

# Introdução ao HTML5 e CSS3 – parte 1

Profª Mª Denilce Veloso  
[denilce.veloso@fatec.sp.gov.br](mailto:denilce.veloso@fatec.sp.gov.br)  
[denilce@gmail.com](mailto:denilce@gmail.com)

# HTML5

- **HTML** (Hypertext Markup Language ou Linguagem de Marcação de Hipertexto) é uma linguagem para estruturação e apresentação de conteúdo para a World Wide Web e é uma tecnologia chave da Internet originalmente proposto por Opera Software. HTML5 é a quinta versão da linguagem HTML.

\*\* A linguagem HTML foi criada por Tim Berners Lee na década de 1990. As especificações da linguagem são controladas pela W3C (World Wide Web Consortium).



# HTML

- Atenção: Linguagem de Marcação é diferente de Linguagem de Programação

Boa noite, meu nome é

`<nome>Denilce</nome>`



Tags/etiqueta/marcação



# Editores

Pessoal da  
Fatec 

**Exemplos: (Atom, Sublime, NotePad++, Visual Studio Code, Adobe Dreamweaver CC, Free Editor ...**

## *Editores Online*

<https://vscode.dev/>

<http://Repl.it/>

<https://codepen.io/>

<https://jsfiddle.net/>

<https://jsbin.com/?html,output>



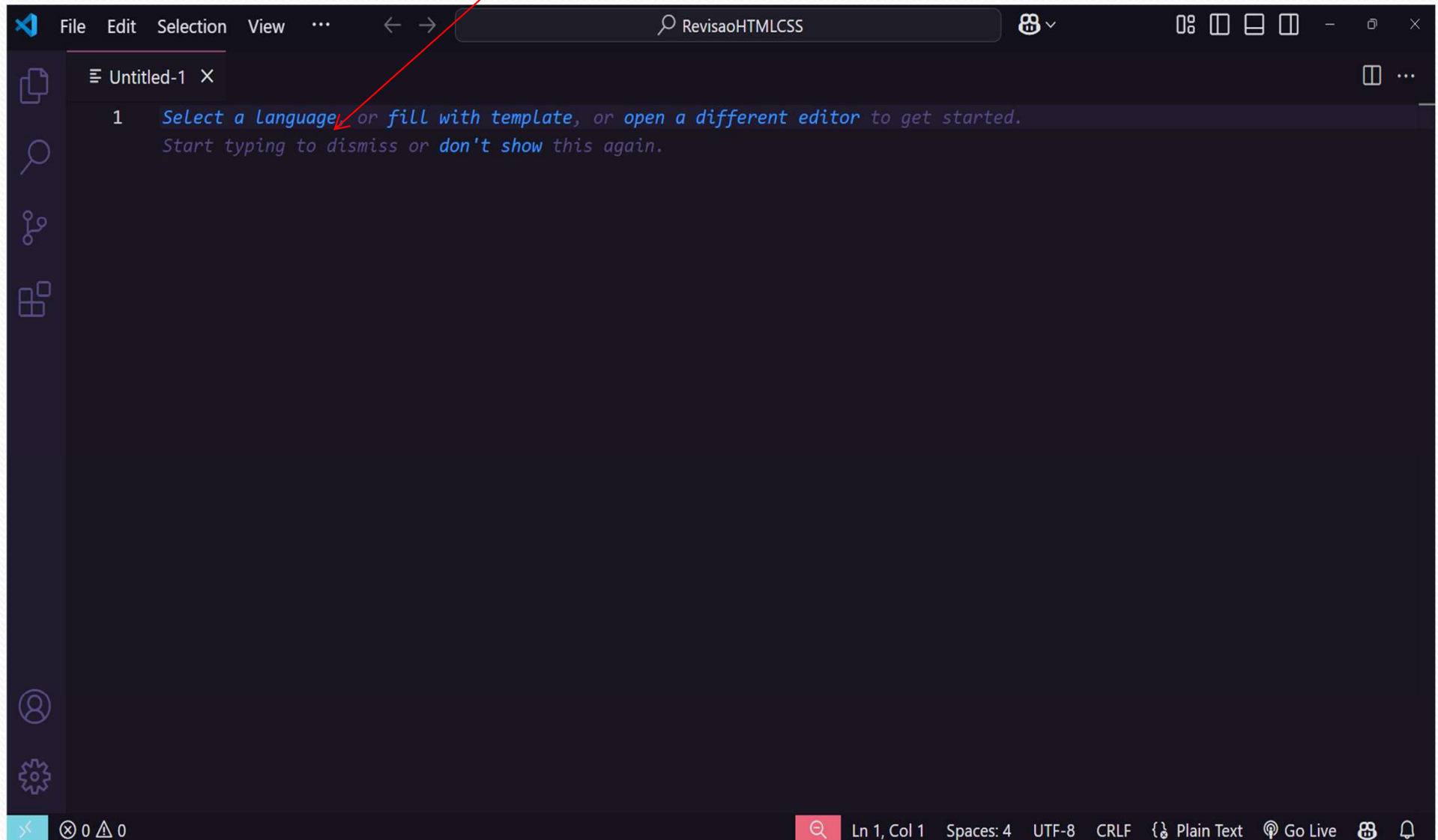
# (parênteses) Editores

## Visual Studio Code (VS Code)

O VS Code é mais do que um editor de código-fonte, sendo uma plataforma de desenvolvimento completa com recursos. É executado no Windows, macOS e Linux, com extensões que dão suporte a várias linguagem de programação.

# VS Code

Criar novo arquivo, escolher linguagem





# VS Code (Dicas)

- ! Ou HTML:5 → cria a estrutura básica do HTML (extensão do **EMMET** já vem pré-instalada)

- Tags Básicas

<tag> + Enter

Por exemplo:

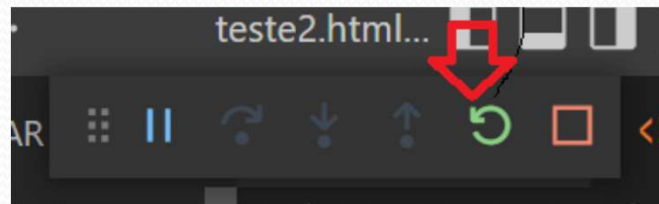
p + enter → <p></p>

div + #container → <div id="container"></div>

li\*5

Executar (F5 ou executar)

Reiniciar (quando faz alterações -> ctrl + shift + f5) ou

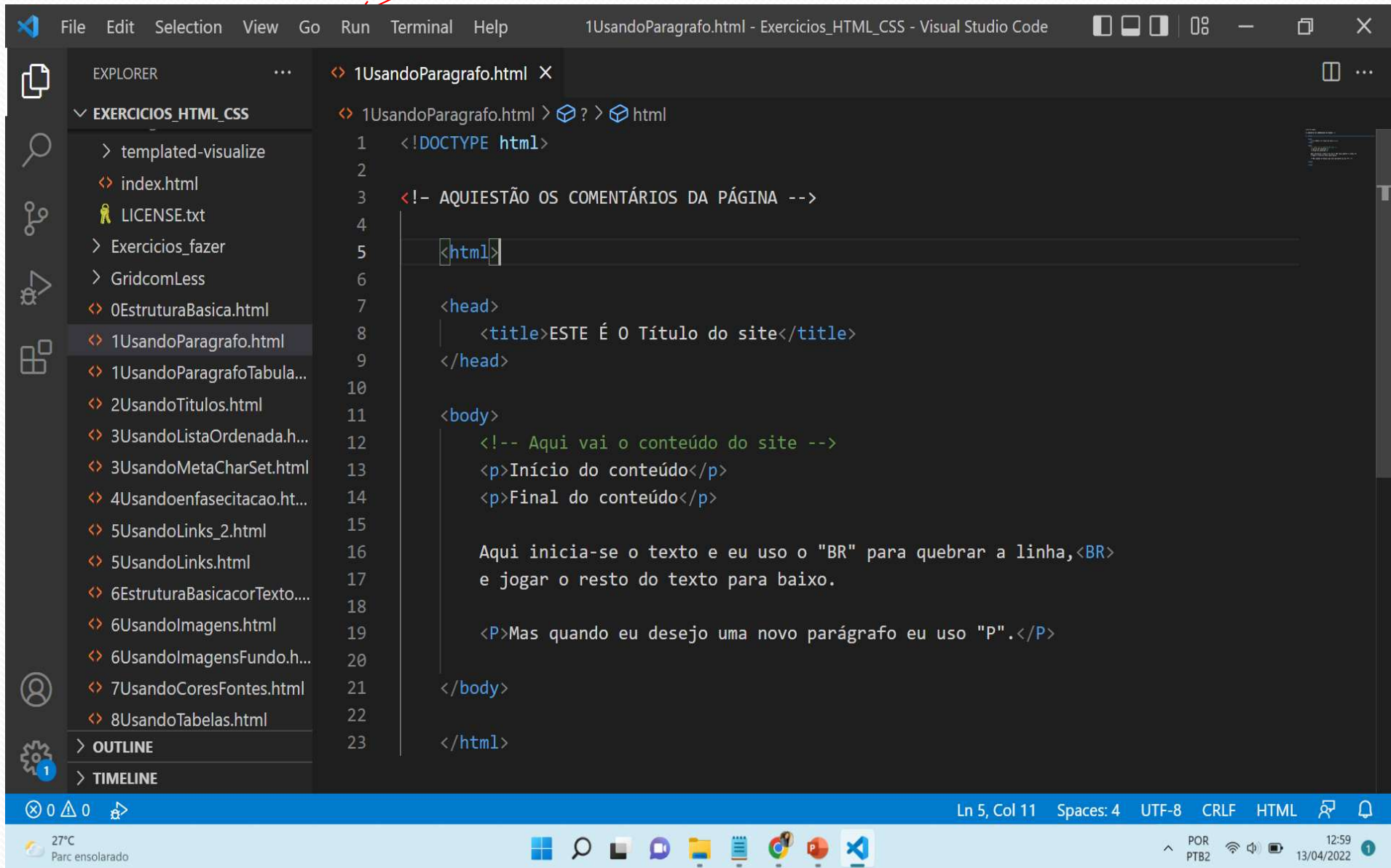


Ver mais:

- <https://cibersistemas.pt/tecnologia/como-escrever-html-css-mais-rapido-com-codigos-de-trapaca-emmet-no-codigo-vs/#:~:text=Para%20criar%20tags%20HTML%20ob%C3%AAsicas,h1%20Enter%20%2C%20ou%20p%20Enter%20>

# VS Code

Com arquivo aberto, para executar RUN\Start ou F5



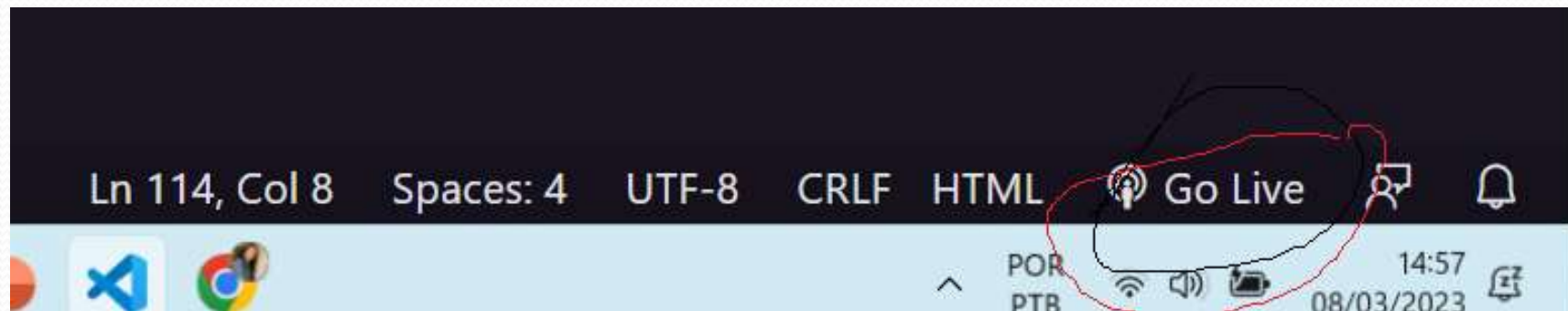


# VS Code x Browser

Dica:

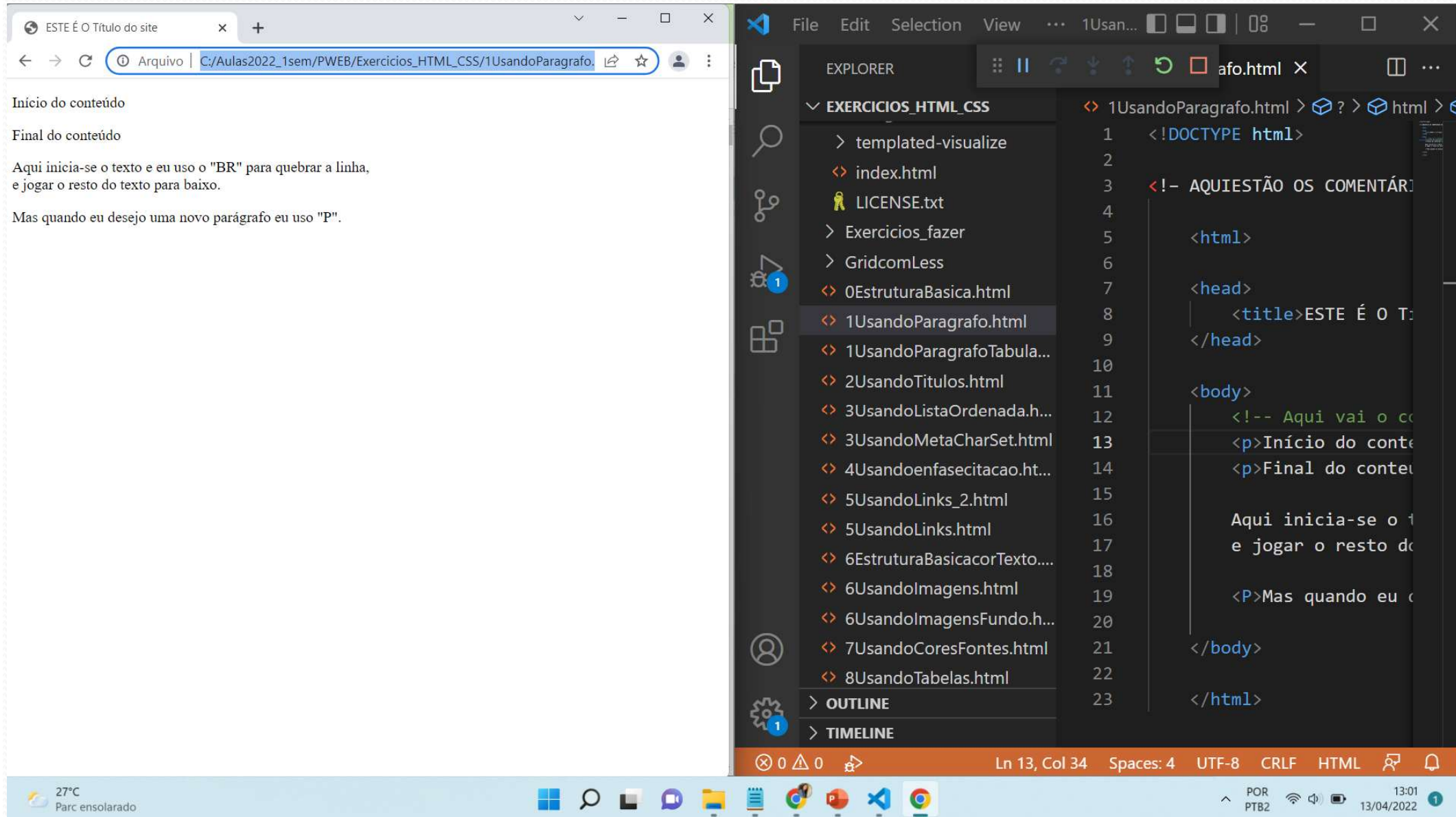
Caso não funcione o F5, instalar a extensão Live Server Preview (cria um servidor local para visualização). E para executar digitar F1 e escolher Open With Live Server.

Para executar pode usar o botão Go Live no canto inferior à direita ou (ALT+L+O).



# VS Code x Browser

Dividir a tela e mostrar o código e o navegador





# VS Code - Dica de Extensão - IA

O GitHub Copilot para o VS Code é uma ferramenta de inteligência artificial que ajuda os desenvolvedores a escrever código mais rápido e eficientemente. Ele pode ser usado para:

- Sugerir códigos
- Corrigir erros
- Refatorar códigos
- Encontrar informações
- Geração de testes

\* *instalar extensão*

- <https://docs.github.com/pt/copilot>



# VS Code – Dica de Extensão – Omni Theme, Dracula Theme etc

Os temas trazem cores, fontes e configurações que se adaptam as necessidades de cada dev, o Omni Theme é uma dessas extensões.



# VS Code – Dica de Extensão - CodeSnap

É uma extensão para VS Code muito utilizada para tirar screenshots (prints) mais elegantes dos seus códigos, com ela é possível selecionar o trecho que deseja capturar e então basta salvar a foto.

<https://marketplace.visualstudio.com/items?itemName=adpyke.codesnap>

**\*\*** para instalar extensões CTRL + SHIFT + P



# VS Code – Dica de Extensão - MobileView

A “MobileView for VS Code” é uma ferramenta poderosa para desenvolvedores móveis, pois simplifica o processo de teste e depuração de aplicativos em vários dispositivos.

*Fonte: <https://www.dio.me/articles/desvendando-a-extensao-mobileview-for-vs-code>*



# HTML – Estrutura Básica

```
<html>  
  <head>  
    <title>Título da Página</title>  
  </head>  
  <body>  
    Esta é minha primeira página.  
  </body>  
</html>
```

Toda a estrutura da página deve estar entre uma tag `<html>`. E dentro dessa tag, é necessário declarar outras duas tags: `<head>` e `<body>`. A tag `<body>` contém o corpo do documento, que é exibido pelo navegador em sua janela.

# HTML5 - DocType, comentários e Parágrafo

`<!DOCTYPE html>`

O DOCTYPE não é uma tag, mas uma instrução especial. Indica para o navegador qual versão do HTML deve ser utilizada para renderizar a página.

`<!DOCTYPE html>`, indica a versão 5. Navegadores modernos suportam HTML5 por padrão.

`<!-- Comentários` Todo o texto que colocar dentro está comentado, ou seja, não aparecerá na página. `-->`

`<p>` - PARÁGRAFOS - Se existirem vários parágrafos de texto, é recomendado usar várias tags `<p>` para separá-los.

*No VS Code → ctrl + : → coloca a tag de comentário*

**\*\*as tags podem estar em minúsculas ou maiúsculas não é case sensitive, mas a convenção de usar minúsculas é amplamente adotada para melhor legibilidade e organização do código.**



# HTML5 – Linguagem (idioma)

Indicar sempre o idioma principal da página:

`<html lang="pt-br">` (Português do Brasil)

Quando houver mudança de idioma no conteúdo, o ideal é sempre identificar a mudança, por exemplo:

`<p lang="en">Hello world</p>`



# HTML5 –Exemplo: 1Usandoparagrafo.html

```
<!DOCTYPE html>
```

```
<!-- AQUI ESTÃO OS COMENTÁRIOS  
      DA PÁGINA  
-->
```

```
<html>
```

```
    <head>
```

```
        <title>Título do site</title>Exemplo: 1Usandoparagrafo.html
```

```
    </head>
```

```
    <body>
```

```
        <!-- Aqui vai o conteúdo do site -->
```

```
        <p>Início do conteúdo</p>
```

```
        <p>Final do conteúdo</p>
```

Aqui inicia-se o texto e eu uso o "BR" para quebrar a linha,<BR>  
e jogar o resto do texto para baixo.

```
<P>Mas quando eu desejo uma novo parágrafo eu uso "P".</P>
```

```
    </body>
```

```
</html>
```



# HTML5 – Tabulação

A tag <pre>

Através da tag <pre></pre> do HTML, pode-se incluir um texto que ficará com a formatação original, ou seja, todos os espaços digitados no elemento serão incluídos na renderização. Além disso, a tag <pre> traz consigo um valor de margem superior e inferior de **1em**, assim como a tag <p>.

Os caracteres &nbsp; , &ensp e &emsp - indica respectivamente 1, 2, ou 3 espaços na tabulação

\*\*

*Outros*

*caracteres*

*veja:*

<https://www.dm.ufscar.br/~waldeck/curso/html/apend/tabcod.html>



# HTML5 – Tabulação - Exemplo

Exemplo: 1UsandoparagrafoTabulado.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Título do site</title>
```

```
</head>
```

```
<body>
```

```
  <p>Teste de tabulação</p>
```

```
  <pre >
```

```
    Teste de Tabulação</pre>
```

```
  <p>&nbsp;Teste de Tabulação para 1 espaço</p>
```

```
  <p>&ensp;Teste de Tabulação para 2 espaços</p>
```

```
  <p>&emsp;Teste de Tabulação para 3 espaços</p>
```

```
</body>
```

```
</html>
```



# HTML5 - Títulos

- <H1 ... H6> - Quando um texto é um título costuma utilizar-se as tags de heading em sua marcação. O h1 é utilizado para descrever o principal título da página.
- Mecanismos de busca (ex. Google, Yahoo e Bind) utilizam essas tags como meio de identificação do seu Site.
- Geralmente utiliza-se a tag <h1> para representar o título(logo) do site, pois será o primeiro elemento no qual os mecanismos de busca irão interpretar.O <h2> colocamos para a descrição do site. O <h3> torna-se o título dos artigos do site e assim por diante.
- Os navegadores especiais para acessibilidade também interpretam o conteúdo dessas tags de maneira a facilitar a navegação do usuário pelo documento.



# HTML5 - Títulos

Em questão de semântica o h1 é o mais forte, então dica:

H1 – use 1 vez

H2 – usado em subtítulos

H3 – só será usado depois de h2

E assim por diante...



# HTML5 – Exemplo: 2Usandotitulos.html

```
<html>
```

```
<head>
```

```
<title>Testando Títulos</title>
```

```
<meta name="author" content=Denilce/>
```

```
</head>
```

```
<body>
```

```
<H1>Aqui será Exibido o texto com a cor padrão da página,e no  
maior tamanho possível</H1>
```

```
<H2>Aqui será exibido um texto menor...</H2>
```

```
<H3>...aqui será exibido um texto menor...</H3>
```

```
<H4>...aqui será exibido um texto menor...</H4>
```

```
<H5>...aqui será exibido um texto menor...</H5>
```

```
<H6>...E aqui será exibido o menor texto possível com a tag "h6"
```

```
</H6>
```

```
</body>
```

```
</html>
```



# HTML5 – Listas Ordenadas

As listas são muito úteis, servem para organizar assuntos em tópicos, números, ou menus.

<OL> é a tag que define lista ordenada, com números.

<UL> é a tag que define lista ordenada, com marcadores.

Nos dois casos, LI" é a tag que define os itens da lista.

```
<OL>
```

```
<LI>ADS
```

```
<LI>Mecânica
```

```
<LI>Logística
```

```
</OL>
```

<DL> é a tag que define lista não ordenada (**OU LISTA DE DEFINIÇÕES**), ou seja, tópicos, e DT, é a tag que define os itens da lista (termos) e DD os sub tópicos (descrição).

```
<DL>
```

```
<DT>Linguagens</DT>
```

```
<DD>Java</DD>
```

```
<DD>C#></DD>
```

```
<DT>SGBD</DT>
```

```
<DD>Oracle</DD>
```

```
</DL>
```



# HTML5 – Exemplo: 3Usandolistaordenada.html

```
<html>
<head>
<title>ListaOrdenada</title>
<meta name="author" content="Denilce"/>
<meta name="keywords" content="aula,HTML"/>
</head>
<body>

<OL>
<LI>Introdução
<LI>Dedicatória
<LI>Capítulo 1
</OL>

<UL>
<LI>Introdução
<LI>Dedicatória
<LI>Capítulo1
</UL>

<DL>
<DT>Introdução</DT>
<DD>Subitem1 da Introdução</DD>
<DD>Subitem2 da Introdução</DD>
<DT>Dedicatória</DT>
<DD>Aqui vai a dedicatória</DD>
</DL>
</body>
</html>
```



# HTML5 – META TAG

Meta tags são linhas de código HTML que descrevem o conteúdo do site para os buscadores. Nelas é possível inserir por exemplo as **palavras chave** que ajudarão os usuários encontrar o site por exemplo. Também é possível declarar a **autoria** sobre o código fonte.

```
<head>  
<title> Meta tag - O que são e como usar?</title>  
<meta name="author" content="Denilce">  
<meta name="description" content="Meta Tags - O que são e como utilizá-las">  
<meta name="keywords" content="sites, web, meta tag, código, html">  
</head>
```

Read more: <http://www.linhadecodigo.com.br/artigo/3595/codigo-tag-o-que-e-meta-tag.aspx#ixzz4oHGhrLx5>



# HTML5 – META TAG CHARSET

HTML5 – Exemplo: 4Usandometacharset.html

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8"/>
    <title>Resolvendo problemas de acentuação em seu website</title>
  </head>
  <body>
    <p>Documentação!</p>
  </body>
</html>
```

## •UTF-8 ou ISO-8859-1?

ISO 8859-1 é o modelo padrão de codificação de caracteres do alfabeto latino. Possui no total 191 tipos de caracteres. Já o modelo de codificação utf-8, tem se tornado muito popular mundialmente, pois ele é capaz de codificar todos os tipos de caracteres.



# HTML5 – Ênfase, Negrito etc.

- `<em>` - para dar ênfase a um trecho de texto (itálico).
- `<strong>` ou `<b>` - ambos são negrito, a diferença está na semântica para auxiliar leitor “não visual” quando usar strong.
- `<i>` ou `<em>` - itálico, a diferença é que `<em>` representa ênfase importante de seus conteúdos, enquanto que a tag `<i>` representa o texto que é uma palavra estrangeira, pensamentos de um personagem ficcional, ou quando o texto se refere a definição de uma palavra em vez de representar seu significado semântico.
- `<small>` - Diminuir o tamanho do texto .
- `<u>` - sublinhar.



# HTML5 – Deletado, Sobreescrito, Subescrito

- `<del>` indica texto apagado (deletado)
- `<mark>` indica texto selecionado (marcado)
- `<sub>` indica texto subescrito
- `<sup>` indica texto superescrito
- `<abbr>` indica texto abreviado



# HTML5 – Citação

CITAÇÃO - Tem o propósito de fornecer informações sobre algo dito por alguém.

**<blockquote>** - fornece a delimitação de um parágrafo citado que pode conter várias linhas. Destaca o parágrafo, deslocando-o um pouco para a direita (definido como inadequado pela W3C em favorecimento ao uso de folhas de estilo)

**<q>** - ela apenas delimita uma pequena porção dentro do texto (descontinuado).



# HTML5 – Exemplo: 4Usandoenfasecitacao.html

```
<html>
<head>
<title>ênfase</title>
<meta name="author" content=Denilce/>
</head>
<body>
<p><em> Bill Gates,6 foi admitido na prestigiosa Universidade Harvard, mas abandonou os
cursos de Matemática e Direito no terceiro ano, para dedicar-se à Microsoft.</em></p>
<blockquote cite="http://www.fulanodetal.com.br">
    Texto dito pelo fulano de tal no site http://www.fulanodetal.com.br que fala sobre
    algumas coisas.
</blockquote>

<p><small>Gates nasceu em uma família de classe média de Seattle.</small></p>
<q> Informação obtida no google </q>

</body>
</html>
```



# HTML5 – LINKS

Os links, ou ligações, estabelecem um caminho traçado entre dois pontos.

- Da página de um site para outra página de outro site.
- Entre páginas de um mesmo site.
- Para determinados pontos de uma página (âncoras).



# HTML5 – LINKS

## TAG <A + ATRIBUTO HREF>

Atributo href utilizado para **informar o endereço** para ser redirecionado, arquivo de vídeo que será assistido, endereço da imagem que será aberta na tela e etc. Caso o link não possua um endereço, deve-se informar um “link falso” (usar sinal “#” ).

- Caminho Absoluto , Endereço completo do local.

```
<a href="http://www.fatecsorocaba.edu.br">Home da Fatec Sorocaba</a>
```

- Caminho Relativo , Endereço relativo do local que estava antes
- ```
<a href="./principal/institucional/institucional.html">Home da Empresa</a>
```

→ Sendo: ./ pasta atual                      ou                      ../ pasta superior



# HTML5 – LINKS

## ATRIBUTO TITLE

O atributo “title” funciona da mesma forma em praticamente todas as Tags do HTML. Ele representa o título daquela Tag, uma breve descrição do que aquele elemento faz ou representa.

Esse atributo faz com que o browser represente em forma de “balão” o título informado.

## ATRIBUTO TARGET

“Target” em inglês significa “Alvo”. Com esse atributo pode-se definir onde o link será aberto, se em uma nova aba, na mesma página ou até mesmo em uma nova janela.

O

| Valor   | Definição                                          |
|---------|----------------------------------------------------|
| _blank  | O link será aberto em uma nova aba do seu browser. |
| _self   | O link será aberto na mesma janela.                |
| _top    | O link alvo é aberto na mesma janela, sem frames   |
| _parent | O link alvo é aberto no frameset principal (Pai)   |

```
<p><a href="http://www.google.com" title="Site da Google " target="_blank">Link  
_blank</a></p>
```



# HTML – Exemplo: 5UsandoLinks.html

```
<html>
<head>
<title>Testando links</title>
<meta name="author" content="Denilce"/>
<meta name="keywords" content="aula,HTML"/>
</head>
<body bgcolor="#6495ED" text="#CC0000">

<h1 id="teste">Este elemento recebe o redirecionamento</h1>

<a href="oEstruturaBasica.html">link para primeira página</a>

<a href="1UsandoParagrafo.html"></a>

<p><a href="http://www.fatecsorocaba.edu.br">Home da Fatec Sorocaba</a><p>

<p><a href="#teste">Clique aqui para ser direcionado para o H1</a><p>

<p><a href="http://www.google.com" title="site da google" target="_blank">Link Google</a></p>

</body>
</html>
```



# HTML – Exemplo: 5UsandoLinks\_2.html

```
<html>
<head>
<title>Testando links 2</title>
<BODY text="#88oooo" link="#AAAAoo" vlink="ooDDoo" alink="#ooooAA">
```

Enviando links para outros lugares

```
<a href="mailto:denilce@gmail.com">Envie um e-mail para a professora</a> <BR>
```

```
<P><a href="ExemploArquivo.docx">Baixe o arquivo de exemplo .doc</a></p>
```

Os links podem apontar para seções de uma mesma página.

```
<a href="#secao1">Vá para a Seção 1</a>
```

```
<P>
```

Links -Plus

As cores no link podem ser mudadas com os comandos link(CordosLinks),alink (Cor de um link que foi pressionado) evlink(Cor de um link já visitado),podem ser colocados no body.

```
</P>
```

```
<p>Teste Links Visitados</p>
```

```
<H1>
```

```
<A href="oEstruturaBasica.html">Estrutura Básica</A><BR>
```

```
<A href="1Usandoparagrafo.html">Usando Parágrafo</A><BR>
```

```
</H1>
```

```
<a name="#secao1">A secao 1 é aqui</a>
```

```
</body>
```

```
</html>
```

Cliente e-mail padrão,  
melhor criar um  
formulário mesmo



# HTML5 – IMAGENS

- Para especificar uma imagem de fundo para o site.

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

Atributo *alt* - texto contido nela é exibido no lugar de uma imagem caso ela não seja exibida. Usado também pelos motores de busca.

**\*\*background-attachment** - especifica se uma imagem de plano de fundo é fixa ou rola com o resto da página: Scroll ou fixed.

**background-repeat** - estabelece se uma imagem de plano de fundo será repetida ou não.  
repeat, repeat-x, repeat-y, no-repeat

- Para inserir figuras, fotos e imagens usar elementos **<img>** e **<figure>**
- *Em resumo, enquanto a tag <img> é usada apenas para exibir uma imagem, a tag <figure> é usada para agrupar uma imagem com sua legenda, indicando que aquele conteúdo visual representa uma figura.*



# HTML5 – IMAGENS

Dentro destas tags podem ser especificar os atributos : "src", "alt", "title", "width" e "height". o HTML 5 INTRODUZIU FIGCAPTION PARA COLOCAR LEGENDA

Ex.:

```
<figure> 
</figure><figcaption>Legenda </figcaption>
```



# HTML5 - Exemplo: 6Usandoimagemfundo.html

```
<html>
  <head>
    <title>Imagem de fundo</title>
    <meta name="author" content="Denilce"/>
  </head>

  <body background="imagemfundo.jpg">
    <h1>Olhe a imagem de fundo</h1>
  </body>
</html>
```



# HTML5 - Exemplo: 6Usandoimagens.html

```
<!DOCTYPE html>  
<html>
```

```
<head>  
  <title>Teste com imagem</title>  
  <meta name="author" content="Denilce" />  
</head>
```

```
<body>  
  <figure>   
    <figcaption>Legenda da minha imagem</figcaption>  
  </figure>
```

```
</body>
```

```
</html>
```



# HTML5 - Exemplo:Base64

- Outra forma de carregar imagens é utilizar o **Base64**, que é um método para codificação de dados para transferência na Internet (codificação MIME para transferência de conteúdo). É utilizado para transmitir dados binários por meios de transmissão que lidam apenas com texto, como por exemplo para enviar arquivos anexos por e-mail. Não é uma utilização de uso geral.
- O método de **Base 64** é constituído por 64 caracteres, sendo eles [A-Za-z0-9, / e + que deram origem ao seu nome.

**Texto original: hello world**

**Texto convertido para Base64: aGVsbG8gd29ybGQK**

Ex

CSS

```
background-image: url(data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSJA  
AAADU1EQVR42mNkYPhfDwAChwGA60e6kgAAAABJRU5ErkJggg==);
```

HTML

```

```



# HTML5 - Exemplo:Base64

- Esse método tem vantagens e desvantagens para utilização nas páginas
- As imagens da página podem começar a carregar e renderizar mesmo sem a página ter completado o download.
- O motor interno do browser ter que decodificar a imagem.
- Imagens base64 usando o protocolo (data URI scheme) data: aumentam o tamanho da página html em muito, o que torna a página mais lenta de ser renderizada.
- A vantagem está em não ter que abrir outra conexão e fazer uma solicitação HTTP ao servidor para a imagem. Esse benefício é perdido muito rapidamente, portanto, há apenas uma vantagem: para um grande número de imagens individuais muito pequenas.



# Abrindo um parênteses

## FORMATO DE ARQUIVOS

O gimp possibilita que você exporte o arquivo em vários tipos de formatos, mas qual a diferença entre eles?

- JPG é o formato fechado de imagem que pode ser no padrão de cor RGB (tela) ou CMYK (impressão).
- PNG é o formato com fundo transparente padrão de cor RGB (tela).
- TIFF é o formato com fundo transparente padrão de cor CMYK (impressão).
- PDF formato de arquivo parcialmente fechado é possível o texto ser editado.
- XCF é o formato de arquivo do gimp, formato editável.
- PSD é o formato de arquivo do Photoshop, formato editável.

**WEBP** - formato desenvolvido pelo Google que oferece alta qualidade de imagem e tamanho de arquivo menor.

Ver mais em: <https://kinsta.com/pt/blog/tipos-arquivo-imagem/>



## Slide 43

---

**DDAOV1**

**\*\* o GIMP é um software gratuito parecido com o Photoshop**

DENILCE DE ALMEIDA OLIVEIRA VEL; 2025-02-20T01:15:35.014



# HTML5 – Alterando cores, fontes

Para inserir as cores, fontes e tamanhos pode se usar assim  
(***VEREMOS DEPOIS UTILIZANDO CSS***):

Para as cores de fundo e do texto

Ex.

```
<body bgcolor="#6495ED" text="#CC0000">
```

Para mudar somente a cor , fonte e tamanho do texto atual

```
<font color="#7FFF00" face="arial" size="12"> Texto
```

```
Exemplo </font>
```



# HTML5 - Exemplo: 7UsandoCoresFontes.html

```
<html>
```

```
<head>
```

```
  <title>Fontes, cores, tamanhos </title>
```

```
</head>
```

```
<body bgcolor="#6495ED" text="#CC0000">
```

```
  <font color="#7FFF00" face="arial" size="12"> Onde Face é a  
  letra,
```

```
    Size, é o tamanho da letra e color é a cor da letra</font>
```

```
  <p>Aqui está outro teste!!!!!!!!!!<p>
```

```
</body>
```

```
</html>
```



# HTML5 – Tabelas – Dados tabulares

**<table>** Usar exclusivamente para exibir dados tabulares, ou seja, informações que fazem sentido em formato de linhas e colunas, como tabelas de dados, calendários, grades de preços, estatísticas, etc.

Na maioria dos casos o ideal é utilizar o CSS com display:

```
.container {  
  display: table;  
  width: 100%;  
}
```



# HTML5 – Tabelas – Dados Tabulares

`<caption>` Caption em inglês é cabeçalho. Insere o cabeçalho de título da tabela.

`<th>` é um acrônimo de Table Header ou cabeçalho de tabela. Pode ser usado em substituição da TD. *A melhor aplicação desta tag é em títulos de linhas ou colunas.*

`<thead>` `<tbody>` `<tfoot>` Estas tags são usadas para agrupar linhas de uma tabela e defini-las. Por exemplo, `<thead>` define a primeira linha da tabela e `<tfoot>` define a última linha, `<tbody>` é o corpo da tabela propriamente dito.

*No HTML5 não é necessário fechar as tags: `<tbody>`, `<thead>`, `<tfoot>`, `<tr>`, `<th>`, `<td>`.*



# HTML5 – Tabelas – Dados Tabulares

`<tr>` Define as linhas das tabelas.

`<td>` - **A estrutura da tabela obedece a ordem linha e coluna**, é preciso adicionar a tag `<td>` em todas as linhas para criar as colunas desejadas e inserir o conteúdo a ser exibido.



# HTML5 - Exemplo: 8UsandoTabelas.html ->1ª parte

```
<html>
<head>
<title>Testando Tabelas - SOMENTE PARA TABULAÇÃO</title>
<metaname="author" content="Denilce"/>
<meta charset="UTF-8"/>
</head>
<body>
<table>
  <caption>Tabela de Adjetivos</caption>
  <thead> <tr> <th>Adjetivo<th>Superlativo Absoluto Sintético</tr>
  <!-- primeira linha, th é mesma coisa que td -->
  <tfoot> <tr> <th>Adjetivo <th>Superlativo Absoluto Sintético</tr>
  <!-- ultima linha -->

  <tbody>
    <tr><td>Ágil</td><td>Agílimo<br>Agilíssimo</td> </tr>

    <!-- TH dado do cabeçalho fica em negrito, TD dado da célula -->
    <br>
    <tr><td>Agradável</td><td>Agradabilíssimo</td></tr>
    <br>
```



# HTML5 - Exemplo: 8UsandoTabelas.html ->2ª parte

```
<tr><td>Capaz</td><td>Capacíssimo</td></tr>
<br>
<tr><td>Humilde</td><td>Humílimo</td></tr>
<br>
</tbody>
</table>
</body>
</html>
```



# HTML5 - Elementos Estruturais

O conjunto de tags do HTML é bem vasto mas é também limitado.

Quando não consegue achar a tag certa para aquele conteúdo pode ser utilizar alguns recursos, por exemplo as tags `<div>` e `<span>`. São tags sem nenhum significado especial mas que podem servir para agrupar um certo conteúdo, tanto um bloco da página quanto um pedaço de texto.

É possível estilizar esses divs e spans com CSS customizado. ***Por padrão, eles não têm estilo algum.***



# HTML5 - Elementos Estruturais

<DIV> – Vem de “divisão”, é possível criar um bloco, uma divisão, e dentro deste bloco ter uma imagem, links, textos e o que mais desejar. É possível aplicar o CSS nesse bloco, e tudo que estiver dentro dela vai receber aquelas regras de estilo.

As *divs* são importantes para agrupar elementos dentro ou fazer outros detalhes que não precisam de significado semântico, apenas **visual**.



# Exemplo 15UsandoDivs.html (1)

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title>Testando Divs CSS</title>
```

```
<meta name="author" content=Denilce/>
```

```
<link href="15UsandoDivs.css" rel="stylesheet">
```

```
</head>
```

```
<body>
```

```
<div class="todo">
```

```
    <div class="topo">
```

```
        Aqui é o topo
```

```
    </div>
```

```
    <div class="meio">
```

```
        <div class="esquerda">
```

```
            <div class="menu">
```

```
                <div class="itemMenu">Principal</div>
```

```
                <div class="itemMenu">Empresa</div>
```

```
            </div>
```

```
        </div>
```

**\*\*ATENÇÃO: ALTERAR NO CSS A COR DA CLASSE MEIO PARA FICAR IGUAL A DOS ITENS DO MIOLO. CRIAR MAIS UM ITEM DO MIOLO (PODE REDUZIR O TAMANHO DAS ANTERIORES SE DESEJAR)**



# Exemplo 15 Usando Divs.html (2)

```
<div class="miolo">  
    <div class="item_miolo">Seção 1 miolo</div>  
    <div class="item_miolo">Seção 2 miolo </div>  
</div>
```

```
<div class="direita">  
    <div class="item_direita">Seção 1 direita</div>  
    <div class="item_direita">Seção 2 direita</div>  
</div>
```

```
</div>
```

```
<div class="rodape">  
    <div class="rodape_direita">  
        Aqui está o rodapé  
    </div>
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```



# Exemplo 15UsandoDivs.css (1)

```
body {  
    margin:0px;  
    background-color: #cccccc;  
    font-family:"Trebuchet MS", Tahoma, Arial, Verdana;  
    font-size:12px;  
    color:#000;  
}  
  
.todo {  
    width: 750px;  
    border: 1px solid black;  
    margin:0 auto; /* topo e bottom = o direita e esq. automatica */  
    background-color: greenyellow;  
}  
  
.topo {  
    width: 746px;  
    height: 50px;  
    border: 2px solid red;  
}
```



# Exemplo 15UsandoDivs.css (2)

```
.meio {  
    width: 750px;  
    min-height: 400px;  
    background-color: deeppink;  
}
```

```
.rodape {  
    width: 746px;  
    height: 75px;  
    border: 2px solid brown;  
    background-color: blue;  
}
```

```
.esquerda {  
    float: left;  
    width: 150px;  
    min-height: 400px;  
    border: 2px solid brown;  
}
```



# Exemplo 15 Usando Divs.css (3)

```
.menu {  
    width: 150px;  
    height: 400px;  
}
```

```
.itemMenu {  
    width: 140px;  
    height: 21px;  
    background-color: lightyellow;  
    padding: 3px 0px 0px 10px;  
    border-bottom: 1px solid black;  
}
```

```
.miolo {  
    float: left;  
    width: 438px;  
    min-height: 400px; /* altura minima, importante se tem imagem no mesmo  
div. usando float por ex.*/  
    border: 2px solid blue;  
}
```



# Exemplo 15 Usando Divs.css (4)

```
.item_miolo {  
    width: 428px;  
    height: 150px;  
    text-align: center;  
    background-color: #cccccc;  
    margin: 2px 5px 4px 5px;  
}
```

```
.direita {  
    float: right;  
    width: 150px;  
    min-height: 400px;  
    border: 2px solid yellow;  
}
```

```
.item_direita {  
    width: 140px;  
    height: 100px;  
    text-align: center;  
    background-color: #cccccc;  
    margin: 4px 5px 4px 5px;  
}
```



# HTML5 - Tag Span

- Ao usar os seletores ID e CLASS em uma tag qualquer para estilizar, estará estilizando todo o conteúdo da tag, o que nem sempre é o que desejado.
- A tag **<span>**, permite que se use os seletores ID e CLASS em apenas um trecho da tag, ou seja, a tag *span* é uma tag *inline*, pois ela é usada dentro de outras tags, é uma espécie de elemento interno de outro elemento (de outra tag).



# Exemplo 15UsandoSpan.html

```
<html>
```

```
  <head>
```

```
    <title> Teste de Span </title>
```

```
    <link rel="stylesheet" href="15UsandoSpan.css">
```

```
  </head>
```

```
<body>
```

```
<p>Neste exemplo do <span class="principal">Curso de  
HTML</span>, vamos mostrar uma exemplo do <span  
class="destaque">SPAN</span> e sua importância.</p>
```

```
</body>
```

```
</html>
```



# Exemplo 15 Usando Span.css

```
p{  
    font-size: 30px;  
    font-family: "Arial";  
}
```

```
.principal {  
    font-weight: bold;  
    color: red;  
    font-size: 25px;  
}
```

```
.destaque {  
    font-style: italic;  
    font-weight: bold;  
    font-size: 45px;  
    color: blue;  
}
```



# HTML Semântico

- As tags <header>, <section> e <footer> - são tags do HTML5.
- Facilita a leitura do HTML.
- HTML semântico carrega significado independente da sua apresentação visual. Isso é importante quando o conteúdo será consumido por clientes não visuais. Neste caso, a estrutura semântica do HTML é essencial para ele entender as partes do conteúdo. Os robôs de busca como o Google também são leitores não visuais da sua página.



main - define o conteúdo principal dentro do <body>.Conteúdo principal aquele relacionado diretamente com o tópico central da página ou com a funcionalidade central da aplicação. Deve ser usada apenas uma vez na página.

- nav - representação de um bloco principal de links e botões de navegação - nem todo grupo de links ou botões deve ser tratado como um nav. Ex.: (home, sobre, contato), links na mesma página ou outra.



## HTML Semântico

- section - utilizado para representar uma seção genérica, geralmente com um cabeçalho próprio e o seu conteúdo;
- aside - a tag aside pode ser utilizada para representar uma seção de conteúdo secundário ou auxiliar a outro pedaço de maior importância. Por ex.: Citações, links de referência ou notas adicionais.
- header - referente ao cabeçalho de uma seção específica (ou da própria página) contendo títulos, introduções e outros elementos similares;

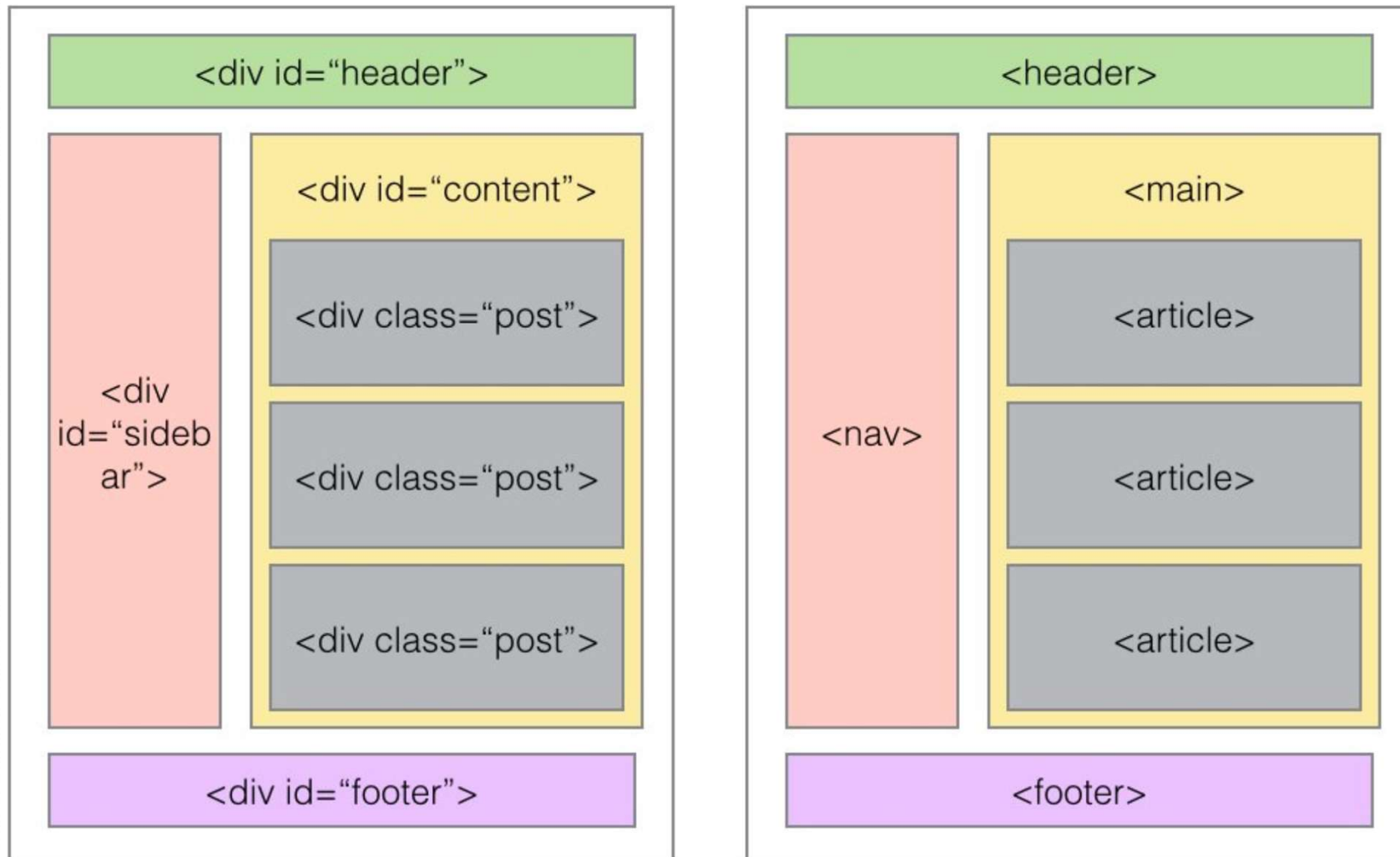


- footer - o rodapé referente a um bloco de conteúdo, pode ter um nav nele também;
  - article - normalmente identifica o conteúdo em si, como uma notícia de um portal, um post em blog ou um comentário em uma lista de comentários.
- \*\* Cada desenvolvedor deve usar quais desejar, porém lembrando que elas tem significado, por ex. não faz sentido colocar um Aside sem ser do lado.**



# HTML Semântico

## DIVs x Semântico



Fonte: <https://i.stack.imgur.com/w9ADa.jpg>



## HTML Semântico – Exemplos

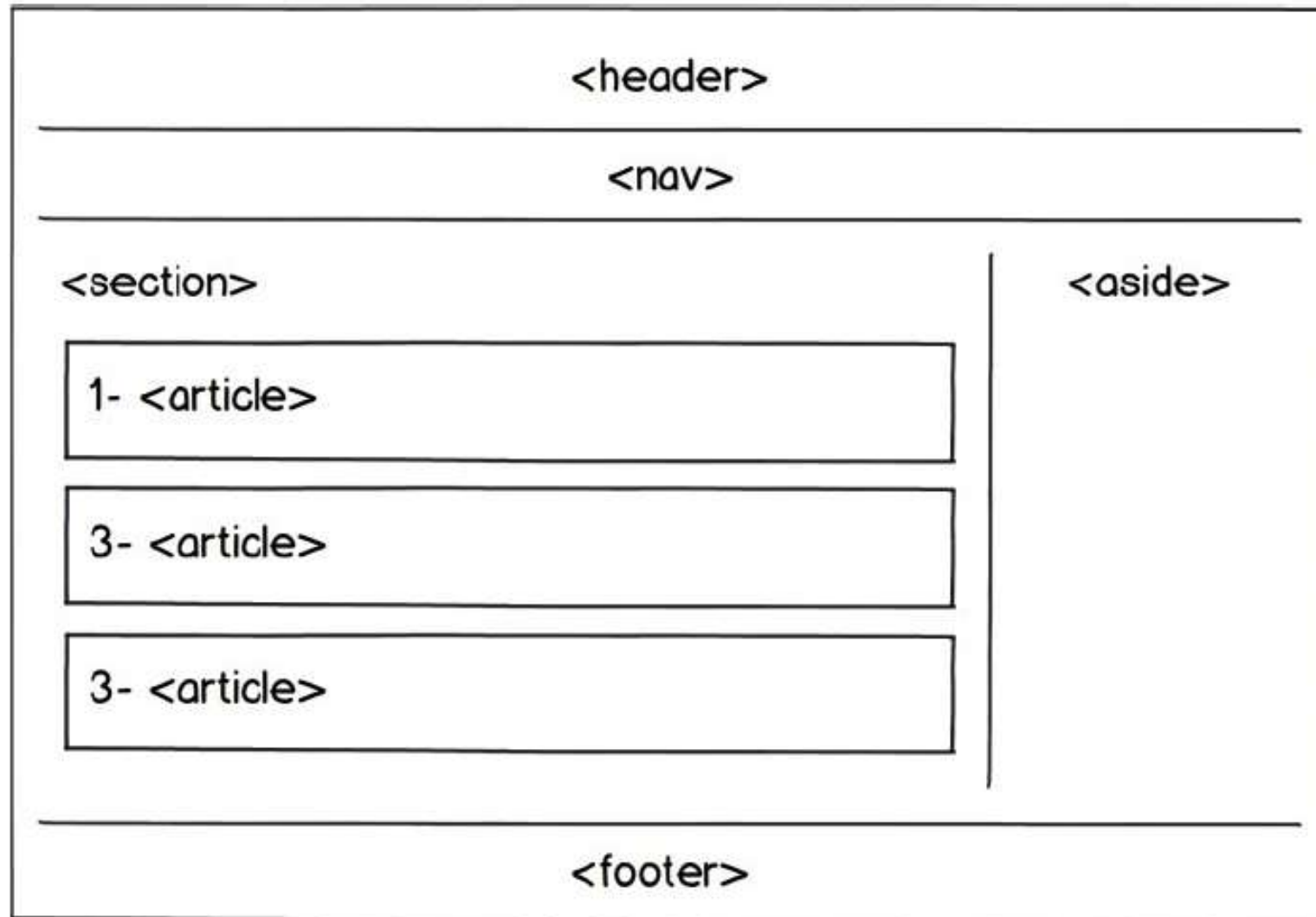
ver [16\\_3UsandoHtmlSemantico.html](#)

[16\\_4UsandoHtmlSemanticoComFlexBox.html](#)

**\*\*** com o flexbox, ele fica flexível, responsivo.



# HTML Semântico





# HTML Semântico

```
<header> O meu blog sobre HTML5 </header>

<nav> Página principal | Sobre o autor | Contato </nav>

<article>
  <header> Utilizando as tags <header> e <article>... </header>
  Publicado a 3 dias atrás

  ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore
  et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
  ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit
  esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt
  in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur
  adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad
  minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
  consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat

</article>
```

Adaptado de (Mazza,2012)



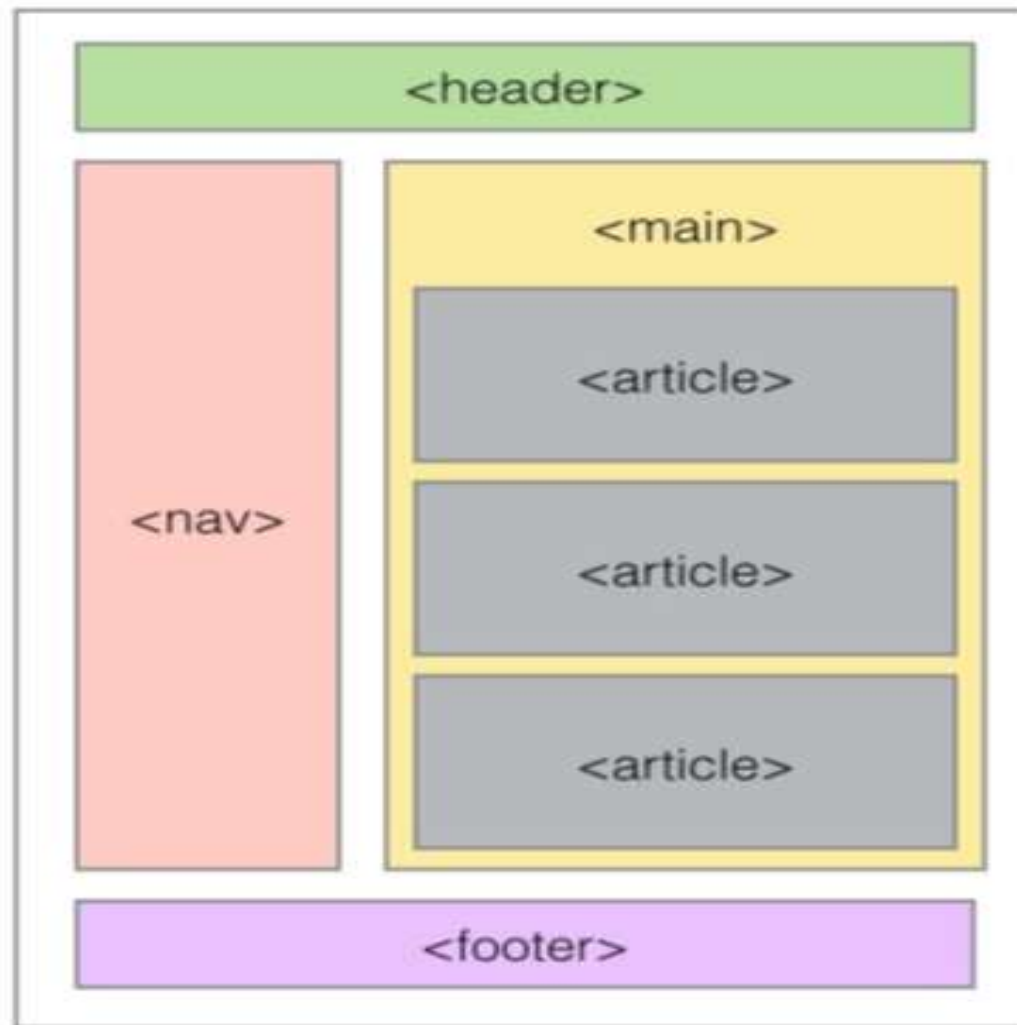
# HTML Semântico



Fonte: <https://danieldigital.com.br/blog/seo-tecnico-a-importancia-do-html5-semantico-para-melhorar-o-trafego-do-seu-site/>



# HTML Semântico



Fonte: <https://i.stack.imgur.com/w9ADa.jpg>

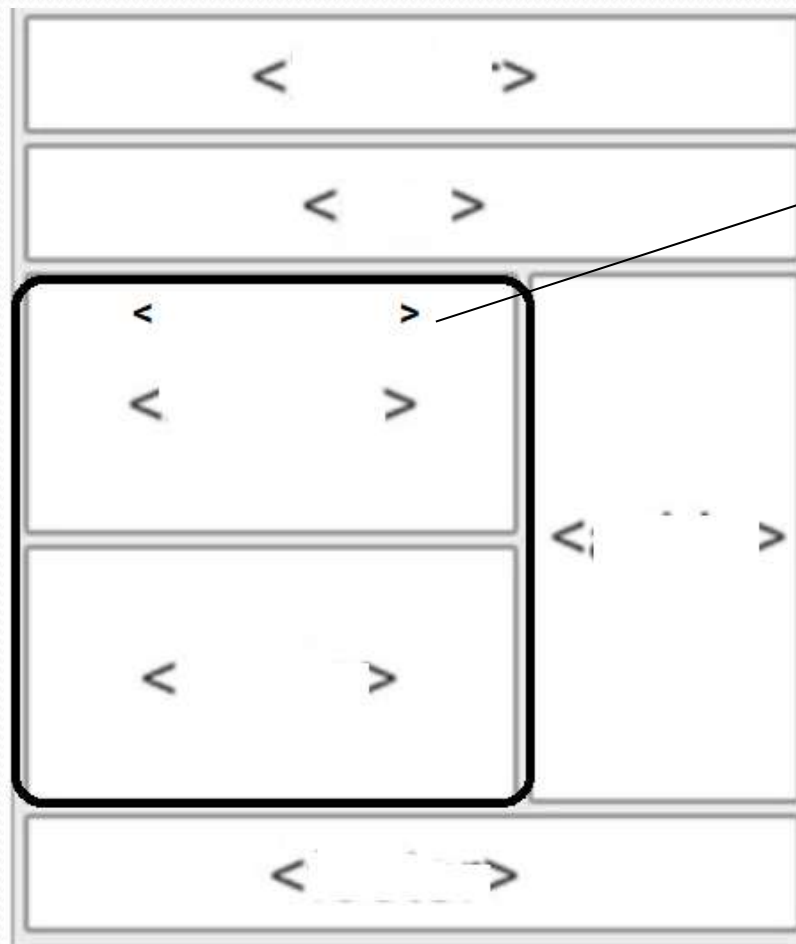


**EXERCÍCIO** – Quais seriam os elementos semânticos HTML5 dentro de cada box: footer, header, nav, article, section, aside?





EXERCÍCIO – Quais seriam os elementos do HTML5 semântico dentro de cada box (article, section, header, footer, nav) ?



Em negrito, é mais externo, tem 2 boxes dentro dele.

# HTML5 – ATIVIDADE3 (HTML)

1. Monte a sua página com as informações da Atividade2 (pesquisa), certifique-se de utilizar recursos vistos na aula: títulos, listas, cores, parágrafos, quebras de linha, imagens, links, **html semântico** etc.
2. Crie os documentos index.html e styles.css utilizando o editor de sua preferência.
3. Após finalizado e testado o documento, disponibilize-o no Github na pasta Atividade3.

**Seuusuario\PWEB\Atividade3**

*Obs.: uma sugestão para criar a pasta Atividade3 no GitHub, é utilizar a opção criar arquivo, ele não permite criar pasta vazia.*



# HTML5 – Formulários

Formulário- utilizado para entrada de dados, interação entre o usuário e o web site.

Definido pela tag **<form>** o que estiver dentro dela será compreendido como elementos de formulário.

`<form>`

`<!-- Aqui se informam os campos. -->`

`</form>`

Nunca será utilizada sozinha sempre terá outras, por exemplo:

`<label>`

`<input>`

`<textarea>`

`<button>` ou  $\rightarrow$  input do tipo button

`<select>`

`<fieldset>`

`<legend>`

...

*A grande vantagem do button é que pode ter Tags HTML dentro dele, enquanto que o input button é auto-fechado e o texto que ele mostra é definido no atributo value.*



# HTML5 – Formulários

O **<label>** é o título, rótulo do campo.

Os campos para entrada de dados mais utilizados são definidos pela tag **<input>**. Para definir o tipo é necessário informar o atributo (type).

```
<form>
```

```
<p><label>Nome:</label> <input type="text"></p>
```

```
<p><label>Email:</label> <input type="email"></p>
```

```
<p><label> Aluno?:</label><input  
type="checkbox"></p>
```

```
</form>
```



# HTML5 – Formulários (tipos de input)

Tipo	Finalidade	Exemplo
text	Inserção de texto	<input type="text"/>
checkbox	Caixa de seleção. Muito utilizado para que o usuário possa selecionar mais de uma opção	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
radio	Campo radial de opção. Muito utilizado para restringir o usuário a somente uma seleção de opção.	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
password	Campo para inserção de senhas. Idêntico ao "text", mas os dados informados são de asteriscos (*) e por uma série de asteriscos (*).	<input type="password"/>
email	Campo para inserção de email.	<input type="email"/>

Veja: <https://developer.mozilla.org/pt-BR/docs/Web/HTML/Element/Input>  
[http://www.w3schools.com/html/html\\_form\\_input\\_types.asp](http://www.w3schools.com/html/html_form_input_types.asp)

# HTML5 – Formulários (tipos de input)

submit	Cria um botão que é utilizado para enviar as informações do formulário. Necessita do atributo "value" para definir o nome do botão.	<input type="submit" value="Submit"/>
button	Cria um simples botão. Necessita do atributo "value" para definir o nome do botão.	<input type="button" value="Botão"/>
reset	Cria um botão, que ao ser clicado, elimina todos os dados informados nos campos do formulário. Necessita do atributo "value" para definir o nome do botão.	<input type="reset" value="Limpar"/>
file	Cria campo para seleção de arquivos.	<input type="file" value="Escolher arquivo"/> Nenhum arquivo selecionado



# HTML5 – Formulários

Tipo de entrada	Finalidade	Observações
tel	Usado para inserir um número de telefone.	O tel não impõe nenhuma sintaxe particular, portanto, se quiser determinar um formato específico, use <code>pattern</code> ou <code>setCustomValidity()</code> para validação adicional.
search	Solicita aos usuários que digitem o texto pelo qual desejam buscar.	A diferença entre <code>search</code> e <code>text</code> é basicamente estilística. O uso de uma entrada do tipo <code>search</code> pode fazer com que o campo de entrada fique com o mesmo estilo dos campos de pesquisa da plataforma usada.
url	Usado para inserir um URL único.	A função do <code>url</code> é inserir um URL único e <u>absoluto</u> (link em inglês), que representa uma grande extensão de valores.

Exemplo: Pattern para telefone = `"\([0-9]{2}\) [0-9]{4,6}-[0-9]{3,4}$"`

# HTML5 – Formulários

email	Usado para inserir um único endereço de e-mail ou uma lista de endereços de e-mail.	Se for especificado o atributo <code>multiple</code> , podem ser inseridos vários endereços de e-mail, separados por vírgulas.
datetime	Usado para inserir data e hora com o fuso horário em UTC.	
date	Usado para inserir uma data sem fuso horário.	
number	Para entrada numérica.	Os valores válidos são <u>números de ponto flutuantes</u> ( <a href="#">link em inglês</a> ).
range	Para entrada numérica, mas ao contrário de <code>number</code> , o valor real não é importante.	A implementação do controle de intervalo é um controle deslizante na maioria dos navegadores que têm suporte.
color	Usado para escolha de Cor	Especificar cor inicial

Depen  
de do  
step  
(passo)



# HTML5 – Formulários (novos tipos input)

## EXPRESSÕES REGULARES

Expressão Regular é uma linguagem de busca de padrões, uma linguagem onde informa-se o padrão (*pattern*) do texto que deseja-se encontrar, passa-se o texto alvo (*target*) e pede-se para uma *Regex engine* (motor) fazer essa busca.

<https://tableless.com.br/o-basico-sobre-expressoes-regulares/>

<http://wbruno.com.br/html/validando-formularios- apenas-com-html5>

<https://rubular.com/> → permite testar a expressão

<https://regex101.com/> → permite testar a expressão



# HTML5 – Formulários (atributos)

Para utilizar o identificador FORM, necessário primeiro conhecer os atributos:

**ACTION:** neste atributo informar a URL para a qual o conteúdo do formulário será submetido, se este atributo não for especificado, a URL corrente será usada.

**METHOD:** especificar o método de submissão usado.

**GET:** é o método default, conteúdo do formulário é anexado ao final da URL especificada no atributo ACTION (é enviada como string anexada a URL).

**POST:** conteúdo do formulário seja codificado exatamente como no método GET, é enviado ao servidor em um bloco de dados endereçado à URL especificada no atributo (mais recomendado - encapsulado junto ao corpo da requisição HTTP e não pode ser vista).



# HTML5 – Formulários (atributos)

O atributo **enctype** em um formulário HTML (<form>) define como os dados do formulário devem ser codificados antes de serem enviados ao servidor. **Ele é usado principalmente quando o método de envio (method) é POST.**

Principais valores do enctype:

**application/x-www-form-urlencoded (Padrão)**

Os dados são codificados como pares chave=valor, separados por &. Útil para formulários simples (texto, números, etc.)

```
<form action="processa.php" method="POST" enctype="application/x-www-form-urlencoded">
```

```
<input type="text" name="nome" value="João">
```

```
<input type="number" name="idade" value="25">
```

```
<input type="submit" value="Enviar"> </form>
```



# HTML5 – Formulários (atributos)

## **multipart/form-data**

Utilizado para o envio/upload de arquivos.

```
<form action="upload.php" method="POST" enctype="multipart/form-data">  
  <input type="text" name="nome" value="João">  
  <input type="file" name="arquivo">  
  <input type="submit" value="Enviar"> </form>
```



# HTML5 – Formulários (atributos)

## **text/plain**

Os dados são enviados como texto simples, sem codificação especial. Ele não é muito utilizado, pois pode causar problemas na interpretação dos dados e não é seguro para transmissões na web.

```
<form action="mailto:exemplo@email.com" method="POST" enctype="text/plain">  
<label>Nome:</label>  
<input type="text" name="nome" value="João">  
<label>Mensagem:</label>  
  <textarea name="mensagem">Olá, gostaria de mais informações.</textarea>  
<input type="submit" value="Enviar"> </form>
```



# HTML5 - Exemplo: 9UsandoFormularios.html

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title>Criando Formulários</title>
```

```
</head>
```

```
<body>
```

```
<FORM method="post">
```

```
<p>
```

```
    <label>Nome:</label><br>
```

```
    <input type="text" name="nome" required size="50" maxlength="50" pattern="[a-z\s]+$" autofocus>
```

```
  </p>
```

```
  <p>
```

```
    <label>Email:</label><br>
```

```
    <input type="email" name="email" required size="50" maxlength="50" pattern="[a-zo-9._%+-]+@[a-zo-9.-]+\.[a-z]{2,4}$">
```

```
  </p>
```

```
  <p>
```

```
    <label>Assunto:</label><br>
```

```
    <input type="text" name="assunto" size="50" maxlength="50" placeholder="vazio">
```

```
  </p>
```

```
  <p>
```

```
    <label>Mensagem:</label><br>
```

```
    <textarea name="mensagem" cols="100" rows="8"></textarea>
```

```
  </p>
```

```
  <p><input type="checkbox"> Aceita receber nossas novidades</p>
```

```
  <p><input type="submit" value="Enviar"></p>
```

```
<INPUT TYPE="submit" VALUE="Gravar">
```

```
</form>
```

```
</body>
```

```
</html>
```

**\*\*ATENÇÃO: INCLUIR INPUT DE SENHA\*\***

DDAOV1

**\*\* Ver <http://html5pattern.com/>**

**<https://www.devmedia.com.br/iniciando-expressoes-regulares/6557>**

**Quando o valor do MAXLENGTH for maior que SIZE, é criado automaticamente recurso de rolamento de tela**



**DDAOV1**

Por exemplo: O padrão `pattern="[a-z\s]+$"` em HTML significa que o valor do campo de entrada deve ser composto por:

Letras minúsculas de a a z (a-z)

Espaços em branco (\s)

Um ou mais caracteres (+)

O final da string (\$)

O padrão `pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$"` é uma expressão regular usada para validar endereços de e-mail em HTML. Vamos decompor a expressão:

`[a-z0-9._%+-]+`: Essa parte corresponde a um ou mais caracteres entre os seguintes: letras minúsculas (a-z), números (0-9), underline (\_), ponto (.), símbolo de porcentagem (%), sinal de adição (+) ou hífen (-). Isso corresponde ao nome de usuário do endereço de e-mail.

`@`: Corresponde ao símbolo "arroba" (@), que separa o nome de usuário do nome do domínio em um endereço de e-mail.

`[a-z0-9.-]+`: Essa parte corresponde a um ou mais caracteres entre os seguintes: letras minúsculas (a-z), números (0-9), underline (\_), ou hífen (-). Isso corresponde ao nome do domínio do endereço de e-mail.

`\.[a-z]{2,4}`: Essa parte corresponde a um ponto literal (.) seguido por duas a quatro letras minúsculas (a-z). Isso corresponde ao domínio de topo (TLD) do endereço de e-mail, como .com, .org ou .net.

# HTML5 - Exemplo: 10UsandoFormularios\_2.html (1)

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title>Criando Formulários 2</title>
```

```
</head>
```

```
<body>
```

```
<FORM method="post">
```

```
Nome      :<input type="text" name="nome" size="50"
maxlength="50" pattern="[a-z\s]+$" autofocus>
```

```
<br><h4> Escolha a marca </h4>
```

```
<select name="marca">
```

```
<option value="vw">Volkswagem</option>
```

```
<option value="Ford">Ford</option>
```

```
<option value="Renault">Renault</option>
```

```
<option value="Chevrolet"> Chevrolet </option>
```

```
</select>
```

```
<br><h4> Escolha o modelo </h4>
```

```
<select name="modelo">
```

```
<option value="gol">Gol</option>
```

```
<option value="EcoSport">EcoSport</option>
```

```
<option value="Duster">Duster</option>
```

**\*\*ATENÇÃO: INCLUIR UM COMBOBOX  
PARA ESCOLHA DE POTÊNCIA (DEPOIS  
DO MODELO)**

<https://tableless.com.br/o-basico-sobre-expressoes-regulares/>

Poderia deixar um  
item selecionado no  
Option

```
<option value="nada"
selected>Selecione um
item</option>
```



## HTML5 - Exemplo: 10UsandoFormularios\_2.html (2)

```
</select>
```

```
<br><br><h4> Escolha o Tipo de Pagamento </h4>
```

```
<INPUT TYPE="RADIO" NAME="Tipo" VALUE="op1"> Consorcio
```

```
<INPUT TYPE="RADIO" NAME="Tipo" VALUE="op2"> Financiamento Banco
```

```
<INPUT TYPE="RADIO" NAME="Tipo" VALUE="op3"> Vista
```

```
<br><br><h4> Seguradora </h4>
```

```
<INPUT TYPE="checkbox" NAME="PortoSeguro" VALUE="op1">Porto Seguro
```

```
<INPUT TYPE="checkbox" NAME="Itau" VALUE="op2">Itau
```

```
<INPUT TYPE="checkbox" NAME="HDI" VALUE="op3">HDI
```

```
<br><br><h1>Por favor deixe seu comentário</h1>
```

```
<form>
```

```
<textarea name="obs" rows="10" cols="100">
```

```
</textarea>
```

```
</form>
```

```
<br><br><INPUT TYPE="submit" VALUE="Gravar">
```

```
<INPUT TYPE="reset" VALUE="Limpar">
```

```
</form>
```

```
</body>
```

```
</html>
```



## HTML5 - Exemplo: 10UsandoFormularios\_2.html (3)

```
</select>
```

```
<br><br><h4> Escolha o Tipo de Pagamento </h4>
```

```
<INPUT TYPE="RADIO" NAME="Consortio" VALUE="op1"> Consorcio
```

```
<INPUT TYPE="RADIO" NAME="Financiamento Banco" VALUE="op2">
```

```
Financiamento Banco
```

```
<INPUT TYPE="RADIO" NAME="Vista" VALUE="op3"> Vista
```

```
<br><br><h4> Seguradora </h4>
```

```
<INPUT TYPE="checkbox" NAME="PortoSeguro" VALUE="op1">Porto Seguro
```

```
<INPUT TYPE="checkbox" NAME="Itau" VALUE="op2">Itau
```

```
<INPUT TYPE="checkbox" NAME="HDI" VALUE="op3">HDI
```

```
<br><br><h1>Por favor deixe seu comentário</h1>
```

```
<form>
```

```
<textarea name="obs" rows="10" cols="100">
```

```
</textarea>
```

```
</form>
```

```
<br><br><INPUT TYPE="submit" VALUE="Gravar">
```

```
<INPUT TYPE="reset" VALUE="Limpar">
```

```
</form>
```

```
</body>
```

```
</html>
```



## HTML5 - Form – Fieldset e Legend

**FIELDSET** - serve para agruparmos um conjunto de campos de um formulário, que estão interligados.

**LEGEND** - legenda do campo



# HTML5 - Exemplo: 10UsandoFormularios\_3.html

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
  <title>Criando Formulários 3</title>
```

```
</head>
```

```
<body>
```

```
  <FORM method="post">
```

```
    <fieldset>
```

```
      <legend>Login</legend>
```

```
      <label for="email">E-mail: </label>
```

```
      <input type="text" name="email"> <br />
```

```
      <label for="senha">Senha: </label>
```

```
      <input type="password" name="senha">
```

```
    </fieldset>
```

```
  </form>
```

```
</body>
```

```
</html>
```



# HTML5 - Exemplo: 10UsandoFormularios\_4.html

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
  <title>Criando Formulários 4</title>
```

```
</head>
```

```
<body>
```

```
  <FORM method="post">
```

```
    <input type="checkbox" name="opcoes" value="html" /> HTML<br />
```

```
    <input type="checkbox" name="opcoes" value="css" /> CSS<br />
```

```
    <input type="checkbox" name="opcoes" value="javascript" /> JavaScript
```

```
    <br />
```

```
    <input type="checkbox" name="opcoes" value="php" /> PHP
```

```
  </FORM>
```

```
</body>
```

```
</html>
```



## HTML5 – Input – Diferença entre name e id nos inputs

A principal diferença é a seguinte: id é usado quando o nome é usado uma única vez, e name quando pode ser usado mais de uma vez.



# Storage API do HTML5 - 1

O HTML5 nos provê algumas formas de armazenamento de dados no lado do cliente:

- sessionStorage
- localStorage
- SQL Database (IndexedDB)
- Cookies.

Cada método tem suas próprias vantagens e limitações.



## Storage API do HTML5 - 2

Os sistemas de `localStorage` e `sessionStorage`, fazem parte da API Storage. Seu tipo de armazenamento é de chave: valor, sendo a chave sempre uma `STRING`. O objeto desta API possui 4 métodos:

- `getItem(chave)` //obtém um valor armazenado no Storage
- `setItem(chave,valor)` //guarda um valor no Storage
- `removeItem(chave)` //exclui um valor do Storage
- `clear()` //limpa o Storage

**\*\*** No quesito suporte dos navegadores, há diferenças para a API Storage e Web SQL, é necessário sempre estar atento.



A **sessionStorage** ficará disponível apenas para a janela que criou o dado até que seja fechada (por exemplo, quando a sessão for encerrada).

A **localStorage**, por outro lado, não depende da sessão. Se setar um dado no local storage esse ficará automaticamente disponível para qualquer janela. Além disso, permanecerá disponível até que seja explicitamente deletada pelo programador do site ou pelo o usuário.



# HTML5 - Exemplo: 10\_1UsandoStorage.html

**\*\*ATENÇÃO: INCLUIR MAIS UMA VARIÁVEL  
SESSIONSTORAGE → NUMERORA**

```
<body>
<script>
  // Verifica se o navegador suporta local storage
  if (typeof (Storage) !== "undefined") {

    // Armazena um valor na variavel nome
    localStorage.setItem("nome","Ana Maria");
    // OU localStorage.nome = "Ana Maria";

    // Le o valor da variavel nome
    alert(localStorage.getItem("nome"));    }
    //OU alert(localStorage.nome);
  else {
    alert("Seu navegador nao suporta armazenamento local!");
  }
</script>
</body>
```



- Um banco de dados transacional baseado em chaves e valores, com modelo de dados flexível.

```
var request = indexedDB.open('NomeDoBancoDeDados', 1);
```

```
request.onupgradeneeded = function(event) {  
    var db = event.target.result;  
    var objectStore = db.createObjectStore('NomeDaStore', { keyPath:  
'id' });  
};
```

```
request.onsuccess = function(event) {  
    var db = event.target.result;  
    var transaction = db.transaction(['NomeDaStore'], 'readwrite');  
    var objectStore = transaction.objectStore('NomeDaStore');  
    objectStore.add({ id: 1, dados: 'valor' });  
};
```

**DDAOV1**

Não precisa instalar nada na maquina, é uma API do navegador.

DENILCE DE ALMEIDA OLIVEIRA VEL; 2024-09-02T12:37:11.386



# Storage API do HTML5 - 5

- Cookies - Armazena pequenas quantidades de dados que são enviadas entre o navegador e o servidor em cada solicitação HTTP. Possuem um limite de tamanho (geralmente 4 KB por cookie).

```
document.cookie = 'chave=valor; expires=DataDeExpiracao; path=/';  
var valor = document.cookie;
```



Olha que interessante:

[https://www.colab55.com/collections/tendencias-de-design-em-2023?utm\\_source=google&utm\\_medium=cpc&utm\\_campaign=pareto.in.pmax.br&gclid=CjoKCQIAjbagBhD3ARIsANRrqEvVzxrSP1l24oXMcq1atROsZ\\_XY2SawNtcnSTdUzAE6vuFVxlUkCxwaAps8EALw\\_wcB](https://www.colab55.com/collections/tendencias-de-design-em-2023?utm_source=google&utm_medium=cpc&utm_campaign=pareto.in.pmax.br&gclid=CjoKCQIAjbagBhD3ARIsANRrqEvVzxrSP1l24oXMcq1atROsZ_XY2SawNtcnSTdUzAE6vuFVxlUkCxwaAps8EALw_wcB)

<https://webnus.net/pt/top-5-website-design-trends-for-2024/#:~:text=Em%202024%2C%20espere%20over%20tons,que%20mant%C3%A9m%20os%20visitantes%20envolvidos.>

<https://elementor.com/blog/pt-br/tendencias-e-melhores-praticas-de-design-web-para-2025/>



# Dicas sobre Front-End

*Recursos gráficos para todos - Vetores, PSD (Photoshop e Gimp), Background e Fotos grátis. Consegue baixar em formato arquivo aberto ou de software de edição de imagem (vetor)*

*Formato AI (Adobe Illustrator, Corel e Inkscape)*

- <https://br.freepik.com/>



# Dicas sobre Front-End

*Existem vários **formatos** sugeridos para imagens nas redes sociais*

*Exemplos: site [https://www.canva.com/pt\\_br/](https://www.canva.com/pt_br/)*



# DICAS DE COMO CRIAR SEU SITE NO GITHUB PAGES

- <https://docs.github.com/pt/pages/getting-started-with-github-pages/creating-a-github-pages-site>
- <https://www.youtube.com/watch?v=WqIaqUqQXdc>



# Referências (1)

- Alura. Apostilas Cursos Gratuitas Disponível em: <<https://www.alura.com.br/apostilas>>. Acesso em: 02.FEV.2024.
- CodeAcademy. Cursos Gratuitos. Disponível em: <<https://www.codecademy.com/pt>> Acesso em: 02.FEV.2024.
- DUCKETT, Jon. *HTML & CSS: Design and Build Websites*. John Wiley & Sons, 2014.
- LGPD. Disponível em: <https://computerworld.com.br/2018/09/19/lgpd-10-pontos-para-entender-a-nova-lei-de-protecao-de-dados-no-brasil/> Acesso: 24 fev 2019.
- GOOGLE. Disponível em: <<https://www.google.com/webmasters>> Acesso em: 02.FEV.2024.
- INPUT. O elemento Input. Disponível em: <<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input#attr-pattern>> Acesso em: 11.AGO.2018.



# Referências (2)

- JOBSTRAIBIZER, Flávia. Criação de Sites com Css - Desenvolva páginas Web mais leves e dinâmicas em menos tempo. 1ª Ed. São Paulo: Digerati Books.,2009.
- REGEXP. RegExp. Disponível em: <[https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/RegExp](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp)> Acesso:11.AGO.2021
- STORAGE. Utilizando a Storage API do HTML5. Disponível em: <http://imasters.com.br/artigo/20315/desenvolvimento/utilizando-a-storage-api-do-html5/?trace=1519021197&source=single> Acesso em: 13.MAR.2016.
- ZEMEL, Tércio. Web Design Responsivo: Páginas adaptáveis para todos os dispositivos. 1ª Ed. São Paulo: Casa do Código, 2012.
- W3C. Site da W3C oficial. Disponível em: <<http://www.w3.org>> Acesso em: 02.FEV.2024