

### Estrutura de arquivos

É sempre importante estabelecer um padrão para a organização dos arquivos de um site. Não é obrigatório, e não existe um determinado padrão para ser seguido. Como um site é um conjunto de várias páginas, recomenda-se que todas as páginas estejam dentro de uma pasta só.

Como um livro, é necessário que exista uma capa (*página inicial de um site*) ou seja o índice: que indica ao visitante quais são as outras páginas que fazem parte do projeto deste website.

Esse índice é denominado de **index.html**.

> Documentos > web > html				Pesquisar ...
Nome	Data de modificação	Tipo	Tamanho	
css	13/04/2020 13:40	Pasta de arquivos		
imagens	08/04/2020 19:05	Pasta de arquivos		
js	13/04/2020 13:40	Pasta de arquivos		
aula1	23/03/2020 08:44	Arquivo HTML	1 KB	
conteudoImagem	06/04/2020 20:06	Arquivo HTML	1 KB	
exercicio4_formatacao	03/04/2020 13:33	Arquivo HTML	1 KB	
exercicio5_formatacao	30/03/2020 08:20	Arquivo HTML	1 KB	
exercicio5_formatacao	26/03/2020 13:36	Arquivo PNG	200 KB	
formatacao	03/04/2020 14:39	Arquivo HTML	7 KB	
imagens_exemplo	30/03/2020 09:21	Arquivo HTML	1 KB	
links_exemplo	02/04/2020 19:56	Arquivo HTML	3 KB	
lista	09/04/2020 08:20	Arquivo HTML	2 KB	
tabela	09/04/2020 22:14	Arquivo HTML	2 KB	
tabela_Imagens	09/04/2020 22:14	Arquivo HTML	1 KB	

Juntamente com a pasta do seu site, onde estão as páginas, é importante criar as subpastas, como: imagens, downloads, js, css, etc.

> Documentos > web > html				Pesquisar ...
Nome	Data de modificação	Tipo	Tamanho	
css	13/04/2020 13:40	Pasta de arquivos		
imagens	08/04/2020 19:05	Pasta de arquivos		
js	13/04/2020 13:40	Pasta de arquivos		
aula1	23/03/2020 08:44	Arquivo HTML	1 KB	
conteudoImagem	06/04/2020 20:06	Arquivo HTML	1 KB	
exercicio4_formatacao	03/04/2020 13:33	Arquivo HTML	1 KB	
exercicio5_formatacao	30/03/2020 08:20	Arquivo HTML	1 KB	
exercicio5_formatacao	26/03/2020 13:36	Arquivo PNG	200 KB	
formatacao	03/04/2020 14:39	Arquivo HTML	7 KB	
imagens_exemplo	30/03/2020 09:21	Arquivo HTML	1 KB	
links_exemplo	02/04/2020 19:56	Arquivo HTML	3 KB	
lista	09/04/2020 08:20	Arquivo HTML	2 KB	
tabela	09/04/2020 22:14	Arquivo HTML	2 KB	
tabela_Imagens	09/04/2020 22:14	Arquivo HTML	1 KB	

## Curso Desenvolvimento WEB

Observe o exemplo acima: Na pasta **documentos**, existe a subpasta **web**. Dentro da subpasta **web**, existe a pasta **html**. Dentro da pasta **html**, estão todas as páginas do projeto do site.

Dentro da pasta **html**, tem as subpastas: **css**, **imagens** e **js**. No decorrer no desenvolvimento do site, poderão ser criadas outras subpastas, dependendo do conteúdo que será armazenado.

O DOCTYPE não é uma tag HTML, mas uma instrução especial. Ela indica para o navegador qual versão do HTML deve ser utilizada para renderizar a página. Utilizaremos `<!DOCTYPE html>`, que indica para o navegador a utilização da versão mais recente do HTML - a versão 5, atualmente.

Usaremos: `<!DOCTYPE html>`

É importante também configurarmos qual codificação utilizar em nossas páginas através de **charset na tag <meta>**. No HTML5 é utilizado o valor **UTF-8**, também chamado de **Unicode**.

**Exemplo:** `<meta charset="utf-8">`

A tag **<body>** é onde colocamos o conteúdo de nossa página, e que será exibido pelo navegador.

Atributos de cor da tag body:

`<body bgcolor="#ffff00">`

`<body bgcolor="rgb(30,10,100)">`

`<body bgcolor="black">`

Atributos de imagem da tag body:

`<body background="imagens/fundo3.jpg">`

`<body background="https://santaportal.com.br/sistema/Arquivos/Fotos/248201884036117.jpg">`

### HTML – Tags básicas

Tag	Descrição
<code>&lt;html&gt;</code>	Define um documento HTML
<code>&lt;body&gt;</code>	Define o corpo de um documento
<code>&lt;h1&gt; ... &lt;h6&gt;</code>	Define cabeçalhos 1 a 6
<code>&lt;p&gt;</code>	Define um parágrafo
<code>&lt;br&gt;</code>	Insere uma quebra de linha
<code>&lt;hr&gt;</code>	Define uma linha horizontal
<code>&lt;!-- --&gt;</code>	Define um comentário

### Tags de formatação

Tag	Descrição
<code>&lt;b&gt;</code>	Formata um texto em negrito
<code>&lt;big&gt;</code>	Formata um texto com fonte maior
<code>&lt;em&gt;</code>	Formata um texto com ênfase
<code>&lt;i&gt;</code>	Formata um texto em itálico
<code>&lt;small&gt;</code>	Formata um texto com fonte pequena
<code>&lt;strong&gt;</code>	Formata um texto em destaque
<code>&lt;sub&gt;</code>	Formata um texto subscrito
<code>&lt;sup&gt;</code>	Formata um texto sobrescrito
<code>&lt;ins&gt;</code>	Formata um texto sublinhado

<del>	Formata um texto anulado
-------	--------------------------

### Alguns caracteres especiais em HTML

Saída	Descrição	Nome
	Espaço sem quebra	&nbsp;
<	Menor que	&lt;
>	Maior que	&gt;
&	E comercial	&amp;
"	Aspas	&quot;
'	Apóstrofo	&apos; (não funciona no IE)
©	Copyright	&copy;

### HyperLink

Permite a ligação da página atual com outros documentos, que podem ser páginas, arquivos. É através da **tag** “a href” que é feita a ligação entre as páginas de sua estrutura de páginas web.

**Carrega um link externo (site externo).**

```
<a href= "https://www.sp.senac.br/bebedouro">Senac Bebedouro/SP</a>
```

Link Interno - Carregando uma página dentro de sua estrutura de páginas de seu website

```
<p><a href= "aula1.html">Aula1 em HTML</a>
```

**Destino → indica se o link abrirá na mesma aba; em nova aba por exemplo**

**Exemplo:**

```
<a href= "https://www.sp.senac.br/bebedouro" target="_self">Senac Bebedouro/SP</a>
```

**\_self →** é o padrão, ou seja: senão informarmos será aberto na mesma aba

```
<a href= "https://www.sp.senac.br/bebedouro" target="_blank">Senac Bebedouro/SP</a>
```

**\_ blank →** será aberto em outra aba

**Link com âncora - Abre o conteúdo dentro da mesma página**

**Abaixo é definido o link para a âncora (se for não utilizar o símbolo hashtag, o navegador vai tentar abrir uma página salva dentro da sua estrutura de página):**

```
<a href="estruturaHTM.html">Estrutura básica do HTML</a><br>
```

```
<a href="#dicas">Dicas de HTML</a><br>
```

```
<hr>
```

```
<a name= "estruturaHTML">Estrutura básica do HTML</a>
```

```
<p>A <strong>Linguagem de Marcação HTML</strong>, tem sua estrutura básica na versão <em>5</em>, conforme abaixo:</p>
```

```

```

```
<hr>
```

```
<a name= "dicas">Dicas de HTML</a>
```

Existem vários sites onde você pode estudar e aprender a utilizar outras tags, na medida que você ir desenvolvendo suas páginas.<br>

```
<pre>No link: https://www.w3schools.com/html/</pre>
<hr>
```

### Link para e-mail:

```
<a href="mailto:edson davidp@yahoo.com.br">Contato</a>
```

Neste exemplo de link para e-mail, o usuário deverá ter configurado um cliente de e-mail como o Aplicativo **Outlook** da família **Office**, para que os e-mails sejam encaminhados para o mesmo.

### Link com imagens

```
<a href="formatacao.html"></a> <br>
```

No exemplo acima ao passar o mouse sobre a imagem, será exibida a mãozinha. Ao clicar sobre a mesma, será aberta a página que está entre aspas.

### Link com arquivo

```
<a href="html.zip">Páginas de exemplo em html</a><br>
```

No exemplo acima quando passar o mouse sobre o texto Páginas de exemplo em html, será baixado para a máquina do usuário, o arquivo htmp.zip.

### Link com arquivo texto

```
<a href="receita.txt">Receita</a>
```

No exemplo acima quando passar o mouse sobre o texto Receitas, será exibido na tela do navegador, o conteúdo do arquivo receita.txt.

### Imagens

```

```

Exibe a imagem com texto alternativo (texto alternativo é quando a imagem não pode ser exibida), com a largura (width) de 200 pixels e altura (height) de 150 pixels.

Você também pode abrir uma imagem diretamente de uma url. **Exemplo:**

```

```

### Listas ordenadas e não ordenadas

Uma lista ordenada é uma lista de itens. Estes itens podem estar em ordem de números, algarismos romanos maiúsculos ou minúsculos e letras maiúsculas ou minúsculas. Lista não ordenada são em formato de **bullets** (*símbolos*). **Exemplo:**

```
<h3>Lista Ordenada</h3>
<ol>
  <li>Café</li>
  <li>Leite</li>
  <li>Bolo</li>
</ol>
```

A lista será exibida assim:

## Lista Ordenada

1. Café
2. Leite
3. Bolo

Para exibir a lista com o tipo em algarismo romano:

```
<h3>Lista Ordenada</h3>  
<ol type="I">  
  <li>Café</li>  
  <li>Leite</li>  
  <li>Bolo</li>  
</ol>
```

## Lista Ordenada

- I. Café
- II. Leite
- III. Bolo

Para exibir a lista com o tipo alfabético:

```
<h3>Lista Ordenada</h3>  
<ol type="A">  
  <li>Café</li>  
  <li>Leite</li>  
  <li>Bolo</li>  
</ol>
```

## Lista Ordenada

- A. Café
- B. Leite
- C. Bolo

**Obs:** Como o padrão é números, não é necessário especificar no atributo **type** o tipo 1. Pode especificar também algarismo romano “i” e alfabético “a”.

```
<h3>Lista não ordenada</h3>  
<ul>  
  <li>Café</li>  
  <li>Leite</li>  
  <li>Bolo</li>  
</ul>
```

## Lista não ordenada

- Café
- Leite
- Bolo

Este tipo é o disc (*que é o padrão*). Abaixo estão os outros 2 tipos:

```
<h3>Lista não ordenada</h3>
<ul type="circle">
  <li>Café</li>
  <li>Leite</li>
  <li>Bolo</li>
</ul>
```

### Lista não ordenada

- Café
- Leite
- Bolo

```
<h3>Lista não ordenada</h3>
<ul type="square">
  <li>Café</li>
  <li>Leite</li>
  <li>Bolo</li>
</ul>
```

### Lista não ordenada

- Café
- Leite
- Bolo

### Metas tags HTML e SEO

Meta tags são importantes ferramentas de comunicação entre o webmaster e o site de busca. O **SEO** para muitas pessoas é somente isso: **Meta Tags**. Contudo existe cerca de 250 variáveis para determinar os resultados da busca, e as meta tags são apenas algumas delas.

#### Exemplo:

```
<meta name="author" content="Edson David Pereira">
<meta name="description" content="Exemplos de listas ordenadas e não Ordenadas">
<meta name="keywords" content="lista, ordenada">
```

### Tabelas

As tabelas são definidas com a tag <table>. Uma tabela é dividida em linhas (com a tag <tr>), e cada linha é dividida em células de dados (com a tag <td>). As letras td significam "table data," que é o conteúdo de uma célula de dados. As colunas de cabeçalho são definidas com a tag <th> </th>. Para mesclar em uma determinada linha, utilizando mais de uma coluna, é utilizado o atributo colspan. Com elas pode-se fazer alinhamentos que dificilmente seriam possíveis com simples comandos.

Em anos atrás vários layouts de websites eram construídos com tabela onde a mesma fazia com que um determinado site tinha um aspecto mais profissional e o usuário podia navegar de forma mais eficiente, pois os objetos podem eram melhor posicionados na home-page.

#### Exemplo básico de uma tabela:

## Curso Desenvolvimento WEB

```

<table border="1" width="1000" align="center" cellspacing="3" cellpadding="5">
  <thead>
    <tr>
      <th colspan="3" bgcolor="#0ff"><h3>Lista de Produtos</h3></th>
    </tr>
    <tr>
      <th align="right" bgcolor="#ccc">Código</th>
      <th align="left" bgcolor="#ccc">Produto</th>
      <th align="right" bgcolor="#ccc">Preço</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td align="right">1</td>
      <td align="left">Dual Core 2.5GHz Memória 2GB HD 320GB DVD-
RW Linux - Qbex + Monitor LCD Widescreen 15,6"
      </td>
      <td align="right">R$ 1.099,00</td>
    </tr>
    <tr>
      <td align="right">2</td>
      <td align="left">Quad Core 2.3GHz Memória 4GB HD 500GB DVD-
RW Linux - Space BR + Monitor LCD 21,5" Widescreen</td>
      <td align="right">R$ 1.299,00</td>
    </tr>
    <tr>
      <td align="right">3</td>
      <td align="left">Impressora + Copiadora + Scanner</td>
      <td align="right">R$ 299,00</td>
    </tr>
    <tr>
      <td align="right">4</td>
      <td align="left">Notebook Atom N270 1.6GHz Memória 1GB HD 160GB LED 10"
Cinza</td>
      <td align="right">R$ 1.049,00</td>
    </tr>
    <tr>
      <td align="right">5</td>
      <td align="left">Impressora + Copiadora + Scanner + Fax</td>
      <td align="right">R$ 299,00</td>
    </tr>
    <tr>
      <th colspan="3" bgcolor="#0ff"><strong>System.Net - Produtos e Assistência em Inf
ormática - Contato: 17-99999-0000</strong></th>
    </tr>
  </tbody>
</table>

```

## Curso Desenvolvimento WEB

**Border:** Define a espessura da borda da tabela. Exemplo: `<table`

`border="1">`

**Width:** Define a largura de uma tabela, bem como de uma coluna. **Exemplo:** `<table border="1" width="1000">`

**Align=** Define o alinhamento de uma tabela, bem como de uma coluna. **Exemplo:**

`<table border="1" width="1000" align="center">`

**Cellspacing:** Define o espaçamento entre uma e outra coluna.

**Exemplo:** `<table border="1" width="1000" align="center" cellspacing="3">`

**Cellpadding:** Define o espaçamento interno de uma coluna. **Exemplo:**

`<table border="1" width="1000" align="center" cellspacing="3" cellpadding="5">`

**Thead:** Indica o cabeçalho da tabela. Ou seja: os cabeçalhos onde vão ser colocados o título da tabela, bem as colunas de cabeçalho. **Exemplo:** `<thead>`

```
<tr>
  <th colspan="3" bgcolor="#0ff"><h3>Lista de Produtos</h3></th>
</tr>
<tr>
  <th align="right" bgcolor="#ccc">Código</th>
  <th align="left" bgcolor="#ccc">Produto</th>
  <th align="right" bgcolor="#ccc">Preço</th>
</tr>
</thead>
```

**Bgcolor:** Define a cor de fundo da coluna.

**Bordercolor:** Define a cor da borda, que pode ser de toda a tabela, ou de uma determinada linha.

**Tbody:** Indica o corpo da tabela. Ou seja: onde serão inseridos os conteúdos, referente a cada cabeçalho. **Exemplo:**

```
tbody>
  <tr>
    <td align="right">1</td>
    <td align="left">Dual Core 2.5GHz Memória 2GB HD 320GB DVD-
RW Linux - Qbex + Monitor LCD Widescreen 15,6"</td>
    <td align="right">R$ 1.099,00</td>
  </tr>
</tbody>
```

No exemplo acima, foi definido com uma linha. Você pode especificar quantas linhas deseja, sempre entre as tags `<tbody>` e `</tbody>`.

**Observação:** Uma tabela inicia com a tag `<table>`, e deverá ser fechada com `</table>`.

**Exemplo de tabela com imagens:**

```
<table border="0" width="300" align="center" cellspacing="3" cellpadding="5" bordercolor="#f00"
>
  <thead>
    <tr>
      <th colspan="3" bgcolor="#0ff"><strong>Lista de Imagens</strong></th>
```



```
</tr>
</thead>
<tbody>
<tr>
<td align="left"></td>
<td align="left"></td>
<td align="left"></td>
</tr>
<tr>
<td align="center" bgcolor="#ccc">Logo Senac</td>
<td align="center" bgcolor="#ccc">Informática</td>
<td align="center" bgcolor="#ccc">Clínica</td>
</tr>
</tbody>
</table>
```

### Formulários

É possível a criação de formulários na linguagem de marcação html, através de dois métodos: **POST e GET**.

**POST:** Este método é sigiloso, ou seja: envia os parâmetros no corpo da requisição HTTP, escondendo-os da URL. Muito utilizado em formulários web. Ele permite proteger os dados submetidos pelo formulário. Se desejarmos proteger, temos que utilizar ao invés do **HTTP o HTTPS**, pois aí conseguimos criptografar os dados. Com o método **POST**, é possível enviar informações um pouco maior, como imagens. No método **GET**, algumas partes podem ser perdidas no caminho.

As requisições **POST** são mais utilizadas para enviar informações para serem processadas, como por exemplo, criar algum recurso, como um produto, ou um cliente.

**GET:** Quando utilizamos o GET, os parâmetros são passados no cabeçalho da requisição. Por isso, podem ser vistos pela URL, como no caso de um formulário de login. As requisições do tipo GET são recomendadas para obter dados de um determinado recurso. Como em um formulário de busca ou em uma listagem de todos os produtos cadastrados.

#### Tipos utilizados em um formulário:

```
<form method="POST" action="dados_recebidos.html">
```

#### label

Rótulo. Nome que será exibido para que o usuário identifique o que será informado em uma caixa. **Exemplo:** `<label for="nome">Nome:</label>`

#### input

Entrada de dados, que podem ser de: **text, password, hidden, image, button, submit, reset, checkbox, radio, select**. **Exemplo:**

```
input type="text" name="nome">
```

onde **name** indica o nome da caixa (nome do campo)

## Curso Desenvolvimento WEB

**method:** Método de envio.

**action:** ação que pode ser o nome de um arquivo, ou mesmo estar em branco, indicando que os dados serão processados na mesma página, sendo neste caso deverá estar em branco o action.

**button:** Insere um botão. **Exemplo:**

```
<input type="submit" name="enviar" value="Enviar">
```

**Submit:** Encaminha os dados para algum arquivo;

**Name:** nome do objeto que identifica para o arquivo (como se fosse o nome do campo);

**Value:** nome que será exibido para o usuário.

```
<input type="reset" name="limpar" value="Limpar">
```

**reset:** Limpa todas as caixas (campos) do formulário.

**hidden:** Tipo de campo que também é utilizado no formulário, entretanto não é exibido.

**Exemplo do método POST:**

```
<form method="POST" action="dados_recebidos.html">

    <label for="nome">Nome:</label>

    <input type="text" name="nome"><br>

    <label for="sobrenome">Sobrenome:</label>

    <input type="text" name="sobrenome"><br>

    <input type="submit" name="enviar" value="Enviar">

</form>
```

**Atributos:**

**required:** Deixa como obrigatório o preenchimento do item.

**placeholder:** Deixa uma ajuda para o preenchimento do item. Quando o usuário começar a preencher desaparece.

**fieldset:** Exibe uma moldura (caixa) envolvendo os itens do formulário.

**legend:** Exibe a legenda dentro do fieldset.

**Outros tipos:**

**checkbox:** Insere uma caixa de verificação para vários itens. **Exemplo:**

```
<h3>Desenvolvimento desktop/web. Selecione os que utiliza:</h3>
    <input type="checkbox" name="html" value="html" checked><label
1 for="html">HTML</label><br>
    <input type="checkbox" name="css" value="css">CSS<br>
```

### Desenvolvimento desktop/web. Selecione os que utiliza:

☒ HTML  
☐ CSS

O atributo **checked** deixa selecionado. Para retirar a seleção, é só retirar este atributo.

**inputbox:** Insere um similar ao checkbox, de aspecto circular, entretanto só permite selecionar um item, desde que o name seja o mesmo. **Exemplo:**

```
<h3>Sexo:</h3>
    <input type="radio" name="sexo" value="Masculino"><label for=
"masculino">Masculino</label>
```

```
<input type="radio" name="sexo" value="Feminino"><label for="feminino">Feminino</label>
```

**Sexo:**

☐ Masculino ☐ Feminino

**Obs.:** O atributo checked pode ser utilizado também, para que determinado item esteja selecionado como padrão.

O tipo select criar uma caixa com uma lista de opções. O atributo size, quando igual a 1, exibe somente 1 item da lista. <h3>Cursos oferecidos:</h3>

```
<label for="escolaridade">Escolaridade: </label>
<select name="cursos" size="1">
  <option value="#">Selecione</option>

  <option value="html_css">HTML5/CSS3</option>

  <option value="javascript">Javascript</option>
  <option value="jquery">jQuery</option>
  <option value="php">PHP</option>
  <option value="aspnet">ASP-Net</option>
  <option value="python">Python</option>
</select>
```

O tipo textarea é uma caixa de texto com várias linhas.

```
<fieldset>
  <legend>Sugestões</legend>
  <label for="sugestao"></label>
  <textarea cols=60 rows="10" name="sugestoes" maxlength="500"></text
area>
</fieldset>
```

Neste exemplo são criadas 60 colunas, 10 linhas com tamanho máximo de caracteres 500.

**Exemplo de botões:**

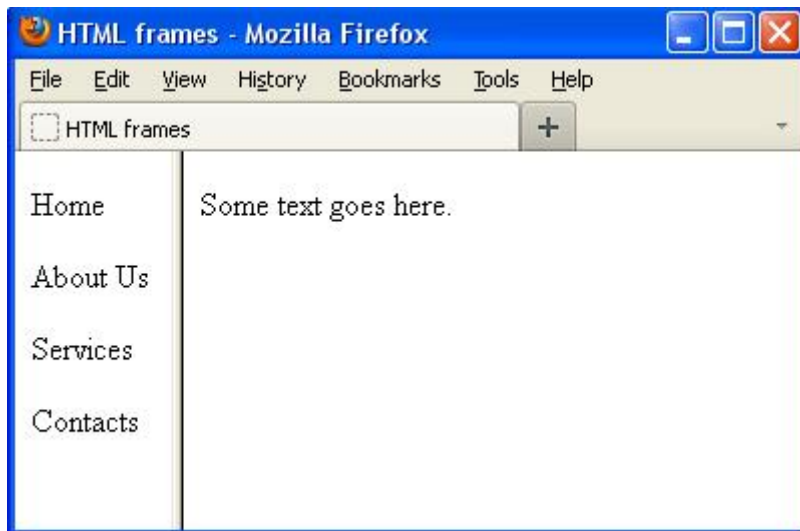
```
<p>
  <input type="submit" name="enviar" value="Enviar">
  <input type="reset" name="limpar" value="Limpar">
</p>
```

Enviar

Limpar

### iframe

Há muitos anos atrás uma das tecnologias mais comuns e muito utilizadas por desenvolvedores de layout para web, era utilizar o **frameset**, ou seja: conjunto de **frames (quadros)**. Consiste na divisão da janela do navegador em várias partes, e cada uma delas era colocada uma página diferente.



Observe na imagem acima, a divisão com a parte superior (denominada de banner ou topo), a parte lateral esquerda com o menu, e a lateral a direita onde eram exibidos os conteúdos referente a cada link que era clicado.

A **W3C** descontinuou isso a partir do HTML5, ou seja: não é mais aceito. Com isso o **iframe**, permaneceu e é possível utiliza-la para criação de layouts.



O **iframe** é um elemento HTML muito poderoso. Através da tag **<iframe>**, é possível incluir vídeos, imagens e conteúdos de outras páginas. Ele pode integrar o conteúdo em qualquer parte do seu layout, sem interferir na estrutura do seu site. Dessa forma, podemos estilizá-lo diretamente através do **CSS**. Sendo assim, ele possui o comportamento de um elemento HTML tradicional e pode ser modificado a fim de se adequar ao layout da sua página.

Portanto, imagine o **iframe** como parte do conteúdo, porém lembre-se que não faz parte da sua página. Quando encontramos vídeos do YouTube incorporados dentro de uma página, esses

elementos foram integrados utilizando o iframe HTML. Da mesma forma, os mapas do Google geralmente também são incluídos através de um iframe HTML.

### Sintaxe:

Apesar da maioria dos navegadores atuais suportarem esse elemento sem problemas, algumas versões mais antigas ainda podem encontrar dificuldades. Para incluir uma mensagem caso o navegador não tenha suporte, basta acrescentar um conteúdo dentro das tags **<iframe>**. Assim, caso o navegador não suporte o iframe, será mostrada a mensagem, caso ele suporte a mensagem não será exibida. Podemos incluir apenas uma mensagem, ou ainda incluir um código HTML. Vejamos então os exemplos abaixo:

### Exemplo:

```
<iframe name="principal" src="home.html" frameborder="0" scrolling="auto" width="1200" height="1000"></iframe>
```

<p>Seu navegador não suporte à tecnologia necessária para exibição desta página.</p>

**name:** identifica o nome do iframe;

**src:** página que será carregada automaticamente, quando a página principal (normalmente denominada **index**), for carregada;

**frameborder:** indica se vai ter uma borda frameborder="1", ou sem borda: frameborder="0";

**scrolling:** indica se vai exibir barra de rolagem tanto vertical quanto horizontal (se houver **yes**, senão houver **no**, ou **auto**, caso o conteúdo ultrapasse na horizontal e/ou na vertical) caso o conteúdo seja maior do que a área definida, nos atributos width e height.

É obrigatório o fechamento do **iframe**.

### Exemplo em html:

```
<body>
  <h1>EXEMPLO DA TAG IFRAME</h1>
  <div id="menu">
    <p><a href="home.html" target="principal">Home</a></p>
    <p><a href="link1.html" target="principal">Link 1</a></p>
    <p><a href="link2.html" target="principal">Link 2</a></p>
    <p><a href="link3.html" target="principal">Link 3</a></p>
    <p><a href="link4.html" target="principal">Link 4</a></p>
  </div>

  <iframe name="principal" src="home.html" frameborder="0" scrolling="a
uto" width="1000" height="800">
    <p>Seu navegador não suporte à tecnologia necessária para exibição
desta página.</p>
  </iframe>
</body>
```

**Definição o CSS interno** (*O CSS será explicado no capítulo seguinte. Ele permite estilizar, ou seja: é o que dá aparência para a página*):

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Exemplo da tag iframe</title>
  <style type="text/css">
    #menu {
      width:110px;
      background-color: #fff;
      float: left;
    }
  </style>
</head>
```

**Obs.:** a propriedade float é que permite deslocar a div menu para a esquerda, para que o iframe fique alinhado ao lado dos links do menu.

### Página Home

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Home Page</title>
  <style type="text/css">
    #texto1 {
      color: #00f;
      font-size: 16px;
      font-family: 'Courier New', Courier, monospace;
    }
    #texto2 {
      color:chartreuse;
      font-style: italic;
      font-variant: small-caps;
      background-color:#0f0;
      font:"Comic Sans MS";
      font-size:25px;
    }
  </style>
</head>
<body>
  <h1>HOME PAGE</h1>
  <marquee direction="left">Seja bem-vindo</marquee>
  <p id="texto1">O iframe é um elemento HTML muito poderoso. Atravé
s da tag iframe, é possível incluir vídeos, imagens e conteúdos de outras
páginas. Ele pode integrar o conteúdo em qualquer parte do seu layout, s
```

em interferir na estrutura do seu site. Dessa forma, podemos estilizá-lo diretamente através do CSS. Sendo assim, ele possui o comportamento de um elemento HTML tradicional e pode ser modificado a fim de se adequar a o layout da sua página.</p>

```
<p id="texto2">Portanto, imagine o iframe como parte do conteúdo, porém lembre-se que não faz parte da sua página. Quando encontramos vídeos do YouTube incorporados dentro de uma página, esses elementos foram integrados utilizando o iframe HTML. Da mesma forma, os mapas do Google geralmente também são incluídos através de um iframe HTML.</p></body></html>
```

Para que no CSS seja declarado o seletor **p** (que permite modificar características da tag **p**), é necessário criar o identificador através de: `<p id="texto1">` e `texto2`.

No CSS através da hashtag (#) ele identifica o nome deste id.

### HTML5 – Tags semânticas

A semântica é importante para quem está visualizando a estrutura de seu código html.

Novos elementos O HTML5 possui uma série de elementos que permitem definir setores principais na página do código. Por meio desses elementos é possível diferenciar diretamente pelo código HTML5 áreas importantes do site como 'rodapé cabeçalho e barra lateral. Conseguimos nos reportar à área do site indicando exatamente rodapé, deve estar o conteúdo mencionado. Fazendo uma comparação: por exemplo, quando você está lendo um livro, é muito fácil diferenciar um capítulo, um título ou um parágrafo apenas pelo tamanho da letra, da fonte ou da cor. Isso seria muito complicado para um mecanismo de busca utilizando apenas o go, pois esses detalhes passariam despercebidos. Com o HTML5, os buscadores têm seu trabalho simplificado e conseguem obter do código informações mais exatas, levando menos tempo para estocar essa informação. Diante desse cenário, um novo termo é utilizado na programação e na construção de páginas em HTML5 que é a semântica. Ela pode ser compreendida como o estudo do significado, ou seja, esses novos elementos representam um significado para o navegador e o desenvolvedor. Com a inserção desses elementos, o HTML5 oferece novos elementos semânticos para definir diferentes partes de uma página para web:

<header>	
<nav>	
<section>	<aside>
<article>	
<footer>	

Veja a definição de cada um desses elementos:

**<header>**: especifica um cabeçalho de um documento ou uma seção. Deve ser utilizado quando seu conteúdo tiver o significado introdutório para a página. A página pode ter vários elementos <header> em um mesmo documento;

**<nav>**: define um conjunto de links de navegação. Esse elemento destina-se a grande, blocos de links de navegação. No entanto, nem todos os links em um documento precisam estar dentro de um elemento <nav>;

**<section>**: define uma seção do documento, um agrupamento temático de um conteúdo que geralmente vem com um título. A página pode ser dividida em seções nas quais o desenvolvedor pode inserir informações, como: introdução, conteúdo e informações de contato;

especifica um conteúdo próprio e independente de outros conteúdos. "Toda

vez que a página tiver uma informação que faça sentido por si própria de forma independente do resto do site, esse conteúdo deve ser colocado nesse elemento da página- Por exemplo: postagem de blogs, artigos para jornal, fóruns de discussão;

**<aside>**: define um conteúdo secundário que pode ser colocado como uma barra lateral, um conteúdo que pode agregar ou não as informações relativas à página;

**<footer>**: esse elemento especifica um rodapé de um documento ou de uma seção. Ele deve conter informações sobre o autor do documento, os direitos autorais, os links para termos de uso, as informações de contato, etc. É possível ter vários elementos <footer> em uma página.

Com o uso desses elementos, os motores de busca poderão identificar o verdadeiro sentido de conteúdo em uma página, permitindo que esses dados possam ser compartilhados e reutilizados em empresas, comunidades e em outras aplicações.

### Elemento **<main>**

Apesar da inclusão de novos elementos semânticos, não existia uma maneira de agrupá-los e colocar o conteúdo principal em um bloco. Anteriormente, os desenvolvedores utilizavam o elemento <div> para isso. No final de 2012, o grupo de trabalho do W3C incluiu um novo elemento na documentação padrão, e essa inclusão o tornou o novo elemento principal do corpo do documento. Esse elemento foi o <main> e com ele foi possível dar mais significado a outros elementos da página.

Não deve haver mais do que um elemento <main> em uma mesma página.

O elemento <main> não deve ser utilizado dentro de outros elementos como <article>, <aside>, <footer>, <header> ou <nav>.

Inserindo o elemento <main>

### **Exemplos:**



**<header>**

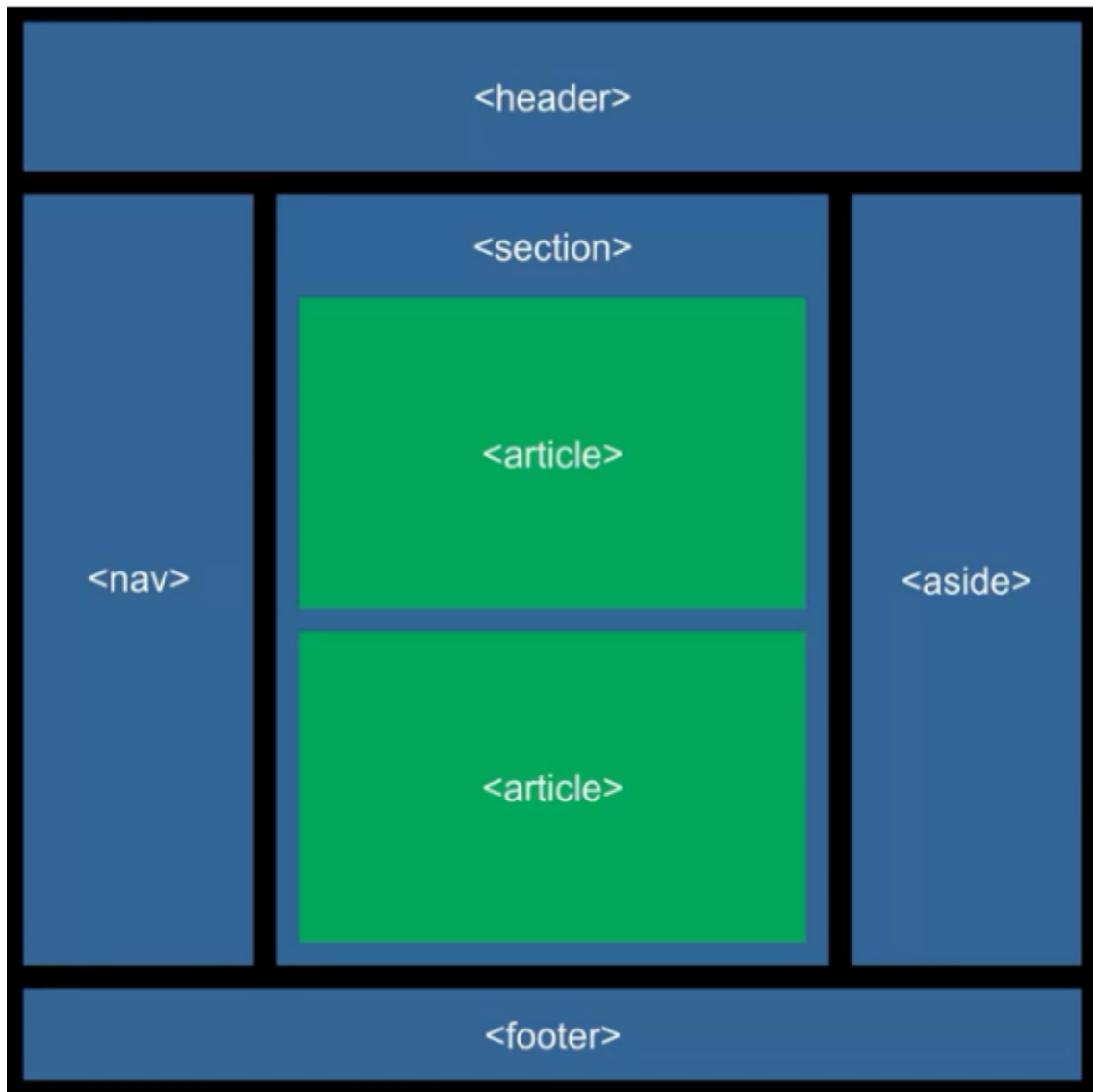
**<nav>**

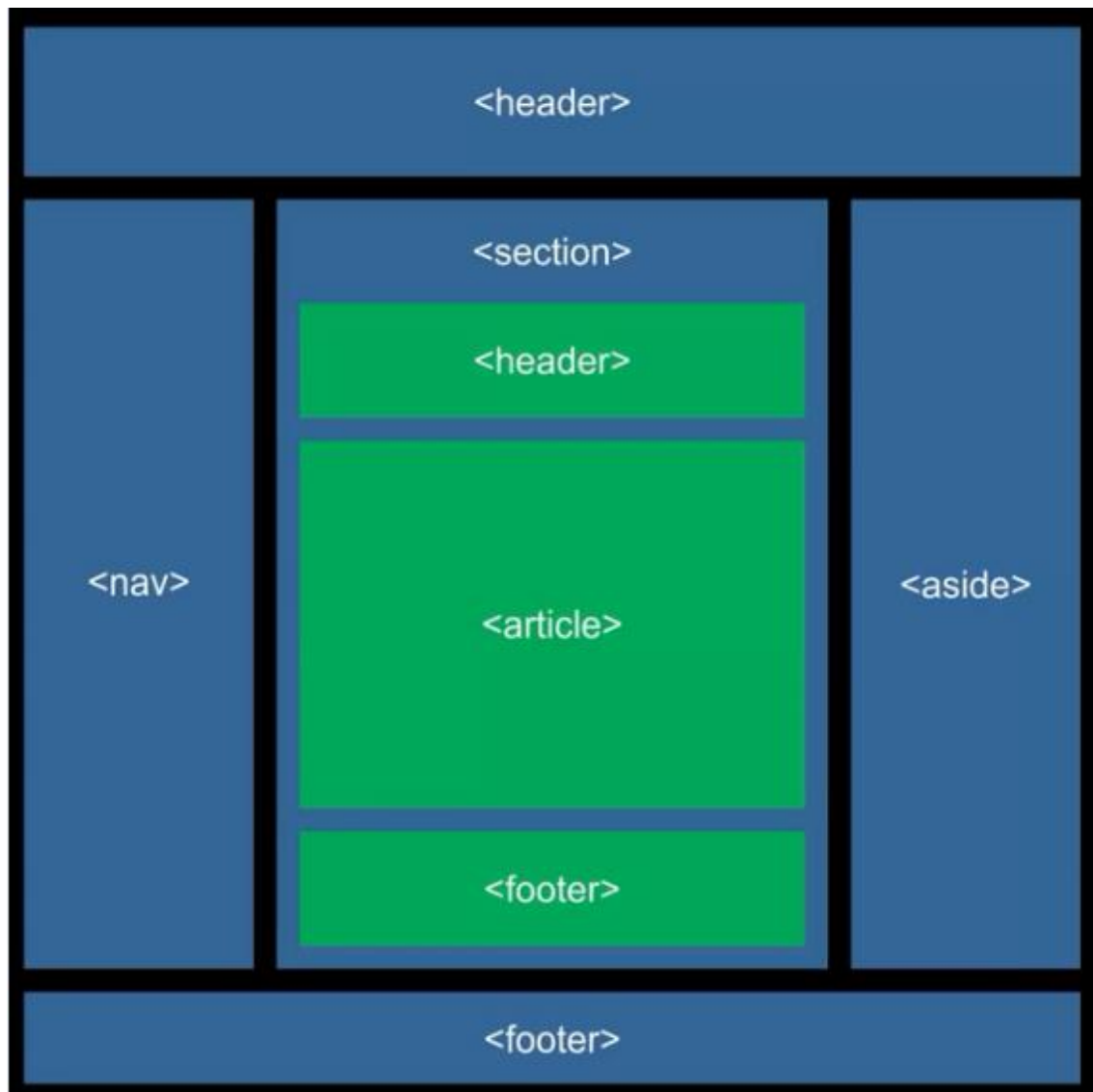
**<section>**

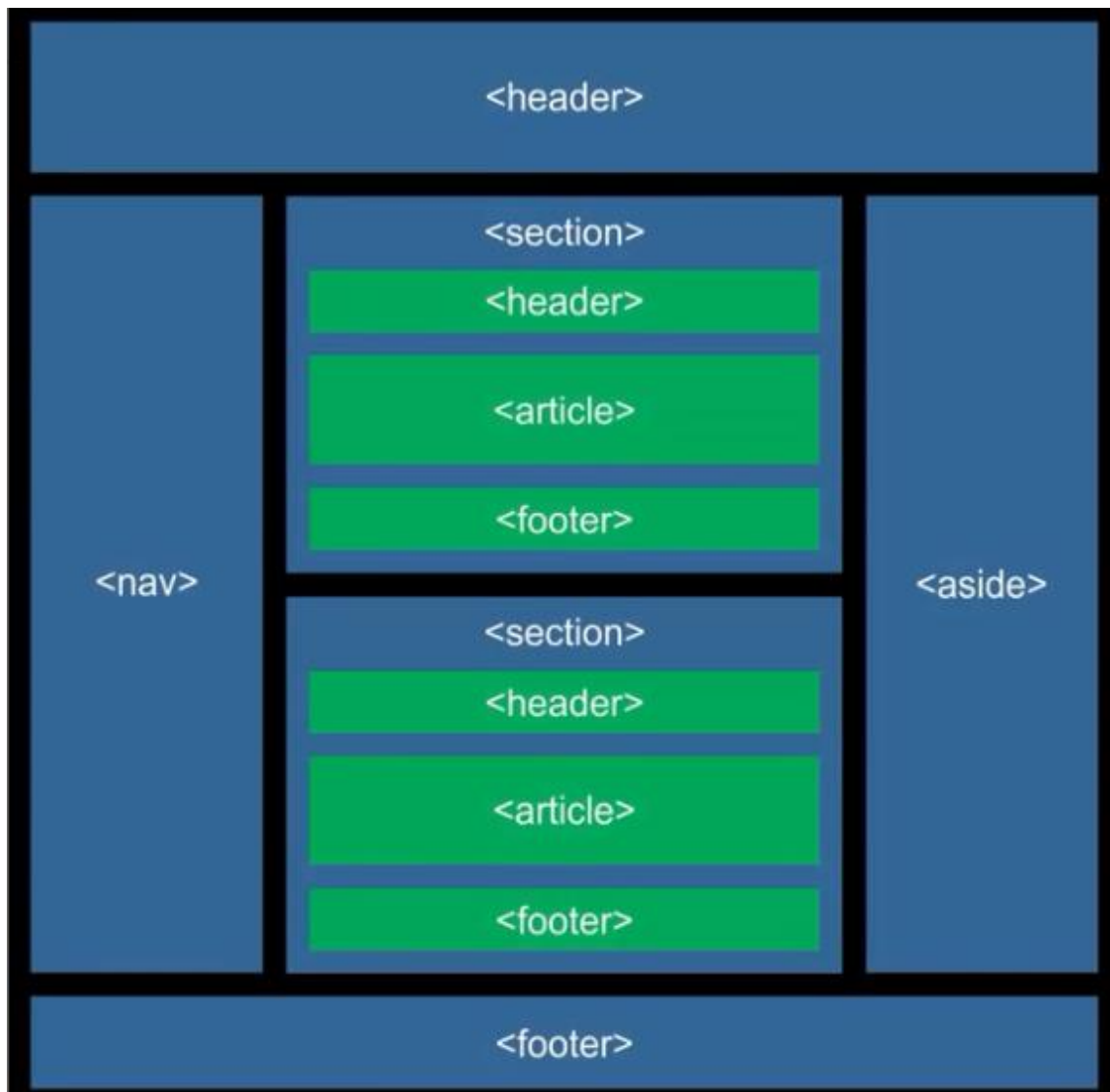
**<aside>**

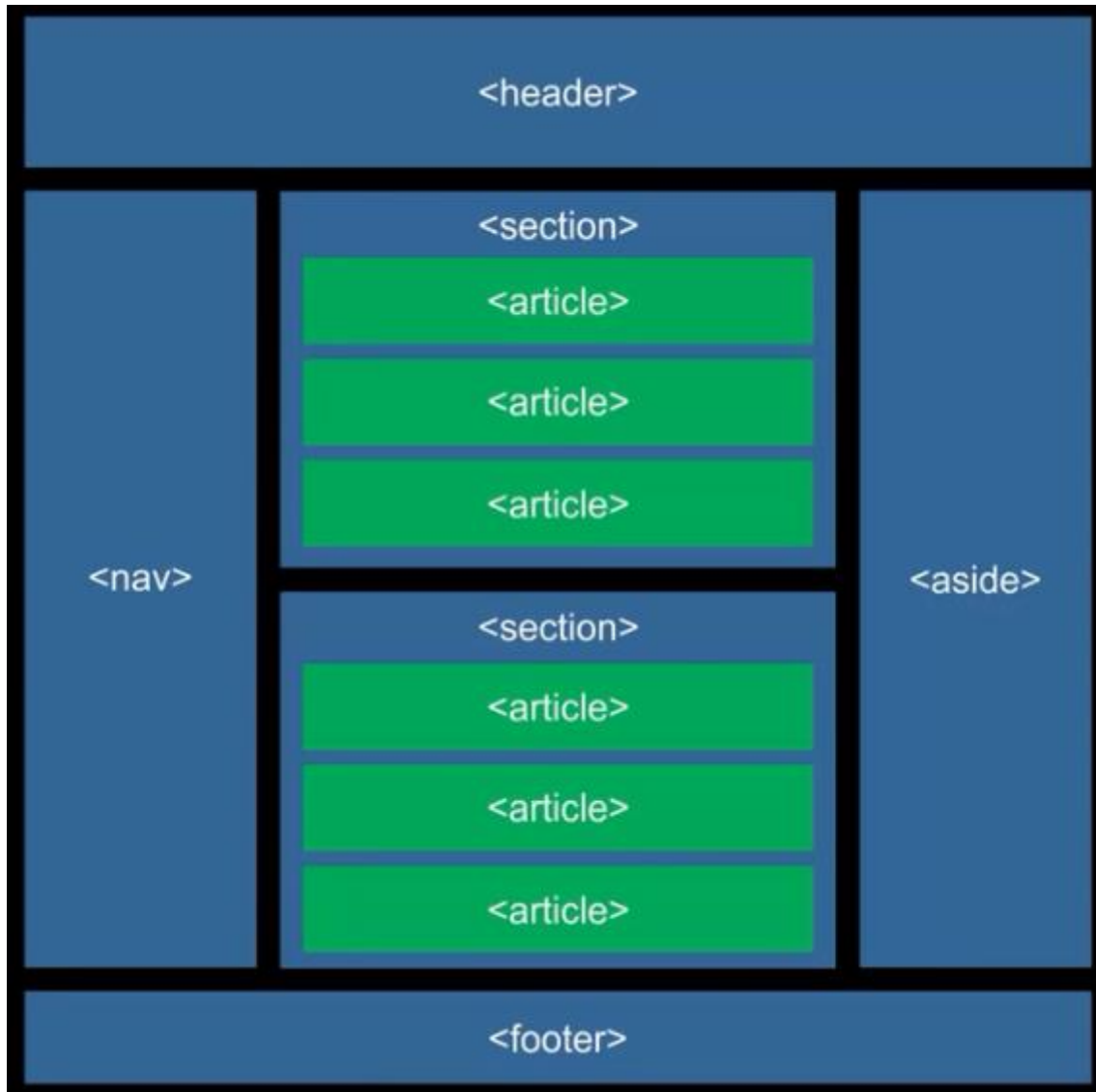
**<article>**

**<footer>**









### CSS – estilos importantíssimos e fundamentais

**CSS** não é uma linguagem de programação, e sim uma linguagem com estilos. Existem inúmeras regras (declarações) que indica como o conteúdo deve ser apresentado. Significado de **CSS** – Cascading Style Sheet – Folhas de estilo em cascata.

O conjunto destas regras pertencente a folha de estilos, pode ser feita de três maneiras pelo desenvolvedor. A mais utilizada consiste em um arquivo externo com extensão `css`, com todas as declarações em **CSS**. A segunda é criar estas declarações na mesma página única (como são feitos em blogs) entre as tags `<head>` `</head>` e a terceira é em casos mais específicos, aplicados junto ao elemento (**tag**) que é denominado de **CSS Inline**.

#### Primeira opção – chamando um arquivo externo:

```
<link href="estilos.css" type="text/css" rel="stylesheet">
```

#### Segunda opção – declarado entres as tags `<head>` `</head>`:

```
<style type="text/css">
  p {
```

```

        color:#0f0; /* cor do texto */
    }
    body {
        font-family: Verdana, Geneva, Tahoma, sans-serif; /* fonte pela
ordem */
    }
</style>

```

p: seletor (refere-se dentro do **CSS**) e indica a tag em **HTML** que terão suas características alteradas;

color: propriedade;

#0f0: valor que será passado a esta propriedade.

Observe que depois do seletor tem que abrir a chave ( { )indicar as propriedades, sempre finalizando com ponto e vírgula (;) e encerrar com o fechamento da chave ( } ).

### Terceira opção: declarado junto ao elemento (tag) - CSS Inline

Aplicado por exemplo somente na linha que você deseja alterar (modificar) as características de uma determinada tag. Veja exemplo abaixo:

```
<p style= "color: #ff0; font-size:30px">CSS Inline</p>
```

style: estilo que poderá receber uma ou mais propriedades;

color: propriedade do **css**;

#ff0: valor que esta propriedade irá receber

### Tabela de algumas propriedades/valores

#### Lista

Propriedade	Descrição	Exemplo
list-style-image	Define a imagem que será exibida	list-style-image: url(bullets/list.gif);
list-style-position	Define o posicionamento da imagem	list-style-position: inside; /* A imagem fica na parte interna */ list-style-position: outside; /* A imagem fica na parte externa */

### Tabela de unidades

Unidade	Descrição
%	Porcentagem (geralmente relativas ao elemento pai)
in	Polegada (2,54 cm)
cm	Centímetro
mm	Milímetro
em	1em é igual ao tamanho da fonte atual. 2em significa 2 vezes maior que o tamanho atual
pt	Pontos (1pt é o mesmo que 1/72)
px	Pixels (um ponto na tela do computador/dispositivo)

## Cores

Propriedade	Descrição
color:red	Nome da cor. No exemplo <b>vermelha</b>
rgb(x,x,x) → três cores: red, green, blue	Ex.: rgb(255,0,0)

## Fonte

Propriedade	Descrição	Exemplo
font-family	Família de algumas fontes, pela ordem.	<b>font-family:</b> Arial, Helvetica, sans-serif;
font-size	Tamanho da fonte	<b>font-size:</b> 18px;
font-style	Estilo da fonte	<b>font-style:</b> italic;
font-variant	Fonte com tamanho maiúsculo reduzido →	<b>font-variant:</b> small-caps;
font-weight	Configura a fonte para negrito	<b>font-weight:</b> bold;

## Texto

Propriedade	Exemplo
color	<b>color:</b> #f00; /* Cor do texto */
line-height	<b>line-height:</b> 1px; /* distância entre as linhas */
letter-spacing	<b>letter-spacing:</b> 1px; /* espaço entre os caracteres */
text-align	<b>text-align:</b> center; /* alinha o texto ao centro */
text-decoration	<b>text-decoration:</b> none; /* retira o sublinhado */
text-indent	<b>text-indent:</b> 1.2cm; /* indenta (ideal para criar parágrafo) */

## Fundo

Propriedade	Exemplo
background-image	<b>background-image:</b> url(imagens/webdesign.png); /* Imagem de fundo */
background-repeat	<b>background-repeat:</b> no-repeat; /* Não repetir a imagem */
background-position	<b>background-position:</b> center center; /* Define a posição da imagem */ <b>background-position:</b> 100px 300px; /* Define a posição da imagem 100px à esquerda e 300px em relação a parte superior */

## Borda

Propriedade	Exemplo
<b>border-bottom</b>	<code>border-bottom: thin dotted #0f0;</code>  <code>/* Define a borda inferior pontilhado fino, cor verde */</code>
<b>border-radius</b>	<code>border-radius: 15px;</code> <code>/* arredonda todas as margem */</code>
<b>border</b>	<code>border: thin solid #C0262C;</code>

## Margem

Propriedade	Exemplo
<b>margin-left</b>	<code>margin-left: 50px; /* Distância da margem esquerda em 50px */</code>
<b>margin-right</b>	<code>margin-right: 50px; /* Distância da margem direita em 50px */</code>
<b>margin-top</b>	<code>margin-top: 50px; /* Distância da margem superior em 50px */</code>
<b>margin-bottom</b>	<code>margin-bottom: 50px; /* Distância da margem inferior em 50px */</code>

## Espaçamento

Propriedade	Exemplo
<b>padding-left</b>	<code>padding-left: 50px; /* espaçamento interno esquerda em 50px */</code>
<b>padding-right</b>	<code>padding-right: 50px; /* espaçamento interno direita em 50px */</code>
<b>padding-top</b>	<code>padding-top: 50px; /* espaçamento interno superior em 50px */</code>
<b>padding-bottom</b>	<code>padding-bottom: 50px; /* espaçamento interno inferior em 50px */</code>



## Links

Propriedade	Exemplo
<b>a</b>	<pre>a {     font-family: Arial, Helvetica, sans-serif;     font-weight: bold;     text-decoration: none; } /* Define as propriedades gerais do link */</pre>
<b>a:link</b>	<pre>a:link {     color: #00f;     background-color: aqua; } /* Cor do link */</pre>
<b>a:visited</b>	<pre>a:visited {     color: #3b4758;     background-color: #99b9e4; } /* Cor do link visitado */</pre>
<b>a:hover</b>	<pre>a:hover {     color: #3b4758;     background-color: #0f0; } /* quando passa o mouse sobre o link */</pre>

Outros	Exemplo
Aplica o estilo no elemento input, somente no tipo de botão submit: altera o ponteiro do mouse, arredonda as quatro margens e aplica sombra	<pre>input[type="submit"]{     cursor: pointer;     border-radius: 30px;     text- shadow: 2px 2px 5px #000;</pre>
<b>float</b> → posiciona o elemento à esquerda	<pre>float: left;</pre>

## CSS3 X Flexbox

Propriedade	Exemplo
rgba(x,x,x,x) → além das três cores, transparência	Ex.: rgba(255,255,0,0.5)
<b>height:100vh</b>	height definindo o espaçamento na vertical, onde vh é o viewport height
<pre>@media (min-width:400px) {     body {         font-size:14px;     } }</pre>	Definindo o valor mínimo da largura através da propriedade <b>min-width</b> . No exemplo ao lado quando a largura da página estiver em 400px, a fonte do corpo da página será de 14px. É como se fosse uma estrutura condicional.

## Curso Desenvolvimento WEB

Neste outro exemplo: se a largura estiver em 600px, a fonte será de 18px.	<pre>@media (min-width:600px) {     body {         font-size:18px;     } }</pre>
<code>display: flex</code>	Definindo os links como coluna através da propriedade
<code>justify-content: center;</code>	Alinhamento centralizado
<code>align-items: center;</code>	Alinhamento na vertical
<code>justify-content: space-around;</code>	Espaçamento entre as colunas
<code>display: grid;</code>	Exibição em formato de grid
<code>gap: 2em;</code>	Define o espaçamento entre cada coluna ou linha do grid. Isso acontece com o <code>display: grid</code> é declarado
<code>grid-template-columns: 240px 240px 240px;</code>	Define a largura de cada coluna.
<code>grid-template-columns: repeat(3, 260px);</code>	Tem o mesmo efeito que o exemplo acima.
<code>transition: background-color 1s, color 1s;</code>	Controla a velocidade de uma animação quando há mudanças de propriedades CSS
<code>transform: scale(0.2);</code>	especifica como os elementos aninhados são renderizados no espaço 3D
<code>animation: linear 2s;</code>	animação entre estilos.

### Programando online:

<https://codepen.io/pen>

HTML; CSS e Javascript

<https://repl.it/@edsondavid/AttentiveTintedLivedistro>

Tags, CSS, Javascript, etc...

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

Procure no google: Color Picker (Seletor de cores)

<https://wordpress.com/>

DotNet

<https://dotnetfiddle.net/>

HTML, Javascript, PHP

<https://repl.it/languages>

Várias linguagens, Mobile, Jogos, etc...

<https://code.org/>

### **Sobre o CSS - Função**

<https://www.chiefdesign.com.br/css/>

Exemplos de HTML5

<https://flatschart.com/html5/>

Fontes

<https://fonts.google.com/>

MDN

<https://developer.mozilla.org/pt-BR/>

### **Propriedade clip-path**

MDN Color Picker

[https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS\\_Colors/seletor\\_de\\_cores](https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Colors/seletor_de_cores)

CSS Clip-path

<https://bennettfeely.com/clippy/>

Indica em que navegadores os elementos html/css podem ser utilizados

<https://caniuse.com/>

Link com sites de imagens **pagas**

<https://stock.adobe.com/br/>

<https://www.shutterstock.com/>

### **Gratuitas**

<https://www.pexels.com/pt-br/>

<https://pixabay.com/>

<https://unsplash.com/>

ícones

<https://iconarchive.com/>

[https://www.w3schools.com/icons/fontawesome5\\_intro.asp](https://www.w3schools.com/icons/fontawesome5_intro.asp)

<https://material.io/resources/icons/>

<https://www.flaticon.com/>

### **Criar jogos**

<https://editor.p5js.org/>

<https://fontmeme.com/netflix-font/>

### **Criar sites**

<https://sites.google.com/new>

### **Criação de jogos**

<https://www.makeindiegames.com.br/>

### **Mapeamento de imagens**

<https://www.image-map.net/>

### **Desenvolvimento de sistemas**

[GDB online Debugger | Compiler - Code, Compile, Run, Debug online C, C++ \(onlinegdb.com\)](#)