

HTML - HyperText Markup Language

- **Cabeçalhos:**

O HTML possui seis níveis de cabeçalhos (header), numerados de 1 a 6, sendo o número 1 o de maior destaque. Este número determina o tamanho da fonte do cabeçalho a ser apresentado no documento. Cabeçalhos são exibidos em letras maiores e em negrito.

Dessa forma, é possível usar seis formatos diferentes de cabeçalhos por meio das tags **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** e **<h6>**. À medida que o valor numérico aumenta, o texto diminui. As tags de cabeçalho também inserem uma linha em branco antes e depois do cabeçalho.

Exemplo:

```
<h1>HTML</h1>
```

```
<h3>HyperText Markup Language</h3>
```

```
<h6>Linguagem de Marcação de Hipertexto</h6>
```

HTML

HyperText Markup Language

Linguagem de Marcação de Hipertexto

- **Linhas Horizontais:**

Para separar partes de um texto com linhas horizontais que atravessam a largura da página, utiliza-se a tag **<hr>**, que significa *horizontal rule* (linha horizontal). Por ser considerada uma tag órfã, ela deve ser escrita com a barra **<hr />**

- **Parágrafos:**

Para se definir parágrafos, é necessário usar as tags **<p>** e **</p>** (*paragraph*) para cada parágrafo desejado. O navegador vai sempre avançar uma linha em branco, posicionando-se na segunda linha seguinte ao comando.

Exemplo:

```
<p>HTML: linguagem de marcação  
utilizada na construção de páginas  
Web.</p>  
<hr />  
<p>Documentos HTML podem ser  
interpretados por navegadores.</p>
```

HTML: linguagem de marcação utilizada na
construção de páginas Web.

Documentos HTML podem ser interpretados
por navegadores.

- **Quebras de Linha:**

Usa-se a tag **
** (*break*) quando for preciso quebrar uma linha em um parágrafo. A partir do ponto de definição de **
** o texto é deslocado para a próxima linha. A tag **
** é órfã, pois não possui comando de finalização, portanto, deve ser escrita a barra de finalização. O comando **
** avança (quebra) para a linha imediatamente após àquela em que se encontra, sem deixar uma linha em branco adicional.

Exemplo:

```
H <br /> T <br /> M <br /> L <br /> <br /> Versão 5
```

H
T
M
L

Versão 5

- **Texto Predefinido:**

Para apresentar o texto exatamente na forma em que ele foi digitado, usa-se a tag **<pre>** e **</pre>** (*predefined*). Com esta tag serão mantidos os espaços e as tabulações do texto original.

Exemplo:

<pre> <pre> Com esta tag, o texto aparece no Navegador exa- tamente como foi digitado: H T M L </pre> </pre>	<pre> Com esta tag, o texto aparece no Navegador exa- tamente como foi digitado: H T M L </pre>
--	--

- **Lista Não Ordenada:**

A tag **** em conjunto com a tag ****, é usada quando há necessidade de apresentar uma lista de dados não ordenada. As tags **** e **** definem o bloco do documento que será usado para apresentar elementos como tópicos, e as tags **** e **** são usadas para definir o texto que será apresentado.

Exemplo:

<pre> Cores: Amarelo Azul Celeste Marinho Royal Verde Vermelho </pre>	<pre> Cores: • Amarelo • Azul ◦ Celeste ◦ Marinho ◦ Royal • Verde • Vermelho </pre>
---	---

- **Lista Ordenada:**

A tag **** em conjunto com a tag ****, é usada quando há necessidade de apresentar uma lista de dados ordenada. As tags **** e **** indicam o trecho do documento que será usado para apresentar os elementos como tópicos, e as tags **** e **** são usadas para definir o texto que será apresentado. Pode-se alterar os marcadores, utilizando no parâmetro type da tag **** as opções: "decimal" ou 1, "upper-alpha" ou "A", "lower-alpha" ou "a", "upper-roman" ou "I" e "lower-roman" ou "i".

Exemplo:

<pre> Navegadores mais usados: <ol type="1"> Chrome Internet Explorer Firefox Edge Safari </pre>	<pre> Navegadores mais usados: 1. Chrome 2. Internet Explorer 3. Firefox 4. Edge 5. Safari </pre>
---	---

- **Lista de Definições:**

A tag **<dl>** em conjunto com as tags **<dt>** ou **<dd>**, são usadas quando há necessidade de apresentar uma lista de definições, tipo um glossário. As tags **<dl>** e **</dl>** indicam o trecho do documento que será usado para apresentar os tópicos, e as tags **<dt>** e **</dt>** ou **<dd>** e **</dd>** são usadas para definir o texto que será apresentado.

</dd> são usadas, respectivamente para apresentar as palavras ou termos a serem explicados (<dt> e </dt>) e a explicação da palavra ou termo indicado (<dd> e </dd>).

Exemplo:

Glossário:

```
<dl>
  <dt>URL</dt>
  <dd>Uniform Resource Locator - endereço Web no qual se encontra algum recurso.</dd>
  <dt>W3C</dt>
  <dd>World Wide Web Consortium - consórcio mundial fundado em 1994, responsável pela criação das normas que regem a Web.</dd>
  <dt>XHTML</dt>
  <dd>eXtensible Hypertext Markup Language - reformulação da linguagem de marcação HTML, baseada em XML.</dd>
  <dt>XML</dt>
  <dd>eXtensible Markup Language - recomendação da W3C para gerar linguagens de marcação para necessidades especiais.</dd>
</dl>
```

Glossário:

URL

Uniform Resource Locator - endereço Web no qual se encontra algum recurso.

W3C

World Wide Web Consortium - consórcio mundial fundado em 1994, responsável pela criação das normas que regem a Web.

XHTML

eXtensible Hypertext Markup Language - reformulação da linguagem de marcação HTML, baseada em XML.

XML

eXtensible Markup Language - recomendação da W3C para gerar linguagens de marcação para necessidades especiais.

- **Citações:**

A tag <q> e </q> (*quotation*) é usada para definir uma citação curta. Os navegadores costumam inserir aspas em torno do texto delimitado pela tag <q>.

Exemplo:

A missão do W3C é: <q>levar a World Wide Web ao seu potencial máximo, desenvolvendo protocolos e diretrizes que garantam seu crescimento a longo prazo da web.</q>

A missão do W3C é: "levar a World Wide Web ao seu potencial máximo, desenvolvendo protocolos e diretrizes que garantam seu crescimento a longo prazo da web."

A tag <blockquote> e </blockquote> é usada para definir um bloco de citação de uma fonte externa. Os navegadores costumam recuar os blocos de textos delimitados pela tag <blockquote>.

Exemplo:

Com relação à Web Semântica, o W3C afirma:

<blockquote cite="http://http://www.w3c.br/Padroes/WebSemantica"> O termo <q>Web Semântica</q> refere-se à visão do W3C da Web dos Dados Conectados. A Web Semântica dá às pessoas a capacidade de criarem repositórios de dados na Web, construírem vocabulários e escreverem regras para interoperarem com esses dados.</blockquote>

Com relação à Web Semântica, o W3C afirma:

O termo "Web Semântica" refere-se à visão do W3C da Web dos Dados Conectados. A Web Semântica dá às pessoas a capacidade de criarem repositórios de dados na Web, construírem vocabulários e escreverem regras para interoperarem com esses dados.

A tag **<abbr>** define uma abreviatura ou um acrônimo. As abreviaturas podem fornecer informações úteis para navegadores, sistemas de tradução e mecanismos de pesquisa. O atributo **title** pode ser usado na tag **<abbr>** para mostrar a versão completa da abreviatura / sigla quando se passa o mouse sobre o elemento.

Exemplo:

```
O <abbr title="Consórcio World Wide Web">W3C</abbr> foi fundado em 1994.
```

O *W3C* foi fundado em 1994.

A tag **<address>** define informações de contato do autor de um documento ou de um artigo. O texto delimitado por esta tag geralmente é exibido em itálico. A maioria dos navegadores adiciona uma quebra de linha antes e depois do elemento.

Exemplo:

```
<address>
  Escrito por W3C Brasil. <br />
  Contate-nos: <br />
  http://www.w3c.br <br />
  Av. das Nações Unidas, 11.541, 2º andar <br />
  São Paulo, SP - Brasil
</address>
```

*Escrito por W3C Brasil.
Contate-nos:
http://www.w3c.br
Av. das Nações Unidas, 11.541, 2º andar
São Paulo, SP - Brasil*

A tag **<cite>** define o título de uma obra. Os navegadores geralmente exibem elementos **<cite>** em itálico.

Exemplo:

```
<cite>Weaving The Web</cite> de Tim Berners-Lee. Publicado em 1999.
```

Weaving The Web de Tim Berners-Lee. Publicado em 1999.

A tag **<bdo>** (*bi-directional override*) é usada para substituir a direção de texto atual. Esta tag deve ser usada com o atributo **dir**, que permite dois valores: "ltr" (*left to right*) ou "rtl" (*right to left*).

Exemplo:

```
<bdo dir="rtl">Este texto é escrito da direita para a esquerda</bdo>
```

adreuqse a arap atierid ad otircse é otxet etsE

• Links:

O uso e a definição de links (hipertextos) são obtidos por meio das tags **<a>** e **** (*anchor*), que estabelecem os pontos de ancoragem e de ligação internos ou externos.

A partir de um link externo é possível "pular" para outro documento.

O parâmetro **href=""** especifica a localização e o nome do arquivo. Se os arquivos de origem e destino estiverem armazenados no mesmo diretório, basta informar o nome do arquivo html de destino entre as aspas. Se o link for para um site disponível na Internet, basta colocar o endereço do site entre as aspas.

Para criar um link para um arquivo localizado em outro diretório (pasta) é necessário indicar o caminho para o arquivo. Regras de diretórios:

- O ponto de partida para localizar um arquivo é o diretório atual;

- A barra utilizada para separar diretórios é a barra convencional (/);
- Para subir um nível, você deve utilizar os sinais "../". Estes sinais devem ser usados para cada nível que for necessário subir

Sintaxe:

Se for necessário acessar um diretório:

```
<a href="diretório/arquivo.extensão">Texto do link</a>
```

Se for necessário subir um nível de diretório:

```
<a href = "../diretório/arquivo.extensão">Texto do link</a>
```

O texto que fica envolvido pelos dois comandos <a> e aparecerá sublinhado e na cor azul, indicando que é um link.

Nota: Um link não precisa ser um texto. Pode ser uma imagem ou qualquer outro elemento HTML. Basta adicionar a imagem ou elemento desejado entre a abertura e o fechamento da tag que cria o link <a>.... Exemplo:

```
<a href="Imagens/logo.png"></a>
```

Usa-se o parâmetro **target="_blank"** da tag <a> para abrir o link em uma nova janela ou, dependendo do browser, em uma nova aba. O parâmetro **target="_self"** é utilizado para abrir o link na mesma janela, porém, é opcional.

O ponto de ligação local (**link interno**) é utilizado quando um documento possui tópicos importantes ou é muito longo. Ao clicar sobre este tipo de link, ocorre uma rolagem na página até o ponto que foi definido como destino. Deve-se definir o destino, usando o parâmetro **id="nome_inventado"**. Portanto, é necessário adicionar o parâmetro id ao elemento HTML que será o destino do link interno. Entre as aspas, inventar um nome simples e curto (sem caractere especial e sem espaço) para identificar o local de destino. Não pode haver mais de um elemento na página com o mesmo id.

Para criar o link de ligação local, o parâmetro **href** da origem deve usar o caractere # antes do id que foi definido como destino: **href="#nome_inventado"**.

Se o destino do link interno for o início da página, é possível utilizar a palavra reservada **"top"** no link de origem, após o caractere #, conforme exemplo abaixo:

```
<a href="#top">Início da Página</a>
```

Importante: Para um **link interno** funcionar corretamente, é obrigatório que haja conteúdo suficiente de rolagem abaixo do elemento de destino. Caso contrário, o elemento de destino não será posicionado na parte superior da tela do navegador (vai ficar na parte inferior). Portanto, no geral, o link interno só irá funcionar corretamente após adicionar o conteúdo de toda a página. Se não houver conteúdo suficiente após o elemento de destino, adicionar linhas em branco ao final do conteúdo da página, conforme exemplo de link interno mostrado mais abaixo.

Exemplos:

Links Externos:

```
<a href="www.google.com" target="_blank">Google</a>
<br />
<a href="Detalhes.html">Detalhamento</a>
<br />
<a href="Index.html">Voltar</a>
<br />
<a href="Imagens/logo.png">Logotipo</a>
```

[Google](http://www.google.com)
[Detalhamento](#)
[Voltar](#)
[Logotipo](#)

Link Interno:

```
<a href="#resumo">Ir para o Resumo</a>
<br /><br />
...
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />

<h2 id="resumo">Resumo do Capítulo</h2>
...
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
<br /><br /><br /><br /><br /><br /><br /><br />
```

[Ir para o Resumo](#resumo)

Resumo do Capítulo

- **Link para Enviar E-mail:**

No parâmetro **href=""** da tag **<a>**, usa-se **mailto:** antes do endereço de e-mail que será indicado como destino.

Para enviar mensagens para mais de um destinatário, basta separar os endereços de e-mail por vírgulas.

É possível preencher previamente o campo "Assunto" ou "Subject" da mensagem, acrescentando após o(s) endereço(s) do e-mail a variável "?subject=" seguida do texto que se deseja usar como assunto (o assunto não precisa estar entre aspas).

Para preencher o campo CC (com cópia), basta adicionar ao final "&cc=" e o endereço de e-mail do destinatário.

Sintaxe:

```
<a href="mailto:endereço1,endereço2?subject=assunto&cc=endereço3">Texto/Link</a>
```

Exemplo:

```
<a href="mailto:email@loja.com.br,contato@empresa.com?subject=Sugestão&cc=cliente@fornecedor.br">E-mail para a Empresa</a>
```

[E-mail para a Empresa](mailto:email@loja.com.br,contato@empresa.com?subject=Sugestão&cc=cliente@fornecedor.br)

- **Imagens**

Para inserir imagens, deve-se utilizar a tag órfã **** em conjunto com os atributos:

- **src:** identifica o nome do arquivo e o local em que se encontra armazenado (endereço). É obrigatório;
- **alt:** adiciona um texto alternativo explicando o que é a imagem para o usuário. Este recurso é útil para as situações em que a imagem não pode ser visualizada por conta de conexão lenta, erro no atributo "src" ou quando se utiliza um leitor de tela. Desta forma, este atributo também auxilia as pessoas com deficiência visual. É obrigatório.
- **title:** adiciona um texto explicativo (descrição) que aparece quando o mouse passa sobre a imagem;
- **width:** permite definir a largura (em pixels);
- **height:** permite definir a altura (em pixels);

Se apenas uma dimensão (largura ou altura) for definida em uma imagem, a proporção da imagem será preservada.

Para fazer uso da imagem como hiperlink é só definir a tag **** entre as tags **<a>** e ****.

Exemplo:

```

```



Legenda de Figuras:

O elemento **<figure>** pode ser utilizado (não é obrigatório) para marcar uma figura em um documento em conjunto com um elemento **<figcaption>** para definir a legenda da respectiva figura. Um elemento **<figure>** deve ser criado para cada imagem/figura que se deseja adicionar a legenda.

Exemplo:

```
<figure>
  
  <figcaption>Logotipo da Linguagem HTML versão 5</figcaption>
</figure>
```



• Vídeos

As tags **<video>** e **</video>** permitem incluir vídeos numa página Web. Em HTML5, há três formatos de vídeos suportados por este elemento: MP4, WebM e Ogg. Atributos:

- **src:** identifica o endereço do arquivo;
- **width:** permite definir a largura (em pixels);
- **height:** permite definir a altura (em pixels);
- **controls:** adiciona ao vídeo controles como os de "play", "pause" e "volume";
- **autoplay:** permite iniciar um vídeo automaticamente assim que ele estiver pronto/carregado. Se este atributo for omitido, o vídeo não será executado automaticamente. Este atributo não funciona em equipamentos móveis, como iPads e iPhones;
- **loop:** indica que o vídeo deverá ser reiniciado cada vez que ele finalizar;
- **muted:** indica que o áudio do vídeo estará mudo ao ser carregado;
- **poster:** especifica o endereço de uma imagem para ser exibida enquanto é realizado o download do vídeo ou até o usuário apertar o "play". As dimensões da imagem serão ajustadas às dimensões do vídeo, independentemente do tamanho original da imagem;
- **preload:** indica ao navegador se o vídeo deve ou não ser carregado junto com a página. Este atributo é ignorado se a opção **autoplay** for utilizada. Valores permitidos: **"none"** – o vídeo não deve ser carregado, **"auto"** – o navegador deve carregar todo o vídeo junto com a página ou **"metadata"** – apenas metadados (dimensões, duração, lista de faixas, etc.) devem ser carregados;

Entre a abertura e o fechamento das tags é possível utilizar o elemento **source** para relacionar os endereços de formatos alternativos do arquivo de vídeo. Este recurso é importante considerando a incompatibilidade que pode ocorrer entre os diferentes navegadores e os formatos dos vídeos. O navegador irá usar o primeiro formato que for reconhecido/suportado. O elemento **<source ... >** funciona como uma nova tag dentro da tag **<video>** e deve ser acompanhado do atributo **src**. Quando o elemento **source** é utilizado, é possível incluir o

atributo **type** para indicar o tipo de mídia utilizada (*MIME-type*). Valores possíveis: "video/mp4", "video/webm" e "video/ogg". Exemplo:

```
<video ... >
```

```
<source src="..." type="video/mp4" >
```

```
<source src="..." type="video/webm" >
```

```
...
```

O seu navegador não suporta a tag video...

```
</video>
```

Caso o navegador não suporte a tag **<video>**, o texto ou imagem que for adicionada entre a abertura e o fechamento das tags será exibido.

Exemplo:

```
<video src="Videos/IFSP.mp4" width="320"
height="240" controls></video>
```



• Áudios

As tags **<audio>** e **</audio>** funcionam de forma semelhante à tag de vídeo. Em HTML5, há três formatos de áudio suportados por este elemento: MP3, Wav e Ogg. Obs.: O navegador Internet Explorer suporta somente o formato MP3.

A relação de atributos é igual à tag **<video>**, exceto os atributos "poster" e as definições de largura ("width") e altura ("height") que não são utilizados.

O elemento **source** também segue as mesmas definições da tag **<video>**, sendo que os valores possíveis para o tipo de mídia (*MIME-type*) são: "audio/mp3", "audio/wav" e "audio/ogg".

Exemplo:

```
<audio src="Audios/Musica.mp3" controls></audio>
```



As tags **<audio>** e **<video>** são tags de mídia que foram criadas, entre outros motivos, para acabar com a dependência de plug-ins para a reprodução de alguns formatos de áudio e vídeo nas páginas. Dessa forma, nos sites que exigiam a instalação do Flash, do Media Player ou do Quick Time, por exemplo, basta que sejam implementadas essas tags e que o navegador seja compatível com HTML5. Sendo assim, áudios e vídeos podem ser manipulados de forma nativa, da mesma forma que ocorre com as imagens.

• Recursos Externos:

Para vincular à página documentos ou recursos de diversos tipos, deve-se utilizar a tag **<embed />** em conjunto com os atributos:

- **src:** identifica o endereço do arquivo ou recurso;
- **type:** atributo opcional que indica o *MIME type* do recurso. O termo *MIME type*, também conhecido como *Internet Media Type*, é usado para indicar o tipo de mídia que será transmitido/apresentado. Os navegadores geralmente usam o *MIME type* para determinar qual ação deve ser executada quando um recurso é carregado (qual plug-in utilizar). A estrutura de um *MIME type* consiste de um tipo e um subtipo, geralmente escritos em letras minúsculas, separados por uma '/'. Nenhum espaço é permitido. Tipos e subtipos (*MIME types*) mais comuns:

Texto	text/html, text/css, text/javascript, text/plain
--------------	--

Imagem	image/gif, image/png, image/jpeg, image/bmp, image/webp
Áudio	audio/midi, audio/mpeg3, audio/webm, audio/ogg, audio/wav
Vídeo	video/mp4, video/webm, video/mpeg, video/avi, video/quicktime
Aplicação (binários)	application/pdf, application/xml, application/octet-stream, application/vnd.ms-powerpoint, application/xhtml+xml, application/zip, application/x-shockwave-flash

- **width:** permite definir a largura (em pixels);
- **height:** permite definir a altura (em pixels);
- **autoplay:** para arquivos de áudio e vídeo é possível definir se o arquivo vai executar assim que o usuário abrir a página ou só quando apertar o botão play. Valores possíveis: "true" ou "false". Obs.: Alguns navegadores somente entendem valores 1 ou 0 para estes atributos. Existe uma variação deste atributo que funciona de forma semelhante: autostart. Alguns navegadores podem ignorar estes atributos e sempre iniciar os arquivos automaticamente ao carregar a página.

Para vincular um vídeo do YouTube ou incorporar outro site HTML à página atual é possível utilizar as tags **<iframe>** e **</iframe>** em conjunto com os atributos:

- **src:** identifica o endereço (URL) que será incorporado;
- **width:** permite definir a largura (em pixels);
- **height:** permite definir a altura (em pixels);

Caso o navegador não suporte a tag **<iframe>**, o texto ou imagem que for adicionada entre a abertura e o fechamento das tags será exibido.

Para vincular um vídeo do YouTube à página é necessário adaptar o endereço do vídeo que será informado no atributo **src**. Deve-se incluir no endereço (URL) o parâmetro "**embed**" para que o player do YouTube seja incorporado à página:

<https://www.youtube.com/embed/xxxxxxxxxxxxxx>

Para iniciar o vídeo do YouTube automaticamente, deve-se adicionar o parâmetro "**?autoplay=1**" ao final do endereço (URL):

<https://www.youtube.com/embed/xxxxxxxxxxxxxx?autoplay=1>

Se desejar que o vídeo do YouTube seja repetido ao final, deve-se utilizar o parâmetro "**&loop=1**" ao final do endereço (URL):

<https://www.youtube.com/embed/xxxxxxxxxxxxxx&loop=1>

O comportamento dos navegadores varia de acordo com os tipos de recursos que serão vinculados e as versões/configurações/plug-ins disponíveis em cada máquina.

Apesar de o HTML5 defender uma estrutura padrão para reproduzir mídias, as próprias mídias não são padronizadas em todos os navegadores. Na prática, isso significa que é possível que alguns visitantes não consigam rodar alguns arquivos de vídeo (ou áudio). Isso ocorre porque os navegadores suportam tipos de vídeos e áudios diferentes. Alguns suportam vídeo WebM, por exemplo, enquanto outros suportam MPEG. Um exemplo é o caso do formato MOV que necessita do QuickTime e que somente é suportado pelo Internet Explorer.

Exemplos:

```
<embed src="https://www.youtube.com/embed/1Q09kEf2Aik"
width="160" height="120" />
```



```
<iframe src="http://ifsp.edu.br" width="260" height="180">
</iframe>
```



Tabelas:

Segundo o consórcio W3C, as tabelas devem ser usadas apenas para dispor e apresentar dados. Em uma tabela, deve-se usar as tags **<table>** e **</table>**, as quais delimitam a área de início e fim de uma tabela. Após a definição da área de apresentação de uma tabela, é necessário um conjunto específico de tags:

- **<caption> ... </caption>**: definem a legenda superior de uma tabela. Entre a legenda e a tabela é incluída uma linha em branco.
- **<tr> ... </tr>**: indicam o início e o fim de uma linha que fará parte de uma tabela. Devem ser incluídas em cada linha que compõe a tabela em uso.
- **<th> ... </th>**: células com destaque - determinam o conteúdo da célula utilizada como título de uma linha ou coluna. O texto do título será formatado em negrito e centralizado.
- **<td> ... </td>**: definem a estrutura de células que conterá dados (conteúdo) de uma tabela, devendo ser utilizado um conjunto dessas tags para cada célula de uma linha.

É possível expandir células usando os parâmetros **colspan** (para ocupar mais de uma coluna) e **rowspan** (para ocupar mais de uma linha) nas tags **<td>** e **<th>**. A quantidade de linhas ou colunas que serão mescladas deve ser definida após o símbolo de igual e entre aspas.

O atributo **border** pode ser usado na tag **<table>** para apresentar as linhas de borda em uma tabela. Como este atributo adiciona um estilo à tabela, é recomendado que este atributo seja explorado apenas nas folhas de estilo (CSS).

Também é possível adicionar tags dentro das células das tabelas para incluir listas, figuras, parágrafos e outras marcações. Inclusive, é possível incluir tabelas dentro de outras.

Ainda existem as tags **<thead>**, **<tfoot>** e **<tbody>**, que servem para agrupar linhas da tabela. Vale ressaltar que dentro do grupo **<thead>** deve-se ter apenas linhas contendo a tag **<th>** como célula.

Outra tag de agrupamento que permite que sejam definidas as colunas é a tag **<colgroup>**. Dentro dessa tag é possível definir uma tag **<col>** para cada coluna e adicionar alguns atributos ou propriedades que influenciarão todas as células daquela coluna. Sendo assim, a tag **<col>** é útil para aplicar estilos a colunas inteiras, em vez de repetir os estilos para cada célula e para cada linha.

Exemplo:

```
<table border>
  <caption>Domicílios que possuem equipamentos</caption>
  <tr>
    <th>Ano</th><th>Equipamento</th><th>%</th>
  </tr>
  <tr><th rowspan=4>2015</th><td>Televisão</td><td>97%</td></tr>
  <tr>
    <td>Celular</td><td>93%</td></tr>
  <tr>
    <td>Computador Portátil</td><td>32%</td></tr>
  <tr>
    <td>Computador Mesa</td><td>25%</td></tr>
  <tr><th rowspan=4>2014</th><td>Televisão</td><td>98%</td></tr>
  <tr>
    <td>Celular</td><td>92%</td></tr>
  <tr>
    <td>Computador Portátil</td><td>30%</td></tr>
  <tr>
    <td>Computador Mesa</td><td>28%</td></tr>
  <tr><th rowspan=4>2013</th><td>Televisão</td><td>98%</td></tr>
  <tr>
    <td>Celular</td><td>90%</td></tr>
  <tr>
    <td>Computador Portátil</td><td>28%</td></tr>
  <tr>
    <td>Computador Mesa</td><td>31%</td></tr>
</table>
```

**Domicílios que possuem
equipamentos**

Ano	Equipamento	%
2015	Televisão	97%
	Celular	93%
	Computador Portátil	32%
	Computador Mesa	25%
2014	Televisão	98%
	Celular	92%
	Computador Portátil	30%
	Computador Mesa	28%
2013	Televisão	98%
	Celular	90%
	Computador Portátil	28%
	Computador Mesa	31%

HTML - Formulários

As tags para criação de formulários são **<form>** e **</form>**, as quais possibilitam iniciar e terminar um formulário. Seus atributos são:

- **action**: informa o endereço (URL) para o qual o conteúdo do formulário será enviado. Se esse atributo for omitido, o endereço corrente será usado.
- **method**: informa o método de transferência dos dados que será usado pelo atributo "action", sendo possível adotar os valores:
 - **get**: valor padrão do atributo "action", usado quando se deseja enviar os dados de um formulário para um endereço (URL de um servidor), seguindo o formato "endereço?nome=ação". Esse tipo de ação apresenta os dados do formulário na barra de endereços do programa de navegação.
 - **post**: faz com que os dados do formulário sejam enviados para o servidor definido como o atributo "action" em um bloco de dados. Este é o método mais indicado, uma vez que os dados não são apresentados na barra de endereço do navegador.
- **enctype**: especifica a forma de submissão dos dados de um formulário quando se usa o valor "post" no atributo "method". O valor padrão desse atributo é "application/x-www-form-urlencoded" (dados do formulário são passados de forma codificada). Caso seja usado com o valor "file" do atributo "type" (será mostrado mais adiante), o valor desse atributo deve ser "multipart/form-data" (dados do formulário são passados em multipartes). É muito comum o uso do valor "text/plain" para envio de dados de um formulário de maneira formatada.

Para criar formulários são necessárias diversas tags para a apresentação dos objetos:

- **<label>**: apresenta o texto de identificação (rótulo) de um campo. Pode-se usar os atributos:
 - **for**: o nome interno associado ao atributo "id" de identificação de um campo da tag **<input>** (será apresentado em seguida).
 - **title**: o texto explicativo que será apresentado quando o ponteiro do mouse permanecer sobre esse elemento por mais de um segundo. Esse atributo é opcional.
- **<input />** (tag órfã): utilizada para criar campos de entrada de dados em um formulário. É o recurso mais comum e mais utilizado. É possível usar os seguintes atributos:
 - **name**: estabelece o nome do elemento (campo, área de texto ou botão) no formulário. Obrigatório quando há mais de um elemento do mesmo tipo.
 - **id**: nome de identificação associado ao atributo "for" da tag **<label>**.
 - **type**: tipo de entrada de dados que o campo vai receber. Aceita os valores:
 - **text**: formato caixa de texto, utilizado para receber dados do tipo caractere.
 - **password**: utilizado exclusivamente para campos de senha.
 - **radio**: tem a forma de um botão circular, utilizado para selecionar apenas uma opção numa lista de opções predefinidas.
 - **checkbox**: tem a forma de um botão quadrado, utilizado para fazer a seleção de mais de uma opção numa lista de opções predefinidas.
 - **submit**: apresenta um botão que terá como objetivo enviar os dados informados no formulário para um endereço. A ação desse botão deve estar associada ao atributo "value" (será apresentado em seguida).
 - **reset**: apresenta um botão de ação que serve para apagar os dados de um formulário, voltando seus campos para os valores padrão. A ação desse botão deve estar associada ao atributo "value" (será apresentado em seguida).
 - **hidden**: cria um campo invisível cujo conteúdo é enviado com os dados de um formulário, mas não é apresentado para o usuário.
 - **file**: possibilita anexar um arquivo para ser enviado como upload. Para tanto, é necessário que a tag **<form>** tenha os atributos **method="post"** e **enctype="multipart/form-data"**. Para este tipo de campo é possível usar o atributo **accept** para especificar os tipos de arquivos que serão aceitos. Os valores possíveis para o atributo **accept** são: a extensão do arquivo (exemplo: ".doc", ".jpg", ".pdf", etc), o tipo da mídia (*media_type*) ou os termos: "audio/*" (todos os arquivos de áudio); "video/*" (todos os arquivos de vídeo); e "image/*" (todos os arquivos de imagem).
 - **image**: permite que uma imagem seja usada como um botão do tipo "submit". Para indicar a figura que será utilizada, adicionar o atributo "src". O clique do mouse é realizado sobre a imagem. Ao enviar os dados de um formulário com "input image",

também são enviadas as coordenadas da posição do clique. Quando este tipo é usado, é obrigatório utilizar também o atributo "alt" para indicar o texto explicativo da imagem.

- **color:** usado para selecionar e armazenar cores (não funciona em todos os navegadores).
- **number:** usado para selecionar números (não funciona em todos os navegadores).
- **date / datetime-local / time / month / week:** usado para selecionar datas / horas (não funciona em todos os navegadores).
- **range:** Define um controle para inserir um número cujo valor exato não é importante (apresenta um controle deslizante). O intervalo padrão é de 0 a 100. No entanto, é possível definir restrições sobre quais números são aceitos com os atributos min, max e step.
- **tel:** usado para campos que devem conter números de telefone (não funciona em todos os navegadores).
- **url:** usado para campos que devem conter endereços URL. Dependendo do suporte do navegador, o campo pode ser validado automaticamente quando enviado (não funciona em todos os navegadores).
- **size:** determina o tamanho do campo apresentado no formulário. Geralmente utilizado com os valores "text" ou "password". Na ausência dessa informação é criado um campo com tamanho padrão médio de 20 caracteres, dependendo do tipo de fonte utilizada.
- **maxlength:** define o tamanho máximo de caracteres a ser informado no campo. Usado em campos "text" ou "password". Quando omitido, o campo aceita um texto de qualquer tamanho, que se desloca para a esquerda caso seja maior que a caixa.
- **min:** indica o valor mínimo para o campo.
- **max:** indica o valor máximo para o campo.
- **step:** indica o intervalo de valores válidos para o campo. Exemplo: se step="3", os números válidos podem ser -3, 0, 3, 6, etc.
- **value:** indica o valor padrão do conteúdo do campo.
- **autocomplete:** Especifica se o elemento deve ter o preenchimento automático ativado. Valores: "on" ou "off". O preenchimento automático permite que o navegador exiba opções para preencher o campo, com base em valores digitados anteriormente.
- **placeholder:** Apresenta uma dica para o valor esperado no campo.
- **autofocus:** Quando presente, este atributo determina que o elemento deve receber o foco quando a página for carregada.
- **checked:** Usado em campos de "radio" e "checkbox" quando há necessidade de manter uma das opções pré-selecionadas. Deve estar associado ao valor "checked", resultando em checked="checked".
- **disabled:** Indica que o elemento deve aparecer desabilitado. Um elemento desabilitado não pode ser preenchido ou clicado. Elementos desabilitados não são submetidos.
- **readonly:** Indica que o campo é "somente leitura". Quando o campo é somente leitura, ele não puder ser modificado (no entanto, é possível selecionar e copiar o seu conteúdo).
- **required:** Especifica que o campo deve ser preenchido antes de submeter o formulário.
- **pattern:** Especifica uma expressão usada para validar um campo. Exemplo: código formado por três letras maiúsculas seguidas por quatro dígitos: pattern="[A-Z]{3}[0-9]{4}". O uso de pattern garante que o valor corresponda ao formato correto para o código.
- **accesskey:** associa uma tecla de atalho ao elemento. Quando a tecla de atalho é acionada, o controle recebe o foco. É possível estabelecer o uso desse recurso com as tags **<button>**, **<input>** e **<textarea>**.
- **list:** associa uma lista de valores predefinidos para um elemento do tipo **<input>**. Este atributo deve fazer referência ao valor do atributo id de um elemento do tipo **<datalist>**.
- **<datalist>**: especifica uma lista de valores predefinidos para um elemento do tipo **<input>**.
Atributo:
 - **id:** nome de identificação da lista.A tag **<datalist>** deve ser usada em conjunto com as tags **<option></option>**. Estas tags devem ser criadas para cada um dos itens da lista. Os valores devem ser inseridos entre a abertura e o fechamento de cada tag. A tag **<option>** permite o seguinte atributo:
 - **value:** o valor interno da opção. Esse recurso é opcional e, quando omitido, usa como valor o texto indicado entre a abertura e o fechamento da tag **<option>**.
- **<select>**: permite criar uma lista de opções predefinidas. Aceita os atributos:

- **name:** nome da variável que vai receber o conteúdo do campo.
- **size:** número de linhas que serão exibidas na tela. Caso a lista seja maior, uma barra de rolagem surge automaticamente.
- **multiple:** permite selecionar mais de um item da lista. Na ausência do atributo "size", o padrão para múltiplas escolhas será de quatro linhas na tela. Seu uso é opcional. Em caso de múltiplas escolhas, para os itens na sequência, o usuário deve clicar em uma das opções e arrastar o ponteiro do mouse; para os itens não sequenciais, clicar em uma opção, pressionar a tecla <Ctrl> (Control) e com o ponteiro do mouse clicar nas outras opções. Deve ser utilizado com o valor "multiple", resultando em multiple="multiple".

Para auxiliar na funcionalidade da tag **<select>**, deve-se usar as tags **<option></option>** e **<optgroup></optgroup>**:

A tag **<option>** permite criar os itens da lista de opções. Utiliza os seguintes atributos:

- **selected:** escolhe a opção que ficará como pré-selecionada na lista apresentada. Deve ser utilizado com o valor "selected", resultando em selected="selected".
- **value:** o texto que identifica uma determinada opção. Esse recurso é opcional e, quando omitido, usa como valor o texto indicado após a tag <option>.

A tag **<optgroup>** é utilizada quando se deseja agrupar as opções definidas com a tag <option>. Em uma lista de seleção, os itens agrupados aparecem um nível a frente dos demais.

- **<textarea>**: delimita uma área de texto (campo) com várias linhas e colunas, para que o usuário possa escrever um texto a ser enviado. A inclusão opcional de um texto inicial na caixa se faz com uma informação escrita entre as tags <textarea> e </textarea>. Atributos:
 - **name:** nome do campo.
 - **rows:** a altura da caixa, determinada pelo número de linhas.
 - **cols:** a largura da caixa, indicada pelo número de colunas.
 - **wrap:** determina por meio dos valores "on" (valor padrão) ou "off" se o texto escrito será ou não quebrado.
 - **placeholder:** Apresenta uma dica para o valor esperado no campo.
- **<button>**: permite apresentar botões customizados (definidos pelo desenvolvedor), tendo como atributos:
 - **type:** o tipo de ação que o botão deve executar. Podem ser aceitos os valores:
 - **submit:** apresenta um botão customizado com o objetivo de enviar para um endereço os dados que foram informados no formulário.
 - **reset:** apresenta um botão customizado que serve para apagar os dados de um formulário, voltando os campos desse formulário para os valores padrão.
 - **name:** o nome que o botão terá no formulário.
 - **value:** o texto apresentado dentro do botão.
 - **accesskey:** associa uma tecla de atalho ao botão.
- **<fieldset>**: permite delimitar áreas de agrupamento de objetos, como botões "radio" e "checkbox", entre outras possibilidades.

Exemplo:

```
<form method="get">
  Usuário: <input type="text" name="usuario" size="10" maxlength="10"
autofocus />
  <br /><br />
  Nível: <br />
  <input type="radio" name="nivel" value="bas" />Básico <br />
  <input type="radio" name="nivel" value="ava" />Avançado
  <br /><br />
  Período: <select name="periodo">
    <option value="" selected disabled>Escolha... </option>
    <option value="man">Manhã</option>
    <option value="noi">Noite</option>
  </select>
  <br /><br />
  <input type="submit" value="Enviar">
  <input type="reset" value="Limpar">
</form>
```

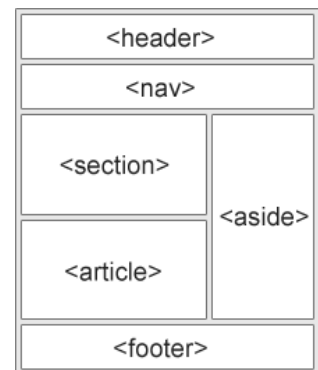
Visual representation of the HTML form code shown on the left. It includes a text input for 'Usuário', radio buttons for 'Nível' (Básico and Avançado), a dropdown menu for 'Período' (Escolha...), and two buttons: 'Enviar' and 'Limpar'.

As tags **<div>** e **</div>** determinam o agrupamento ou divisão de blocos de tags em seções. Permite a aplicação de efeitos de formatação a um determinado bloco de texto.

As tags **** e **** possuem a mesma função das tags **<div>** e **</div>**, com a diferença que a formatação é limitada a um trecho do texto que pode ser uma frase, uma palavra ou uma letra.

O HTML5 incorporou novas tags com significados semânticos. Em vez de agrupar os elementos do cabeçalho em uma div genérica e sem significado, é possível usar uma tag **<header>**, por exemplo, que carrega em si o significado de representar um cabeçalho. Com isso, têm-se um HTML com estrutura baseada no significado de seu conteúdo, o que traz uma série de benefícios, como a facilidade de manutenção e compreensão do documento. Portanto, o HTML5 adicionou novos elementos semânticos para definir diferentes partes de uma página da Web:

<article>	- Define um artigo
<aside>	- Define um conteúdo na lateral da página
<details>	- Define detalhes adicionais que o usuário pode visualizar ou ocultar
<figure>	- Define um conteúdo independente, como ilustrações, diagramas, fotos etc.
<figcaption>	- Define uma legenda para um elemento <figure>
<footer>	- Define um rodapé para um documento ou seção
<header>	- Define um cabeçalho para um documento ou seção
<main>	- Define o conteúdo principal de um documento
<mark>	- Define um texto marcado ou realçado
<nav>	- Define links de navegação
<section>	- Define uma seção em um documento
<summary>	- Define um título visível para um elemento <details>
<time>	- Define uma data/hora



Estas tags adicionadas ao código HTML não modificam a aparência visual da página. Apenas organizam o conteúdo em blocos. Porém, se forem utilizadas em conjunto com a linguagem CSS, permitem que sejam aplicadas formatações de estilos que permite, por exemplo, modificar o posicionamento do bloco na página.

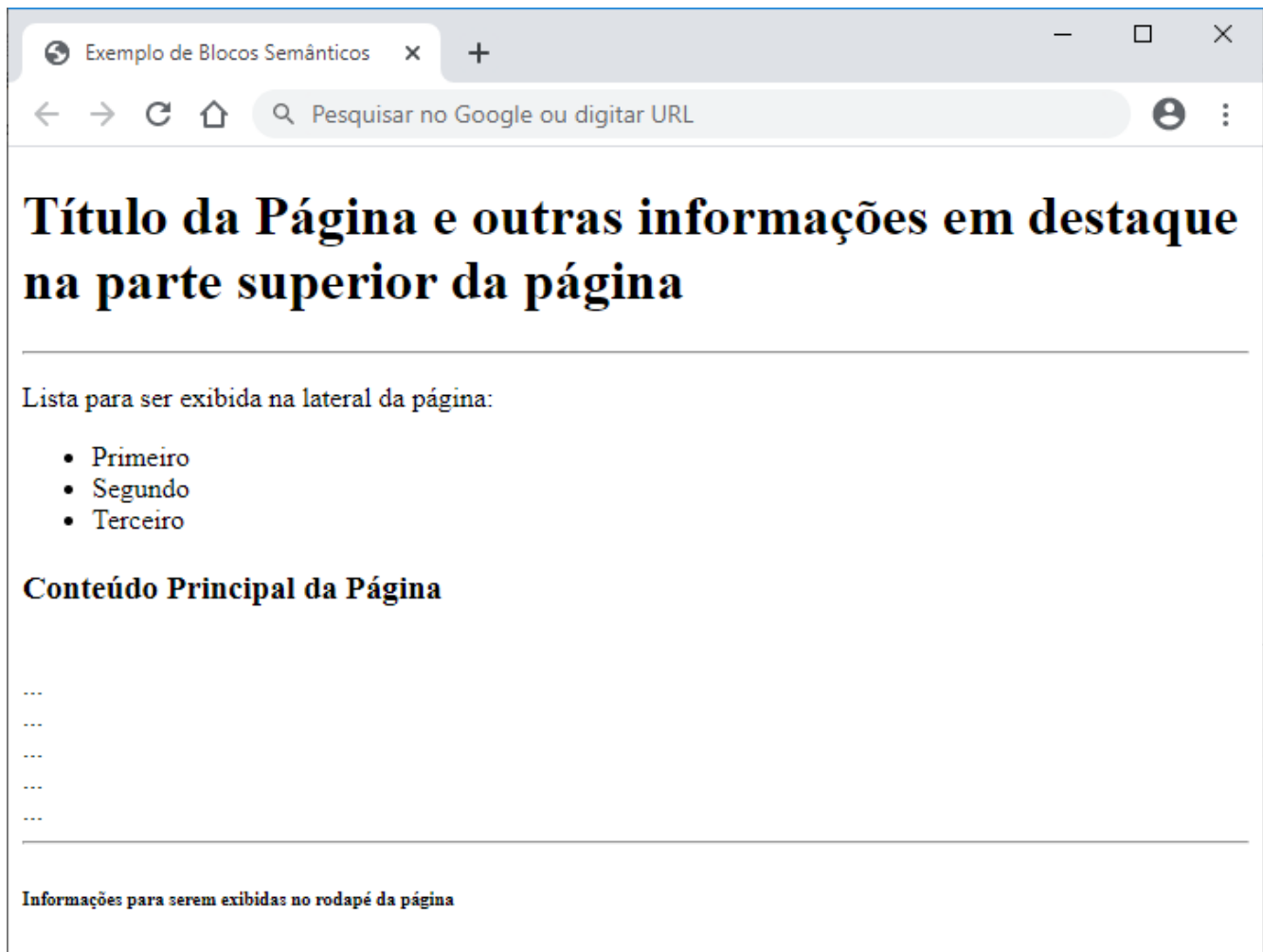
Exemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Exemplo de Blocos Semânticos</title>
  </head>
  <body>
    <header>
      <h1>Título da Página e outras informações em destaque na parte superior da página</h1>
      <hr />
    </header>

    <aside>
      <p>Lista para ser exibida na lateral da página:</p>
      <ul>
        <li>Primeiro</li>
        <li>Segundo</li>
        <li>Terceiro</li>
      </ul>
    </aside>

    <main>
      <h3>Conteúdo Principal da Página</h3>
      <br /> ... <br /> ... <br /> ... <br /> ... <br /> ...
    </main>

    <footer>
      <hr />
      <h6>Informações para serem exibidas no rodapé da página</h6>
    </footer>
  </body>
</html>
```

Em HTML, existem ainda algumas tags que definem o significado estrutural de um bloco de texto. Estes elementos acrescentam informações ao texto, detalhando um conteúdo que deve ser tratado de maneira diferente ou enfatizado (quando esse conteúdo é falado). Apesar de alguns navegadores, a princípio, alterarem o estilo do texto, estas tags não devem ser utilizadas para definir a aparência visual. Aparências visuais e estilos devem ser definidos no CSS.

- <code>** - Código de programação (monoespaçado)
- <dfn>** - Definição de um termo em seu primeiro uso em um documento (itálico)
- ** - Ênfase (itálico)
- <kbd>** - Keyboard Input – o conteúdo é uma entrada do usuário via teclado (monoespaçado)
- <samp>** - Exemplo / Amostra produzido por um programa ou script (monoespaçado)
- ** - Importante (negrito)
- <var>** - Define uma variável (itálico)

HTML – Atributos Globais

Os atributos globais dão significado e contexto aos elementos. Eles podem ser usados em qualquer tag HTML.

- **accesskey**: Especifica uma tecla de atalho para ativar / focar um elemento.
- **class**: Especifica um ou mais nomes de classe para um elemento (refere-se a uma classe no CSS).
- **contenteditable**: Especifica se o conteúdo de um elemento é editável ou não.
- **contextmenu**: Especifica um menu de contexto para um elemento. O menu de contexto aparece quando um usuário clica com o botão direito do mouse no elemento.
- **data-***: Usado para armazenar dados personalizados privados para a página ou aplicativo.
- **dir**: Especifica a direção do texto para o conteúdo de um elemento.
- **draggable**: Especifica se um elemento é arrastável ou não.
- **dropzone**: Especifica se os dados arrastados são copiados, movidos ou vinculados, quando for solto.

- **hidden:** Especifica que um elemento não deve ser exibido. Ele também pode ser usado para evitar que um elemento seja exibido até que alguma outra condição tenha sido atendida. Então, um JavaScript pode remover o atributo hidden e tornar o elemento visível.
- **id:** Especifica uma identificação única para um elemento.
- **lang:** Especifica o idioma do conteúdo do elemento.
- **spellcheck:** Especifica se o elemento deve ter sua ortografia e gramática verificadas ou não.
- **style:** Especifica um estilo CSS "em linha" para um elemento.
- **tabindex:** Especifica a ordem de tabulação de um elemento.
- **title:** Especifica informações adicionais sobre um elemento.
- **translate:** Especifica se o conteúdo de um elemento deve ser traduzido ou não.