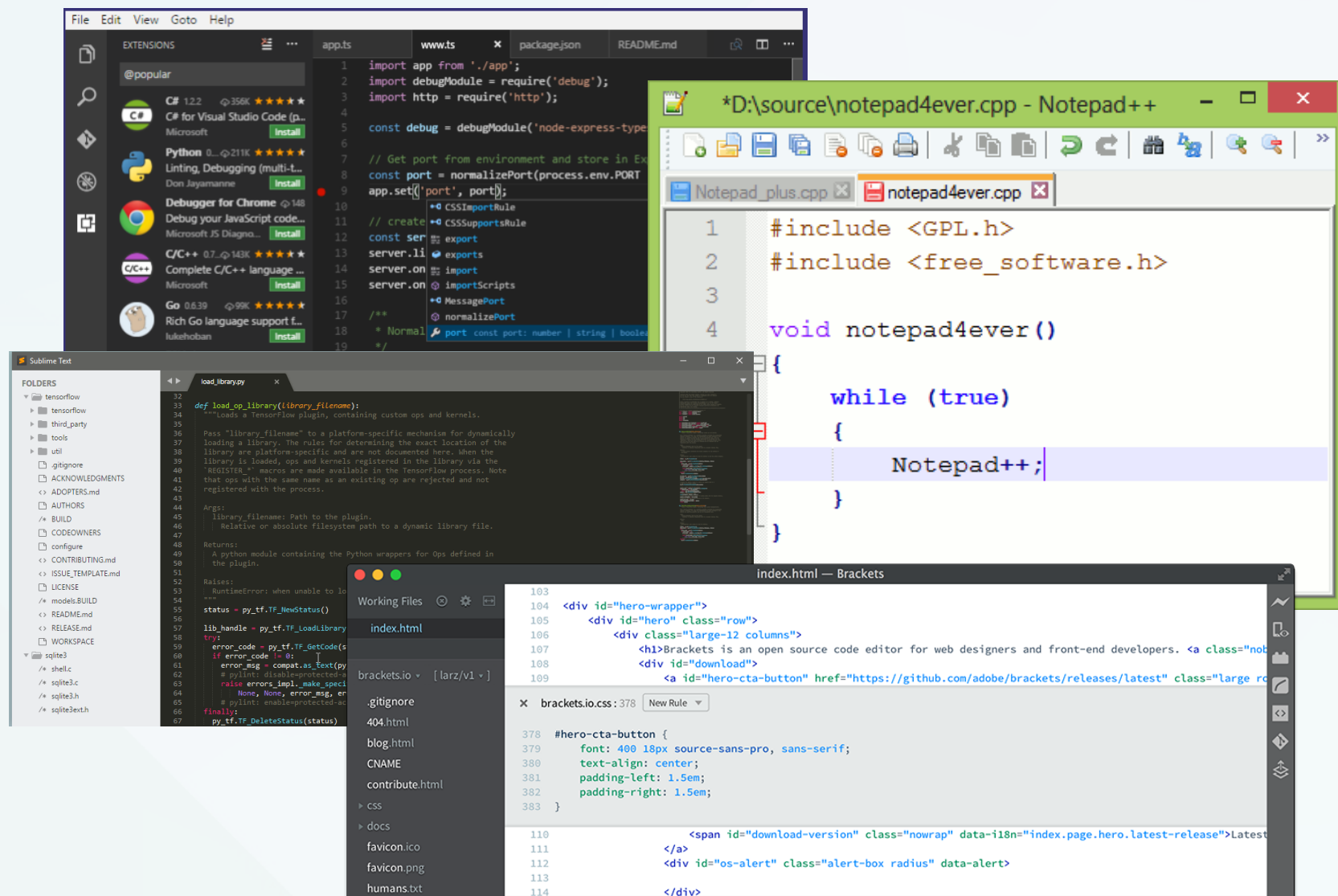
Editores ou IDEs

1. VISUAL STUDIO CODE

2. NOTEPAD++

3. SUBLIME TEXT

4. BRACKETS



<HTML>

Introdução ao HTML

HTML (Hypertext Markup Language), é a linguagem de marcação padrão usada para criar páginas da web.

HTML consiste em elementos que definem as diferentes partes de um documento.

O desenvolvedor usa tags HTML para especificar quais elementos vão em uma determinada página.

Por exemplo, você pode usar tags HTML para especificar cabeçalhos, parágrafos, listas, tabelas, imagens e entre outros.

O que é preciso para criar página HTML?

Você não precisa de nenhum equipamento ou software especial para criar páginas HTML. Você provavelmente já tem tudo o que precisa para começar a criar páginas que é:

Computador e um Editor de texto.

Primeiros passos

Vamos criar uma página da Web usando HTML e visualizá-la em um navegador.

Você aprenderá como é fácil criar uma página da web. De fato, quando você construir esta página, você terá criado sua própria página web!

Quando você vai criar uma página da web, você geralmente faz algo assim:

1. Cria um arquivo HTML
2. Digita algum código HTML neste arquivo
3. Visualiza o resultado no navegador

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Título da página</title>
5   </head>
6
7   <body>
8     <h1>Cabeçalho da minha primeira página.</h1>
9     <p>Este é meu primeiro parágrafo.<p>
10  </body>
11 </html>
```



Primeira página

Apenas codificamos algumas tags HTML . Essas tags dizem ao navegador o que exibir e onde. Note que para cada tag de "abertura" há também uma tag de "fechamento" e o conteúdo que queremos exibir aparece no meio. A maioria das tags HTML tem uma tag de abertura e fechamento.

No próximo slide você verá em detalhes um pouco mais sobre tags HTML e entenderá o código acima.

Elementos HTML

Elementos HTML são os fundamentos do documento HTML, são simplesmente um arquivo de texto composto de elementos HTML. Esses elementos são definidos usando tags.

Um elemento HTML é um componente individual de um documento HTML. Qualquer página da Web consiste em muitos elementos HTML.

As tags HTML informam ao navegador quais elementos apresentar e como apresentá-los. Onde o elemento aparecerá é determinado pela ordem em que as marcas aparecem.

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Título da página</title>
5    </head>
6
7    <body>
8      <h1>Cabeçalho da minha primeira página.</h1>
9      <p>Este é meu primeiro parágrafo.</p>
10   </body>
11 </html>
```

Explicação do código

<!DOCTYPE html> define que este documento seja HTML5

<html> elemento raiz de uma página HTML

<head> elemento contém informações meta sobre o documento

<title> elemento especifica um título para o documento

<body> elemento contém o conteúdo da página

<h1> elemento define um cabeçalho (título)

<p> elemento define um parágrafo

Formatação HTML

Vamos analisar a formatação e a semântica de um documento HTML.

HTML fornece a estrutura do documento consistindo em todos os elementos HTML individuais em cada página e também fornece a semântica de um documento HTML.

Elementos são geralmente usados para um significado particular. Por exemplo, um nível de título 1 é mais importante que um nível de título 2.

A maior parte da formatação e estilo de páginas da Web é feita usando CSS . No entanto, os navegadores formatam determinados elementos HTML de maneira padrão, sem a necessidade de CSS.

CSS é apenas para melhorar a apresentação de um elemento da maneira padrão como o navegador vai apresentá-lo.

No próximo slide, veremos alguns exemplos de elementos HTML comuns, com a explicação de seu uso, bem como um exemplo do seu formato básico quando aplicados a um documento HTML.

Formatação HTML

Cabeçalhos

Esta é uma tag especial para especificar cabeçalhos no HTML. Existem 6 níveis de títulos desde o **<h1>** o mais importante, até o **<h6>** o menos importante.

1	<code><h1>Cabeçalho 1</h1></code>
2	<code><h2>Cabeçalho 2</h2></code>
3	<code><h3>Cabeçalho 3</h3></code>
4	<code><h4>Cabeçalho 4</h4></code>
5	<code><h5>Cabeçalho 5</h5></code>
6	<code><h6>Cabeçalho 6</h6></code>

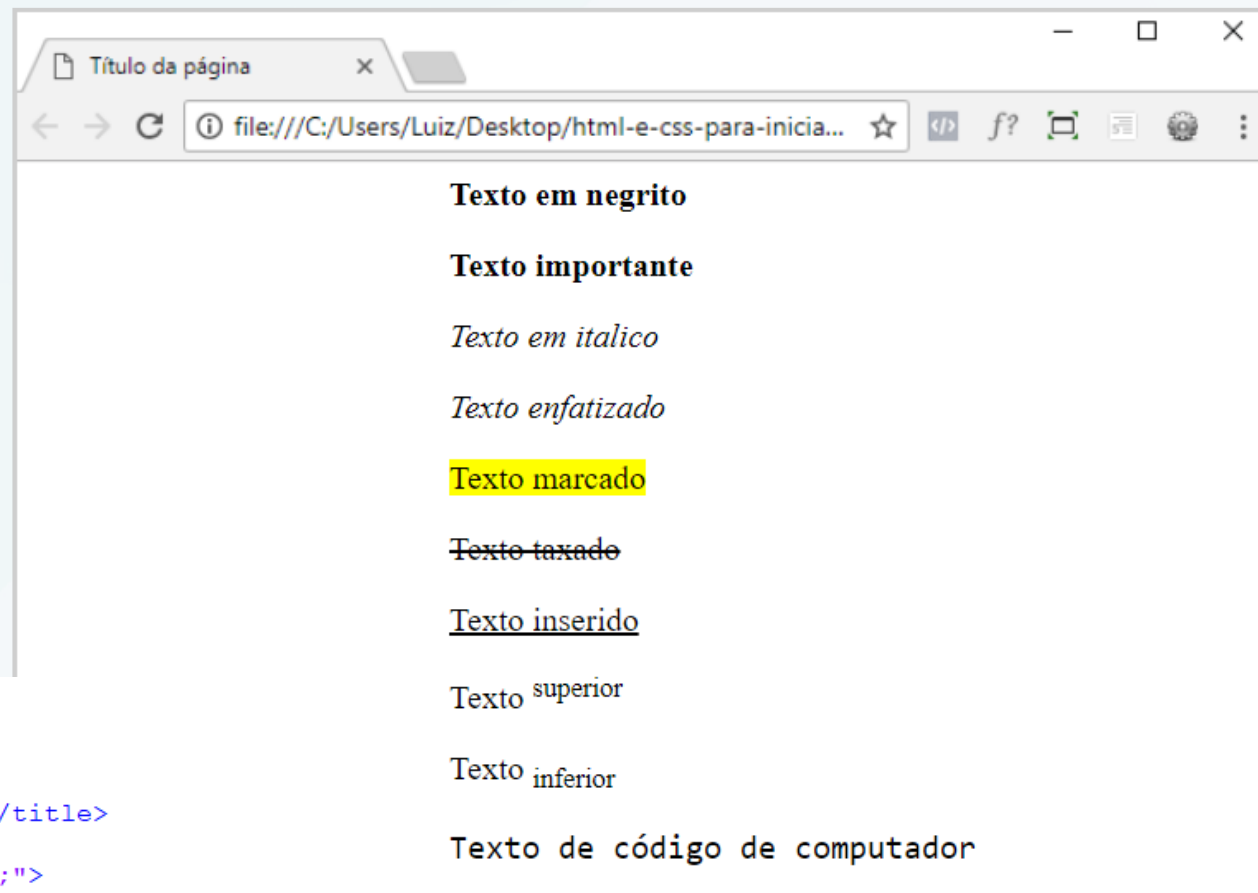


Formatação HTML

HTML tem elementos especiais para a definição de texto com um significado especial. HTML usa elementos como `` e `<i>` para a formatação de saída, como texto negrito ou itálico.

- `` - Texto em negrito
- `` - Texto Importante
- `<i>` - Texto em itálico
- `` - Texto enfatizado
- `<mark>` - Texto marcado
- `` - Texto excluído
- `<ins>` - Texto inserido
- `<sup>` - Texto sobrescrito
- `<sub>` - Texto subscrito
- `<code>` - Texto de código

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body style="font-size: 20px;">
8     <p><b>Texto em negrito</b></p>
9     <p><strong>Texto importante</strong></p>
10    <p><i>Texto em italico</i></p>
11    <p><em>Texto enfatizado</em></p>
12    <p><mark>Texto marcado</mark></p>
13    <p><del>Texto taxado</del></p>
14    <p><ins>Texto inserido</ins></p>
15    <p>Texto <sup>superior</sup></p>
16    <p>Texto <sub>inferior</sub></p>
17    <p><code>Texto de código de computador</code></p>
18  </body>
19 </html>
```



Atributos HTML

Os atributos são elementos adicionados ao HTML para fornecer mais informações sobre esse elemento.

Tags HTML podem conter um ou mais atributos . Atributos são adicionados a uma tag para fornecer ao navegador mais informações sobre como a tag deve aparecer ou se comportar. Os atributos consistem em um nome e um valor separados por um sinal de igual (=).

```
<abbr title="HTML e CSS Pro">HeCP</abbr>
```

Este exemplo usa a tag `<abbr>` que é usada para indicar uma abreviação. Mas veja que foi atribuído o atributo `"title"` com um valor de HTML e CSS Pro.

O atributo `"title"` pode opcionalmente ser usado em qualquer elemento HTML para fornecer informações extras sobre o conteúdo do elemento. Quando usado com a tag `<abbr>`, permite o uso para fornecer uma expansão da abreviação ou sigla.

Ao usar o atributo `"title"`, a maioria dos navegadores exibirá seu valor como uma "dica de ferramenta" quando o usuário passar o mouse por cima do elemento.

Você pode adicionar mais de um atributo a um determinado elemento.

Links HTML

Links são conhecidos como hyperlinks e são definidos usando a tag `<a>`, conhecida como o elemento de âncora.

Para criar um hiperlink, use a tag `<a>` em conjunto com o atributo `href`. O valor do atributo `href` é a URL ou a localização de onde o link está apontando.

Podemos usar URL absoluta e URL relativa.

Absoluta: Isso se refere a uma URL em que o caminho completo é fornecido. Por exemplo:

```
<a href="https://www.htmlecsspro.com">HTML e CSS Pro</a>
```

Relativa: Isso se refere a uma URL em que o caminho é relativo ao local atual. Por exemplo:

```
<a href="/curso/">Curso HTML</a>
```

Links de email

Você pode criar hiperlink para um endereço de e-mail. Para fazer isso, use o atributo `mailto` em sua tag de âncora.

```
<a href="mailto:contato@htmlecsspro.com">E-mail para contato</a>
```

Imagem no HTML

As imagens estão presente na grande maioria dos sites, o HTML torna muito fácil para inserir imagens em uma página web.

Para incorporar uma imagem em uma página da Web, a imagem precisa primeiro existir nos formatos (.jpg, .gif ou .png) .

Você pode criar imagens em um editor de imagens (como o Adobe Photoshop, GIMP , Adobe Illustrator, etc.) e salvá-las em um dos formatos acima.

Depois de criar uma imagem, você precisa incorporá-la à sua página da web. Para incorporar, use a tag ``, especificando a localização da imagem.

Exemplo:

```

```

O ``, elemento acima, contém vários atributos. Esses atributos informam ao navegador tudo sobre a imagem e como exibi-la.

Meta Tags HTML

A tag <meta> fornece metadados sobre o documento HTML. Os metadados não serão exibidos na página, mas poderão ser analisados por máquina. Os elementos meta geralmente são usados para especificar a descrição da página, as palavras-chave, o autor do documento, a última modificação e outros metadados.

Os metadados podem ser usados por navegadores (como exibir conteúdo ou recarregar a página), mecanismos de pesquisa (palavras-chave) ou outros serviços da Web. Podemos adicionar **metadados** às páginas da web colocando entre o elemento <head>.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6
7     <meta name="keywords" content="HTML, CSS, JavaScript"> <!--Palavras-chave-->
8     <meta name="description" content="Aprenda tudo sobre HTML e CSS!"> <!--Descrição-->
9     <meta name="author" content="HTML e CSS Pro"> <!--Autor-->
10    <meta http-equiv="refresh" content="10"> <!--Atualiza a página a cada 10 segundos-->
11  </head>
12
13  <body>
14    CORPO DA PÁGINA
15  </body>
16 </html>
```

Comentário no HTML

O que é um comentário HTML?

Um comentário HTML é uma parte do código HTML que é usado para explicar o código. Isso pode ser útil para outros codificadores HTML ao tentar interpretar o código de outra pessoa.

Também pode ser útil se você tiver que revisitar seu código há meses ou até anos depois. Os comentários não são exibidos no navegador, eles simplesmente estão lá para o benefício do codificador.

Os comentários podem ser usados para introduzir seções principais do código ou podem ser usados para explicar o que uma pequena seção de código faz.

Eles também costumam ser usados na parte superior de um documento para explicar coisas como autor, data de criação, finalidade do arquivo, informações sobre arquivos relacionados e entre outros.

Como escrever comentários no HTML?

Os comentários sempre começam `<!--` e terminam com `-->`. Isso informa ao navegador quando um comentário começa e termina. Volte ao slide anterior e veja os exemplos!

Formulário no HTML

Vamos entender os fundamentos da criação de formulários no HTML.

Um formulário HTML é composto de elementos e esses elementos permitem que o usuário faça coisas como inserir informações ou fazer uma seleção de opções pré-definidas.

Um formulário é definido entre essas duas tags:

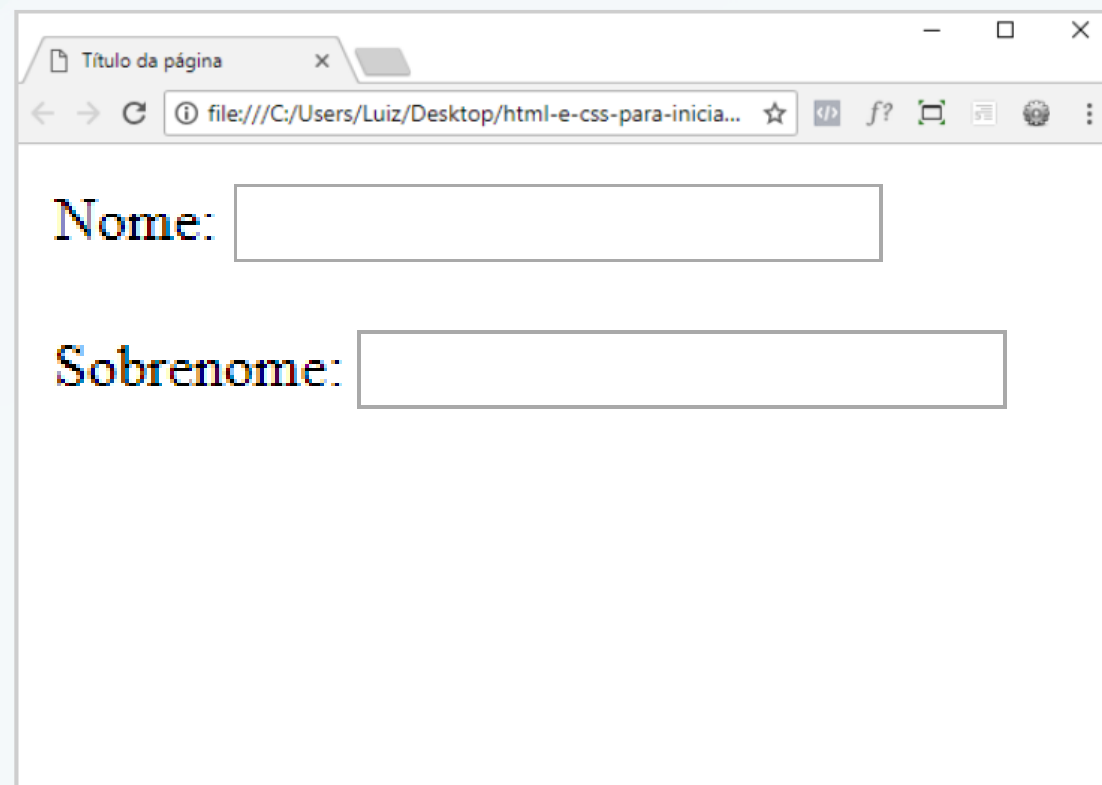
```
<form>  
    ...  
    Elementos do formulário  
    ...  
</form>
```

Formulário no HTML

A tag `<input>`

Essa é a tag mais usada nos formulários HTML. Ela permite que você especifique vários tipos de campos de entrada do usuário, como texto, botões de opção, caixas de seleção e etc.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form>
9       Nome: <input type="text" name="nome"><br /><br />
10      Sobrenome: <input type="text" name="sobrenome">
11    </form>
12  </body>
13 </html>
```



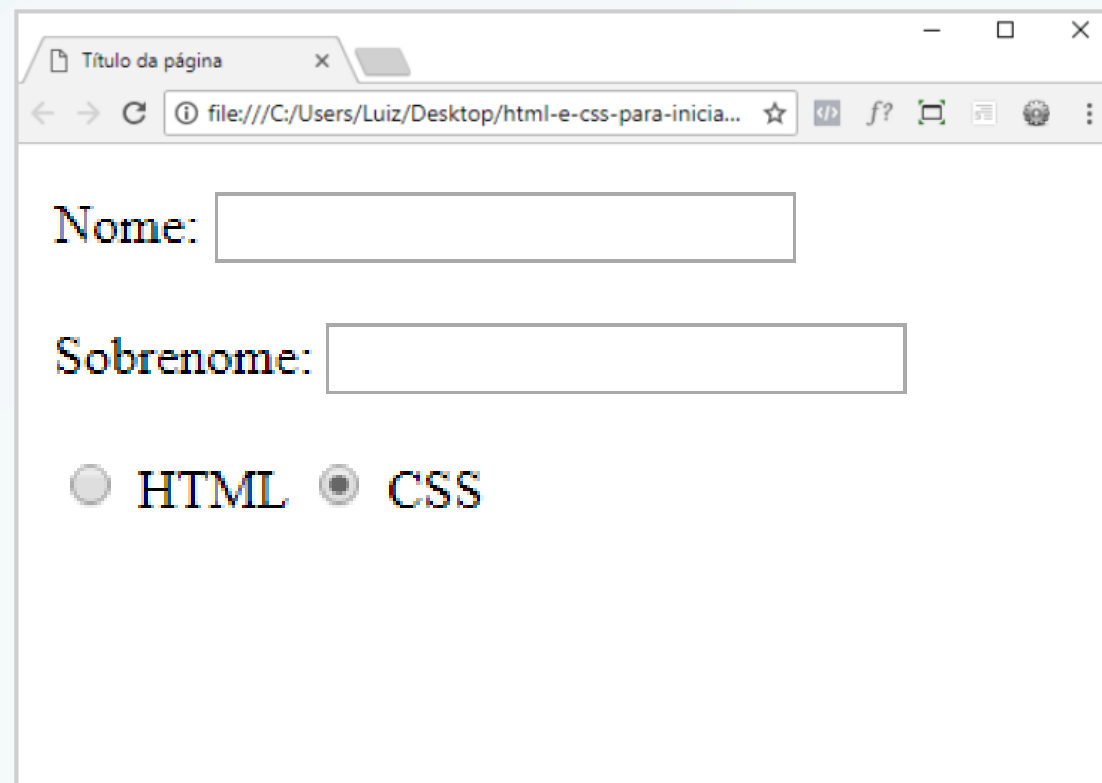
The screenshot shows a web browser window with the title "Título da página". The address bar shows the file path: "file:///C:/Users/Luiz/Desktop/html-e-css-para-inicia...". The browser displays the rendered HTML form, which consists of two text input fields. The first field is labeled "Nome:" and the second field is labeled "Sobrenome:". Both fields are empty and have a light gray border.

Formulário no HTML

Botões do rádio (Radio Button).

Esses tipos de botões, são chamados de botões opção e são usados quando você deseja que o usuário selecione apenas uma opção de um conjunto predeterminado de opções.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form>
9       Nome: <input type="text" name="nome"><br /><br />
10      Sobrenome: <input type="text" name="sobrenome"><br /><br />
11      <input type="radio" name="opcoes" value="html"> HTML
12      <input type="radio" name="opcoes" value="css"> CSS
13    </form>
14  </body>
15 </html>
```

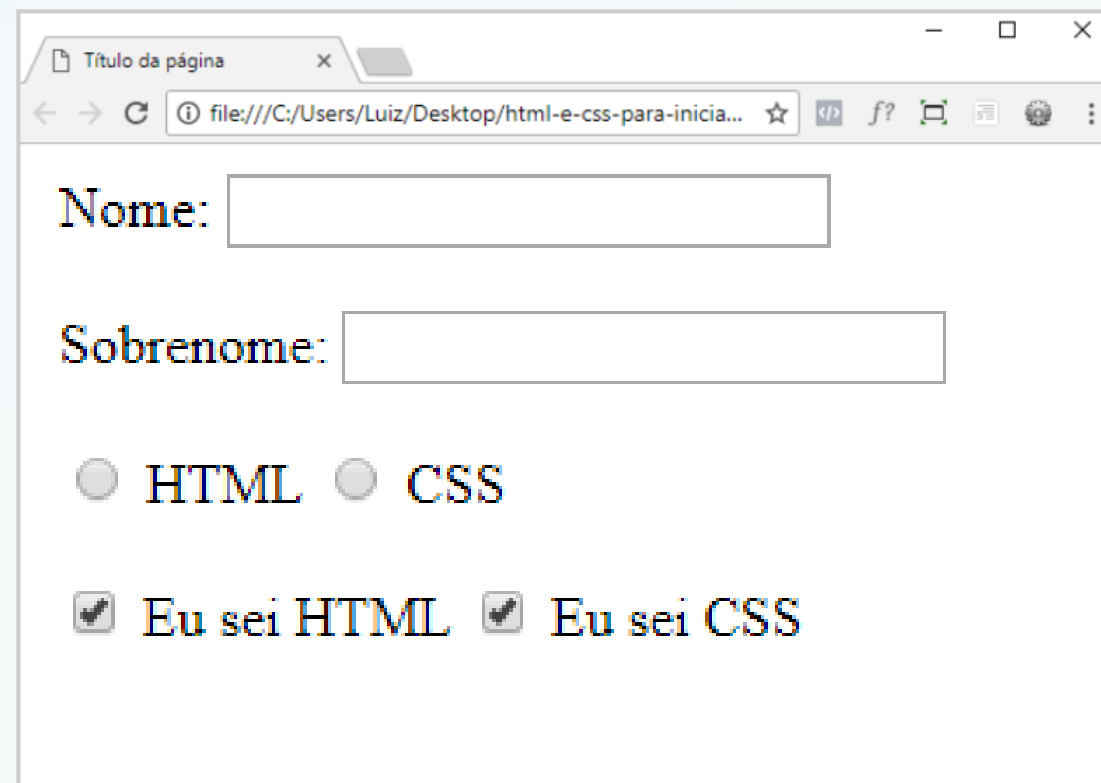


Formulário no HTML

Caixas de seleção (Checkbox).

As caixas de seleção são semelhantes aos botões de opção, porém permitem que o usuário faça várias seleções. Devem ser usadas quando se quiser permitir que os usuários façam mais de uma seleção.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form>
9       Nome: <input type="text" name="nome"><br /><br />
10      Sobrenome: <input type="text" name="sobrenome"><br /><br />
11      <input type="radio" name="opcoes" value="html"> HTML
12      <input type="radio" name="opcoes" value="css"> CSS<br /><br />
13      <input type="checkbox" name="technology" value="HTML"> Eu sei HTML
14      <input type="checkbox" name="technology" value="CSS"> Eu sei CSS
15    </form>
16  </body>
17</html>
```



The screenshot shows a web browser window with the title "Título da página". The address bar shows the file path: "file:///C:/Users/Luiz/Desktop/html-e-css-para-inicia...". The form is rendered as follows:

Nome:

Sobrenome:

☐ HTML ☐ CSS

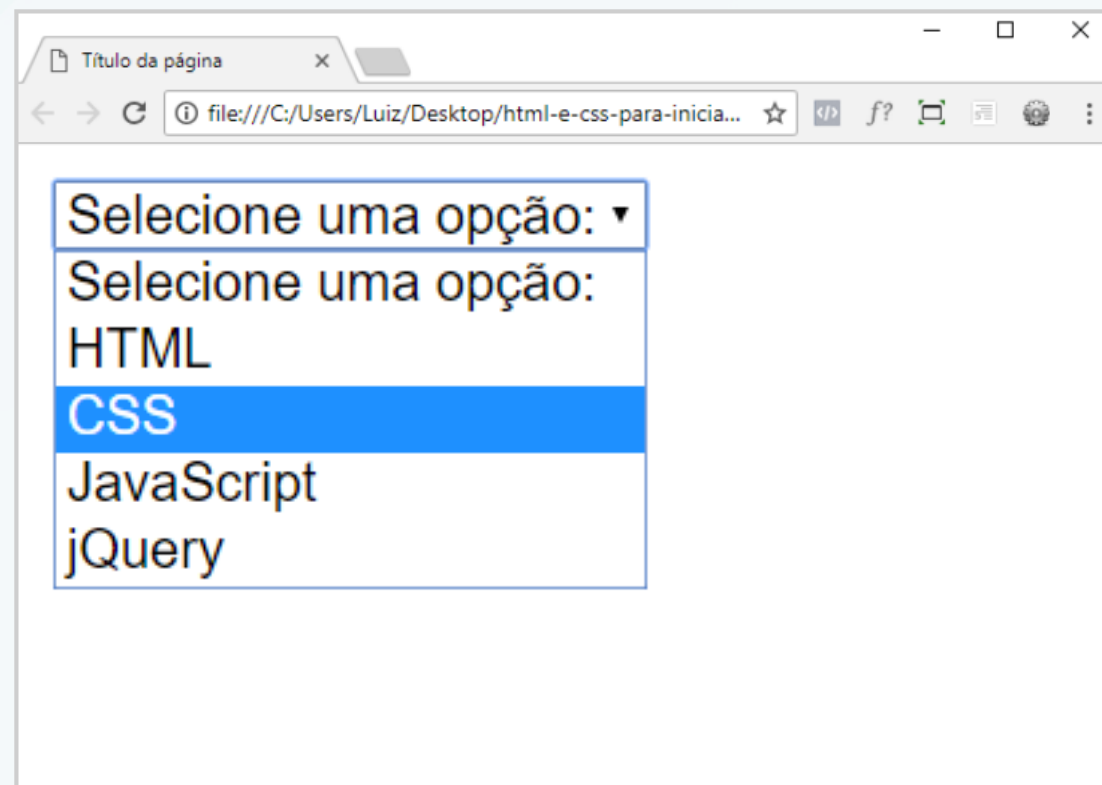
☒ Eu sei HTML ☒ Eu sei CSS

Formulário no HTML

Lista de Seleção (Select).

Lista suspensa com opções que permite que o usuário selecione uma opção em uma lista de opções predefinidas. A lista de seleção é criada usando o `<select>` elemento em conjunto com o `<option>` elemento.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form>
9       <select name="opcoes" style="font-size: 20px;">
10        <option value="" selected>Selecione uma opção:</option>
11        <option value="html">HTML</option>
12        <option value="css">CSS</option>
13        <option value="javascript">JavaScript</option>
14        <option value="jquery">jQuery</option>
15      </select>
16    </form>
17  </body>
```



Formulário no HTML

Textarea.

Pode-se usar o elemento `<textarea>` para permitir que os usuários insiram blocos de texto maiores do que com a tag `<input>`. Geralmente, pode-se usar o atributo **maxlength** para restringir a entrada do usuário a um determinado número de caracteres. Pode-se usar também os atributos **cols** e **rows** para ajustar a largura e a altura.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form>
9       <textarea name="msg" rows="10" cols="30">
10        texto texto texto texto ...
11      </textarea>
12    </form>
13  </body>
14 </html>
```



Formulário no HTML

Ação de Formulário

Geralmente, quando um usuário envia o formulário, o sistema deve fazer alguma ação com os dados. É aí que entra a página de ação. A página de ação é a página para a qual o formulário é submetido.

Esta página pode conter scripts avançados ou programação que insere os dados do formulário em um banco de dados ou envia para um administrador por e-mail.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <form action="envia-dados.php" method="post">
9       Nome: <input type="text" name="nome" />
10      Sobrenome: <input type="text" name="sobrenome" />
11      <input type="submit" value="Submit" />
12    </form>
13  </body>
14 </html>
```

Usamos o atributo method. Este atributo especifica o método HTTP a ser usado quando o formulário é submetido.

Os valores possíveis são:

GET

Os dados do formulário são anexados a URL quando enviados. Isso significa que você pode ver as variáveis de formulário na barra de endereços do seu navegador quando o formulário é enviado.

POST

Os dados do formulário não são anexados ao URL. Este atributo é opcional. Se você não fornecer, o método será postado.

Tabela no HTML

Tabela é usada para apresentar dados tabulares.

As tabelas permitem que você apresente dados tabulares de maneira agradável e estruturada. Pois elas apresentam dados dentro de uma grade com linhas e colunas.

Elementos básicos da tabela:

Para criar tabelas usamos o elemento `<table>`, em conjunto com outros elementos `<tr>` e `<td>`.

Cada conjunto de tags `<tr>` representa uma linha dentro da tabela na qual eles estão aninhados. E cada conjunto de tags `<td>` representa uma célula de dados da tabela na linha em que eles estão aninhados.

Podemos também adicionar cabeçalhos de tabela usando o elemento `<th>`.

Tabela no HTML

Criamos uma tabela e passamos alguns elementos: (cellpadding="5" cellspacing="5" border="1").

1. O "cellpadding", vai dar espaços dentro da coluna.
2. O "cellspacing", vai espaços entre as colunas.
3. A "border", vai colocar uma border na tabela e colunas.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body>
8     <table cellpadding="5" cellspacing="5" border="1">
9       <tr>
10        <th>Nome</th>
11        <th>Telefone</th>
12        <th>Endereço</th>
13      </tr>
14      <tr>
15        <td>Luiz Augusto Vieira</td>
16        <td>11 92222-3333</td>
17        <td>Rua Xpto, 99</td>
18      </tr>
19      <tr>
20        <td>João Martins</td>
21        <td>91 94444-5555</td>
22        <td>Rua Da Paz, 44</td>
23      </tr>
24    </table>
25  </body>
26 </html>
```



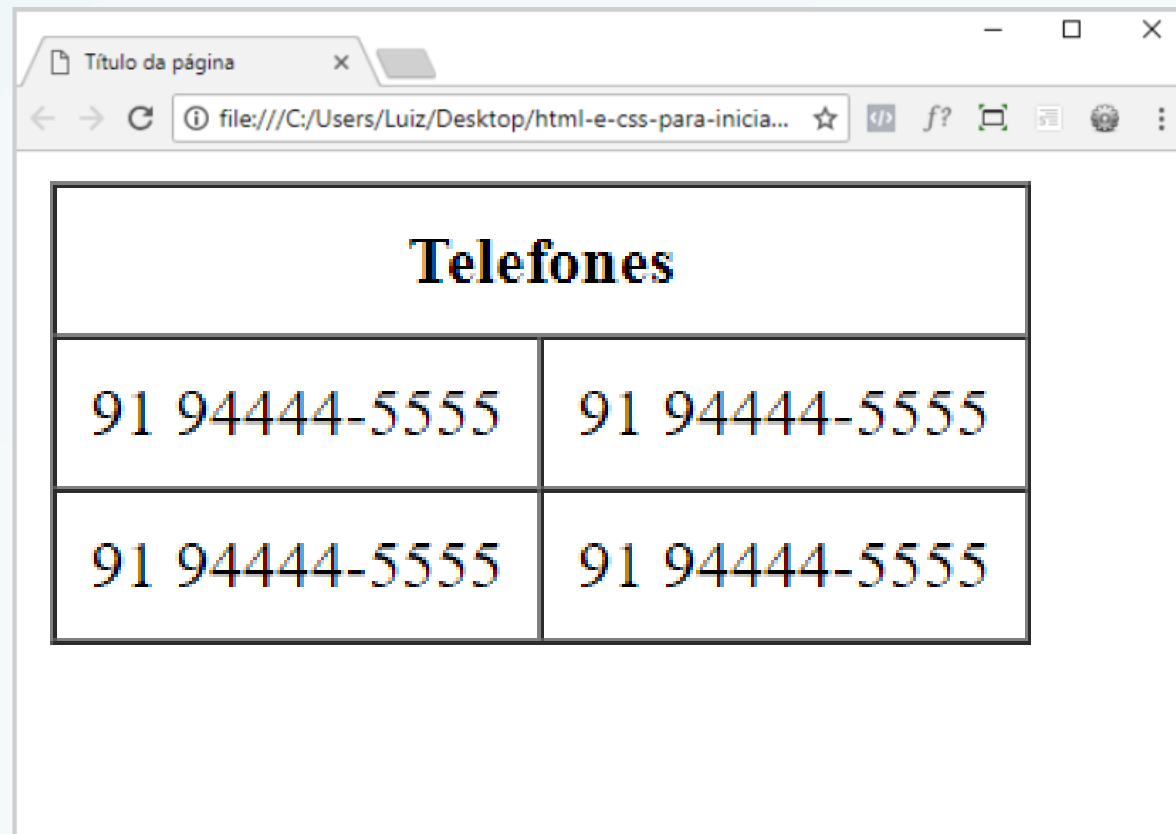
Nome	Telefone	Endereço
Luiz Augusto Vieira	11 92222-3333	Rua Xpto, 99
João Martins	91 94444-5555	Rua Da Paz, 44

Tabela no HTML

Podemos usar o atributo “**colspan**” para mesclar várias colunas com seu valor correspondente.

Vamos ver um exemplo.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body style="font-size: 20px">
8     <table cellpadding="10" cellspacing="0" border="1">
9       <tr>
10        <th colspan="2">Telefones</th>
11      </tr>
12      <tr>
13        <td>91 94444-5555</td>
14        <td>91 94444-5555</td>
15      </tr>
16      <tr>
17        <td>91 94444-5555</td>
18        <td>91 94444-5555</td>
19      </tr>
20    </table>
21  </body>
22 </html>
```



The screenshot shows a web browser window with the title "Título da página". The address bar shows the file path: file:///C:/Users/Luiz/Desktop/html-e-css-para-inicia... The browser displays the rendered HTML table, which has a border and a font size of 20px. The table has a header row with the title "Telefones" and two data rows, each containing two identical phone numbers: "91 94444-5555".

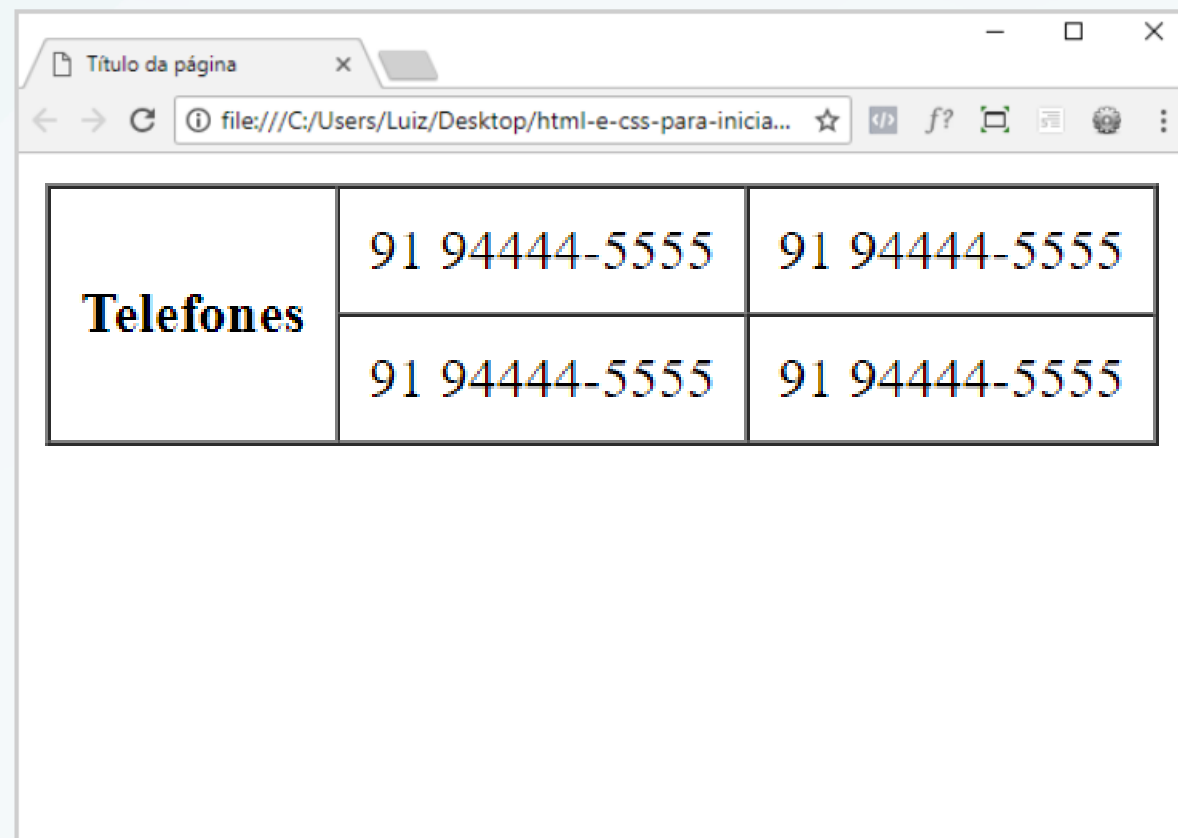
Telefones	
91 94444-5555	91 94444-5555
91 94444-5555	91 94444-5555

Tabela no HTML

Podemos usar o atributo “**rowspan**” para mesclar várias linhas com seu valor correspondente.

Vamos ver um exemplo.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7   <body style="font-size: 20px">
8     <table cellpadding="10" cellspacing="0" border="1">
9       <tr>
10        <th rowspan="4">Telefones</th>
11      </tr>
12      <tr>
13        <td>91 94444-5555</td>
14        <td>91 94444-5555</td>
15      </tr>
16      <tr>
17        <td>91 94444-5555</td>
18        <td>91 94444-5555</td>
19      </tr>
20    </table>
21  </body>
22 </html>
```



The screenshot shows a web browser window with the title "Título da página". The address bar shows the file path: file:///C:/Users/Luiz/Desktop/html-e-css-para-inicia... The browser displays the rendered table from the code block. The table has a border and a font size of 20px. The first column is a header "Telefones" that spans 4 rows. The following three rows each contain two phone numbers: "91 94444-5555".

Telefones	91 94444-5555	91 94444-5555
	91 94444-5555	91 94444-5555
	91 94444-5555	91 94444-5555
	91 94444-5555	91 94444-5555

Tabela no HTML

Vamos desenvolver o Layout ao lado para entendermos melhor a estrutura de tabelas.



Tag <div> no HTML

A tag **<div>** representa uma seção genérica dentro de um documento HTML. Ela permite agrupar seções de elementos **HTML** e formatá-los com **CSS**.

O elemento **<div>** é semelhante ao elemento ****, no entanto, o elemento **** é usado com elementos embutidos, enquanto o elemento **<div>** é usado com elementos no nível de bloco.

O elemento **<div>** aceita conteúdo de fluxo que se refere à maioria dos elementos que podem aparecer dentro do corpo de um documento **HTML**.

```
<div>
  <h1>Este é um cabeçalho dentro da DIV.</h1>
  <p>Este é um parágrafo dentro da DIV.</p>
</div>
```

Este é um cabeçalho dentro da DIV.

Este é um parágrafo dentro da DIV.

Podemos notar que simplesmente colocar o conteúdo dentro de uma tag **<div>** não faz nada ou altera a aparência do conteúdo. Ele simplesmente age como um contêiner que contém o conteúdo incluído.

Tag , e <dl> no HTML

A tag “**UL**”, é usada para especificar uma lista não ordenada (ou lista não numerada).

A tag “**OL**”, é usada para especificar uma lista ordenada (ou lista numerada).

A tag “**DL**”, é usada para declarar uma lista de definição. Uma lista de definições é semelhante as outras listas, mas em uma lista de definições, cada item da lista contém duas entradas; um termo e uma descrição.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>UL, OL e DL</title>
6   </head>
7
8   <body style="font-size: 20px;">
9     <h1>UL</h1>
10    <ul>
11      <li>Item 1</li>
12      <li>Item 2</li>
13      <li>Item 3</li>
14      <li>Item 4</li>
15    </ul>
16
17    <h1>OL</h1>
18    <ol>
19      <li>Item 1</li>
20      <li>Item 2</li>
21      <li>Item 3</li>
22      <li>Item 4</li>
23    </ol>
24
25    <h1>DL</h1>
26    <dl>
27      <dt>Definição de Lista</dt>
28      <dd>Está é uma declaração de termos, suas definições e descrições.</dd>
29      <dt>A UL, é uma lista não numerada.</dt>
30      <dd>Lista não ordenada.</dd>
31      <dt>A OL, é uma lista numerada.</dt>
32      <dd>Lista numerada.</dd>
33    </dl>
34
35  </body>
36 </html>
```

UL

- Item 1
- Item 2
- Item 3
- Item 4

OL

1. Item 1
2. Item 2
3. Item 3
4. Item 4

DL

Definição de Lista

Está é uma declaração de termos, suas definições e descrições.

A UL, é uma lista não numerada.

Lista não ordenada.

A OL, é uma lista numerada.

Lista numerada.

IFrame no HTML

IFrames define um quadro in-line. Um quadro embutido permite que você apresente outro documento HTML dentro da mesma janela.

O **W3C** refere-se a um **iframe** como um contexto de navegação aninhado . Em outras palavras, você pode procurar um documento HTML aninhado dentro de outro. Normalmente, o **iframe** ocupa uma pequena parte do documento, mas você pode torná-lo do tamanho que desejar.

Como criar um iframe?

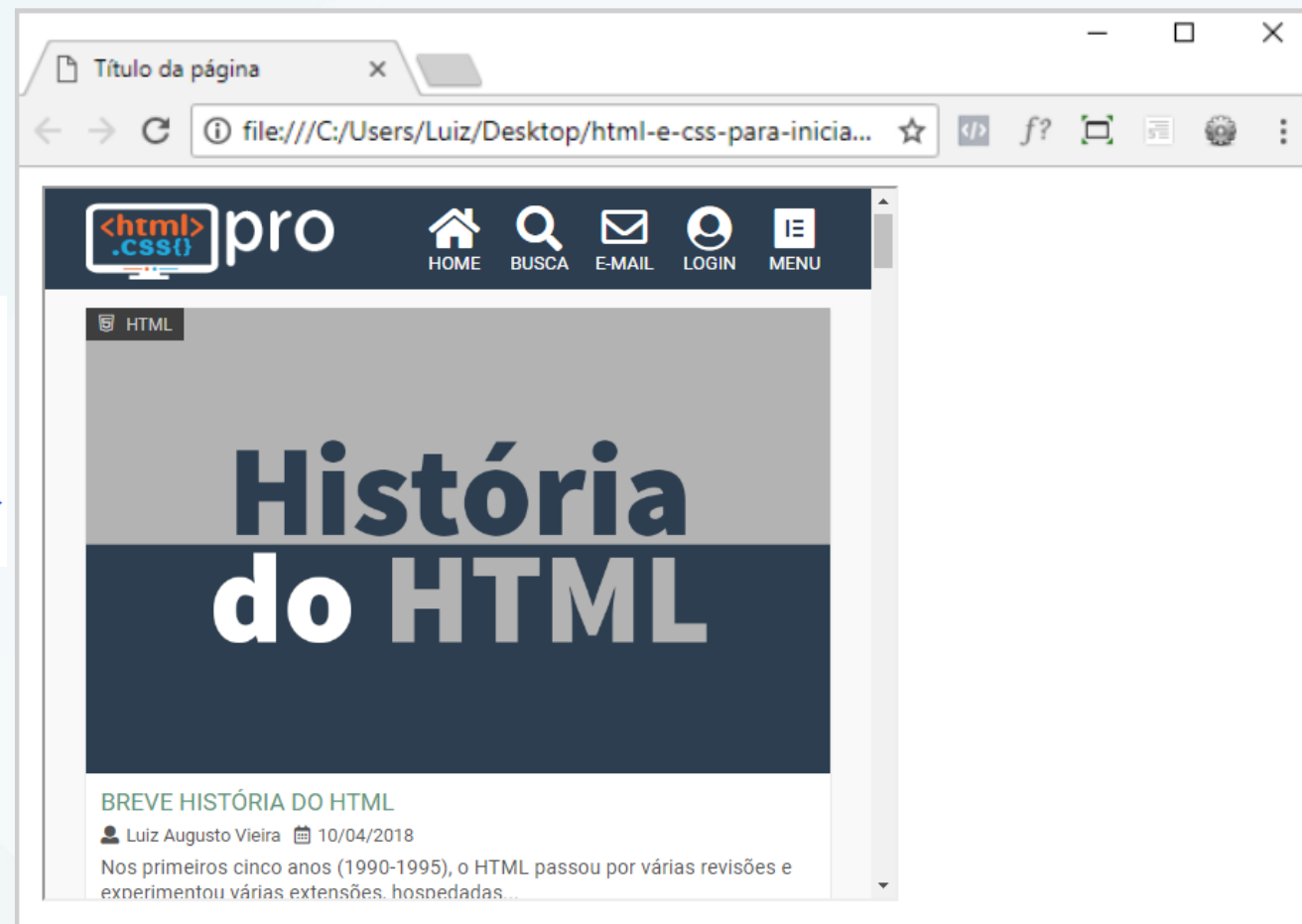
Para criar um **iframe**, use a tag `<iframe>` com atributo **SRC** que fornece uma **URL** do documento externo para usar no **iframe**.

Além disso, você pode definir a largura e a altura usando os atributos **width** e **height**.

IFrame no HTML

Exemplo:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Título da página</title>
6   </head>
7
8   <body>
9     <iframe src="https://www.htmlecsspro.com" width="600" height="500"></iframe>
10
11 </body>
12 </html>
```



{CSS}

O que é CSS?

CSS significa Cascading Style Sheets. É a linguagem padrão para definir estilos em páginas da web. Embora o **CSS** seja mais amplamente conhecido por sua aplicação em documentos HTML, ele pode ser usado para definir estilos para qualquer formato de documento estruturado (como XML, por exemplo).

Os estilos são definidos usando propriedades **CSS**. Por exemplo, você pode definir propriedades de fonte (tamanho, cores, estilo e etc.), imagens de fundo de bloco, estilos de borda e muito mais.

HTML

O objetivo do **HTML** é fornecer estrutura e significado do documento. usamos **HTML** para especificar quais elementos vão na página (por exemplo, títulos, parágrafos, tabelas, imagens e etc.). Cada um desses elementos, assim como seus atributos e os valores dos atributos tem um certo significado.

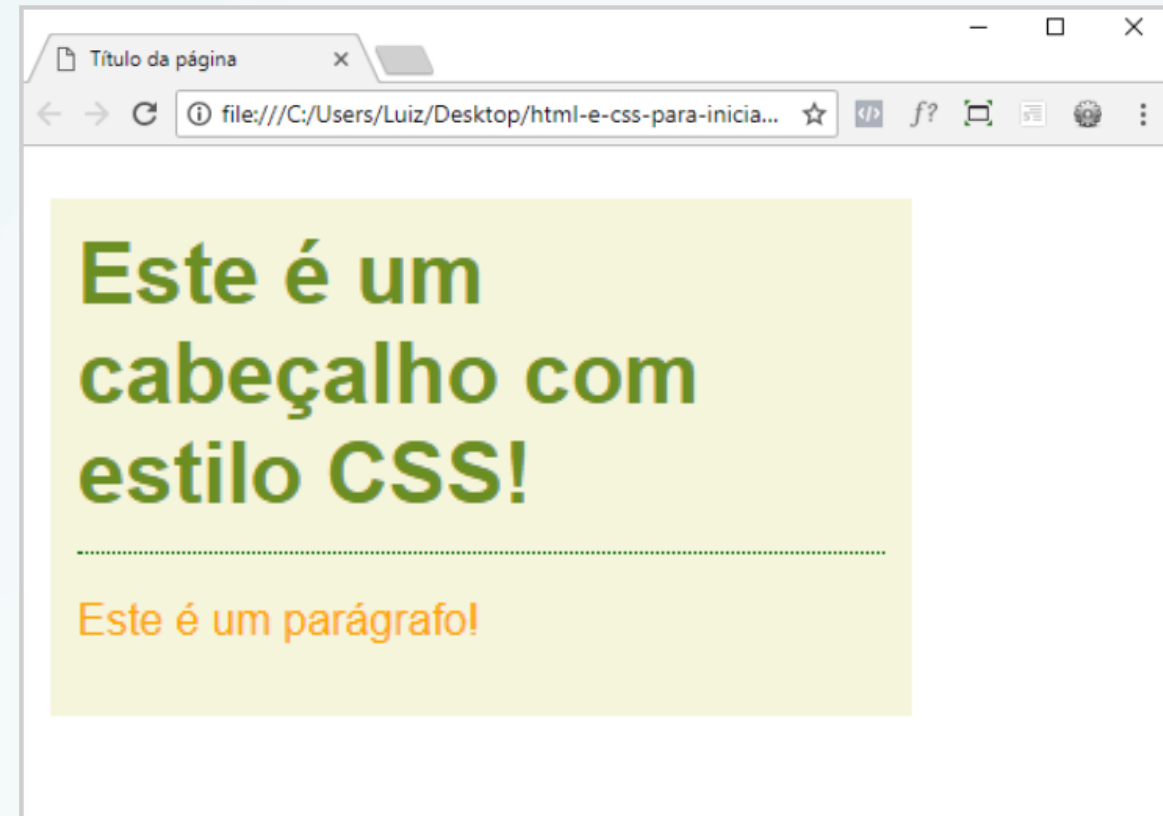
CSS

Podemos usar **CSS** para fornecer estilos para saída de fala (por exemplo, para usuários que usam um leitor de tela). Assim, você pode escrever um documento **HTML** sem se preocupar com sua apresentação, depois usar **CSS** para especificar como ele será apresentado em qualquer contexto. Não só isso, mas você pode alterar o **CSS** sem ter que alterar o **HTML**. Em outras palavras, você pode "conectar" uma folha de estilo em um documento **HTML** e o documento **HTML** assumirá imediatamente os estilos dessa folha de estilo.

Vamos adicionar o seguinte código ao exemplo abaixo:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6
7     <link href="aula1.css" type="text/css" rel="stylesheet" />
8   </head>
9
10  <body>
11    <header>
12      <h1>Este é um cabeçalho com estilo CSS!</h1>
13      <p>Este é um parágrafo!</p>
14    </header>
15  </body>
16 </html>
```

```
1 body{
2   font-family: Arial;
3 }
4 header {
5   width: 300px;
6   padding: 10px;
7   background: beige;
8 }
9
10 h1 {
11   color: olivedrab;
12   border-bottom: 1px dotted darkgreen;
13   margin: 0;
14   padding-bottom: 10px;
15 }
16 p {
17   color: orange;
18 }
```



Sintaxe CSS

A sintaxe do **CSS** consiste em um conjunto de regras. Essas regras têm 3 partes: um seletor, uma propriedade e um valor.

Não precisamos lembrar disso para codificar **CSS**. Depois de começar a codificar o **CSS**, você o fará sem pensar "isso é um seletor" ou "isso é uma propriedade". Isso deve começar a fazer sentido depois de estudar os exemplos neste Curso.

Sintaxe:

```
seletor { propriedade : valor }
```

O seletor representa o elemento **HTML** que você deseja estilizar. **Por exemplo:**

```
h1 { color: blue }
```

O código acima diz ao navegador para renderizar todas as ocorrências do elemento `<h1>` no **HTML** em azul

```
h1, h2, h3, h4, h5, h6 { color: blue }
```

Seletores de agrupamento

Você pode aplicar um estilo a muitos seletores, se desejar. Apenas separe os seletores com uma vírgula.

Sintaxe CSS

Sintaxe:

```
h1 { color:blue; font-family: arial, helvetica, "sans serif" }
```

Aplicando Múltiplas Propriedades

Para aplicar mais de uma propriedade, separe cada declaração com um ponto e vírgula.

```
h1 {  
  color: blue;  
  font-family: arial, helvetica, "sans serif";  
  font-size: 150%;  
}
```

Legibilidade

Você pode tornar seu código **CSS** mais legível espalhando suas declarações de estilo em várias linhas.

Adicionar CSS em uma página da Web

Quando se trata de adicionar **CSS** ao seu documento, você tem várias escolhas de métodos. No entanto, um desses métodos se destaca como o mais comum entre eles.

Existem 4 maneiras de adicionar **CSS** a uma página da Web: **declarar inline** , **incorporar no cabeçalho do documento**, **vincular a um arquivo CSS externo** e **importar um arquivo CSS**.

Estilos Inline

Com estilos inline, as informações da folha de estilo são aplicadas diretamente ao elemento **HTML**. Em vez de definir o estilo em um arquivo único, em seguida, aplicar o estilo a todas as instâncias de um elemento, colocaremos o estilo diretamente ao elemento específico ao qual desejamos aplicar o estilo.

Por exemplo:

```
<p style="color:olive;font-size:24px;">Estilos HTML com CSS</p>
```

Estilos inline podem ser uma maneira rápida e conveniente de adicionar estilos a um documento **HTML**, esse método deve ser usado com moderação.

Adicionando estilos inline em todo o site, o mesmo torna muito difícil de fazer manutenção. Uma pequena alteração pode se tornar um empreendimento importante se o estilo tiver sido aplicado a muitas páginas do site.

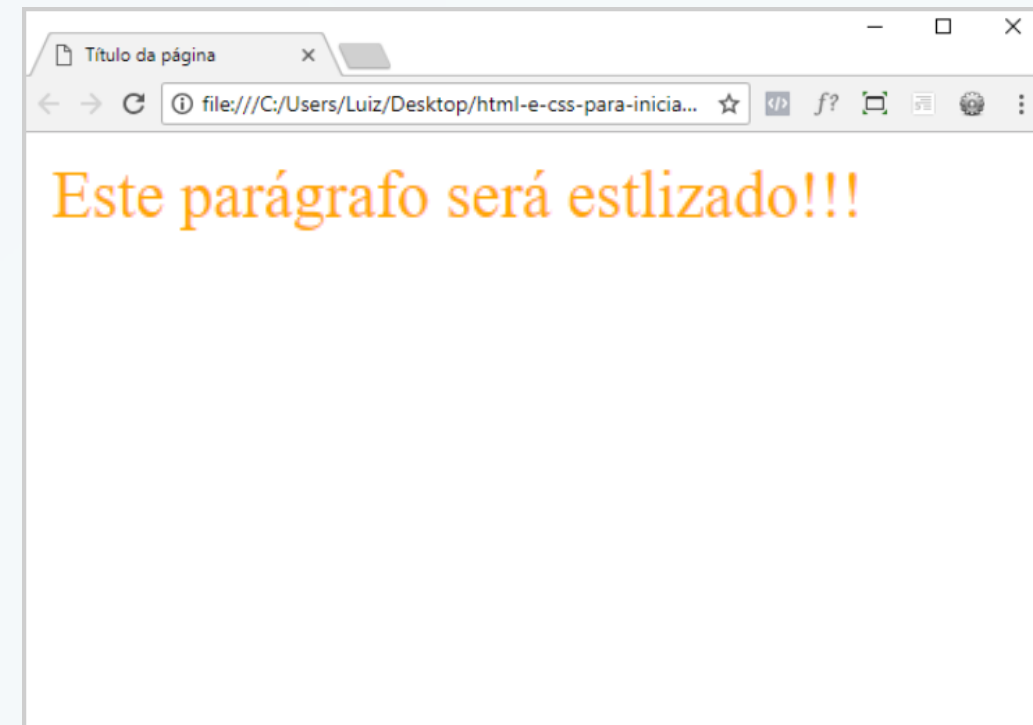
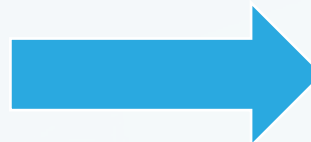
Adicionar CSS em uma página da Web

Estilos Interno

Você adiciona todos os estilos do **CSS** a uma parte do documento, geralmente o topo da página. Isso permite estilizar qualquer elemento na página de um único lugar. Você faz isso incorporando as informações de **CSS** nas tags **<style>** no cabeçalho do documento **HTML**.

Por exemplo, coloque o seguinte código entre as tag **<head>** do documento **HTML**:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6
7     <style>
8       p {
9         color: orange;
10        font-size: 25px;
11      }
12    </style>
13  </head>
14
15  <body>
16    <p>Este parágrafo será estilizado!!!</p>
17
18  </body>
19 </html>
```



Adicionar CSS em uma página da Web

Estilos Externos

Folhas de estilo externas são o método mais comum de aplicar estilos a um site. A maioria dos sites modernos usam uma folha de estilo externa para aplicar estilos em todo o site.

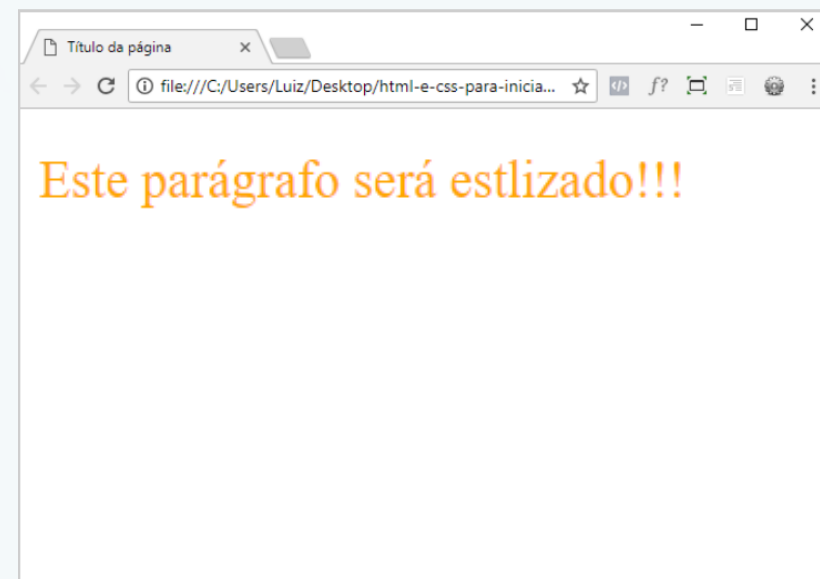
Estilos externos referem-se à criação de um arquivo separado que contém todas as informações de estilos. Este arquivo é então vinculado nas páginas **HTML** que você deseja estilizar.

Para adicionar uma folha de estilo externa a uma página da Web, use a tag `<link>`, fornecendo o **URL** da folha de estilo no atributo **href**, bem como **rel="stylesheet"**.

Eu vou criar um arquivo chamado (aula1.css), salvar na pasta em questão e adicionar os estilos do exemplo de estilo inline.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6
7     <link href="aula1.css" type="text/css" rel="stylesheet" />
8   </head>
9
10  <body>
11    <p>Este parágrafo será estilizado!!!</p>
12
13  </body>
14 </html>
```

```
1 p {
2   color: orange;
3   font-size: 25px;
4 }
```



Adicionar CSS em uma página da Web

Estilos de Importação

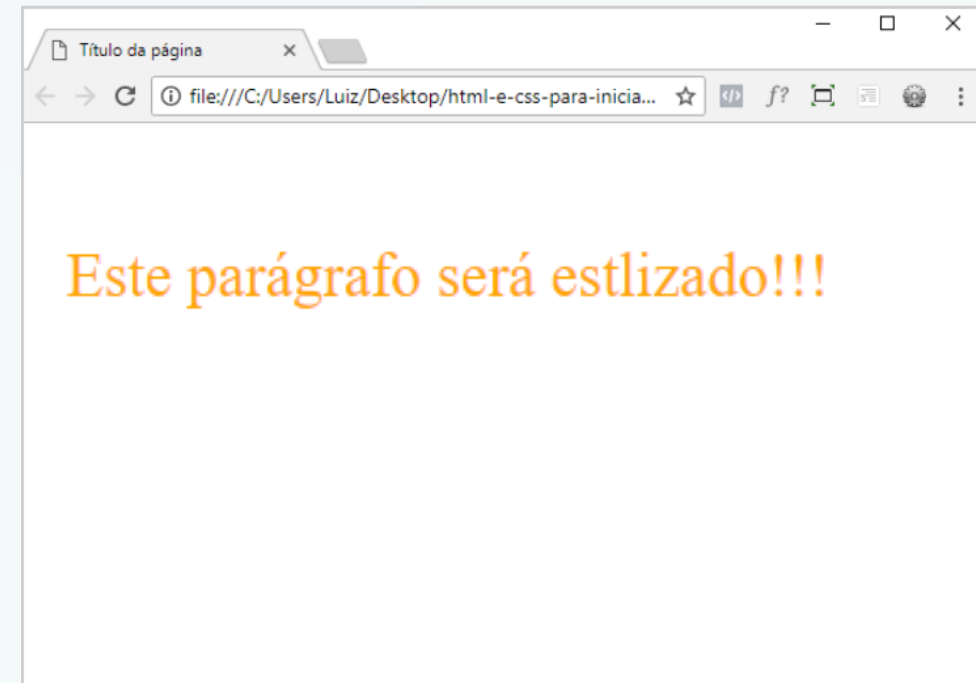
Você pode usar a regra `@import` CSS para importar uma folha de estilos externa.

Para fazer isso, use a tag `<style>`.

Você pode usar uma das seguintes sintaxes:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       @import "aula1.css";
8       @import url("aula1.css");
9     </style>
10  </head>
11  <body>
12    <p>Este parágrafo será estilizado!!!</p>
13  </body>
14 </html>
```

```
1 p {
2   color: orange;
3   font-size: 25px;
4 }
```



Classe CSS

As classes **CSS** oferecem flexibilidade extra para especificar em quais elementos devem ser aplicado estilos específicos.

Em **CSS**, as classes permitem que você aplique um estilo a uma determinada classe de um elemento. Para fazer isso, você vincula o elemento ao estilo declarando um estilo para a classe e atribuindo essa classe ao elemento.

Sintaxe da Classe no CSS

Você declara uma classe **CSS** usando um ponto (.) Seguido do nome da classe. Após dada o nome da classe, basta digitar as propriedades e valores que você deseja atribuir à sua classe.

```
.nome-da-classe { color: blue; }
```

No exemplo acima, criamos a nossa classe com nome de (nome-da-classe) e passamos uma propriedade com seu valor.

Classe CSS

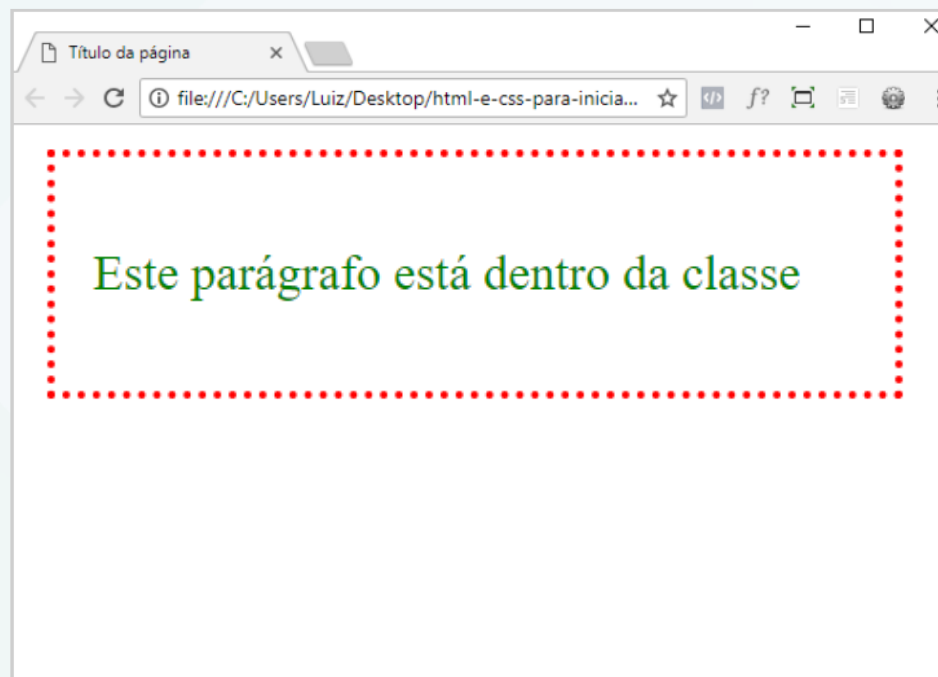
Podemos selecionar elementos ou classes que estão aninhados em outro elemento ou classe.

Por exemplo, `div.minha-classe p` selecionará todos os elementos `<p>` que estão aninhados em um elemento `<div>` que usa a classe `.minha-classe`.

A grande vantagem de fazer isso é que você não precisa aplicar uma classe a cada instância de um elemento quando ele está aninhado em outro elemento que usa uma classe.

exemplo:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div.minha-classe {
8         border:4px dotted red;
9         padding: 20px;
10        width: 400px;
11        font-size: 26px;
12      }
13
14      div.minha-classe p {
15        color:green;
16      }
17    </style>
18  </head>
19  <body>
20    <div class="minha-classe">
21      <p>Este parágrafo está dentro da classe</p>
22    </div>
23  </body>
24 </html>
```



ID CSS

Podemos atribuir um identificador exclusivo a um elemento **HTML** e, em seguida, estilizar esse elemento referenciando seu identificador exclusivo. Isso permite definir um estilo que só pode ser usado pelo elemento ao qual você atribui o **ID**.

Qualquer elemento **HTML** pode ter o atributo ID aplicado a ele. O valor desse atributo é o identificador exclusivo do elemento. Por exemplo, `<h1 id="cabecalho">` dá a esse elemento um identificador exclusivo de cabeçalho.

Podemos referenciar esse **ID** dentro do código **CSS**.

Sintaxe de ID no CSS

A sintaxe para declarar um **ID** no **CSS** é a mesma declarada para classes, exceto que, em vez de usar um ponto(.), você usa uma hash (#).

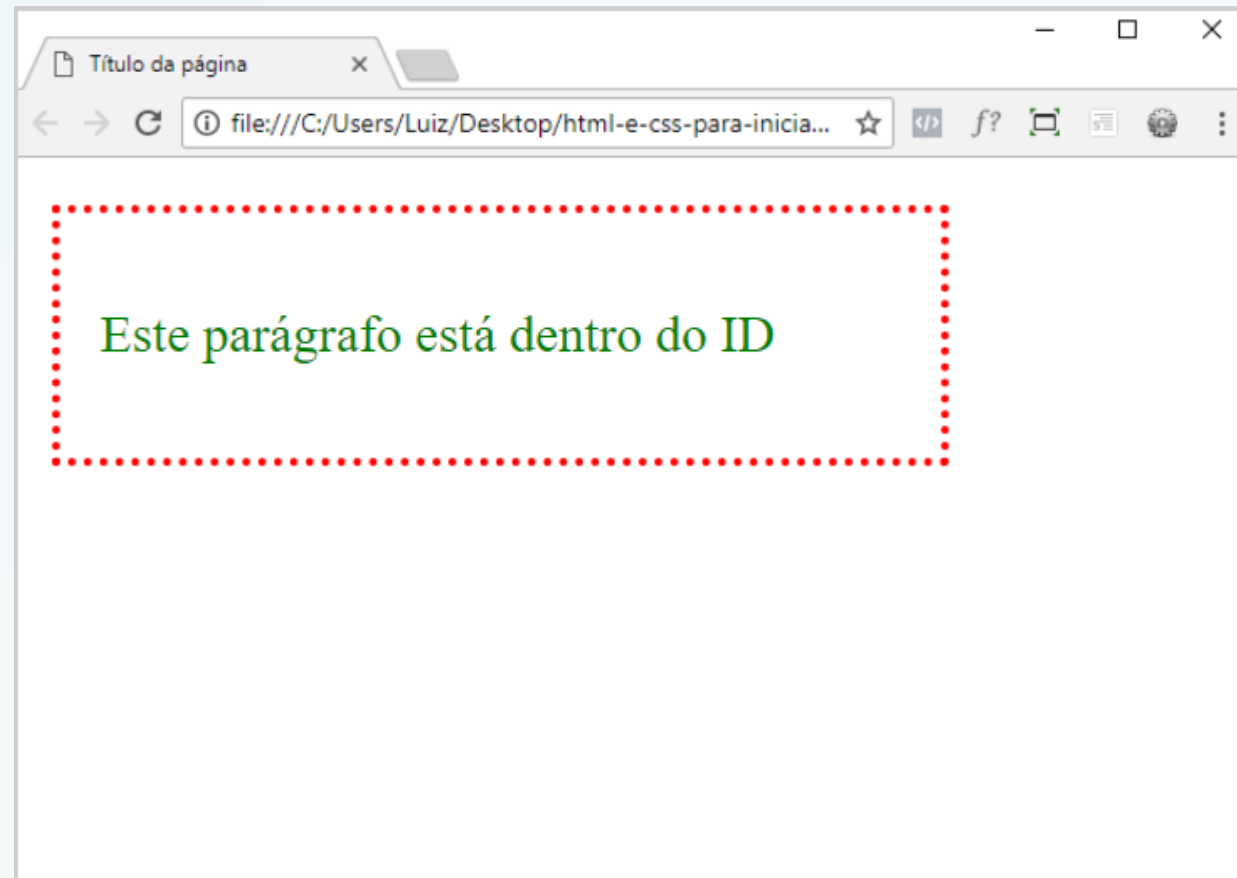
```
#meu-id { color: blue; }
```

Veja que é semelhante a classes.

ID CSS

Você deve usar **IDs** se apenas um elemento na página tiver o estilo aplicado ou se precisar de um identificador exclusivo para esse elemento.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div#meu-id {
8         border:4px dotted red;
9         padding: 20px;
10        width: 400px;
11        font-size: 26px;
12      }
13
14      div#meu-id p {
15        color:green;
16      }
17    </style>
18  </head>
19  <body>
20    <div id="meu-id">
21      <p>Este parágrafo está dentro do ID</p>
22    </div>
23  </body>
24 </html>
```



Propriedade de fonte no CSS

As propriedades da fonte **CSS** permitem alterar a aparência do seu texto. Por exemplo, podemos atribuir uma família de fontes, aplicar formatação em negrito ou itálico, alterar o tamanho da fonte e etc.

Família de Fontes CSS

A propriedade (**font-Family**) nos permite definir a família de fontes para nosso site.

```
<style>
  p {
    font-family: Georgia, Garamond, serif;
  }
</style>

<p>Este texto é renderizado em georgia, garamond ou na fonte serif padrão (dependendo da fonte que o sistema do usuário possui).</p>
```

Este texto é renderizado em georgia, garamond ou na fonte serif padrão (dependendo da fonte que o sistema do usuário possui).

A propriedade font-family aceita diferentes famílias de fontes. Isso porque nem todos os usuários terão as mesmas fontes instaladas em seus computadores. Portanto, podemos fornecer uma lista de fontes, começando com sua primeira escolha e terminando com sua última opção.

Se o usuário não tiver sua fonte de primeira escolha, a segunda opção será usada para esse usuário. Se eles não tiverem a segunda opção, a terceira opção será usada e assim por diante.

Propriedade de fonte no CSS

Tamanho da Fonte CSS

Permite que você defina o tamanho do texto.

```
<style>
  p {
    font-size: 30px;
  }
</style>

<p>Este texto está usando um tamanho de fonte de 30 pixels.</p>
```

Este texto está usando um tamanho de fonte de 30 pixels.

Propriedade de fonte no CSS

Estilo da Fonte CSS

A propriedade (**font-style**) é usada para especificar texto em itálico.

```
<style>
  p {
    font-style: italic;
  }
</style>

<p>Este texto está em itálico.</p>
```

Este texto está em itálico.

Propriedade de fonte no CSS

Variante de Fontes CSS

Permite que você defina seu texto para usar letras maiúsculas pequenas.

```
<style>
  p {
    font-variant: small-caps;
  }
</style>

<p>Este texto está usando pequenas maiúsculas.</p>
```

ESTE TEXTO ESTÁ USANDO PEQUENAS MAIÚSCULAS.

Propriedade de fonte no CSS

Peso da Fonte CSS

Permite que você defina seu texto em negrito. Você pode definir um valor em números.

```
<style>
  p {
    font-weight: bold;
  }
</style>

<p>Este texto é negrito.</p>
```

Este texto é negrito.

Propriedade de fonte no CSS

A propriedade "**font**", é uma propriedade abreviada que permite definir todas as propriedades da fonte de uma só vez.

```
<style>
  p {
    font: italic small-caps bold 20px Georgia, Garamond, serif;
  }
</style>

<p>Os estilos deste texto foram especificados com a propriedade abreviada 'font'.</p>
```

OS ESTILOS DESTES TEXTO FORAM ESPECIFICADOS COM A PROPRIEDADE ABREVIADA 'FONT'.

Propriedades de texto no CSS

Apesar das várias propriedades de fontes no **CSS**, existem outras propriedades que podem ajudar a estilização do texto.

Por exemplo, você pode alterar a cor do texto, alinhar texto, adicionar propriedades de decoração e etc.

Cor do texto no CSS

```
<style>
  p {
    color: blue;
  }
</style>

<p>Essa cor de texto CSS é azul</p>
```

Essa cor de texto CSS é azul

Propriedades de texto no CSS

Alinhamento de texto CSS

A propriedade "**text-align**" CSS é usada para alinhar elementos à esquerda, à direita ou ao centro.

```
text-align: left | right | center | justify
```

Left - Conteúdo é alinhado à esquerda da linha da caixa.

Right - Conteúdo é alinhado à direita da linha da caixa de linha

Center - Conteúdo é centralizado na caixa de linha.

Justify - O texto é justificado de acordo com o método especificado pela propriedade **text-justify**, para preencher exatamente a caixa de linha.

```
<style>
  p {
    text-align: right;
  }
</style>

<p>Este texto CSS está alinhado à direita</p>
```

Este texto CSS está alinhado à direita

Propriedades de texto no CSS

Recuo do texto CSS

A propriedade “**text-indent**” CSS especifica o recuo aplicado a linhas de conteúdo embutido em um bloco.

Ao declarar essa propriedade “**text-indent**”, o recuo é tratado como uma margem aplicada à borda inicial da caixa de linha.

Somente as linhas que são a primeira linha formatada de um elemento são afetadas.

```
<style>
  p {
    text-indent: 50px;
  }
</style>
```

```
<p>Este texto é recuado por 50 pixels. Isso significa que a primeira linha do parágrafo será recuada por 50 pixels, mas as linhas a seguir não serão recuadas. O texto precisará ser quebrado antes que você possa ver o recuo - daí todo esse texto!</p>
```

Este texto é recuado por 50 pixels. Isso significa que a primeira linha do parágrafo será recuada por 50 pixels, mas as linhas a seguir não serão recuadas. O texto precisará ser quebrado antes que você possa ver o recuo - daí todo esse texto!

Propriedades de texto no CSS

Espaçamento de letras CSS

A propriedade “**letter-spacing**” CSS determina o espaço entre as letras.

O valor fornecido para a propriedade “**letter-spacing**” especifica qualquer espaço adicional entre as letras. Podemos usar um valor negativo para aproximar cada letra uma da outra.

```
<style>
  p {
    letter-spacing: 5px;
  }
</style>
```

```
<p>Este texto tem espaçamento entre letras aplicado</p>
```

Este texto tem espaçamento entre letras aplicado

Propriedades de texto no CSS

Espaçamento de palavras CSS

A propriedade “**word-spacing**” CSS controla o espaço entre as palavras.

Essa propriedade “**word-spacing**” especifica espaço adicional entre cada palavra. Portanto, um valor de zero, por exemplo, não tem efeito (o espaçamento normal se aplica). Valores positivos aumentam o espaçamento e valores negativos reduzem o espaçamento.

```
<style>
  p {
    word-spacing: 50px;
  }
</style>
```

```
<p>Este texto tem espaçamento de palavras aplicado</p>
```

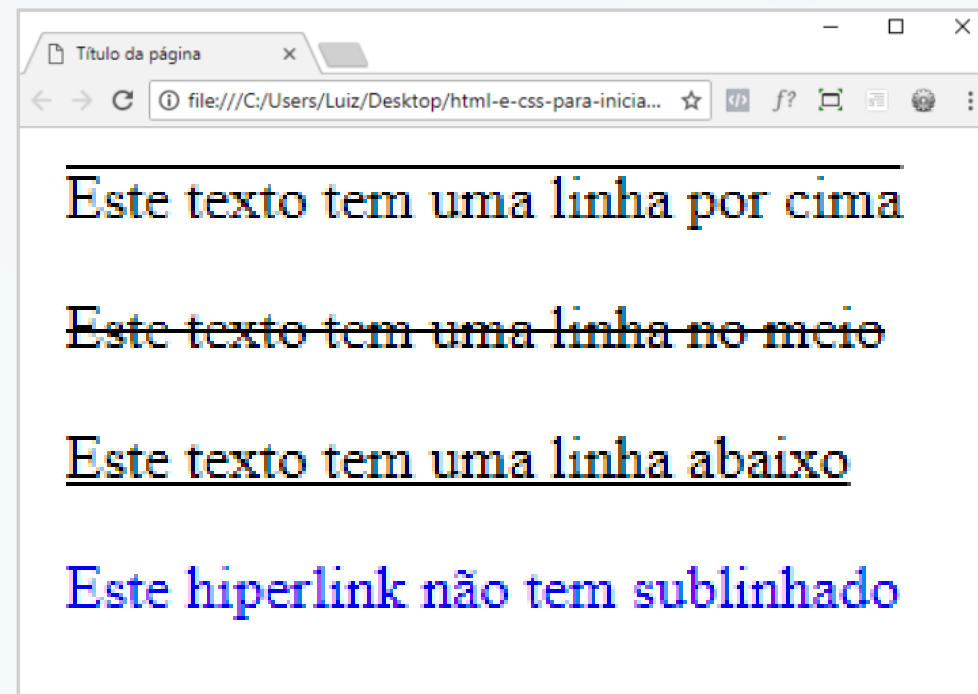
Este texto tem espaçamento de palavras aplicado

Propriedades de texto no CSS

CSS Text-Decoration

A propriedade “**text-decoration**” é um atalho para definir: **text-decoration-line**; **text-decoration-style**; **text-decoration-color** em uma declaração.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       .overline {text-decoration: overline;}
8       .line-through {text-decoration: line-through;}
9       .underline {text-decoration: underline;}
10      a:link {text-decoration: none;}
11    </style>
12  </head>
13  <body>
14    <p class="overline">Este texto tem uma linha por cima</p>
15    <p class="line-through">Este texto tem uma linha no meio</p>
16    <p class="underline">Este texto tem uma linha abaixo</p>
17    <a href="#">Este hiperlink não tem sublinhado</a>
18  </body>
19 </html>
```

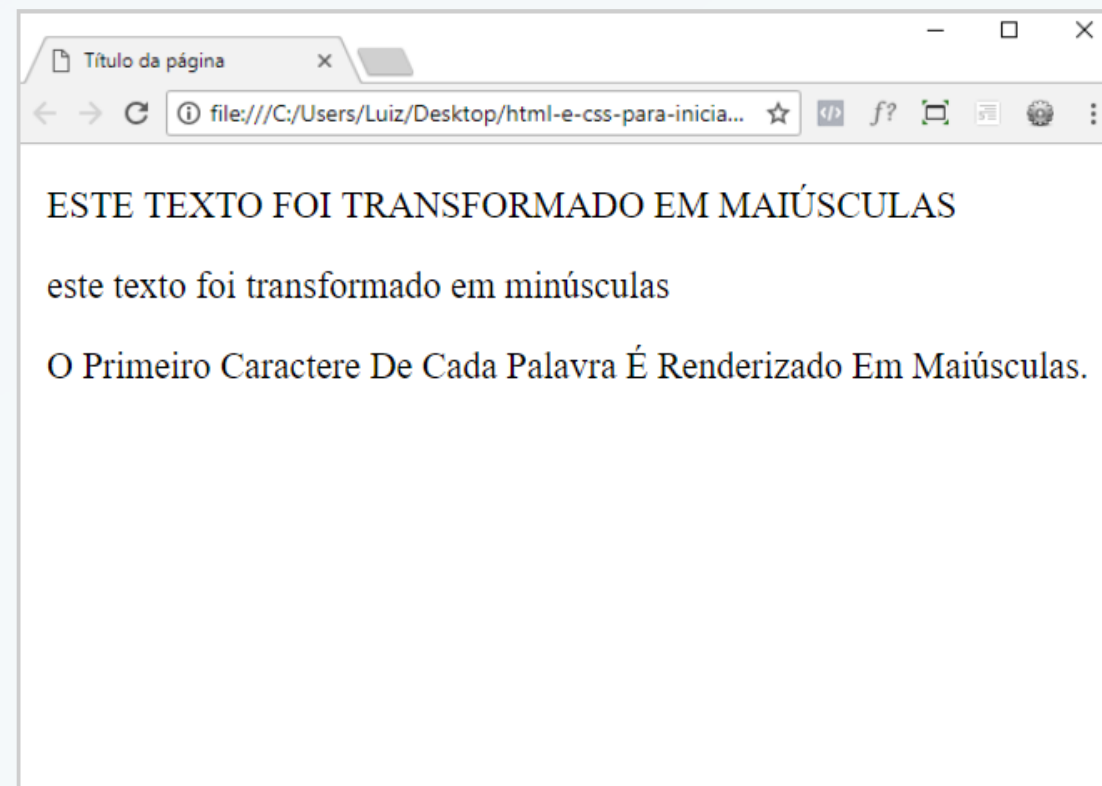


Propriedades de texto no CSS

Transformação de texto CSS

A propriedade “**text-transform**” CSS é usada para controlar a captura do texto. Por exemplo, você pode forçar o texto a maiúscula e minúscula.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       .uppercase {
8         text-transform: uppercase;
9       }
10      .lowercase {
11        text-transform: lowercase;
12      }
13      .capitalize {
14        text-transform: capitalize;
15      }
16    </style>
17  </head>
18  <body>
19    <p class="uppercase">ESTE TEXTO FOI TRANSFORMADO EM MAIÚSCULAS</p>
20    <p class="lowercase">este texto foi transformado em minúsculas</p>
21    <p class="capitalize">O primeiro caractere de cada palavra é renderizado em maiúsculas.</p>
22  </body>
23 </html>
```



Propriedades de texto no CSS

Sombra de texto CSS

A propriedade “**text-shadow**” CSS é usada para aplicar efeitos de sombra ao texto.

Você pode usar a propriedade “**text-shadow**” para aplicar sombras, brilhos externos e outros efeitos de sombra ao texto.

A propriedade text-shadow aceita uma lista de valores. Cada item na lista pode ter dois, três ou quatro valores.

Os dois primeiros valores são valores de comprimento que definem o deslocamento horizontal e vertical da sombra, respectivamente (esses são valores obrigatórios). Um terceiro valor de comprimento pode ser usado para definir o raio de desfoque da sombra (opcional). E um valor de cor pode ser usado para definir a cor da sombra (opcional).

Podemos aplicar vários efeitos de sombra em uma única declaração “**text-shadow**”, separando cada conjunto de valores com uma vírgula. Vários efeitos de sombra são aplicados na ordem especificada e, portanto, podem se sobrepor uns aos outros, mas nunca sobreporão o próprio texto.

As sombras de texto não são cortadas na forma sombreada e podem ser exibidas se o texto for parcialmente transparente. Além disso, as sombras de texto não influenciam o layout e não acionam a rolagem nem aumentam o tamanho da área rolável.

Propriedades de texto no CSS

Sombra de texto CSS

```
<style>
  p {
    text-shadow: 2px 2px 8px orange;
  }
</style>

<p>Se o seu navegador suportar a propriedade de sombra de texto CSS, esse texto terá uma sombra.</p>
```

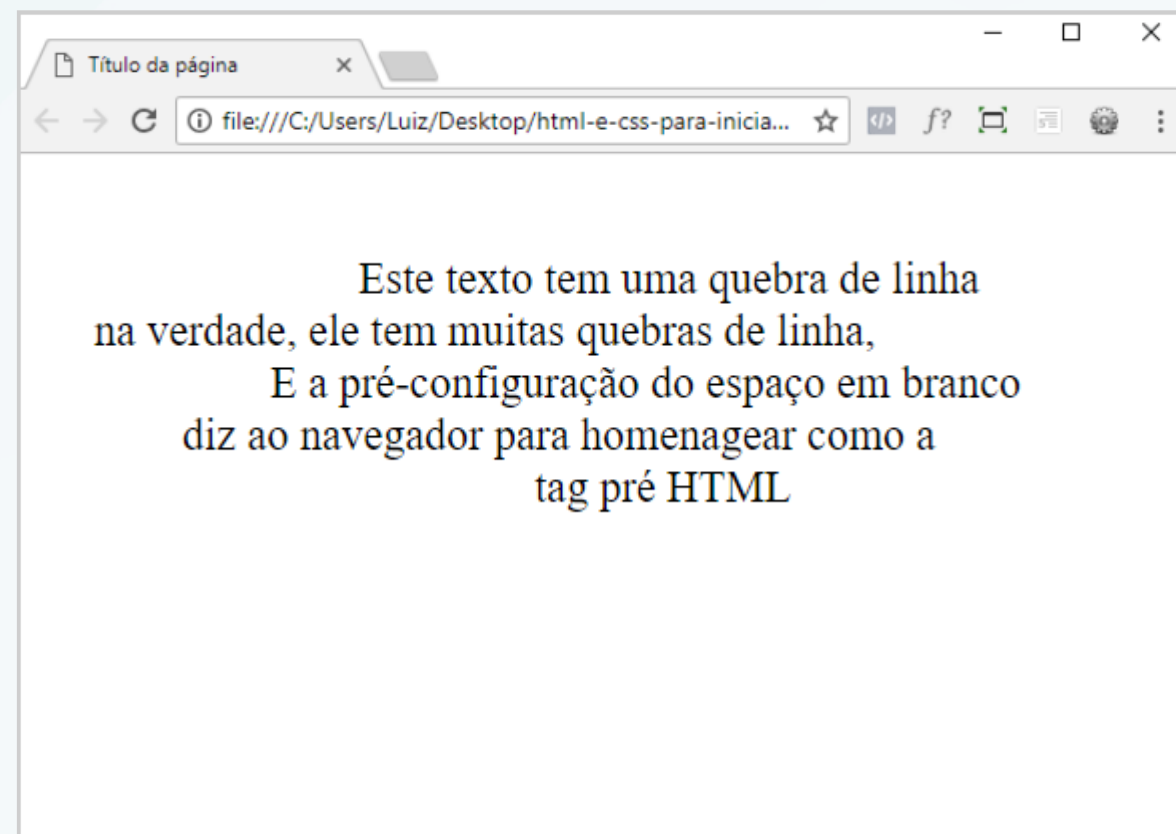
Se o seu navegador suportar a propriedade de sombra de texto CSS, esse texto terá uma sombra.

Propriedades de texto no CSS

Espaço Branco no CSS

A propriedade “**white-space**” CSS é usada para controlar o espaço em branco. Diz ao navegador como lidar com o espaço em branco. Os valores possíveis: **normal**, **pre**, e **nowrap**.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>body{font-size: 22px;}
7       p {
8         white-space: pre;
9       }
10    </style>
11  </head>
12  <body>
13
14    <p>
15      Este texto tem uma quebra de linha
16      na verdade, ele tem muitas quebras de linha,
17      E a pré-configuração do espaço em branco
18      diz ao navegador para homenagear como a
19      tag pré HTML
20    </p>
21  </body>
22 </html>
```



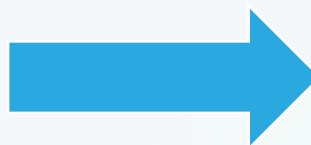
Propriedades de altura e largura no CSS

O **CSS** tem propriedades de altura e largura para ajudar a especificar o tamanho dos elementos.

As propriedades **height** e **width** - Aplica-se a todos os elementos **HTML**, exceto elementos sequenciais não substituídos, colunas de tabelas e grupos de colunas.

Você pode usar uma altura fixa (ou seja, pixels) ou uma altura percentual.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        background-color: blue;
11        color: #fff;
12        font-size: 35px;
13        padding: 15px;
14      }
15    </style>
16  </head>
17  <body>
18    <div>
19      Esta DIV tem altura e largura definida.
20    </div>
21  </body>
22 </html>
```



Background-color no CSS

Você pode usar **CSS** para adicionar efeitos de fundo a qualquer elemento **HTML**.

Para definir a cor de fundo de um elemento, use a propriedade "**background-color**".

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        background-color: blue;
11        color: #fff;
12        font-size: 35px;
13        padding: 15px;
14      }
15    </style>
16  </head>
17  <body>
18    <div>
19      Esta DIV tem uma cor de fundo aplicada.
20    </div>
21  </body>
22 </html>
```

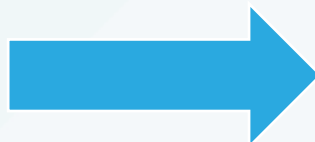


Imagem de plano de fundo no CSS

Imagem de plano de fundo do CSS.

Para adicionar uma imagem de fundo a um elemento, use a propriedade background-image.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        font-size: 35px;
11        padding: 15px;
12
13        background-image: url("bg.png");
14      }
15    </style>
16  </head>
17  <body>
18    <div>
19      Esta DIV tem uma imagem de fundo aplicada.
20    </div>
21  </body>
22 </html>
```

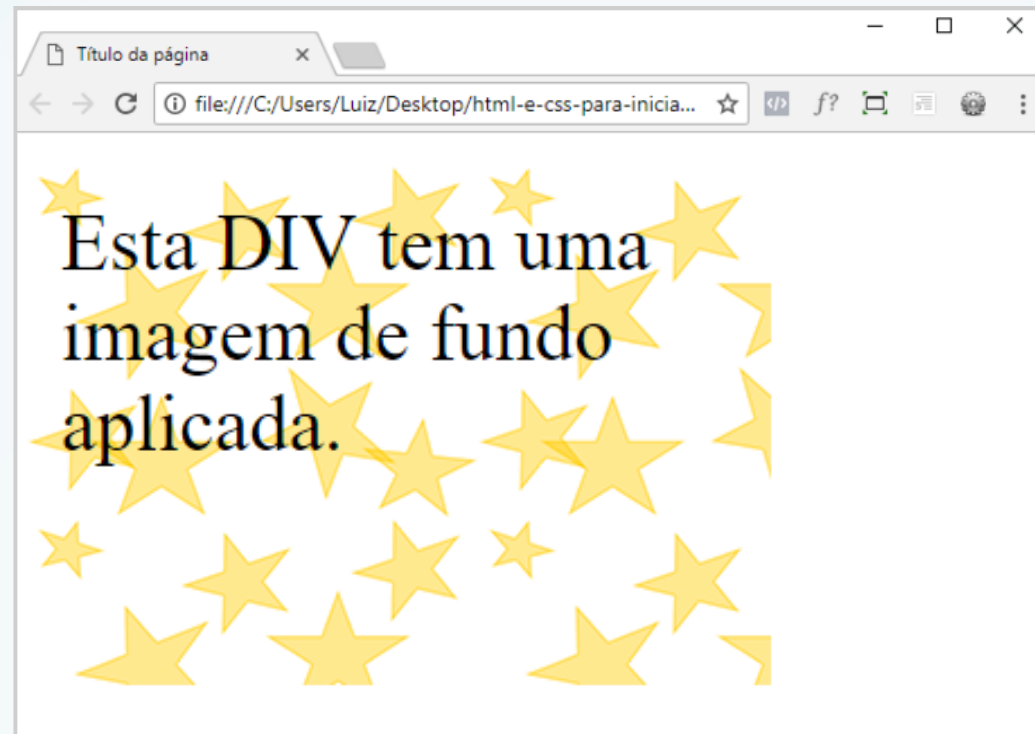
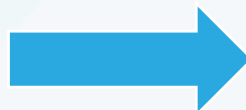


Imagem de plano de fundo no CSS

Repetição de fundo CSS.

Determina se a imagem de fundo se repete dentro do blocos ou não. Podemos usar a propriedade "**background-repeat**".

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        font-size: 35px;
11        padding: 15px;
12
13        background-image: url("bg.png");
14        background-repeat: no-repeat;
15      }
16    </style>
17  </head>
18  <body>
19    <div>
20      Esta DIV tem uma imagem de fundo aplicada, mas não se repete.
21    </div>
22  </body>
23 </html>
```

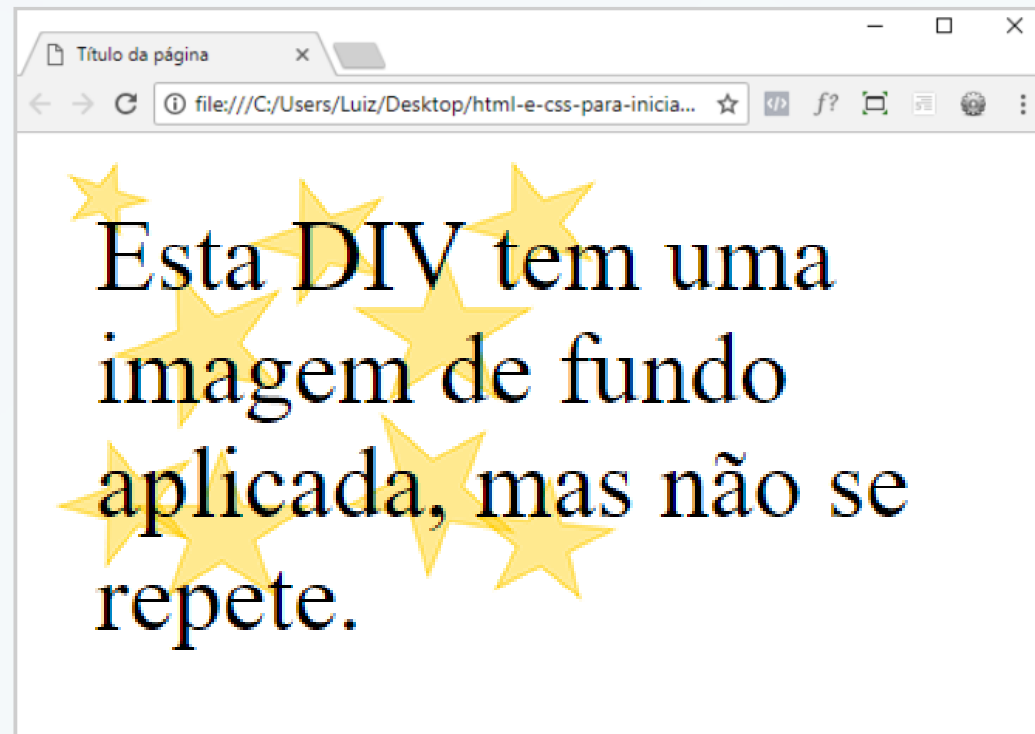


Imagem de plano de fundo no CSS

Posição de fundo do CSS.

Para determina a posição da imagem de fundo no **CSS** use a propriedade "**background-position**".

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        font-size: 35px;
11        background-image: url("bg.png");
12        background-repeat: no-repeat;
13        background-position: 100px;
14      }
15    </style>
16  </head>
17  <body>
18    <div>
19      A imagem de fundo é posicionada a 100 pixels da esquerda.
20    </div>
21  </body>
22 </html>
```

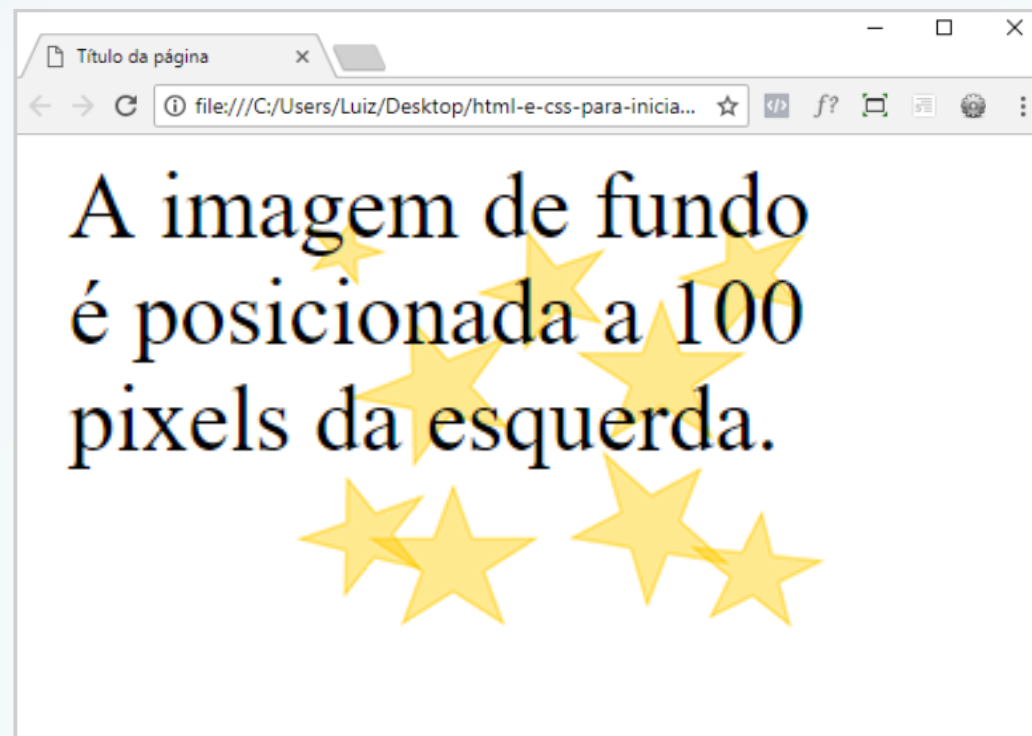


Imagem de plano de fundo no CSS

Anexo de fundo CSS.

Para determinar se a imagem de plano de fundo rola ou não com o contêiner externo, usamos a propriedade **"background-attachment"**.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        font-size: 35px;
11        overflow: auto;
12        background-image: url("bg.png");
13        background-repeat: no-repeat;
14        background-attachment: fixed;
15      }
16    </style>
17  </head>
18  <body>
19    <div>
20      A imagem de fundo é fixa - ela não rola com o contêiner externo.
21      Este exemplo usa a propriedade de estouro de CSS para forçar a
22      caixa a rolar quando há muito texto para caber na caixa.
23    </div>
24  </body>
25 </html>
```

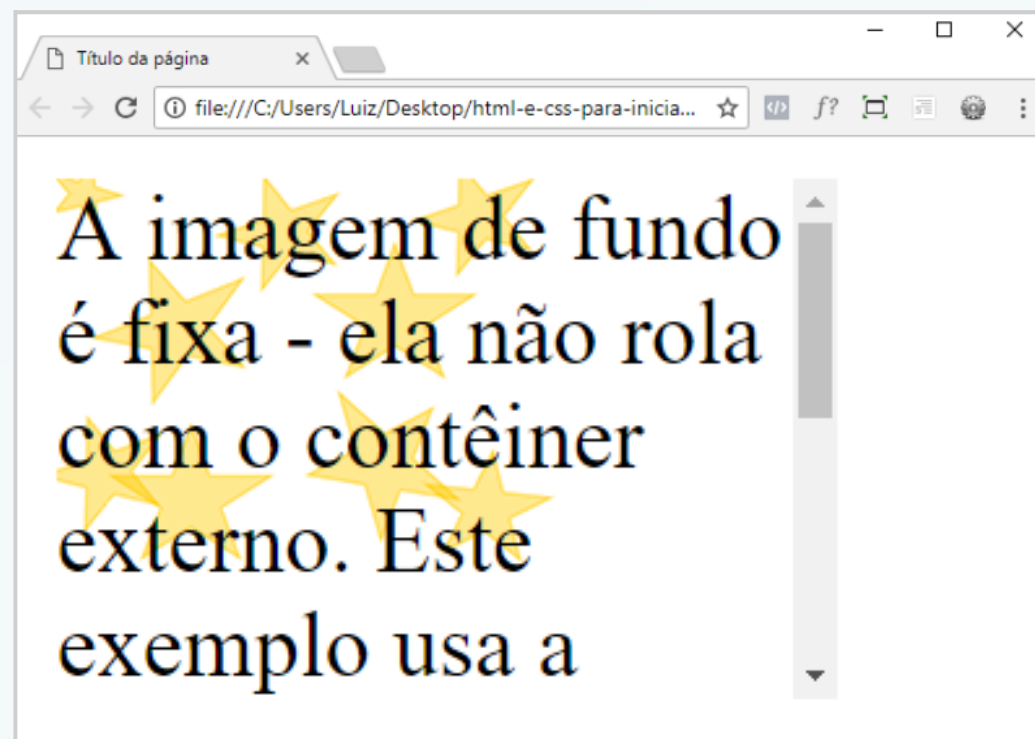
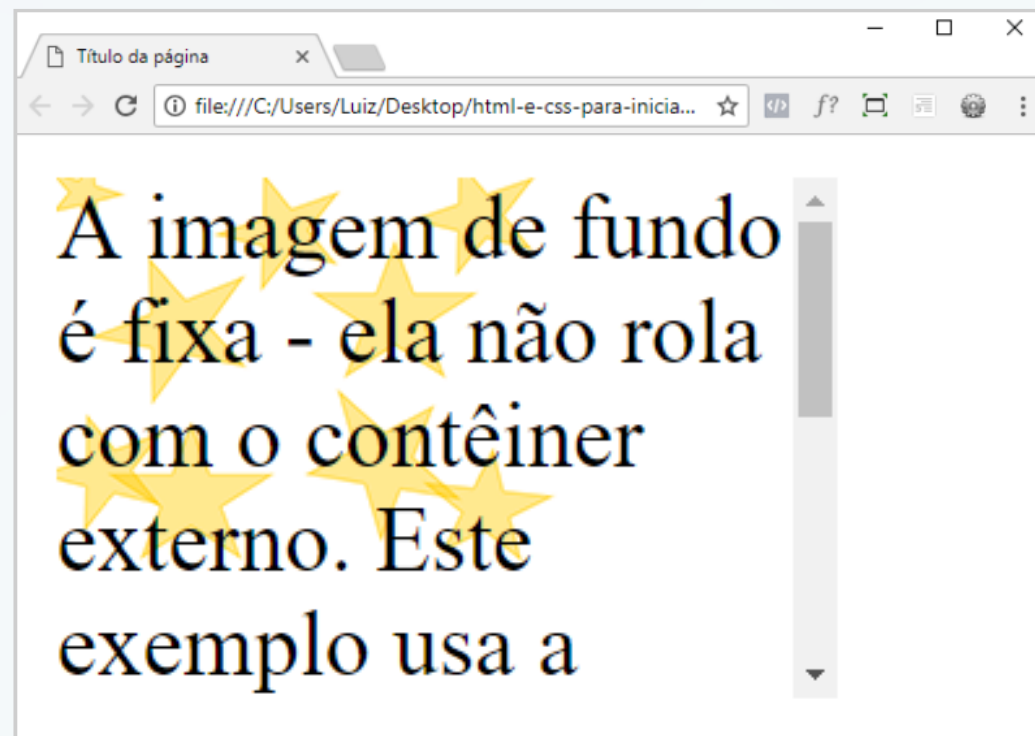


Imagem de plano de fundo no CSS

Código abreviado.

Podemos usar a propriedade background para definir todas as propriedades de plano de fundo de uma só vez.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         height: 200px;
9         width: 300px;
10        font-size: 35px;
11        overflow: auto;
12        background: url("bg.png") no-repeat fixed;
13      }
14    </style>
15  </head>
16  <body>
17    <div>
18      A imagem de fundo é fixa - ela não rola com o contêiner externo.
19      Este exemplo usa a propriedade de estouro de CSS para forçar a
20      caixa a rolar quando há muito texto para caber na caixa.
21    </div>
22  </body>
23 </html>
```



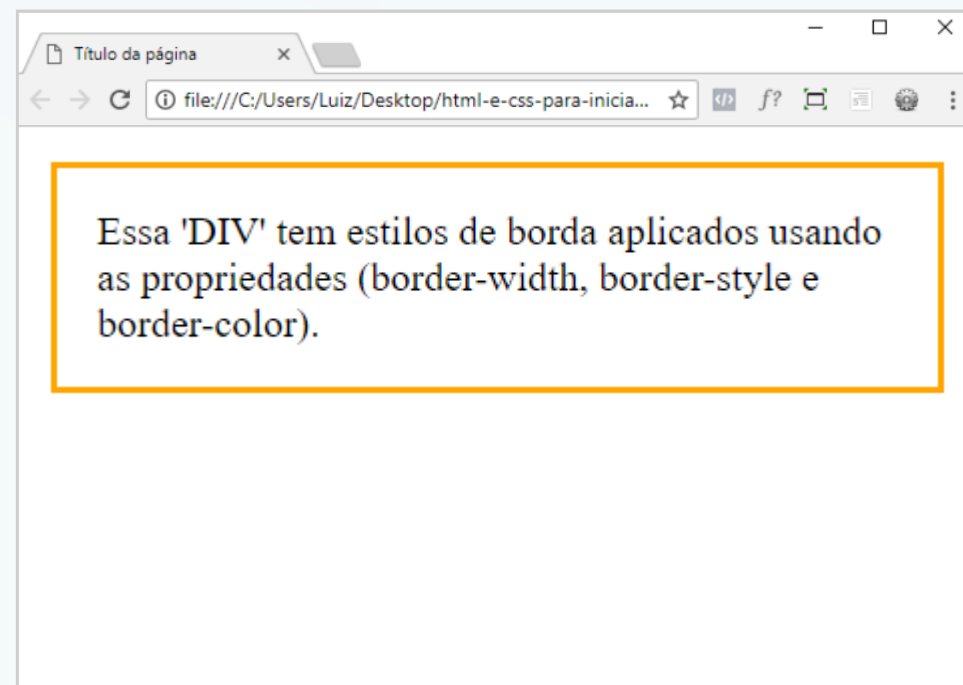
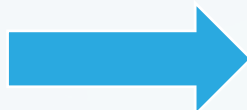
Propriedades de borda no CSS

O **CSS** nos permite que definir estilos para bordas dos elemento **HTML**. E também fornece uma maneira de configurar estilos de borda para um ou mais lados de um elemento.

Como definir bordas em todos os lados?

Para definir estilos de borda para todos os lados de um elemento, podemos usar as propriedades (**border-width**, **border-style** e **border-color**). Você também pode usar a propriedade "**border**" para definir todas as propriedades de uma só vez.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         width: 400px;
9         font-size: 20px;
10        padding: 20px;
11
12        border-width: 3px;
13        border-style: solid;
14        border-color: orange;
15      }
16    </style>
17  </head>
18  <body>
19    <div>
20      Essa 'DIV' tem estilos de borda aplicados usando as
21      propriedades (border-width, border-style e border-color).
22    </div>
23  </body>
24 </html>
```

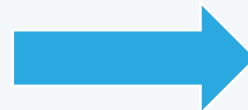


Propriedades de borda no CSS

Propriedade border.

A propriedade “border” é um atalho para a definição de (border-width, border-style e border-color).

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <style>
7       div {
8         width: 400px;
9         font-size: 20px;
10        padding: 20px;
11
12        border: 3px solid orange;
13      }
14    </style>
15  </head>
16  <body>
17    <div>
18      Essa 'DIV' tem estilos de borda aplicados usando a
19      propriedade (border) .
20    </div>
21  </body>
22 </html>
```



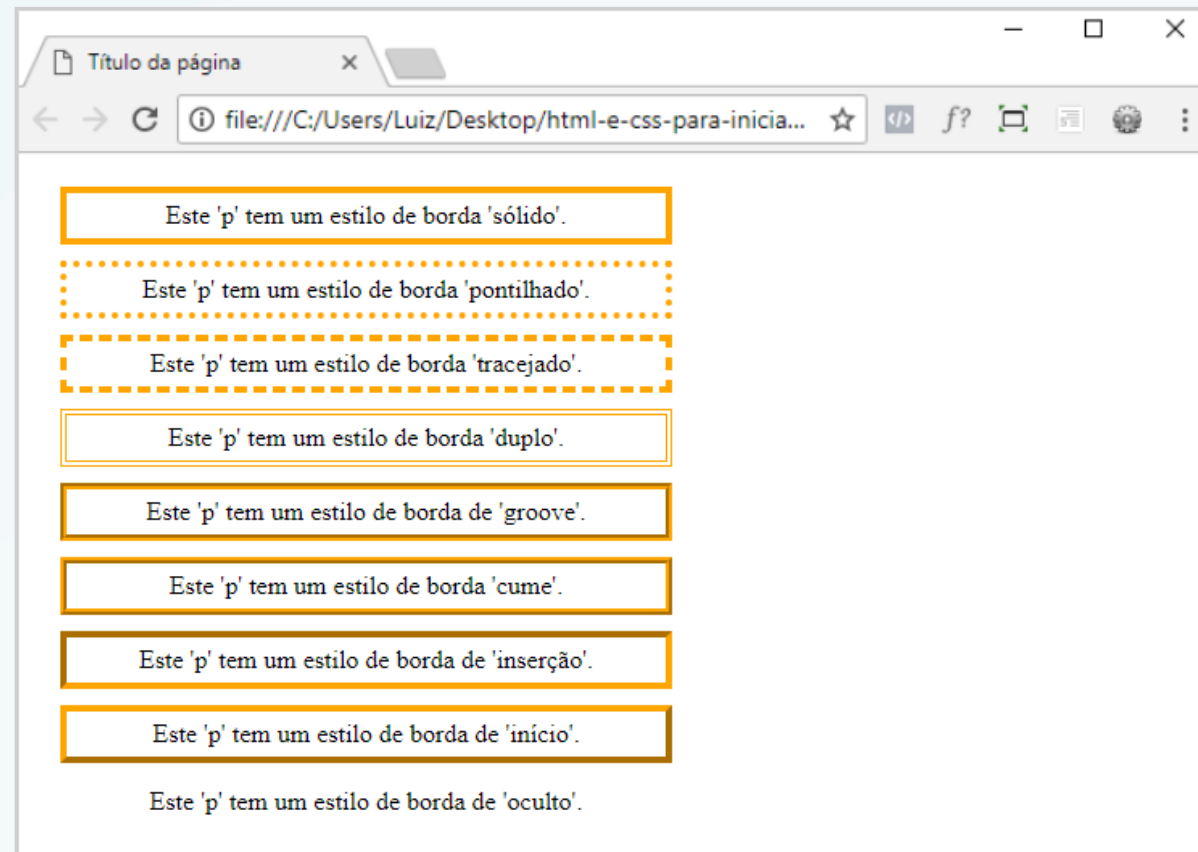
Propriedades de borda no CSS

Estilos de Borda.

As bordas tem os seguintes estilos.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      <p class="solid">Este 'p' tem um estilo de borda 'sólido'.</p>
11      <p class="dotted">Este 'p' tem um estilo de borda 'pontilhado'.</p>
12      <p class="dashed">Este 'p' tem um estilo de borda 'tracejado'.</p>
13      <p class="double">Este 'p' tem um estilo de borda 'duplo'.</p>
14      <p class="groove">Este 'p' tem um estilo de borda de 'groove'.</p>
15      <p class="ridge">Este 'p' tem um estilo de borda 'cume'.</p>
16      <p class="inset">Este 'p' tem um estilo de borda de 'inserção'.</p>
17      <p class="outset">Este 'p' tem um estilo de borda de 'início'.</p>
18      <p class="hidden">Este 'p' tem um estilo de borda de 'oculto'.</p>
19    </div>
20  </body>
21 </html>
```

```
1 div {width: 400px;}
2 p{
3   padding: 5px;
4   margin: 10px;
5   text-align: center;
6   border-color: orange;
7   border-width: 4px;
8 }
9 .solid {border-style: solid;}
10 .dotted {border-style: dotted;}
11 .dashed {border-style: dashed;}
12 .double {border-style: double;}
13 .groove {border-style: groove;}
14 .ridge {border-style: ridge;}
15 .inset {border-style: inset;}
16 .outset {border-style: outset;}
17 .hidden {border-style: hidden;}
```



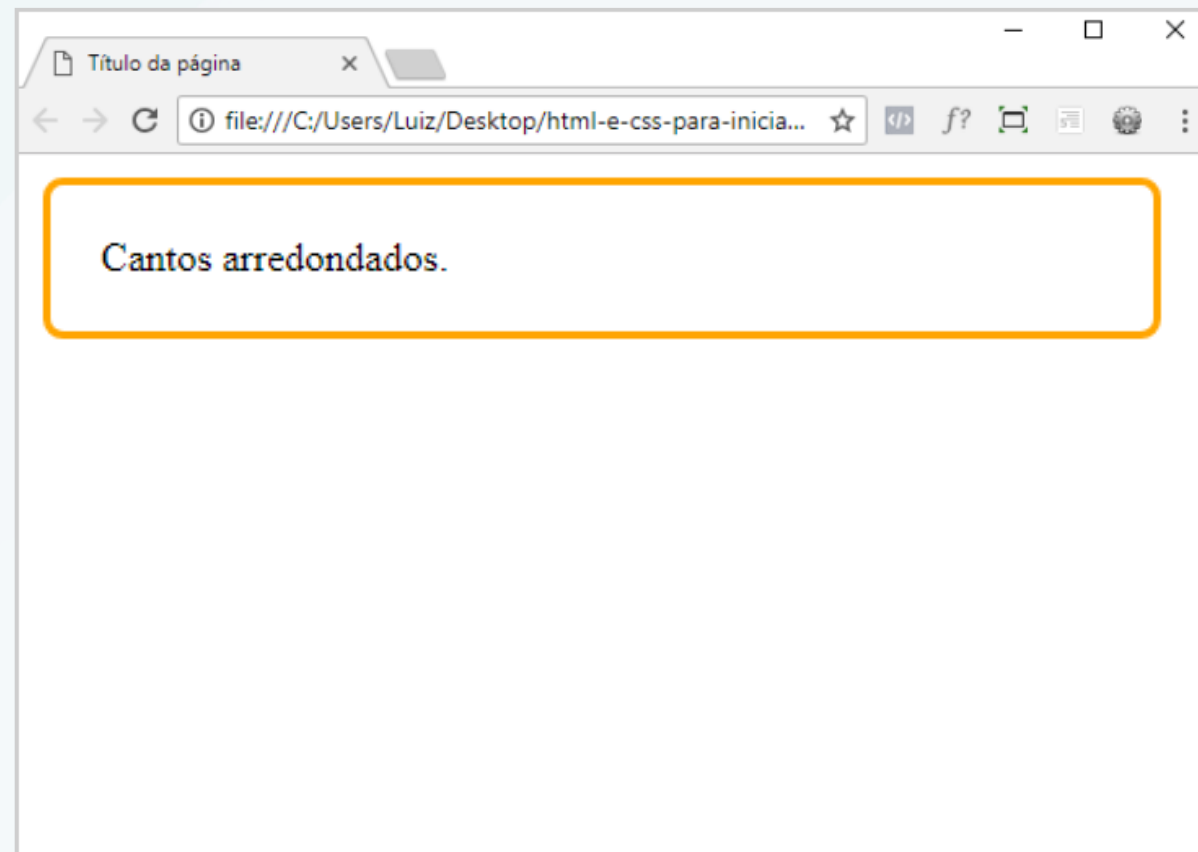
Propriedades de borda no CSS

Raio da Borda.

Podemos arredondar às bordas usando a propriedade "**border-radius**".

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      Cantos arredondados.
11    </div>
12  </body>
13 </html>
```

```
1 div {
2   width: 400px;
3   padding: 20px;
4   border: 3px solid orange;
5   border-radius: 8px;
6 }
```



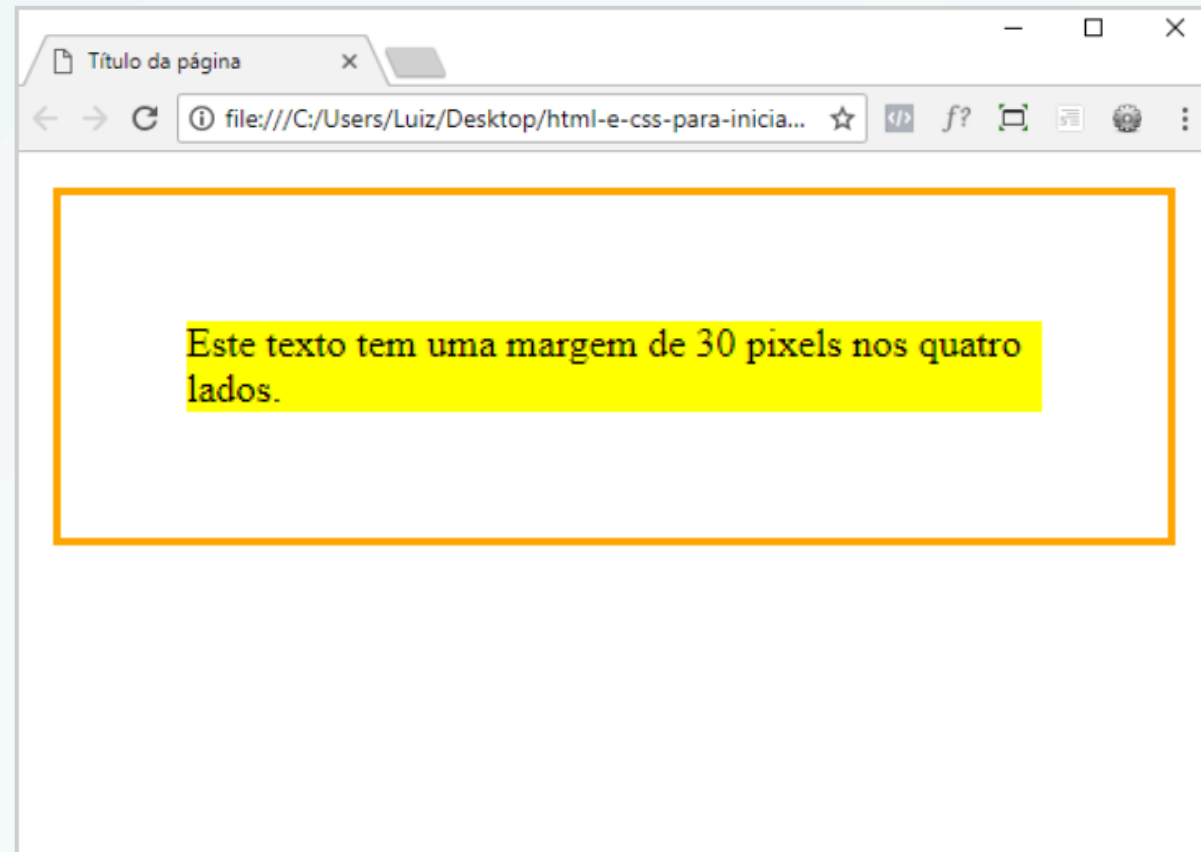
Propriedades de Margem no CSS

As margens definem o espaço ao redor do elemento. Elas são especificadas de maneira semelhante às bordas e podem ser definidas individualmente para cada lado ou todos os lados de uma só vez.

Definindo margens em todos os lados

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      <p>
11        Este texto tem uma margem de 30 pixels nos quatro lados.
12      </p>
13    </div>
14  </body>
15 </html>
```

```
1 div {
2   width: 400px;
3   padding: 20px;
4   border: 3px solid orange;
5 }
6 div p {
7   background: yellow;
8   margin: 30px;
9 }
```



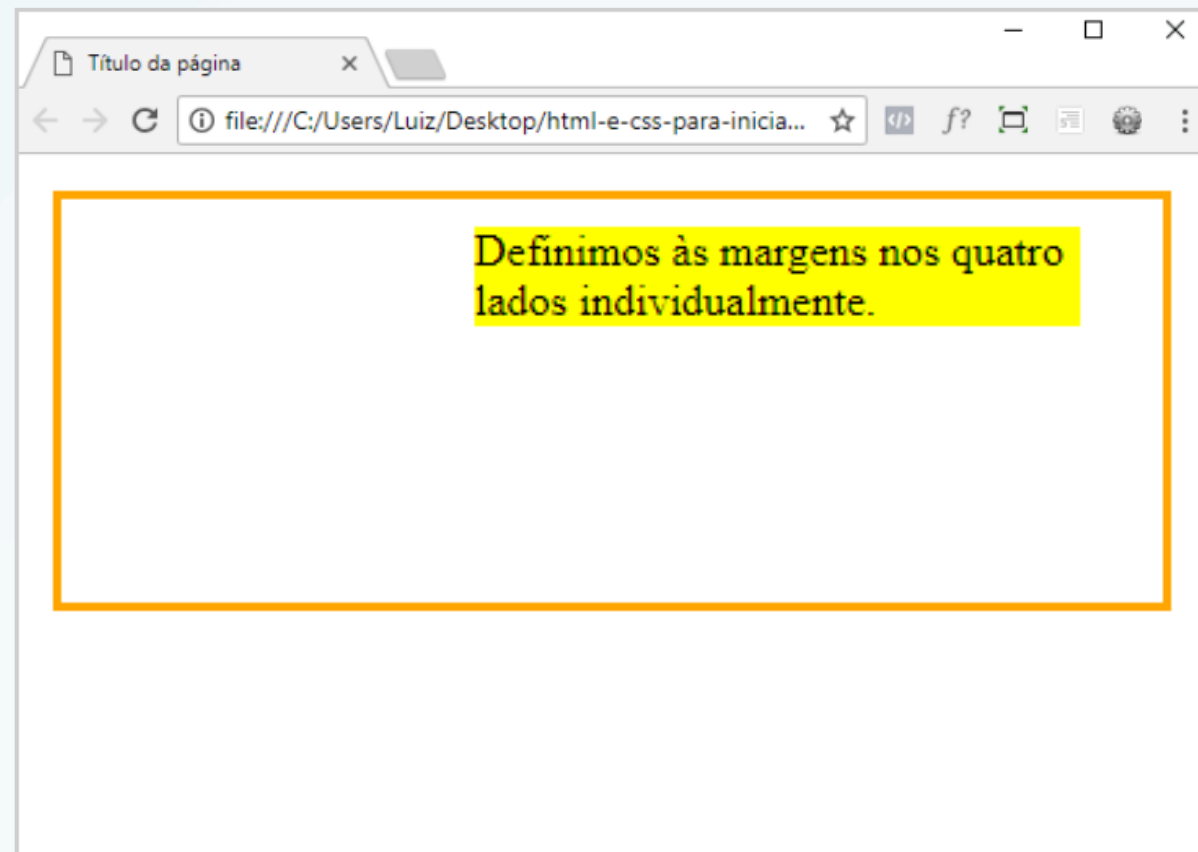
Propriedades de Margem no CSS

Definindo margens para cada lado.

Podemos definir as margens individualmente da seguinte forma.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      <p>
11        Definimos às margens nos quatro lados individualmente.
12      </p>
13    </div>
14  </body>
15 </html>
```

```
1 div {
2   width: 400px;
3   border: 3px solid orange;
4 }
5 div p {
6   background: yellow;
7   margin-top: 10px;
8   margin-right: 30px;
9   margin-bottom: 100px;
10  margin-left: 150px;
11 }
```



Propriedades de preenchimento no CSS

O preenchimento define o espaço entre a borda do elemento e seu conteúdo. O preenchimento é especificado exatamente como as margens e pode ser definido individualmente para cada lado ou todos os lados de uma só vez.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      <p>
11        Preenchimento de 30 pixels em todos os quatro lados.
12      </p>
13    </div>
14  </body>
15 </html>
```

```
1 div {
2   width: 400px;
3   background-color: orange;
4   padding: 30px;
5 }
6 div p {
7   background: white;
8   margin: 0;
9 }
10 }
```



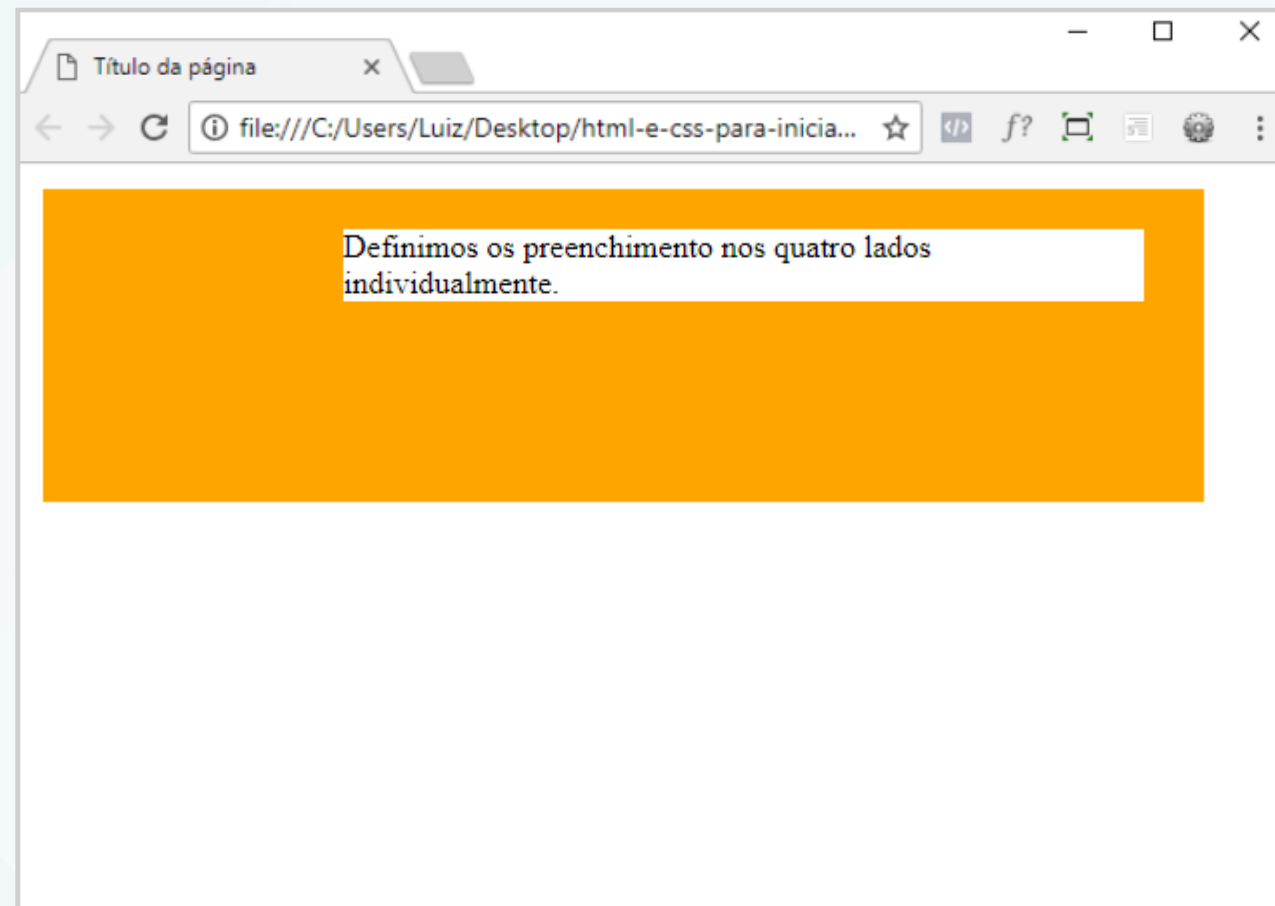
Propriedades de preenchimento no CSS

Preenchimento individualmente para cada lado.

Podemos definir os preenchimentos individualmente da seguinte forma.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      <p>
11        Preenchimento de 30 pixels em todos os quatro lados.
12      </p>
13    </div>
14  </body>
15 </html>
```

```
1 div {
2   width: 400px;
3   background-color: orange;
4   padding-top: 20px;
5   padding-right: 30px;
6   padding-bottom: 100px;
7   padding-left: 150px;
8 }
9 div p {
10  background: white;
11  margin: 0;
12 }
13 }
```

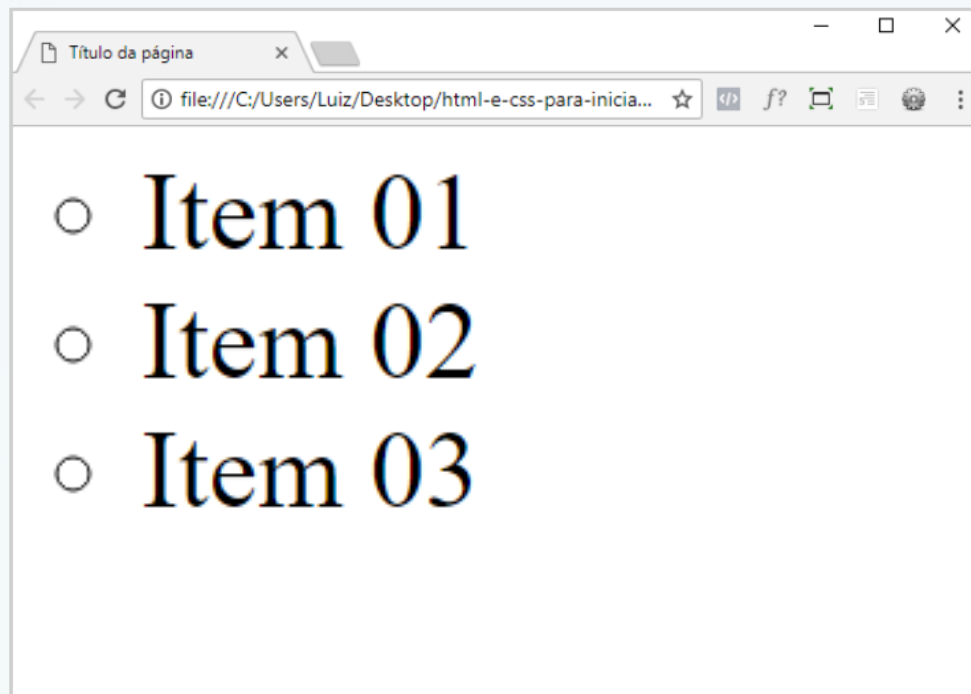


Estilizar listas com CSS

As listas em **HTML** podem ter vários estilos diferentes aplicados usando a propriedade "**list-style-type**" do **CSS** para especificar o tipo de estilo de lista e como os marcadores de item da lista devem ser estilizados.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <ul>
10      <li>Item 01</li>
11      <li>Item 02</li>
12      <li>Item 03</li>
13    </ul>
14  </body>
15 </html>
```

```
1 ul {
2   font-size: 40px;
3   list-style-type: circle;
4 }
```



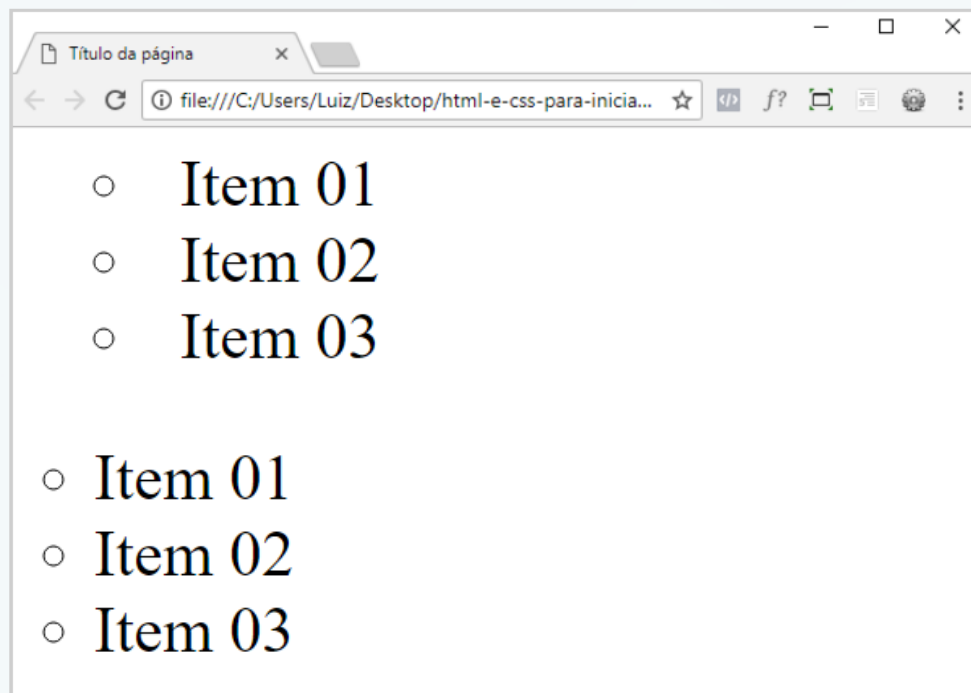
Estilizar listas com CSS

Posição de estilo.

A propriedade "**list-style-position**" determina se o marcador está localizado dentro da caixa contendo a lista ou fora dela.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <ul class="inside">
10      <li>Item 01</li>
11      <li>Item 02</li>
12      <li>Item 03</li>
13    </ul>
14
15    <ul class="outside">
16      <li>Item 01</li>
17      <li>Item 02</li>
18      <li>Item 03</li>
19    </ul>
20  </body>
21 </html>
```

```
1 ul {
2   font-size: 40px;
3   list-style-type: circle;
4 }
5 .inside {
6   list-style-position: inside;
7 }
8 .outside {
9   list-style-position: outside;
10 }
```



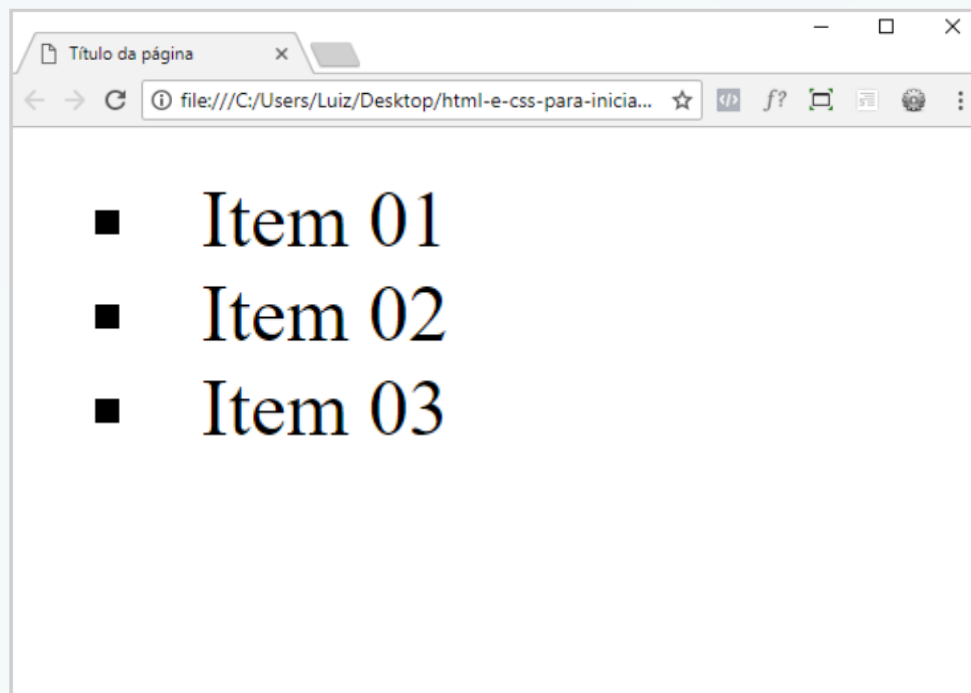
Estilizar listas com CSS

Estilo da lista

A propriedade "**list-style**" é uma forma abreviada de definir as propriedades da lista.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <ul>
10      <li>Item 01</li>
11      <li>Item 02</li>
12      <li>Item 03</li>
13    </ul>
14  </body>
15 </html>
```

```
1 ul {
2   font-size: 40px;
3   list-style: square inside;
4 }
```



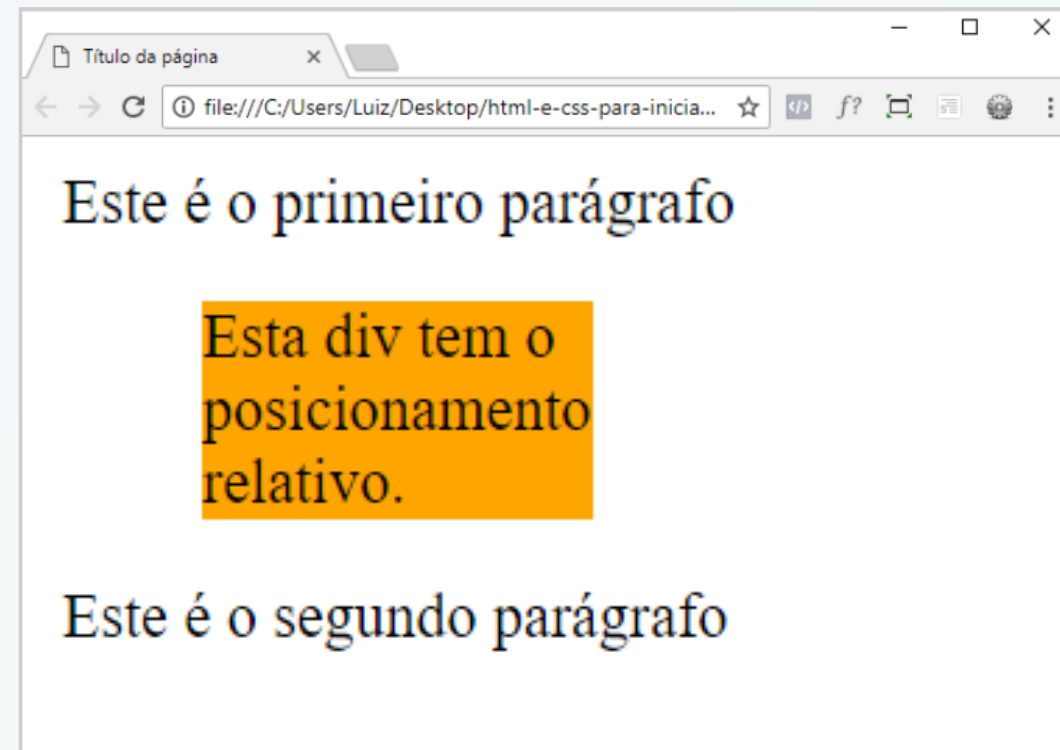
Posicionamento no CSS

Usamos a propriedade “position” do **CSS** para especificar como um elemento é posicionado e onde ele deve estar localizado na página do site.

Posicionamento Relativo.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <p>Este é o primeiro parágrafo</p>
10
11     <div>
12       Esta div tem o posicionamento relativo.
13     </div>
14
15     <p>Este é o segundo parágrafo</p>
16   </body>
17 </html>
```

```
1 body{font-size: 22px;}
2 div {
3   position: relative;
4   left: 50px;
5   background-color: orange;
6   width: 140px;
7 }
```



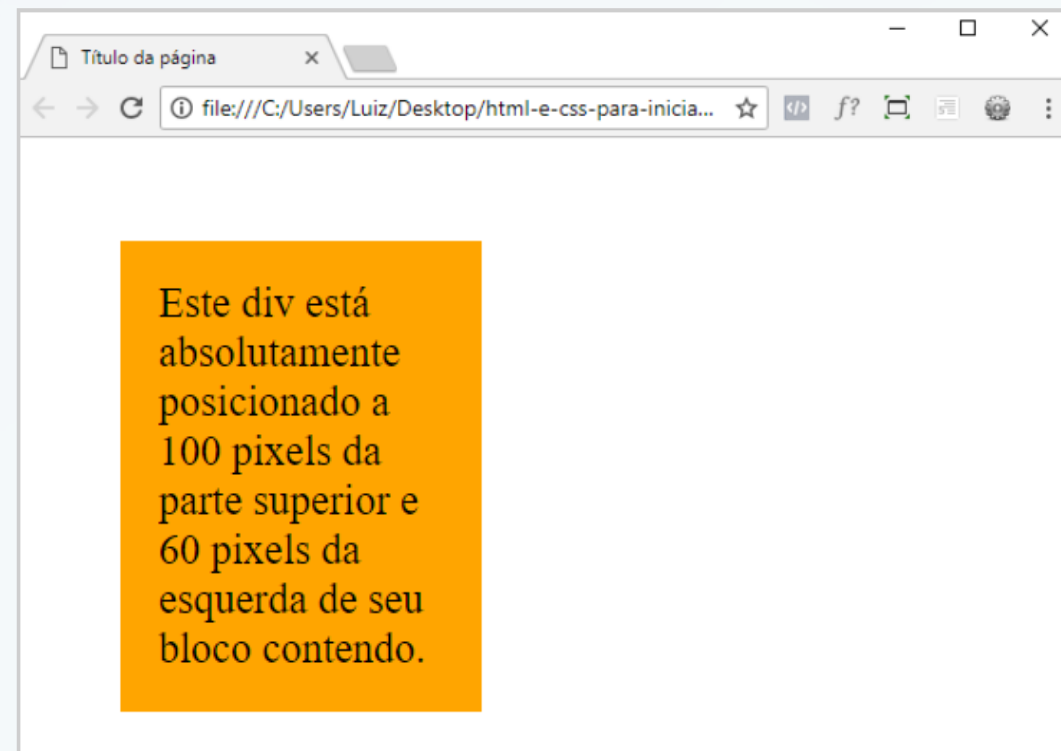
Posicionamento no CSS

Posicionamento Absoluto.

Para realizar o posicionamento absoluto no **CSS**, usamos novamente a propriedade "**position**". Só que, desta vez, usamos o "**position: absolute;**" seguido pelo deslocamento desejado.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      Esta 'div' está posicionado a 50 pixels da parte
11      superior e 50 pixels da esquerda de seu bloco contendo.
12    </div>
13  </body>
14 </html>
```

```
1 body{font-size: 22px;}
2 div {
3   position: absolute;
4   top: 50px;
5   left: 50px;
6   background-color: orange;
7   width: 150px;
8   padding: 20px;
9 }
```



Posicionamento no CSS

Posicionamento Fixo.

O posicionamento fixo permite fixar a posição de um elemento em um ponto específico da página, independente da rolagem.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div>
10      Esta 'div' está posicionado a 50 pixels da parte
11      superior e 50 pixels da esquerda de seu bloco contendo,
12      porém está fixa, e não se moverá ao rolar o scroll.
13    </div>
14  </body>
15 </html>
```

```
1 body{font-size: 22px;}
2 div {
3   position: fixed;
4   top: 50px;
5   left: 50px;
6   background-color: orange;
7   width: 150px;
8   padding: 20px;
9 }
```



Float no CSS

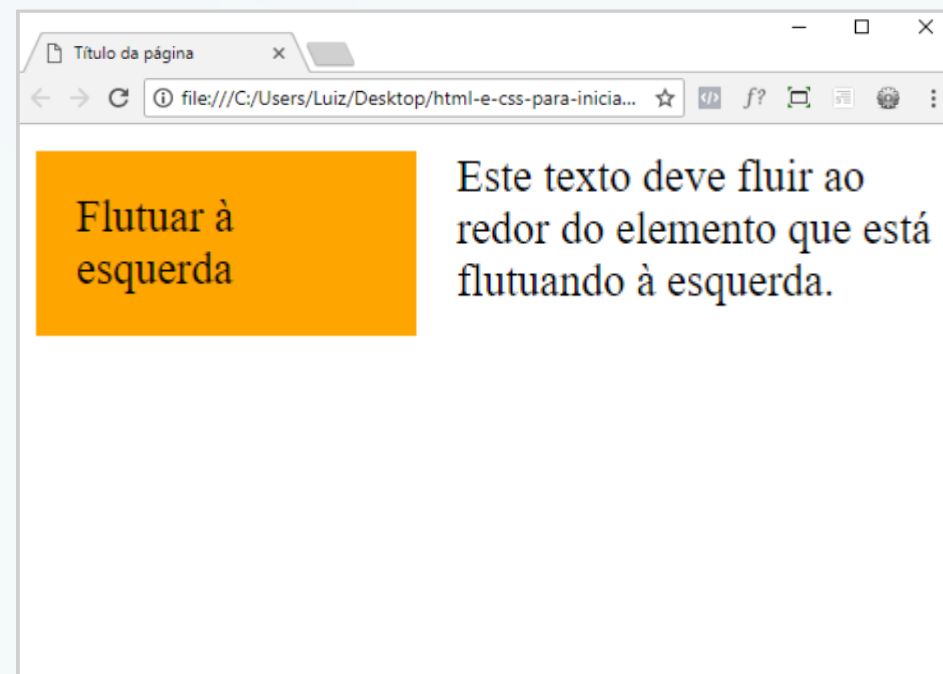
Flutuar à esquerda.

Podemos flutuar um elemento para a direita ou para a esquerda para que outros conteúdos fluam em torno dele. A propriedade “float” nos permite posicionar um elemento em relação aos outros elementos da página.

Supondo que queremos que um elemento “DIV” flutua à Esquerda, então, usamos a propriedade “float: left”, isso faz com que todos os elementos irmãos que não esteja com tamanho definido em 100% fica ao lado.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div class="float-left">
10      Flutuar à esquerda
11    </div>
12
13    <div>
14      Este texto deve fluir ao redor do elemento que está flutuando à esquerda.
15    </div>
16  </body>
17 </html>
```

```
1 body{font-size: 22px;}
2
3 .float-left {
4   float:left;
5   background-color: orange;
6   width: 150px;
7   padding: 20px;
8   margin-right: 20px;
9 }
```



Float no CSS

Flutuar à direita.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div class="float-right">
10      Flutuar à direita
11    </div>
12
13    <div>
14      Este texto deve fluir ao redor do elemento que está flutuando à direita.
15    </div>
16  </body>
17 </html>
```

```
1 body{font-size: 22px;}
2
3 .float-right {
4   float:right;
5   background-color: orange;
6   width: 150px;
7   padding: 20px;
8   margin-right: 20px;
9 }
```



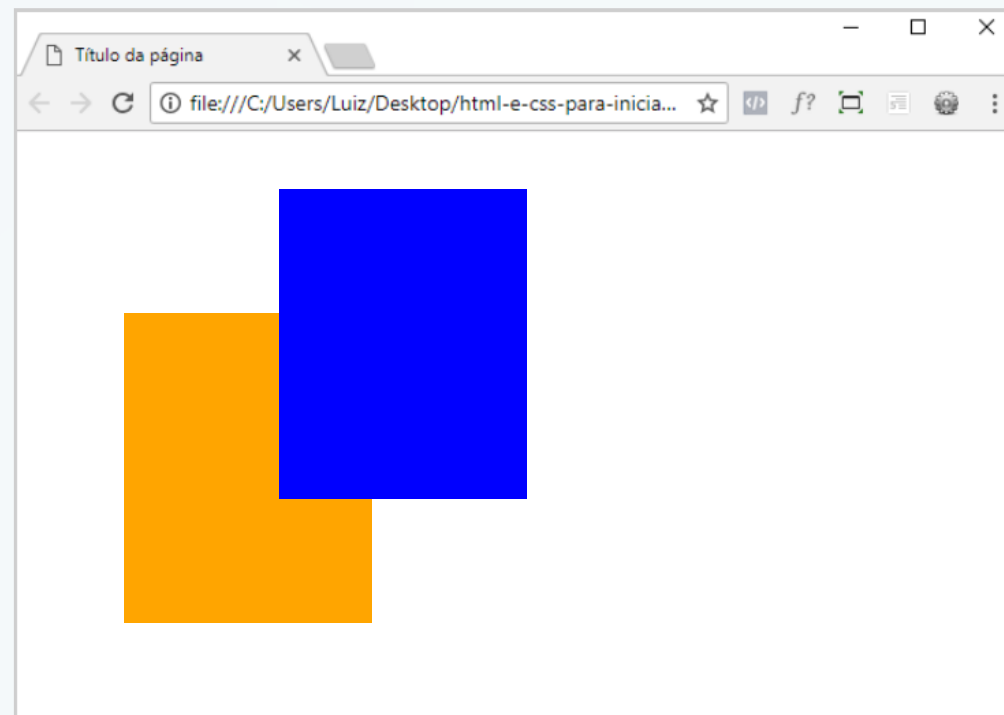
Camadas no CSS

Podemos usar **CSS** para adicionar camadas (layers) de elementos, para que um elemento possa se posicionar na frente ou atrás de outros elementos. Em **CSS**, camadas referem-se à aplicação da propriedade "**z-index**", elemento que se sobrepõem uns aos outros.

A propriedade "**z-index**", quando usada em conjunto com a propriedade "**position**", permite especificar qual elemento deve aparecer na parte superior no caso de uma sobreposição. Uma sobreposição pode ocorrer facilmente ao usar a propriedade "**position**", e isso geralmente é desejável ao criar layouts avançados.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div class="bloco1"></div>
10    <div class="bloco2"></div>
11  </body>
12 </html>
```

```
1 div {
2   width: 80px;
3   height: 100px;
4   position: relative;
5 }
6 .bloco1 {
7   background-color: blue;
8   top: 10px;
9   left: 80px;
10  z-index: 2;
11 }
12 .bloco2 {
13   background-color: orange;
14   top: -50px;
15   left: 30px;
16   z-index: 1;
17 }
```

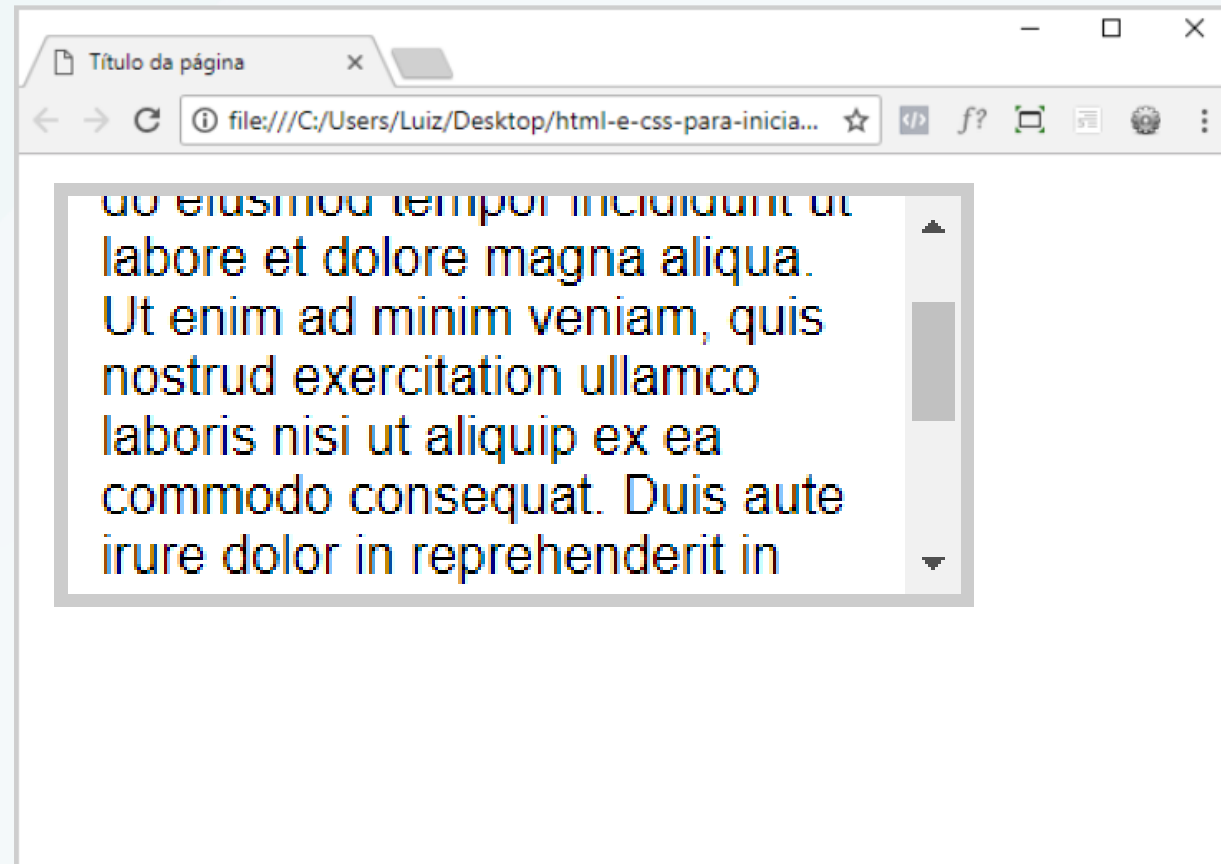


Caixa de rolagem no CSS

Caixa de rolagem no **HTML** é uma caixa que aumenta as barras de rolagem quando seu conteúdo é muito grande para caber na caixa. Podemos criar uma caixa usando um elemento **HTML** com o elemento "**div**". Então, para fazer a rolagem da caixa, aplicamos a propriedade "**overflow**" do **CSS** ao elemento "**div**".

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Aula 1 - CSS</title>
6     <link type="text/css" rel="stylesheet" href="aula1.css" />
7   </head>
8   <body>
9     <div class="caixa">
10      "Lorem ipsum dolor sit amet, consectetur adipiscing elit,
11      sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
12      Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
13      nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in
14      reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
15      Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia
16      deserunt mollit anim id est laborum."
17    </div>
18  </body>
19 </html>
```

```
1 .caixa{
2   height:100px;
3   width:250px;
4   padding: 10px;
5   font: 16px Arial, sans-serif;
6   border:4px solid #ccc;
7   overflow:auto;
8 }
```



Referências

1. <https://www.w3schools.com>

2. <https://developer.mozilla.org/pt-BR/docs/Aprender/HTML>

BLOG:

<https://www.htmlecsspro.com>

Youtube:

<https://www.youtube.com/htmlecsspro>

Facebook:

<https://www.facebook.com/htmlecsspro>

Contato:

contato@htmlecsspro.com

AO SEU SUCESSO. OBRIGADO!!!

- LUIZ AUGUSTO VIEIRA



INSTRUTOR
LUIZ AUGUSTO VIEIRA

