HTML

Basicamente no HTML temos dois tipos de tags: as empty tags e as container tags. Elas diferem pelo que representam: as container tags definem a formatação de uma seção do texto ou do próprio documento. É como se um texto de um determinado processador do Windows, fosse selecionada e definida uma característica para essa seção, e todos os caracteres dessa seção passariam a apresentar as características definidas. Isso também acontece com as container tags, marca-se o inicio e o fim da seção, e todo o texto que estiver entre os marcadores apresentará a formatação definida. A tag de finalização é idêntica a tag de inicio, acrescida de uma barra (/). Alguns exemplos de container tags: <B> e</B>, que deixam o texto em negrito; <OL> e </OL>, que englobam uma lista ordenada, entre outros. As container tags podem estar aninhadas.

As empty tags, por sua vez, representam elemento s que não apresentam princípio, meio e fim. Como exemplo temos: <hr>, que desenha uma linha horizontal; <BR>, que determina uma quebra de linha.

Os elementos podem ter atributos ou argumentos, utilizados para midificar alguma característica de sua funcionalidade. Como por exemplo a empty tag utilizada para inserir uma imagem na página (<img>) utiliza o argumento SRC para indicar a figura que será mostrada.

Exemplo:

<HTML>

<head>

<title> minha pagina HTML </title>

<head>

<body>

<h1> exemplo de <i> tags </i></h1>

<hr>

<ol>

<li> item <b><i> numero </i>< /b> 1

<li> item numero 2

</ol>

<img src=”figura\_1.gif”>

</body>

</HTML>

Quando desejamos colocar comentários em um código HTML, usamos os marcadores <!-- e --> tudo o que estiver entre os marcadores será ignorado pelo navegador.

Exemplo de uso:

...

<!-- texto a ser comentado -->

...

Algumas tags

<p> e </p> 🡪 define um parágrafo.

<b> e </b> 🡪 define um texto em negrito.

<i> e </i> 🡪 define um texto em itálico.

<u> </u> 🡪 define um texto sublinhado.

<hr> 🡪 define uma linha horizontal.

<BR> 🡪 define uma quebra de linha.

<ol> e </ol> 🡪 define uma lista ordenada.

<ul> e </ul> 🡪 define uma lista em tópicos.

<li> 🡪 define os elementos de uma lista.

<sub> 🡪 define um texto subescrito.

<sup> 🡪 define um texto sobrescrito.

<Center> e </Center> 🡪 define a centralização de um texto ou figura.

A tag <FONT>

Essa tag pode ser usada para ajustar a cor do texto, o tipo da fonte ou o tamanho do mesmo, de acordo com o atributo utilizado.

Atributos para <FONT>

* Color , usado para especificar a cor do texto, forma de utilização : <font color=”nome da cor”> texto </font>, o nome da cor pode ser informado por extenso em inglês ou através do código da cor .
* Size, usado para alterar o tamanho do texto, forma de utilização: < font size=”inteiro”> texto </font>, o inteiro determina qual o tamanho da fonte .
* Face, determina qual o tipo de fonte a ser usado, forma de utilização: <font face=” tipo da fonte”> texto </font>, por exemplo o tipo da fonte poderia ser “arial” , “verdadna”, etc.

Podemos utilizar vários atributos da tag <font> de uma só vez, isso depende de quais características queremos atribuir para o texto entre as tags de marcação <font> e </font>.

Exemplo.

...

<font color=”blue” face=”verdana” size=”6” > texto que aparecerá na cor azul, fonte verdadna e tamanho 6 </font>

...

Também podemos utilizar o atributo size e color para a tag <hr>, definindo uma espessura e uma cor para a linha horizontal.

Lista ordenada

A tag <ol> e </ol> define uma lista ordenada. Os itens da lista são definidos pela tag <li>.

A tag <ol>, pode utilizar o atributo type e o mesmo pode assumir um dos valores “ A|a|I|i”.

Exemplo

<HTML>

<head>

<title> minha pagina HTML </title>

<head>

<body>

<p> lista ordenada</p>

<ol type=” A”>

<li> primeiro item

<li> Segundo item

...

<li> n item

</ol>

</body>

</html>

Lista de tópicos

A tag <ul> e </ul> define uma lista de tópicos. Os itens da lista são definidos pela tag <li>.

A tag <ul>, pode utilizar o atributo type e o mesmo pode assumir um dos valores “ circle|square|disc ”.

Exemplo

<HTML>

<head>

<title> minha pagina HTML </title>

<head>

<body>

<p> lista de tópicos</p>

<ul type=” disc”>

<li> primeiro item

<li> Segundo item

...

<li> n item

</ul>

</body>

</html>

**Tebelas**

A container tag <table> é utilizada para apresentar dados tabulares. A estrutura e o conteúdo da tabela gerada são definidos por elemento adicionais que pode aparecer apenas dentro das marcações <table> e </table>.

**Título da tabela**

A container tag <caption> especifica o título da tabela. Cada tabela pode ter apenas um título.

Exemplo

...

<caption> notas da primeira avaliação </caption>

...

**O elemento <TH>**

O elemento <th> é usado para especificar as células de cabeçalho da tabela. Essas células são diferentes das outras, pois seu conteúdo aparece geralmente em negrito. O elemento <th> pode ser apresentado sem conteúdo algum: isso corresponde a uma célula em branco.

**O elemento <TD>**

O elemento <td> especifica as células de dados de uma tabela, por ser uma célula de dados, seu conteúdo é apresentado de forma NORMAL, ou seja, sem destaque. Assim como no <td> pode-se construir células em branco.

**O elemento <TR>**

A container tag <TR> indica o fim de uma linha da tabela. Cada linha da tabela pode conter várias células e, portanto é necessário que se faça uso de uma marcação que indique exatamente o ponto de quebra de uma linha e início de outra.

Toda linha deve terminar com um <TR>, com exceção da ultima linha da tabela, que dispensa o uso do <TR> porque o uso da própria marcação de fim de tabela (</table>) torna implícito o fim da linha.

Exemplo de tabela com marcações básicas

<table border=”3”>

<caption> Notas da Primeira Avaliação</caption>

<td> Notas / aluno </td>

<th> Eduardo </th>

<th> Amanda </th>

<th> Maria </th>

<tr>

<th> Notas</th>

<td> 8,0 </td>

<td> 9,5 </td>

<td> 7,3 </td>

<tr>

<th> Nº de inscrição </th>

<td> 0001</td>

<td> 0002</td>

<td> 0003 </td>

</table>

**Atributos para tabela**

**Border** 🡺 esse atributo é opcional para para ser usado no <table>, se ele estiver presente a tabela será apresentada com as linhas de borda.

Forma de utilização : <table border=”valor”>

**Align**🡺 (align=”left | right | Center”)

Esse atributo pode ser aplicado ao <td>, <th> e ,<tr> e controla o alinhamento do texto dentro da célula, com relação às bordas laterais.

Quando aplicado ao <TR>, ele define o alinhamento de toda a linha da tabela.

Exemplo

<table border=”3”>

<td> primeira célula </td>

<td> segunda célula </td>

<td> terceira célula </td>

<TR>

<td align=”Center”> Centro</td>

<td align=”left”> esquerda</td>

<td align=”right”>Direita</td>

</table>

**Valign**🡺 (valign=”top | middle| bottom”)

Esse atributo pode ser aplicado ao <td> e <th> , ele define o alinhamento do texto nas células com relação às bordas inferior e superior.

Exemplo

<table border=”3”>

<td> teste para alinhamento <BR>

Com relação as bodas<BR>

Inferior e superior<BR>

</td>

<td valign=”top”> em cima</td>

<td valign=”middle”>meio</td>

<td valign=”bottom”>em baixo</td>

</table>

**Colspan**🡺 (colspan= valor)

Para fazer tabelas com números diferentes de células de uma linha para a outra, usa-se o atributo colspan. Ele pode ser aplicado apenas ao <td> e<th>. Esse atributo define quantas colunas de tamanho normal uma célula poderá abranger. Esse atributo simula a função de mesclar das planilhas, mesclado neste caso na horizontal (colunas).

Exemplo

<table border="3">

<td colspan=3>Célula ocupando 3 colunas </td>

<td>Celula normal </td>

<td> Célual normal</td>

<TR>

<td> col1</td>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

</table>

**Rowspan**🡺 (rowspan= valor)

Esse atributo funciona exatamente igual ao colspan. A única diferença é que ele define quantas linhas uma determinada célula deve ocupar. Também simula a função de mesclar das planilhas, só que neste caso mesclando na vertical (linhas).

Exemplo

<table border="3">

<td rowspan=3>Célula ocupando 3 linhas </td>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

<TR>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

<tr>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

</table>

**Cellspacing** 🡺(cellspacing=valor)

Esse atributo é aplicável ao elemento <table> e define o espaço entre cada célula na tabela.

Exemplo

<table border="3" cellspacing = 10>

<td colspan=3>Célula ocupando 3 colunas </td>

<td>Celula normal </td>

<td> Célual normal</td>

<TR>

<td> col1</td>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

</table>

**Cellpadding**🡺 (cellpadding= valor)

Esse atributo é também aplicado ao elemento <table> e define a distância entre o texto e a borda de cada célula.

Exemplo

<table border="3" cellspacing = 10 cellpadding=20>

<td colspan=3>Célula ocupando 3 colunas </td>

<td>Celula normal </td>

<td> Célual normal</td>

<TR>

<td> col1</td>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

</table>

**Nowrap** 🡺 às vezes, o próprio navegador infere pontos de quebra de linha dentro das células da tabelas. Isso ocorre para diminuir o tamanho da tabela, possibilitar que a mesma caiba na tela e evitar uma barra de rolagem horizontal. Para impedir esse comportamento dos navegadores, faz-se uso do atributo nowrap.

Dentro da tag <td> ou <th>

Exemplos

<td nowrap > teste 1 – teste 1 - teste 1 – teste 1 - teste 1 – teste 1 - teste 1 – teste 1 - teste 1 </td>

**Width** 🡺 (width =”valor percentual ou em pixels”)

Esse atributo pode ser aplicado a <table>, <td> e <th>. Quando aplicado ao <table>, ele define o tamanho em pixels ou o tamanho percentual de toda a tabela dentro da página. Quando aplicado ao <td> ou <th>, ele difine o tamanho da célula em relação ao tamanho total da tabela.

Exemplo

<table border="3" width=”50%”>

<td colspan=3>Célula ocupando 3 colunas </td>

<td>Celula normal </td>

<td> Célual normal</td>

<TR>

<td> col1</td>

<td> col2</td>

<td> col3</td>

<td> col4</td>

<td> col5</td>

</table>

**LINKS**

Os links possibilitam que o usuário possa navegar de uma página para outra,dentro ou fora do mesmo site ou página, essa interação pode ser conseguida através da container tag <a>. Essa tag contém informações adicionais(definidas por seus argumentos), que identificam a sua função e, se for o caso, a url do documento de destino.

O elemento <a> tanto pode sinalizar um link para algum documento, quanto pode definir uma posição que será o destino de algum outro link.

O atributo href é usado para fazer referência a um outro documento, ele deve receber o valor da “url” do documento que se deseja referenciar.

**Exemplo**

<a href=”url” > aqui entra o nome do link </a>

Outro atributo da tag <A>, é o Target esse atributo indica em qual janela o browser o link deverá abrir. Se omitirmos esse atributo, ele abrirá o link na mesma pagina em que o mesmo se encontra, se atribuirmos o valor de  **target=”\_blank ”** , o link abrirá em uma nova janela.

Para criarmos links internos, dentro da própria página, necessitamos utilizar o atributo “ID” e o símbolo” #”.

O símbolo **#** informa ao navegador para ficar na mesma página e deve ser seguido do valor do atributo **“ID”**, o **“ID”** é usado para marcar o elemento que é o destino do link, o nome de um “ID” deve sempre começar com uma letra.

**Exemplo**

<HTML>

<head>

<title> meu link </title>

</head>

<body>

<a href=”#cap\_1”> link para o capítulo um</a>

<a href=”#cap\_2”> link para o capítulo dois</a>

Texto texto texto Texto texto texto

Texto texto texto Texto texto texto

<h1 id=”cap\_1”>Capítulo 1</h1>

Texto texto texto Texto texto texto

Texto texto texto Texto texto texto

<h1 id=”cap\_2”>Capítulo 2</h1>

</body>

</HTML>

**Exemplo de um link para um endereço de e-mail**

<A href=”mailto: [fulano@gmail.com.br](mailto:fulano@gmail.com.br)”> link para um e-mail </A>

**Exemplo de uma imagem como link**

<A href=”url”><img src=”foto1.gif”></a>

**o atributo Title**

esse atributo é usado para fornecer uma breve descrição do link. Se você – sem clicar no link- colocar o cursor do mouse sobre o link, vai aparece o texto que você colocou como valor no atributo title.

**Exemplo**

<A href=”url” title=” esse texto vai aparecer quando você colocar o mouse no link”> link</A>

**FOMULÁRIOS**

A linguagem <HTML> permite que o usuário possa interagir com o servidor, através dos formulários é possível a coleta de dados e a interação com o servidor. Apesar dos formulários coletarem os dados, eles não tratam os dados coletados, essa tarefa fica a encargo dos programas chamados de scripts. Os formulários são delimitados pela container tag <FORM>, podem existir vários formulários em uma mesma página, porém não é possível que eles estejam de forma aninhadas, ou seja, uma <form> dentro do outro.

O <FORM> possui dois atributos fundamentais:

**Action (action=”url”)**

Esse atributo especifica a “url” para a qual o conteúdo do formulário será enviado. Usualmente a “url”aponta para o script que deve receber e tratar os dados do formulário.

**Method (method=”Post| Get”)**

Esse atributo indica como os dados serão passados para o script ou programa definido no atributo action.

**Exemplo**

<form action=”url” method= “Post”> ..........</form>

**O element <input>**

A empty tag <input> especifica uma variedade de campos editáveis dentro de um formulário. Ele pode receber diversos atributos de definem o tipo do mecanismo de entrada de dados (botões, janelas de texto, etc)

**Name (name=”nome”)**

O atributo name do elemento <input> é obrigatório, ele atribui o conteúdo do elemento <input> à variável “nome”. Essa atribuição é necessária para que se possa fazer referências a essa variável dentro do script.

**Type (type=”valor”)**

O atributo type pode receber uma variedade de valores que determinam o tipo de campo de entrada de dados.

**Valores para Type**

**Type=”text”**

Campos de texto de linha única, o comprimento visível do campo, em caracteres, pode ser delimitado através do atributo size, que não delimita a quantidade de dados, mas sim o tamanho visível do campo em questão. Já o atributo maxlenght delimita o tamanho dos dados de entrada.

**Exemplo**

<input type=”text” name=”nome” size=”10” maxlenght=”50” value=”digite aqui seu nome”>

O atributo value neste caso traz um valor pré definido para o campo.

**Type=”password”**

O tipo passwor é exatamente como o tipo text, mas o texto digitado aparece como asteriscos, já que se trata de campo de senha. Esse tipo também admite o uso dos atributos maxlenght e size.

**Exemplo**

<input type=”password” name=”senha” maxlenght=”8” size=”6”>

**Type=”radio”**

Os radio buttons é usado quando o usuário deve escolher apenas uma opção, de um conjunto de alternativas. Para esse tipo, o atributo name e value são necessários, neste tipo também é necessário que os valores para o atributo value devem necessáriamente ser iguais, pois a variável receberá apenas uma resposta, ou valor. O atributo checked é opcional, ele faz com que a alternativa que o contém apareça pré selecionada.

**Exemplo**

<input type=”radio” name=”po1” value=”sim” checked> Sim <br>

<input type=”radio” name=”po1” value=”nao” > Não<br>

**Type=”checkbox”**

Esse tipo é usado quando desejamos permitir que o usuário possa selecionar várias respostas de um conjunto de alternativas, neste tipo o atributo name deve ser necessariamente diferentes, pois cada resposta deve ser atribuída a uma variável diferente. O tributo checked também pode ser utilizado para trazer algumas opções pré selecionadas.

**Exemplo**

<input type=”checkbox” name=”op1” value=”ie” checked> internet explore<br>

<input type=”checkbox” name=”op2” value=”fire”>Fire Fox<br>

<input type=”checkbox” name=”op3” value=”chrome”>google

**Type=”submit”**

Esse tipo cria o botão que envia o formulário para a url definida no atributo action, o atributo value neste caso determina o nome do botão.

**Exemplo**

<input type=”submit” value=”enviar”>

**Type=”reset”**

Esse tipo cria um botão que quando pressionado , ele automaticamente limpa todos os campos já preenchidos no formulário, voltando ao estado inicial do formulário.

**Exemplo**

<input type=”reset” value=”limpar”>

**<textarea>**

Essa container tag permite que o usuário entre com um bloco de texto. O bloco de texto não é limitado pelo tamanho do campo visível na página, sendo apresentadas as barras de rolagens quando o texto for maior que a área visível.

Os atributos COLS e ROWS, delimitam o tamanho do campo visível na tela, COLS delimitam as colunas e ROWS delimitam as linhas que se desejam mostrar para o usuário. Para se estabelecer um texto pré defino neste campo, basta que se coloque o texto desejado entre as tags <textarea> e </textarea>

**Exemplo**

<textarea name=c”end” cols=”20 rows=”5”>

Rua capitão solitário, 123

Boa vista – recife – PE

CEP: 08010-034

</textarea>

**Menus e Listas com os elemento SELECT e OPTION**

Menus e scrolling list podem ser definidos a partir da container tag SELECT e da empty tag OPTION, o elemento SELECT deve ainda ser acompanhado do atributo name. O OPTION determina as opções que podem ser selecionadas. O atributo size determina quantas opções serão visíveis na tela. E o atributo MULTIPLE é usado quando se deseja permitir que o usuário possa selecionar mais de uma opção da lista.

**Exemplo**

<select name=”per1” size=”2” multiple>

<option>opção 1

<option>opção 2

<option>opção 3

<option>opção 4

</select>

Quando usamos o atributo selected dentro do campo option, podemos trazer essa opção pré selecionada na lista.

**Exemplo**

<select name=”per1” multiple>

<option selected>opção 1

<option>opção 2

<option selected>opção 3

<option>opção 4

</select>

**<Fieldset> e </fieldset>**

Esse elemento delimita um contorno para o formulário, e o elemento <legend> e </legend> define um nome para seu formulário.

**Exemplo**

<HTML>

<head>

<title> meu formulário </title>

</head>

<body>

<form>

<fieldset> <legend> Formulario </legend>

Senha

<input type=”password” size=”4” maxlenght=”6”>

</fieldset>

</form>

</body>

</html>

**FRAMES**

Os frames são utilizados para possibilitar a divisão da tela em várias partes que terão seu funcionamento de forma independente, como se tivéssemos várias páginas em uma única janela. Para construirmos essas divisões, usamos os frames, descritos em um documento à parte que denominamos “Document Frames”. Os frames utilizam duas tags para sua definição, são elas: <FRAMESET> </FRAMESETE> E <FRAME>.

Os <frameset>, definem as divisões a serem estabelecidas na horizontal e na vertical para isso fazem uso do atributos COLS (vertical, coluna)e ROWS (horizontal, linhas).

**Exemplo dividindo a tela em duas colunas**

<HTML>

<head>

<title> document frame</title>

</head>

<frameset cols=”20%, 80%”> <!—dividindo a tela em duas colunas-->

<frame src=”url” name= “coluna1”><!—informando o que deve abrir no primeiro pedaço de 20%-->

<frame src=”url” name=”coluna2”><!—informando o que deve abrir no Segundo pedaço de 80%-->

</frameset><!—fechando a divisão de duas colunas-->

</html>

**Exemplo dividindo a tela em duas linhas**

<HTML>

<head>

<title> document frame</title>

</head>

<frameset rows=”20%, 80%”><!—dividindo a tela em duas linhas-->

<frame src=”url” name= “linha1”> <!—informando o que deve abrir no primeiro pedaço de 20%-->

<frame src=”url” name=”linha2”> <!—informando o que deve abrir no Segundo pedaço de 80%-->

</frameset><!—fechando a divisão de duas linhas-->

</html>

**Exemplo dividindo a tela em duas linhas e a segunda linha em duas colunas**

<HTML>

<head>

<title> document frame</title>

</head>

<frameset rows=”20%, 80%”><!—dividindo a tela em duas linhas-->

<frame src=”url” name= “linha1”> <!—informando o que deve abrir no primeiro pedaço de 20%-->

<frameset cols=”30%, 70%><!—dividindo a segunda linha em duas colunas-->

<frame src=”url” name=”coluna1”> <!—informando o que deve abrir no primeiro pedaço de 30%-->

<frame src=”url” name=”coluna2”> <!—informando o que deve abrir no segundo pedaço de 70%-->

</frameset><!—fechando a divisão de duas colunas-->

</frameset><!—fechando a divisão de duas linhas-->

</html>

Dentro do <frameset> a representação de divisões podem ser expressas em pixel, o que torna difícil o controle de dimensão pelo fato de não termos a informação de resolução da tela do usuário, ou em percentagem, devendo sempre os pedaços somar o total de 100%. As divisões expressas , devem sempre ser separadas por (,) virgulas.

Já os <frames> , definem qual a página que deverá ser carregada em cada um dos pedaços com o uso do atributo SRC, o atributo name associa um nome ao frame que ele representa. Tornando possível a referência a esse pedaço em específico, esses valores do name será referenciado no atributo target dos links, para informar em qual frame deverá abrir determinado link.