



WEB I

Sumário

1. O QUE É HTML?	4
1.1 O QUE É QUE ESTÁ DENTRO DE UM ARQUIVO HTML?	4
1.2 COMO É QUE O NAVEGADOR FAZ A APRESENTAÇÃO DAS PÁGINAS?	4
1.3 QUEM É RESPONSÁVEL PELA DEFINIÇÃO DOS PADRÕES DA WEB?	4
2. CRIANDO O PRIMEIRO ARQUIVO HTML	5
3. COMPOSIÇÃO DO HTML	6
3.1 TIPO DE DOCUMENTO (DOCTYPE)	6
3.2 ESTRUTURA BÁSICA DE UMA PÁGINA DE WEB	6
3.2 TAG META	7
4. CONTEÚDO DO <BODY>	7
4.1 CORES NO HTML	7
4.2 COR DE FUNDO NA PÁGINA	9
4.3 IMAGEM NO FUNDO NA PÁGINA	9
5. FORMATAÇÃO DE TEXTOS E PARÁGRAFOS	11
5.1 A TAG <P>	11
5.2 A TAG <DIV>	12
5.3 A TAG 	12
5.4 ACENTOS E CARACTERES ESPECIAIS NO HTML	13
5.5 A TAG <H> (CABEÇALHOS)	16
5.6 FORMATAÇÃO DO TEXTO	16
5.7 A TAG 	16
5.7.1 O ATRIBUTO FACE	17
5.7.2 O ATRIBUTO SIZE	17
5.7.3 O ATRIBUTO COLOR: COR DA FONTE	19
5.8 COMENTÁRIOS	20
5.9 LISTAS	20
5.9.1 LISTAS NÃO ORDENADAS	20
5.9.2 LISTAS ORDENADAS	20

5.9.3 LISTAS DE DEFINIÇÕES.....	21
6. INSERINDO IMAGENS.....	21
6.1 O ELEMENTO E O ATRIBUTO SRC	22
6.2 O ATRIBUTO ALT OU TITLE.....	23
7. LINKS.....	23
7.1 A TAG <A> (ÂNCORA) E O ATRIBUTO HREF	23
7.2 CORES DOS LINKS	23
7.3 TIPOS DE LINKS	23
7.4 ESPECIFICANDO A JANELA DE DESTINO DE UM LINK.....	24
7.5 CRIANDO O LINK SOBRE UMA IMAGEM	24
7.6 CRIANDO ÂNCORAS:	25
8. TABELAS.....	26
8.1 ELEMENTOS BÁSICOS DE UMA TABELA	26
8.2 CORES NA TABELA	27
8.3 MESCLANDO CÉLULAS.....	28
9. IFRAMES.....	28
9.1 LINKS PARA IFRAME.....	29
10. FORMULÁRIOS.....	29
10.1 CRIAR UM FORMULÁRIO	29
10.2 ATRIBUTO “ACTION” E MÉTODO	30
10.3 ELEMENTOS PARA FORMULÁRIOS.....	30
10.4 INPUTS.....	30
10.4.1 CAMPO DE SENHA	31
10.4.2 "RADIO BUTTONS"	31
10.4.3 CHECKBOXES	32
10.5 MENU DE SELEÇÃO ÚNICA (<SELECT>).....	32
10.6 BOTÕES DE ENVIO E APAGAR.....	33



1. O que é HTML?

HTML significa HyperText Markup Language (Linguagem de Marcação por Hipertexto). Este nome significa que o HTML é constituído por um texto especial, utilizando um recurso chamado hipermídia. A linguagem HTML sofreu várias atualizações no decorrer dos anos, desde a sua criação e hoje está na versão 4.01.

1.1 O que é que está dentro de um arquivo HTML?

- Um arquivo HTML é constituído por um texto que define os elementos da linguagem HTML usando “Tags de Marcação”.
- As Tags de Marcação dão instruções ao navegador sobre a estrutura do documento e sobre forma como a página deve ser apresentada graficamente.
- As Tags são representadas pelos sinais de “<” e “>” em volta da respectiva Tag, e na sua grande maioria possuem um fechamento com o mesmo nome da Tag.
Exemplo: <tag> </tag>
- Os arquivos HTML podem ser escritos usando um simples editor de texto (como o Bloco de Notas) e os seus nomes devem ter a extensão .html ou .htm.

1.2 Como é que o navegador faz a apresentação das páginas?

- As páginas contêm instruções que descrevem a forma como devem ser mostradas.
- O navegador lê as instruções e usa-as para desenhar uma representação gráfica do conteúdo da página.
- A forma usada para escrever essas instruções baseia-se nas TAGS da linguagem HTML.

1.3 Quem é responsável pela definição dos padrões da Web?

- O órgão que define as regras para a Web é o World Wide Web Consortium (W3C), que conta com a participação ativa dos líderes da indústria e de meios académicos.
- O W3C define as especificações técnicas que formam os padrões da Web.

- Os padrões mais importantes da Web são: HTML, CSS e XML.
- Existe também a linguagem chamada de XHTML. Esta é uma reformulação baseada no HTML 4.01 que obedece aos requisitos de regras do XML.
- Os padrões do XHTML já estão em processo de descontinuação, mas continuam sendo utilizados como a ponte de ligação entre o HTML 4.01 e a nova versão, o HTML 5, ainda sem data prevista para lançamento oficial.

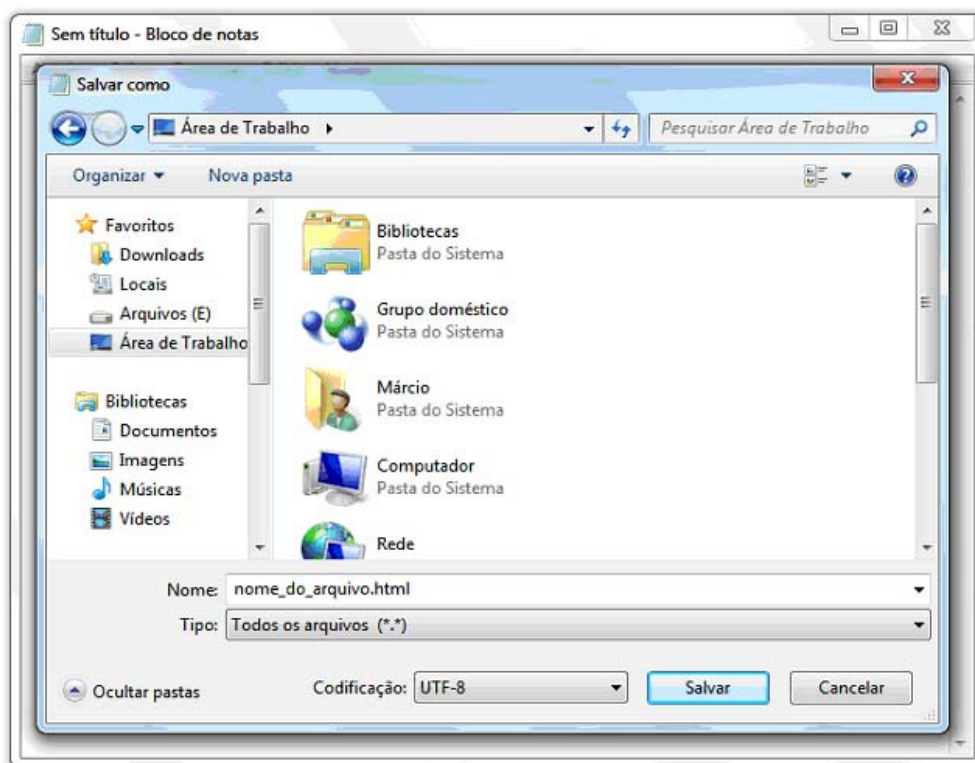
2. Criando o primeiro arquivo HTML

Para que o navegador interprete corretamente um arquivo HTML, deve ser escrito e salvo como arquivo de texto puro. Isso significa que você deve usar um processador de texto simples, como o **Bloco de Notas** do Windows.

Se for feito em algum outro editor, deve-se ter o cuidado de salvar o arquivo como texto puro (**não formatado**).

Os arquivos devem ter extensão **HTM**, ou **HTML**.

- Abra o **BLOCO DE NOTAS**, do Windows.
- Para salvar como HTML, basta clicar no menu **Arquivo, Salvar Como**, trocar na caixa Tipo “Documento de Texto” por “Todos os Arquivos”.
- Na caixa nome, digite o nome do arquivo seguido de um ponto e a extensão **html**. Não esqueça de que é obrigatório que os nomes sejam em **minúsculo e sem espaços** entre os nomes.
- Evite utilizar caracteres especiais nos nomes, como acentos, espaços ou outros caracteres. Para nomes duplos tipo Porto Alegre, use a estrutura: **porto_alegre.html**, já com nomes com outros caracteres, como Natação, use a estrutura: **natacao.html**.



3. Composição do HTML

3.1 Tipo de documento (Doctype)

Não existe um só tipo de HTML, atualmente existem vários: HTML 4.01, XHTML, e muitos outros. Cada um destes tipos está definido em especificação própria no W3C, mas também está definido em linguagem de máquina especificando a estrutura legal, os elementos e os atributos próprios do tipo de HTML. Uma definição assim é chamada de "Document Type Definition", (Definição do Tipo de Documento) ou simplesmente DTD.

Ferramentas para processar documentos, tais como Web navegadores, precisam saber qual é a DTD que o documento HTML usa: esta é razão porque documentos HTML precisam conter logo no seu início uma declaração do doctype.

O mais importante de uma declaração de doctype é que uma família considerável de navegadores não ficará tentando resolver ou "adivinhar" como renderizar o código e ao contrário entrará logo em modo de parseamento "standard", no qual o entendimento (e em consequência a apresentação) do documento é não só mais rápida como também isenta de uma má apresentação ao qual estão sujeitos os documentos sem uma declaração de Doctype.

Ao utilizarmos um código HTML limpo e sem misturar tecnologias, usando o HTML básico ou com CSS podemos apresentar o doctype como nos exemplos abaixo:

<!DOCTYPE html> Este é o modo básico de apresentação (Recomendado)

<!DOCTYPE html public "-//w3c//dtd html 4.01//en" http://www.w3.org/tr/html4/strict.dtd>
Este é o modo com as informações completas (Em fase de descontinuação)

3.2 Estrutura básica de uma página de web

Uma página Web, criada em HTML possui uma estrutura dividida em 4 TAGs básicas:

<html>...</html>

<head>...</head>

<title>...</title>

<body>...</body>

Onde:

<html>...</html> - são usadas para delimitar os comandos HTML, indicam o início e o fim de um documento. O navegador reconhece a TAG <html> e identifica que o documento que virá a seguir deve ser interpretado como HTML. Na tag <HTML>. Também podemos colocar a língua utilizada na nossa página, no caso do português colocamos **lang="pt-br"**.

<head>...</head> - define o cabeçalho de um documento HTML. Nesta seção, são inseridas informações de configuração, como o título que é exibido na barra de títulos do navegador, descrição da página, palavras-chave para buscadores, etc.

<title>...</title> - definem o título da página que é exibido na barra de títulos do navegador. Estas TAGs devem estar sempre dentro das TAGs <head></head>.

<body>...</body> - define a seção mais importante de um documento HTML. Tudo o que estiver entre esta Tag é interpretado como o "corpo" do documento e será exibido na tela do navegador.

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<title>Web I - Escola Alcides Maya</title>
</head>
<body>
Aqui vai o corpo da página
</body>
</html>
```

3.2 TAG Meta

As Tags Meta (ou Meta tags) tem a função de, entre outras coisas, descrever o conteúdo da página para os mecanismos de busca. É nelas que podemos inserir as palavras-chave e a descrição do site para facilitar as pesquisas. Por meio delas, podemos também definir a codificação dos caracteres (no caso do português, devido aos acentos, utilizamos o **charset="UTF-8"**) e também "assinar" o site, declarando sua autoria sobre o código fonte.

Exemplos:

Codificação dos caracteres com acentos:

```
<meta charset="UTF-8">
```

Definindo o autor do site:

```
<meta name="author" content="Escola Alcides Maya">
```

Fornecendo a descrição do site:

```
<meta name="description" content="Meta Tags - O que são e como utilizá-las">
```

Fornecendo as palavras-chave:

```
<meta name="keywords" content="sites, web, HTML, tags, curso">
```

Exemplo de uma estrutura completa de HTML:

```
<!DOCTYPE html>
<html lang="pt-br">
<meta charset="UTF-8">
<meta name="author" content="Escola Alcides Maya">
<meta name="description" content="Descrição completa do site">
<meta name="keywords" content="sites, web, HTML, tags, curso">
<head>
<title>Web I - Escola Alcides Maya</title>
</head>
<body>
Aqui vai o corpo da página
</body>
</html>
```

4. Conteúdo do <body>

Todo o conteúdo inserido ou digitado entre as tags <body> automaticamente fará parte do conteúdo da página a ser visualizado no navegador. Dentro do <body> podemos **ter textos, parágrafos, listas, imagens, tabelas, links, elementos flash** e uma série de outros recursos compatíveis com a linguagem HTML.

4.1 Cores no HTML

As cores mostradas na tabela seguinte têm nomes que são reconhecidos por todos os navegadores famosos. A utilização destes nomes é muito conveniente porque nos permite evitar o uso códigos hexadecimais, mas infelizmente eles não estão definidos em nenhum padrão do W3C. Se quiser que as suas páginas estejam em conformidade com as recomendações do W3C deve usar os códigos hexadecimais equivalentes em vez dos nomes.

white #FFFFFF	silver #C0C0C0	gray #808080	black #000000
aqua #00FFFF	teal #008080	fuchsia #FF00FF	purple #800080
blue #0000FF	navy #000080	lime #00FF00	green #008000
yellow #FFFF00	olive #808000	red #FF0000	maroon #800000

Para entender melhor os códigos hexadecimais basta dividir os seis números em 3 grupos.

Exemplo: #00 00 00 (os dois primeiros números representam a cor **vermelha**, os dois números do meio representam a cor **verde** e os dois últimos números representam a cor **azul**).

A mistura dos valores RGB dá todas as outras cores.

Há uma grande possibilidade de arranjos de cores, para facilitar isso trabalhamos com um número de 216 cores, que ficam divididas as intensidades da seguinte forma:

00 = zero cor (preto)
 33 = intensidade muito fraca
 66 = intensidade fraca
 99 = intensidade média
 CC = intensidade alta
 FF = intensidade máxima

Sempre que tivermos um valor igual nos 3 grupos de cores teremos como resultado as cores branca, preta ou um tom de cinza.

Segue abaixo a tabela completa com as 216 cores e seus respectivos códigos hexadecimais:

330000	333300	336600	339900	33CC00	33FF00	66FF00	66CC00	669900	666600	663300	660000	FF0000	FF3300	FF6600	FF9900	FFCC00	FFFF00
330033	333333	336633	339933	33CC33	33FF33	66FF33	66CC33	669933	666633	663333	660033	FF0033	FF3333	FF6633	FF9933	FFCC33	FFFF33
330066	333366	336666	339966	33CC66	33FF66	66FF66	66CC66	669966	666666	663366	660066	FF0066	FF3366	FF6666	FF9966	FFCC66	FFFF66
330099	333399	336699	339999	33CC99	33FF99	66FF99	66CC99	669999	666699	663399	660099	FF0099	FF3399	FF6699	FF9999	FFCC99	FFFF99
3300CC	3333CC	3366CC	3399CC	33CCCC	33FFCC	66FFCC	66CCCC	6699CC	6666CC	6633CC	6600CC	FF00CC	FF33CC	FF66CC	FF99CC	FFCCCC	FFFFCC
3300FF	3333FF	3366FF	3399FF	33CCFF	33FFFF	66FFFF	66CCFF	6699FF	6666FF	6633FF	6600FF	FF00FF	FF33FF	FF66FF	FF99FF	FFCCFF	FFFFFF
0000FF	0033FF	0066FF	0099FF	00CCFF	00FFFF	99FFFF	99CCFF	9999FF	9966FF	9933FF	9900FF	CC00FF	CC33FF	CC66FF	CC99FF	CCCCFF	CCFFFF
0000CC	0033CC	0066CC	0099CC	00CCCC	00FFCC	99FFCC	99CCCC	9999CC	9966CC	9933CC	9900CC	CC00CC	CC33CC	CC66CC	CC99CC	CCCCCC	CCFFCC
000099	003399	006699	009999	00CC99	00FF99	99FF99	99CC99	999999	996699	993399	990099	CC0099	CC3399	CC6699	CC9999	CCCC99	CCFF99
000066	003366	006666	009966	00CC66	00FF66	99FF66	99CC66	999966	996666	993366	990066	CC0066	CC3366	CC6666	CC9966	CCCC66	CCFF66
000033	003333	006633	009933	00CC33	00FF33	99FF33	99CC33	999933	996633	993333	990033	CC0033	CC3333	CC6633	CC9933	CCCC33	CCFF33
000000	003300	006600	009900	00CC00	00FF00	99FF00	99CC00	999900	996600	993300	990000	CC0000	CC3300	CC6600	CC9900	CCCC00	CCFF00
000000	333333	666666	999999	CCCCCC	FFFFFF												

4.2 Cor de fundo na página

Para que a nossa página possa ter uma cor de fundo, devemos utilizar o parâmetro **bgcolor** a cor desejada dentro da tag <body>, como nos exemplos abaixo:

<body bgcolor="#FF0000"> teremos a página com um fundo **VERMELHO**

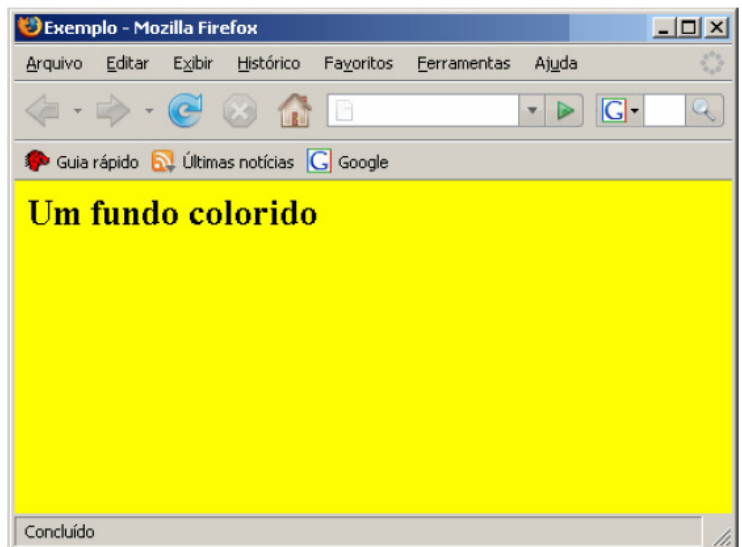
<body bgcolor="#00FF00"> teremos a página com um fundo **VERDE**

<body bgcolor="#0000FF"> teremos a página com um fundo **AZUL**

<body bgcolor="#FFCC00"> teremos a página com um fundo **LARANJA**

Exemplo de uma página com o fundo amarelo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body bgcolor="#FFFF00">
Um fundo colorido
</body>
</html>
```



4.3 Imagem no fundo na página

Para que a nossa página possa ter uma IMAGEM ao invés de uma cor de fundo, devemos utilizar o parâmetro **background** e o caminho da imagem desejada dentro da tag <body>, como mostra nos exemplos que vemos a seguir:

<body background="fundo.jpg"> fundo com uma imagem extensão JPG

<body background="imagem.gif"> fundo com uma imagem extensão GIF

<body background="papel.png"> fundo com uma imagem extensão PNG

Como vimos nos exemplos acima, os formatos de imagem mais recomendados são o JPG, o GIF e o PNG por questões do peso das imagens e da compatibilidade com os navegadores, outros formatos de imagens apesar de serem visualizados em alguns navegadores, não são recomendados.

Se a imagem estiver dentro de uma subpasta devemos colocar o caminho desta junto com o nome da imagem, como nos exemplos.

<body background="imagens/foto.jpg"> Aqui temos uma subpasta chamada imagens

<body background="fotos/paisagens/natureza.jpg"> Aqui temos duas subpastas, uma chamada fotos e outra chamada paisagens.

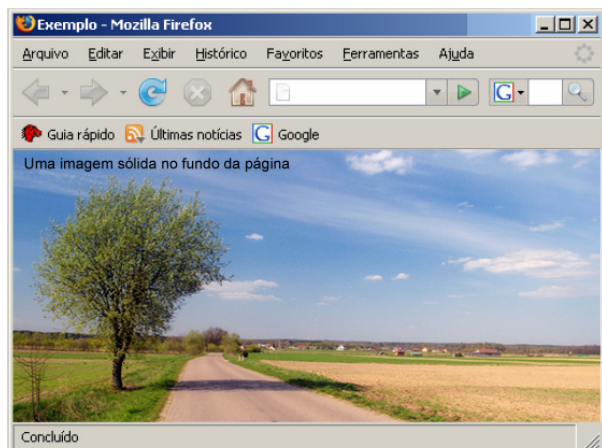
Ao inserirmos uma imagem de fundo devemos ter uma série de cuidados para que a sua visualização esteja correta, afinal temos diferentes resoluções utilizadas nos computadores (800x600, 1024x768, 1280x1024) e ainda monitores de formato quadrado (escala 4:3) e monitores formato wide (escala 16:9). Os monitores wide por serem mais alongados trabalham com resoluções diferentes das vistas acima, sendo as mais comuns as de 1024x600, 1280x768 e 1366x800, podendo ainda ter várias outras resoluções de tela alternativas de acordo com o tamanho e o formato do monitor.

Assim, para que uma imagem ocupe TODA a tela independente de sua resolução, temos duas alternativas:

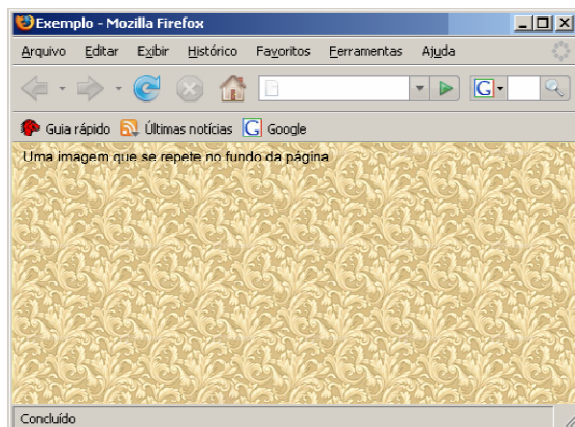
1. Utilizarmos uma imagem com uma grande resolução (tipo 1920x1080) isso fará com que a imagem sempre apareça em tela cheia seja qual for a resolução do monitor, mas em resoluções menores do que a resolução da imagem, a foto será cortada.
2. Utilizarmos uma imagem pequena como um “padrão” que vai se repetindo e preenchendo todo o fundo da tela, dando uma idéia infinita. A única desvantagem é que isso só é possível com imagens que já estejam preparadas para esta auto-repetição, normalmente sendo de imagens abstratas, não podendo utilizar fotos de paisagens, pessoas, etc.

Vejamos abaixo dois exemplos de imagens de fundo na página, uma com uma imagem inteira e outra com uma imagem que se repete:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body background="foto.jpg">
Imagem sólida no fundo da página
</body>
</html>
```



```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body background="foto2.jpg">
Imagem que se repete no fundo da
página
</body>
</html>
```



O atributo `BGPROPERTIES` quando usado com o valor `FIXED`, especifica que a imagem de plano de fundo permanecerá fixa na tela, não acompanhando a rolagem da página. Isto é útil quando queremos que a imagem de fundo fique estática e somente o conteúdo do site tenha rolagem.

Exemplo:

```
<body background="paisagem.jpg" bgproperties="fixed">
```

OBS.: Sempre devemos ter cuidado ao utilizarmos imagens de fundo na nossa página. Muitas vezes temos dificuldades de ler um texto sobre uma imagem de fundo, dificultando a compreensão e a leitura dos conteúdos presentes dentro da página. Então sempre escolha imagens que não tenham cores em excesso e que não prejudiquem a leitura do texto.

5. Formatação de textos e parágrafos

O HTML possui um bom número de tags para formatação de textos e parágrafos. Estas tags resolvem a maior parte do necessário normalmente para exibir textos, cabeçalhos, listas de itens, parágrafos e também de cores e fontes.

5.1 A TAG <p>

É responsável pela quebra de parágrafos, finaliza o parágrafo e insere automaticamente uma linha em branco entre os parágrafos. Também responsável pelo alinhamento dos parágrafos.

<p align="posição">Texto do parágrafo</p>

A posição pode ser:

Left = Alinhamento à esquerda

Center = Alinhamento centralizado

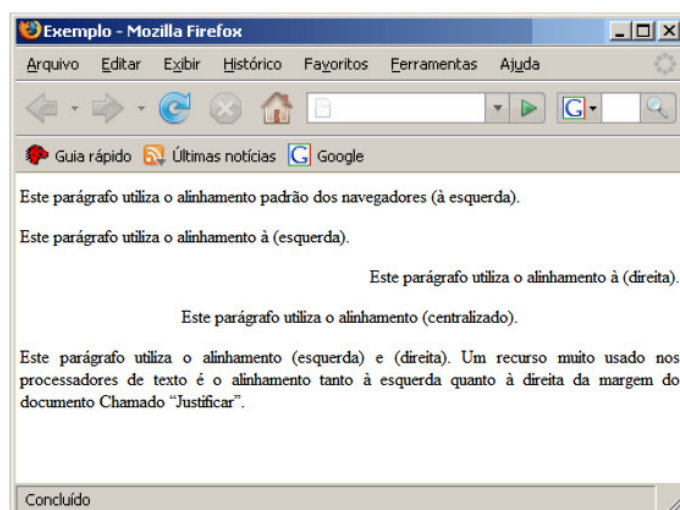
Right = Alinhamento à direita

Justify = Alinha o texto na esquerda e direita

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<p>Este parágrafo utiliza o alinhamento padrão dos navegadores (à esquerda). </p>
<p align="left">Este parágrafo utiliza o alinhamento à (esquerda). </p>
<p align="right">Este parágrafo utiliza o alinhamento à (direita). </p>
<p align="center">Este parágrafo utiliza o alinhamento (centralizado). </p>
<p align="justify">Este parágrafo utiliza o alinhamento (esquerda) e (direita). Um
recurso muito usado nos processadores de texto é o alinhamento tanto à esquerda quanto
à direita da margem do documento Chamado "Justificar".</p>
</body>
</html>
```

Resultado:



5.2 A TAG <div>

Usando isoladamente, sua utilidade é próxima à tag <p>, ou seja, todo o texto entre <div> e </div> será exibido como um parágrafo. A diferença é que <div> insere somente uma quebra de linha simples e depois do parágrafo, e não uma linha em branco como na tag <p>.

O <div> é mais utilizado no alinhamento de elementos diversos e não somente de caracteres de texto, como no <p>. O <div> tem a sua funcionalidade aumentada quando é utilizada formatada através da CSS.

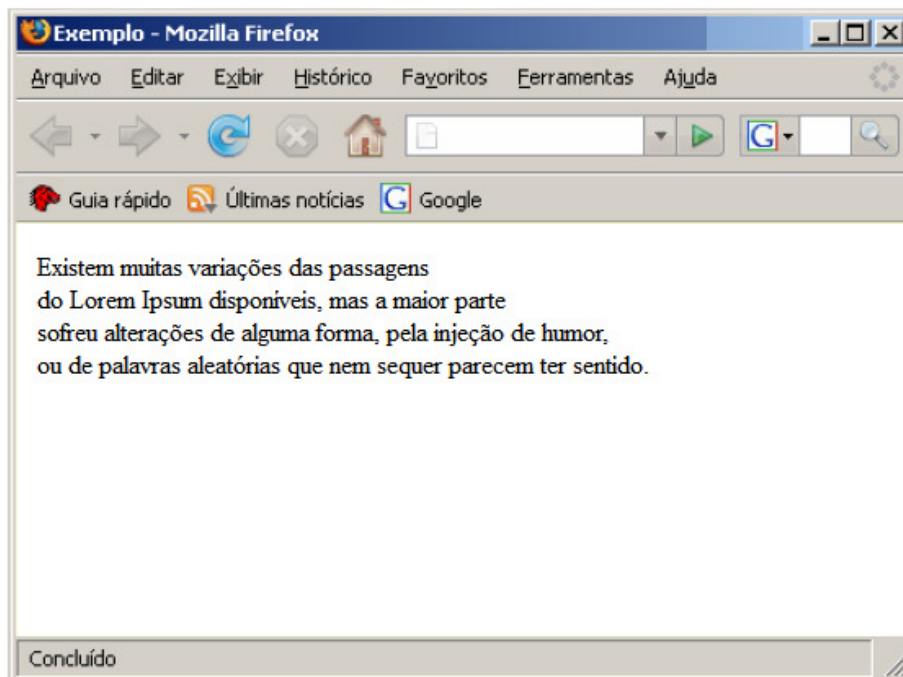
5.3 A TAG

Se você deseja forçar a quebra de linha em texto de parágrafos use esta tag, que não precisa ser fechada. No HTML ela é utilizada somente na forma
 e no XHTML é utilizada na forma </br> na sua abertura e sem fechamento.

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<p>Existem muitas variações das passagens<br>
do Lorem Ipsum disponíveis, mas a maior parte<br>
sofreu alterações de alguma forma, pela injeção de humor,<br>
ou de palavras aleatórias que nem sequer parecem ter sentido.</p>
</body>
</HTML>
```

Resultado:



5.4 Acentos e caracteres especiais no HTML

Às vezes ao digitarmos o texto em um arquivo HTML, quando visualizamos no navegador, alguns caracteres, como acentos ou “ç” são substituídos por outros caracteres ou por um “?”.

Isso acontece quando a codificação dos caracteres não está da forma correta (como vimos nas tags meta).

Para corrigir isso, a linguagem HTML permite que caracteres especiais sejam representados por seqüências de escape, indicadas por três partes: um & inicial, um número ou cadeia de caracteres correspondente ao caractere desejado, e um ; final.

Assim devemos substituir os caracteres acentuados pelo seu respectivo “código HTML”.

Exemplo: Para aparecer um “á”, coloque “´” no código HTML.

ABAIXO SEGUE A TABELA DE CARACTERES:

Acentos:

á	´	Á	Á	ã	ã	Ã	Ã
â	â	Â	Â	à	à	À	À
é	é	É	É	ê	ê	Ê	Ê
í	í	Í	Í	ó	ó	Ó	Ó
õ	õ	Õ	Õ	ô	ô	Ô	Ô
ú	ú	Ú	Ú	ü	ü	Ü	Ü
ç	ç	Ç	Ç				

Caracteres Comerciais:

©	copyright	©
®	marca registrada	®
™	trade mark	™
£	libra esterlina	£
¢	centavo	¢
€	euro	€
¥	iene (yen)	¥
¤	símbolo monetário	¤

Caracteres Especiais:

	Espaço forçado	
&	e comercial	&
>	maior que	>
<	menor que	<
^	acento circunflexo	ˆ
~	acento til	˜
¨	acento trema	¨
´	acento agudo	´
¸	cedilha	¸
"	aspas duplas	"
“ e ”	aspas duplas (esquerda e direita)	“ e ”
‘ e ’	aspas simples (esquerda e direita)	‘ e ’
‹ e ›	aspas angulares simples (esquerda e direita)	‹ e ›
« e »	aspas angulares duplas (esquerda e direita)	« e »
º	ordenal masculino	º
ª	ordinal feminino	ª
—	travessão ‘en’	–
—	travessão ‘em’	—
-	macron	¯
...	reticências	…
	barra vertical	¦
•	marcador (bullet)	•
¶	parágrafo	¶
§	parágrafo legal	§

Caracteres Matemáticos e Lógicos:

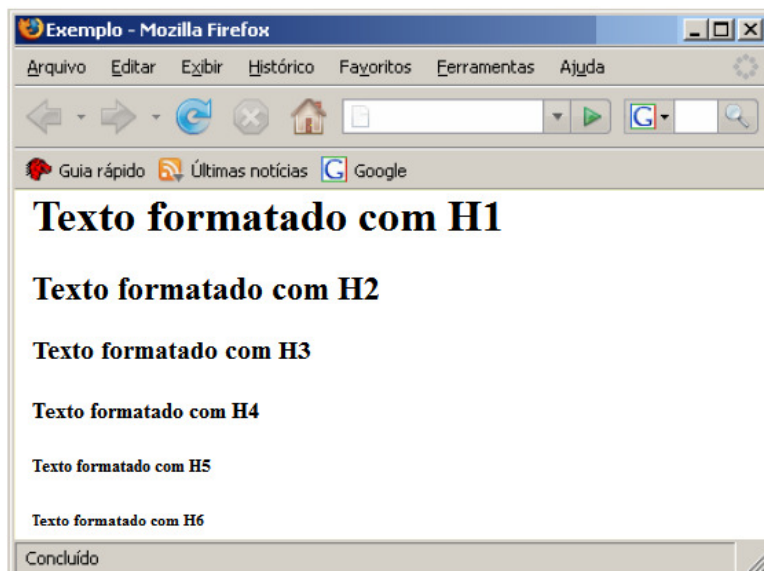
1	elevado a um	¹
2	ao quadrado	²
3	ao cubo	³
1/2	fração um meio	½
1/4	fração um quarto	¼
3/4	fração três quartos	¾
1/8	fração um oitavo	⅛
3/8	fração três oitavos	⅜
5/8	fração cinco oitavos	⅝
7/8	fração sete oitavos	⅞
>	maior que	>
<	menor que	<
±	mais ou menos	±
–	sinal de subtração	−
×	sinal de multiplicação	×
÷	sinal de divisão	÷
*	asterisco	∗
/	barra de fração	⁄
‰	por-mil	‰
∫	sinal de integral	∫
Σ	somatório	∑
Π	PI	∏
√	raiz quadrada	√
∞	infinito	∞
≈	quase igual	≈
≅	aproximadamente	≅

5.5 A TAG <h> (Cabeçalhos)

A tag <h> (Header, ou cabeçalho) é parecida com a dos parágrafos, mas cria um destaque maior no texto. Esta tag possui seis tamanhos diferentes. O tamanho do corpo dos caracteres vai de h6 (menor nível) a h1 (maior nível). A tag <h> também suporta os mesmos parâmetros de alinhamento que a tag <p>.

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<h1>Texto formatado com H1</h1>
<h2>Texto formatado com H2</h2>
<h3>Texto formatado com H3</h3>
<h4>Texto formatado com H4</h4>
<h5>Texto formatado com H5</h5>
<h6>Texto formatado com H6</h6>
</body>
</html>
```



5.6 Formatação do texto

A linguagem HTML define vários elementos para se formatar um texto, como por exemplo, escrever em negrito, itálico, sublinhado, etc.

O quadro abaixo mostra alguns deles:

Elemento	Descrição
	Define texto escrito em negrito
	Define texto enfatizado (outro tipo de itálico)
<i>	Define texto escrito com caracteres itálicos
	Define texto forte (outro tipo de negrito)
<sub>	Texto subscrito (alinhado um pouco mais abaixo)
<sup>	Texto sobrescrito (alinhado um pouco mais acima)
<u>	Texto sublinhado

5.7 A TAG

Através da tag poderemos formatar o tipo de fonte utilizada (face) o tamanho da fonte (size) e a cor da fonte (color), a formatação da cor segue exatamente os mesmos parâmetros utilizados na formatação da cor de fundo da página.

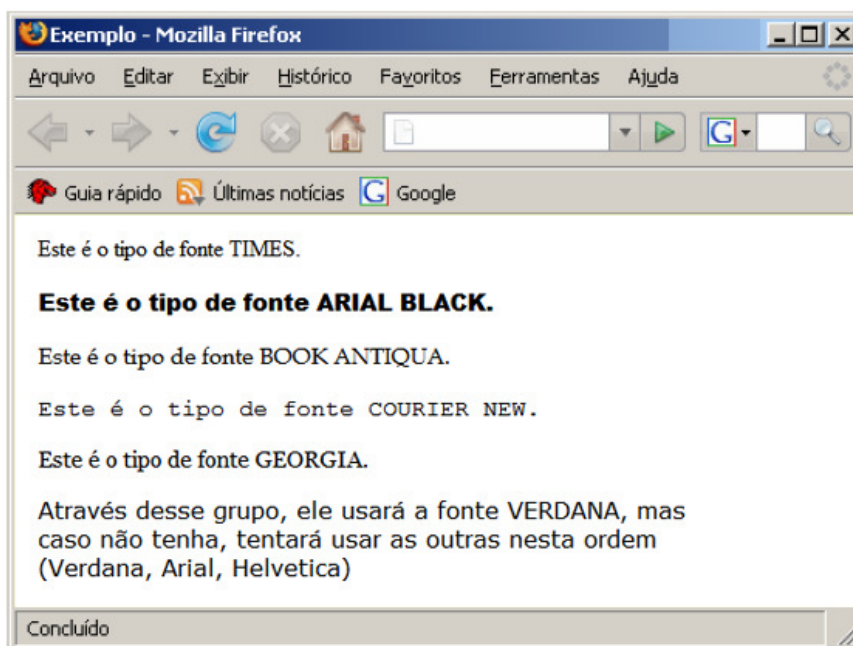
5.7.1 O atributo FACE:

Tipo de fonte - Neste atributo você informa o nome da fonte a ser usada, que deve estar instalada no computador do usuário. Se a fonte informada não estiver presente, você pode especificar uma lista de outras fontes alternativas, separando-as por vírgula. Se o navegador não encontrar a primeira fonte da lista, tentará a segunda e assim por diante.

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<p><font face="Tahoma">Este é o tipo de fonte TAHOMA.</font></p>
<p><font face="Arial Black">Este é o tipo de fonte ARIAL BLACK.</font></p>
<p><font face="Book Antiqua">Este é o tipo de fonte BOOK ANTIQUA.</font></p>
<p><font face="Courier New">Este é o tipo de fonte COURIER NEW.</font></p>
<p><font face="Georgia">Este é o tipo de fonte GEORGIA.</font></p>
<p><font face="Verdana, Arial, Helvetica">Através desse grupo, ele usará a fonte
VERDANA, mas caso não tenha, tentará usar as outras nesta ordem (Verdana, Arial,
Helvetica)</font></p>
</body>
</html>
```

Resultado:



OBS.: Segundo o W3C, a formatação das fontes via a tag e os atributos face, size e color está em desuso, não sendo a forma mais correta de procedimento para a formatação das fontes.

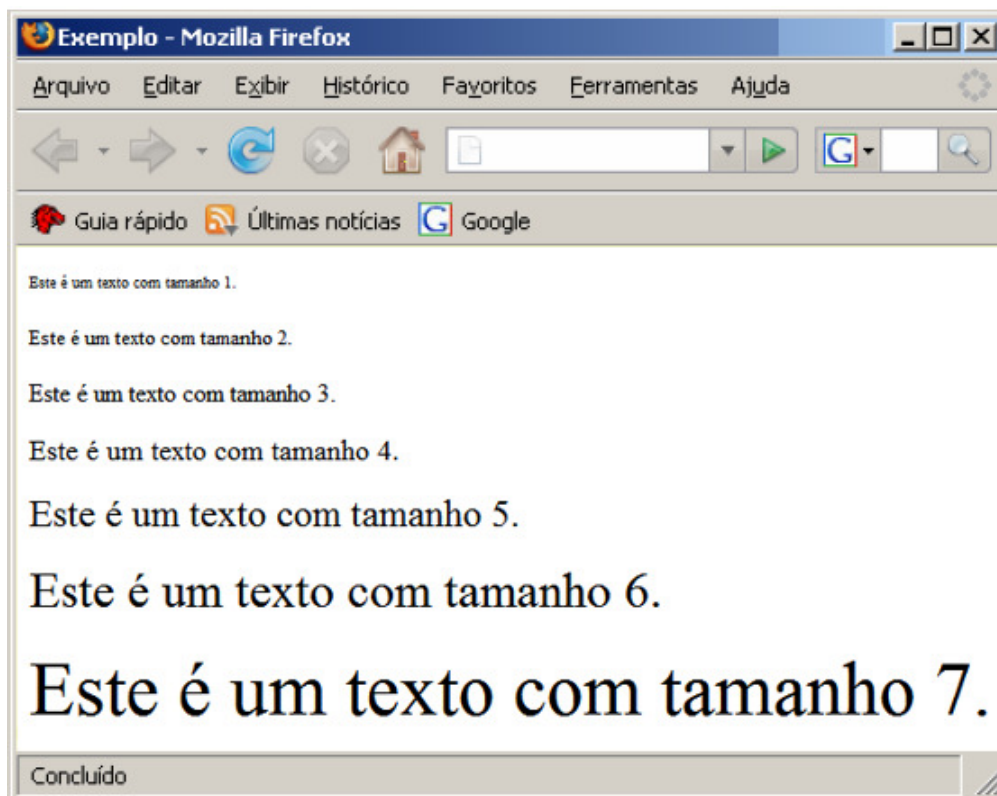
Atualmente, o modo ideal de formatação das fontes é através dos recursos de CSS.

5.7.2 O atributo SIZE

Informa qual o tamanho da fonte. Este tamanho é um nº entre 1 a 7. O valor padrão é 3. É possível também utilizar os valores pequenos (-1 a -7) ou grandes (+1 a +7)

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<p><font size="1"> Este é um texto com tamanho 1. </font></p>
<p><font size="2"> Este é um texto com tamanho 2. </font></p>
<p><font size="3"> Este é um texto com tamanho 3. </font></p>
<p><font size="4"> Este é um texto com tamanho 4. </font></p>
<p><font size="5"> Este é um texto com tamanho 5. </font></p>
<p><font size="6"> Este é um texto com tamanho 6. </font></p>
<p><font size="7"> Este é um texto com tamanho 7. </font></p>
</body>
</html>
```

Resultado:



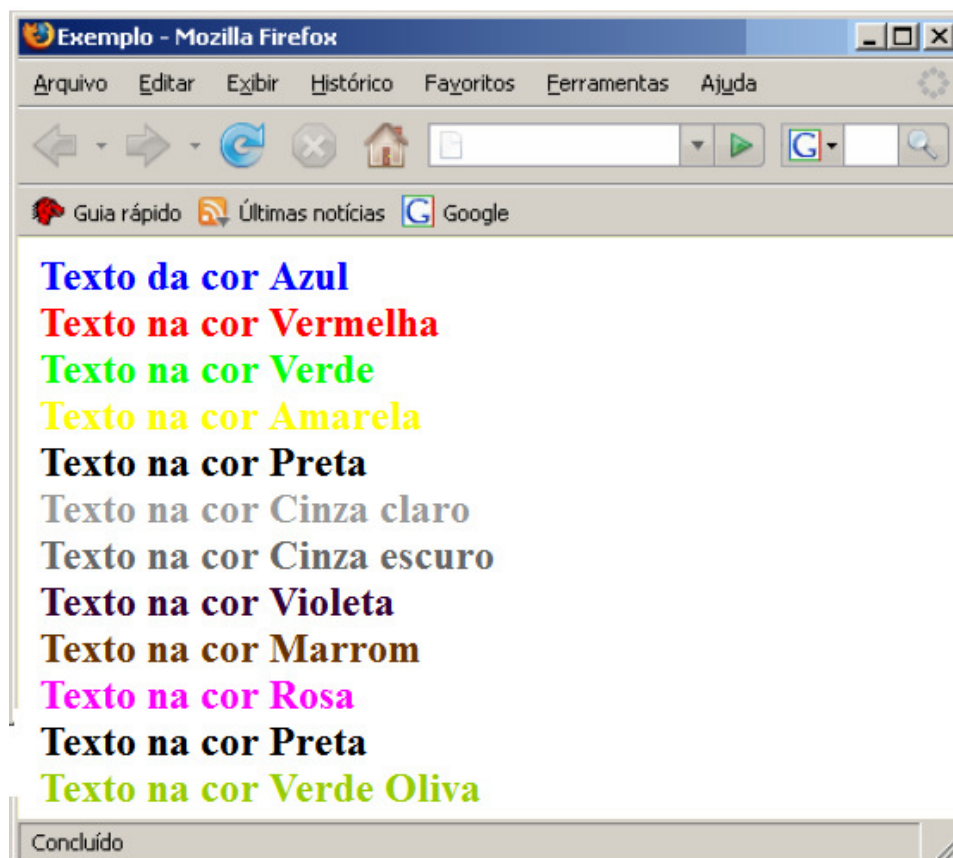
5.7.3 O atributo COLOR: Cor da fonte

O sistema de cores para as fontes segue exatamente o mesmo esquema das cores para o fundo da página, podendo ser utilizado o nome da cor ou o código hexadecimal.

Como no exemplo a seguir:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<h2>
<font color="#0000FF">Texto da cor Azul</font><br>
<font color="#FF0000">Texto na cor Vermelha</font><br>
<font color="#00FF00">Texto na cor Verde</font><br>
<font color="#FFFF00">Texto na cor Amarela</font><br>
<font color="#000000">Texto na cor Preta</font><br>
<font color="#999999">Texto na cor Cinza claro</font><br>
<font color="#666666">Texto na cor Cinza escuro</font><br>
<font color="#330033">Texto na cor Violeta</font><br>
<font color="#663300">Texto na cor Marrom</font><br>
<font color="#FF00FF">Texto na cor Rosa</font><br>
<font color="#000000">Texto na cor Preta</font><br>
<font color="#99CC00">Texto na cor Verde Oliva</font><br>
</h2>
</body>
</html>
```

Resultado:



5.8 Comentários

As tags de comentário são especiais porque não se escrevem da mesma forma que as tags que representam elementos normais. Os comentários servem para dizer ao navegador que o seu conteúdo é um comentário e não pode ser apresentado graficamente. Usa-se essas tags para fazer anotações que explicam a forma como o código fonte está feito para que mais tarde consigamos compreender aquilo que fizemos antes.

O exemplo seguinte mostra um comentário:

```
<!-- Isto é um comentário -->
```

5.9 Listas

A linguagem HTML contém elementos que permitem criar listas de definições, listas ordenadas e listas não ordenadas.

5.9.1 Listas Não Ordenadas

Uma lista não ordenada contém vários itens marcados todos com o mesmo símbolo (normalmente um círculo pequeno ou um quadrado pequeno).

Para criar uma lista não ordenada usa-se o elemento `` ("unordered list"). Dentro desse elemento os vários itens são criados com o elemento `` ("list item"). O exemplo seguinte mostra uma lista simples:

```
<ul>
<li>Janeiro</li>
<li>Fevereiro</li>
<li>Março</li>
</ul>
```

Este é o aspecto com que ficará no seu navegador:

- Janeiro
- Fevereiro
- Março

A tag `` pode ter parâmetros adicionais, como modificar o marcador da lista, para isso temos 3 tipos de marcadores (quadrado, círculo e disco). Abaixo vemos os exemplos com os 3 marcadores:

```
<ul type="circle"> Lista com marcador tipo círculo (padrão)
```

```
<ul type="square"> Lista com marcador tipo quadrado
```

```
<ul type="disc"> Lista com marcador tipo disco (círculo vazado)
```

5.9.2 Listas Ordenadas

Uma lista ordenada contém vários itens numerados e é criada com o elemento `` ("ordered list").

Os itens da lista definem-se com o elemento `` ("list item").

O exemplo seguinte mostra uma lista ordenada:

```
<ol>
<li>Brasil</li>
<li>Argentina</li>
<li>Uruguai</li>
<li>Paraguai</li>
</ol>
```

Este é o aspecto com que ficará no seu navegador:

1. Brasil
2. Argentina
3. Uruguai
4. Paraguai

A tag `` também tem parâmetros adicionais, onde podemos definir que tipo de numerador representará a lista, se numérico, alfabético, romano e se é minúsculo ou maiúsculo. Podemos definir também em que número iniciará a contagem da lista.

Exemplos:

`<ol type="1">` Lista ordenada por números (padrão)

`<ol type="1" start="20">` Lista ordenada por números começando no nº 20

`<ol type="I">` Lista ordenada por números romanos maiúsculos

`<ol type="i">` Lista ordenada por números romanos minúsculos

`<ol type="A">` Lista em ordem alfabética maiúscula

`<ol type="a">` Lista em ordem alfabética minúscula

5.9.3 Listas de Definições

Uma lista de definições não é constituída por uma série de itens mas sim por vários termos acompanhados de descrições dos seus significados.

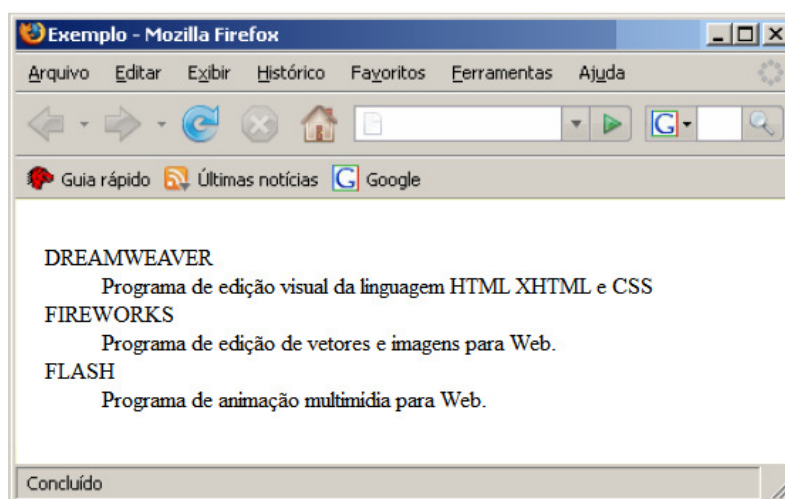
As listas de definições criam-se com o elemento `<dl>` ("definition list"). O nome de cada termo fica dentro de um elemento `<dt>` ("definition term") e a sua descrição fica num elemento `<dd>` ("definition description").

Exemplo:

```
<dl>
<dt>DREAMWEAVER</dt>
<dd>Programa de edição visual da linguagem HTML XHTML e CSS</dd>
<dt>FIREWORKS</dt>
<dd>Programa de edição de vetores e imagens para Web.</dd>
<dt>FLASH</dt>
<dd>Programa de animação multimídia para Web.</dd>
</dl>
```

Resultado no navegador:

OBS.: As listas de definição podem ser formatadas via CSS ficando com um visual extremamente rico e podendo ser utilizado para um grande número de outros recursos.



6. Inserindo Imagens

Usar imagem em sua página pode ser um grande diferencial de atratividade. A tag que utilizaremos para inserir uma imagem é `` (image) e veremos os seus atributos, tais como altura, largura, borda e dimensões.

6.1 O elemento e o atributo src

A inserção de imagens num documento se faz com o elemento . Este elemento tem conteúdo vazio, o que significa que ele apenas possui atributos e também não tem fechamento.

A escolha da imagem para mostrar na página se faz através do atributo src (que é uma abreviação de "source", ou origem) o qual indica o nome e o local em que pode ser encontrado o arquivo que contém a imagem. As imagens devem estar nos formatos JPG, GIF ou PNG.

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>


</body>
</html>
```



O exemplo seguinte ilustra o uso deste elemento:

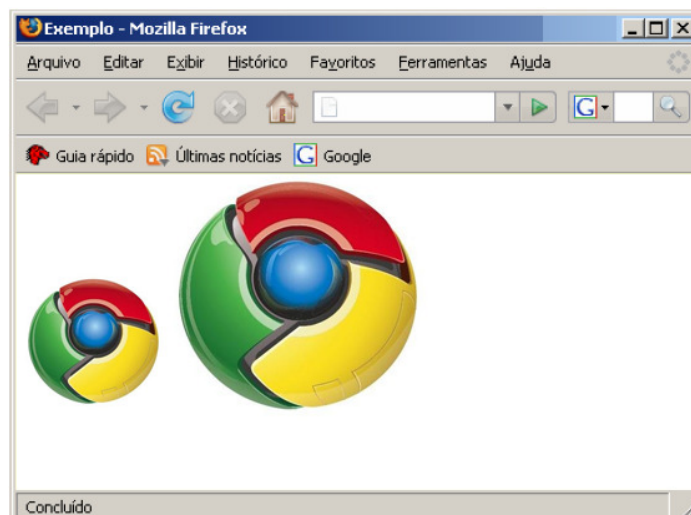
O src pode apontar para um arquivo local ou para uma URL (endereço web) onde o arquivo está hospedado, em qualquer um dos casos a imagem será visualizada normalmente.

É possível também mudarmos as dimensões da imagem através dos atributos **width (largura)** e **height (altura)**, podendo até mesmo mostrar uma mesma imagem de tamanhos diferentes na mesma página.

Exemplo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>


</body>
</html>
```



6.2 O atributo alt ou title

Os atributos alt e title usam-se para definir "texto alternativo" à imagem. Este texto será mostrado em vez da imagem no caso de o browser não conseguir apresentá-la, e usa-se assim:

```
  

```

É importante ressaltar a diferença entre alt e title. O atributo alt funciona somente no Internet Explorer, já o atributo title é usado no Mozilla. Para evitar problemas com a utilização desses atributos, utilize-os juntamente como no exemplo:

```

```

Os atributos alt e title devem conter uma descrição daquilo que a imagem contém para que seja possível compreender o seu conteúdo mesmo sem a ver. A utilização dos atributos alt e title é recomendada porque melhora a acessibilidade das páginas. Este atributo pode ser lido em voz alta pelo software de leitura utilizado pelas pessoas que têm deficiências visuais. Nesses casos, sem os atributos alt e title o conteúdo das imagens seria sempre um mistério.

7. Links

O HTML usa ligações de hipertexto ("hyperlinks") para ligar as páginas da Web umas às outras.

7.1 A Tag <a> (âncora) e o atributo href

A tag <a> serve para criar uma ligação ("link") para outro documento. O link pode apontar para qualquer recurso disponível na Web: uma página escrita em HTML, uma imagem, um arquivo de som, um filme, etc.

Sintaxe para criar um link:

```
<a href="url">Texto a mostrar</a>
```

O atributo href contém o endereço (URL) do recurso ao qual se faz o link. O texto que fica entre as tags <a> é o local visível sobre o qual se deve clicar para entrar no link.

O Link a seguir vai para o site da escola:

```
<a href="http://www.alcidesmaya.com.br">Escola Alcides Maya</a>
```

7.2 Cores dos links

Geralmente o navegador apresenta os links em azul e sublinhados, mas é possível alterar isso através dos parâmetros opcionais do elemento <body>.

LINK: Define a cor dos links ainda não visitados.

VLINK: Define a cor dos links já visitados.

ALINK: Define a cor dos links no instante em que são clicados pelo usuário.

7.3 Tipos de links

Link para uma página web:

```
<a href="http://www.google.com.br">Site do Google</a>
```

Link para um documento HTML:

```
<a href="pagina.html">Página 2</a>
```

Link para uma imagem:

```
<a href="imagem.jpg">Abre a imagem</a>
```

Link para um endereço de e-mail:

```
<a href="mailto:nome@email.com.br">Mande um e-mail</a>
```


7.4 Especificando a janela de destino de um link

Por padrão, o navegador exibe a página destino de um hyperlink na mesma janela, substituindo a página atual. Usando o atributo TARGET, podemos especificar outra janela.

```
target="_blank"
```

Como no exemplo:

```
<a href="http://www.google.com.br" target="_blank">Google</a>
```

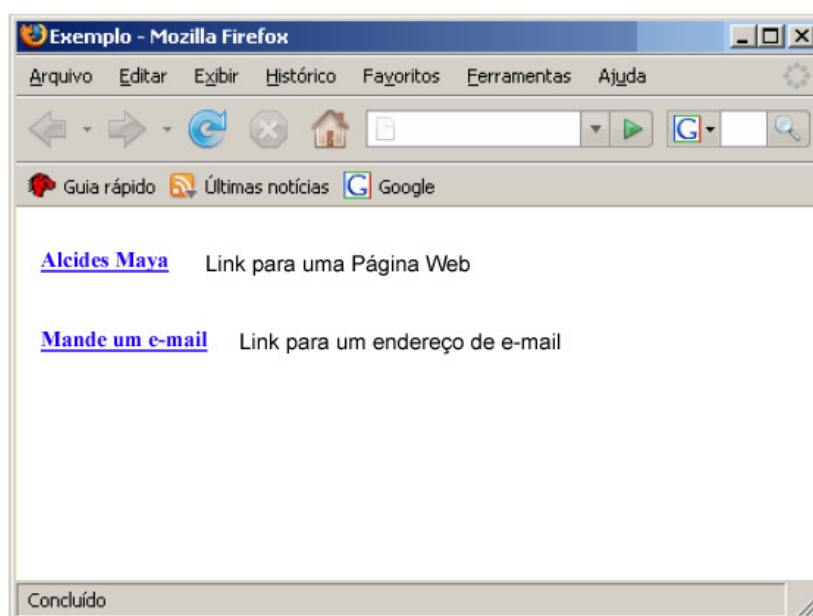
Visualizando no navegador:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>

<a href="http://www.alcidesmaya.com.br">Alcides Maya</a>

<a href="mailto:nome@email.com.br">Mande um e-mail</a>

</body>
</html>
```



7.5 Criando o link sobre uma imagem

É possível utilizar os links não apenas com o texto, mas também utilizar os links em imagens, botões, banners, etc. Para que possamos ter um link sobre uma imagem é necessário que a tag <a> envolva a tag que fornece o caminho da imagem.

A sintaxe do código para criar links em imagens ficará assim:

```
<a href="pagina.html"></a>
```


Ao criarmos o link sobre a imagem ela ficará marcada com uma borda azul (que identifica os links), Essa borda pode ser retirada se utilizarmos o atributo `border="0"` dentro da tag da imagem.

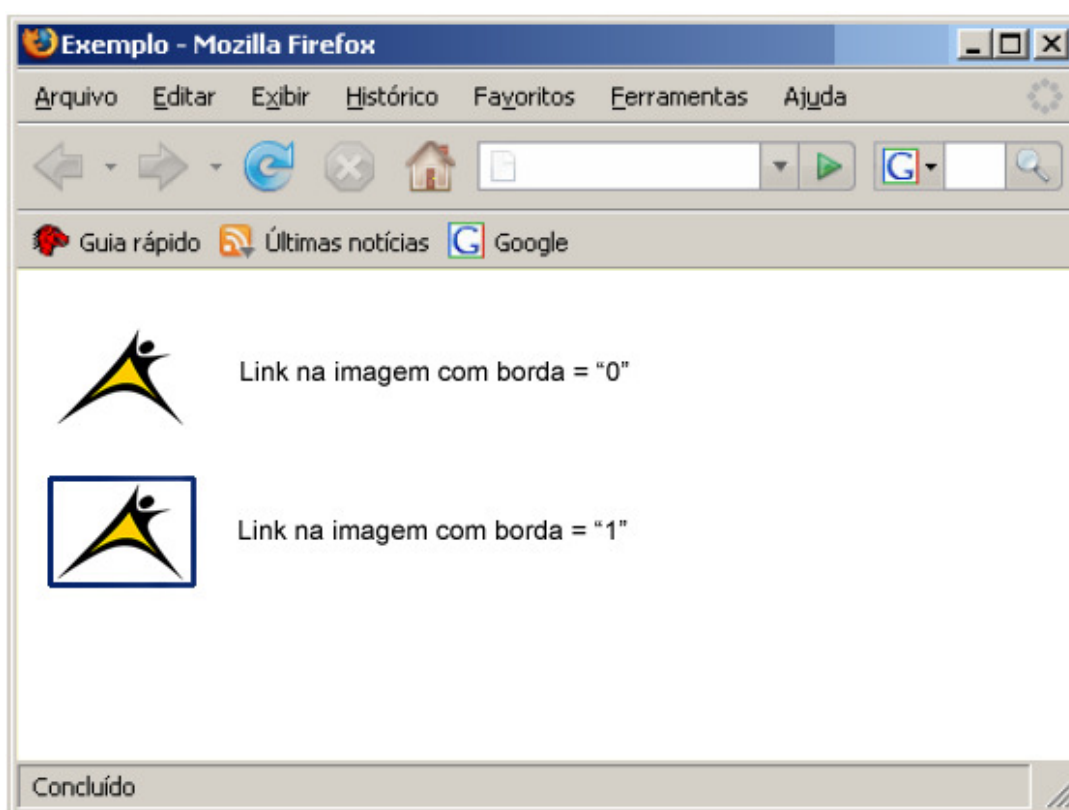
Veja os exemplos:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>

<p><a href="alcides.html"></a></p>

<p><a href="alcides.html"></a></p>

</body>
</html>
```



7.6 Criando âncoras:

Âncoras nada mais são do que links para dentro da própria página, permitindo que se navegue por dentro de uma página, quando ela possui um conteúdo muito extenso, é permitido também criar links para cada capítulo e ainda links para voltar ao topo da página.

Para definirmos uma âncora também utilizamos a tag `<a>` mas sem o atributo `href`, ao invés disso utilizamos o atributo **name**, que posteriormente será a referência do link.

A sintaxe para a criação de uma âncora ficará assim:

```
<a name="ancora">Texto da âncora</a>
```

A âncora pode ser criada em qualquer ponto do código HTML (mas normalmente é definida em pontos mais abaixo na página, e o link serve para que cheguemos rapidamente neste determinado ponto.

Para fazermos um link até a âncora usamos o seguinte código:

```
<a href="#ancora">Vai até âncora</a>
```

Onde o sinal de “#” representa que o link será para a própria página, mas que deve procurar nela a respectiva âncora. No caso de querermos criar um link de volta para o topo da página, não é necessária a criação de uma âncora para isso, basta criarmos um link com o código:

```
<a href="#top">voltar para o topo</a>
```

O link para “#top” sempre remeterá automaticamente para o topo da página

8. Tabelas

As tabelas definidas em HTML têm permitido apresentar dados em forma tabular e construir layouts. Apesar de as recomendações modernas darem preferência à utilização das Folhas de Estilos em Cascata (CSS) ao invés das tabelas quando se trata de fazer layouts complexos. A verdade é que as tabelas continuam a ser muito utilizadas para dados tabulares e a utilização para ajustes de layout por tabelas, apesar de não ser o recomendado pelo W3C, pode ser feito como forma de exercício de formatação.

8.1 Elementos básicos de uma tabela

A tabelas definem-se com a tag `<table>`.

A estrutura básica tabela divide-se em linhas (com o elemento `<tr>`, que o nome vem de “table row”), e cada linha divide-se em células (definidas com o elemento `<td>`, que o nome vem de “table data”). É possível inserir também um cabeçalho de texto com a tag `<th>`, que o nome vem de “table header”. Uma célula pode conter dados, texto, imagens e outros elementos do HTML.

Atributos Básicos de uma tabela:

Border: Define o número de pixels da borda principal da tabela.

Cellpadding: Define, em pixels, o espaço entre as bordas da célula e o conteúdo da mesma.

Cellspacing: Define o espaço entre as bordas (em pixels).

Align: Alinha horizontalmente a tabela em relação ao seu entorno.

Valign: Controla o posicionamento do conteúdo dentro das células.

Height: Define a altura da tabela em pixels ou porcentagem.

Width: Define a largura da tabela em pixels ou porcentagem.

Background: Permite-nos colocar um fundo para a tabela desde um link a uma imagem.

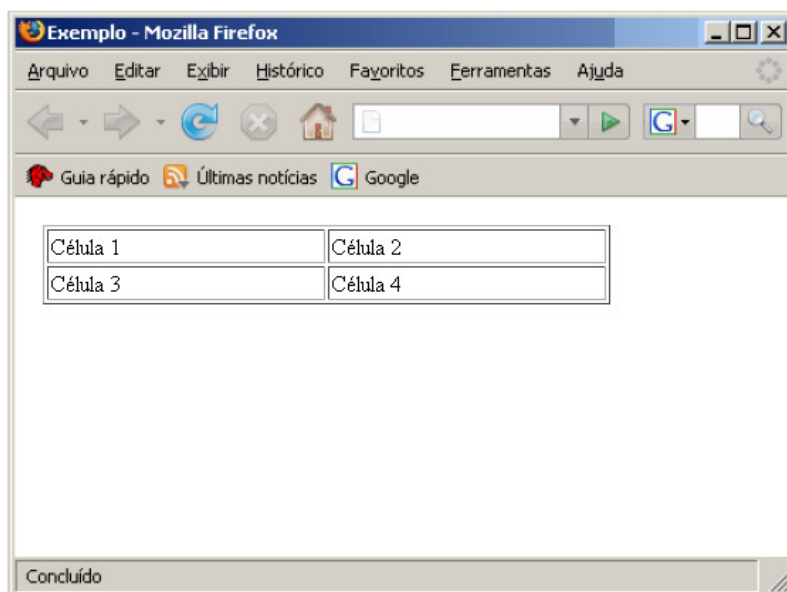
Bgcolor: Dá cor de fundo à tabela.

Rowspan: Define quantas linhas uma célula pode mesclar.

Colspan: Define quantas colunas uma célula pode mesclar.

Exemplo de uma tabela simples:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<table border="1" width="400">
<tr>
<td>Célula 1</td>
<td>Célula 2</td>
</tr>
<tr>
<td>Célula 3</td>
<td>Célula 4</td>
</tr>
</table>
</body>
</html>
```



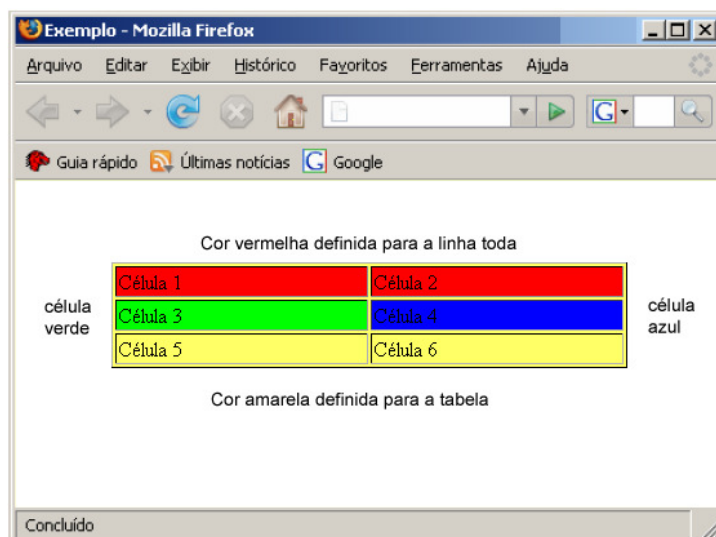
No exemplo acima criamos uma tabela com borda 1 (caso a borda não fosse definida não ficaria visível), com 400 pixels de largura e com duas linhas <tr>.

Na primeira <tr> colocamos as células 1 e 2 e na segunda <tr> colocamos as células 3 e 4.

8.2 Cores na tabela

Para trabalharmos com cores de fundo na tabela, utilizamos o parâmetro bgcolor e o código da cor. O bgcolor pode ser colocado como atributo da tag <table> se quisermos que toda a tabela tenha essa cor de fundo. Se quisermos que somente uma linha receba determinada cor de fundo, colocamos o bgcolor na tag <tr> e se for somente uma célula, colocamos o bgcolor na tag <td>.

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<table border="1" width="400"
bgcolor="#FFFF66">
<tr bgcolor="#FF0000">
<td>Célula 1</td>
<td>Célula 2</td>
</tr>
<tr>
<td bgcolor="#00FF00">Célula 3</td>
<td bgcolor="#0000FF">Célula 4</td>
</tr>
<tr>
<td>Célula 5</td>
<td>Célula 6</td>
</tr>
</table>
</body>
</html>
```



8.3 Mesclando células

Para mesclar linhas utilizamos o parâmetro `rowspan` e o número de linhas desejadas, já para mesclar colunas utilizamos o parâmetro `colspan` e o número de colunas.

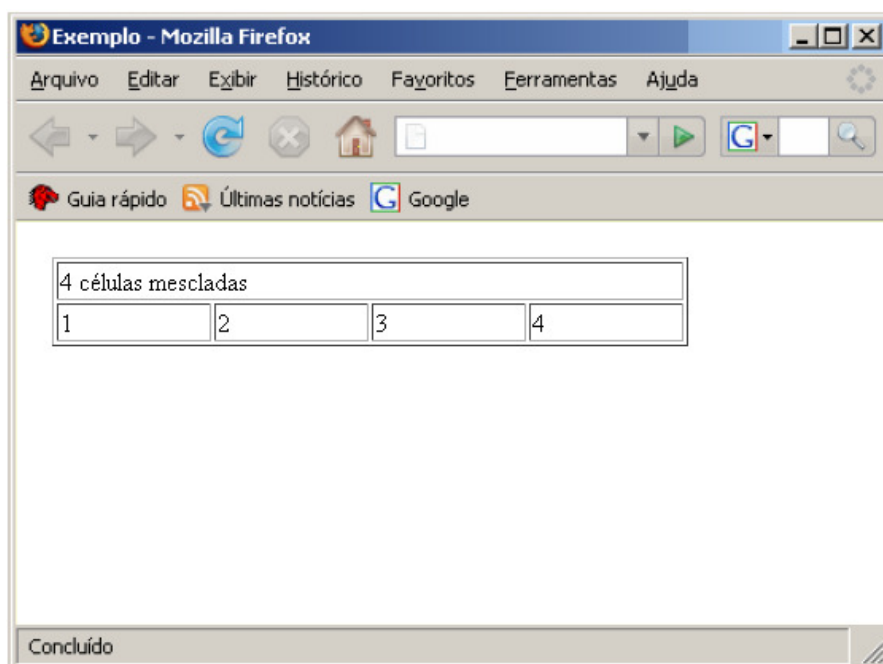
Exemplos:

`<td colspan="2">` **Mescla duas colunas**

`<td rowspan="4">` **Mescla quatro linhas**

Segue abaixo um exemplo de tabela com células mescladas

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<table border="1" width="400">
<tr>
<td rowspan="4">4 células
mescladas</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</table>
</body>
</html>
```



9. IFRAMES

iFrames (inline frames) é um recurso muito utilizado em websites. Consiste na inserção de páginas web (arquivos) dentro de páginas web (arquivos). A tag `<iframe>` é extremamente útil quando queremos criar uma rolagem em um local específico do site, não deixando o site inteiro muito longo.

Os parâmetros da tag `<iframe>` são:

name = nome dado ao iframe, servirá como o target dos links

src = é a página que abrirá inicialmente dentro do iframe

width = largura que o iframe terá, definida em pixels

height = altura largura que o iframe terá, definida em pixels

frameborder = borda do frame (se não quisermos borda, basta colocar "0")

scrolling = se haverá barra de rolagem no iframe ou não (as opções são "yes", "no" e "auto")

Para utilizarmos um iframe, devemos inserir no local desejado o seguinte código por exemplo:

```
<iframe name="iframe" src="pagina.html" width="600" height="400"
frameborder="0" scrolling="no"></iframe>
```

9.1 Links para Iframe

Para determinarmos um iframe como destino do hiperlink, ou seja, podemos fazer com que uma página especificada no atributo href da tag <a> seja exibida em um iframe.

Vejam os exemplos:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<title>Exemplo</title>
</head>
<body>
<p><a href="exemplo1.html" target="iframe"> brasil </a></p>
<p><a href="exemplo2.html" target="iframe"> porto alegre </a></p>
<div><iframe name="iframe" src="pagina.html" width="600" height="400"
frameborder="0" scrolling="no"></iframe></div>
</body>
</html>
```

10. Formulários

Os formulários servem para recolher dados introduzidos pelos usuários e enviá-los para processamento em um banco de dados ou para envio em um e-mail.

10.1 Criar um formulário

Um formulário é uma seção da página HTML que contém elementos que permitem ao usuário introduzir dados (elementos <textarea> e vários tipos de elementos <input>, <option> e <select>.) Estes elementos permitem inserir dados numéricos, textos curtos, textos extensos, selecionar elementos numa lista com várias escolhas, responder facilmente com respostas do tipo "sim" ou "não", selecionar rapidamente uma opção num grupo pequeno, etc.

Os formulários criam-se com o elemento <form>. Dentro desse elemento principal colocamos diversos elementos para a inserção dos dados. A tag <form> sempre deve ter um nome atribuído à ela.

```
<form name="dados">
<input>
.
.
.
<input>
</form>
```

O elemento <form> por si só não faz com que o browser desenhe nada na página nem permite inserir dados. Ele contém elementos que recolhem os dados e possui atributos que dizem ao browser como e para onde deve enviar os dados para processamento.

10.2 atributo “action” e Método

Para que o conteúdo de um formulário possa ser processado, dentro da tag <form> deverá haver um atributo chamado action="" onde dentro das aspas deverá ter o arquivo que processará o formulário. Este arquivo pode ser de extensão php, asp, cgi, pl, dependendo do sistema que foi utilizado. Também devemos definir o método do formulário, que pode ser “get” quando buscamos informações ou “post” quando enviamos informações.

É possível também criarmos uma página de resposta em html, apenas para testar se o formulário foi construído com sucesso. Esse exemplo ficaria assim:

```
<form name="dados" method="post" action="resposta.html">
```

10.3 Elementos para Formulários

<form> Define um formulário para recolher dados inseridos pelo usuário

<input> Insere um campo para introduzir dados

<textarea> Define uma área de texto (permite inserir texto com várias linhas)

<label> Define um nome para um elemento

<fieldset> Agrupa elementos num formulário

<legend> Define uma legenda para um grupo de elementos do formulário

<select> Define uma lista com várias opções selecionáveis

<option> Insere mais uma opção numa lista com várias opções selecionáveis

<button> Define um botão que pode ser pressionado

10.4 Inputs

O elemento que encontramos com maior frequência em formulários é o elemento <input>. Os campos <input> SEMPRE devem ter um nome atribuído para si, para sua identificação. O exemplo seguinte mostra um formulário simples com dois elementos input:

```
<form name="dados" method="post" action="resposta.html">  
Primeiro nome:  
<input type="text" name="primeiro_nome"><br>  
Último nome:  
<input type="text" name="ultimo_nome">  
</form>
```

Aspecto do formulário quando visualizado no seu navegador:

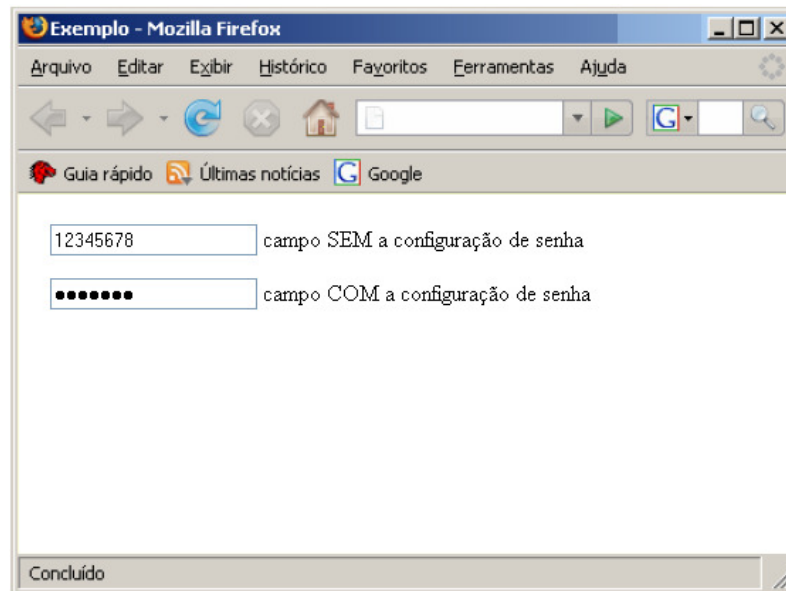
Primeiro nome:
Último nome:

10.4.1 Campo de senha

Podemos atribuir a um campo a função específica de campo de senha, para que não mostre os caracteres que foram digitados. Para isso devemos formatar o campo de texto da seguinte forma:

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

Um exemplo no navegador:



10.4.2 "Radio Buttons"

Os "Radio Buttons" ou "Botões de Rádio" são usados para criar um grupo (pequeno) de opções em que pode-se selecionar apenas uma das opções. Nos "Radio Buttons", para que isso funcione, todos eles devem ter o mesmo nome, além disso é importante que o atributo "value" corresponda à resposta marcada.

```
<form name="dados" method="post" action="resposta.html">  
Primeiro nome:  
<input type="radio" name="time" value="Gremista"> Gremista<br>  
<input type="radio" name="time" value="Colorado"> Colorado  
</form>
```

No navegador visualiza-se assim:

- ☐ Gremista
- ☐ Colorado

10.4.3 Checkboxes

As caixas de validação ("checkboxes") devem ser usadas sempre que queremos que o usuário aprove (ou não) itens dentro de um pequeno grupo. É permitido validar mais do que uma opção simultaneamente.

```
<form name="dados" method="post" action="resposta.html">
<input type="checkbox" name="roller">Eu ando de roller<br>
<input type="checkbox" name="skate">Eu ando de skate
</form>
```

No navegador visualiza-se assim:

- ☐ Eu ando de roller
☐ Eu ando de skate

10.5 Menu de Seleção única (<select>)

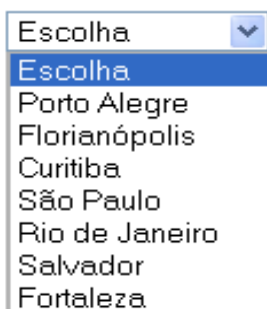
Estes tipos de menus, são particularmente práticos pois possibilita a inclusão de listas de opções longas, ocupando pouco espaço na tela.

Esse tipo de controle é criado pela tag `<select>...</select>`, de fechamento obrigatório. Cada opção do menu é identificada pela tag `<option> ... </option>` e dentro de cada option devemos ter o atributo "value" com a resposta referente a cada item.

Desta maneira:

```
<form name="dados" method="post" action="resposta.html">
<p>Escolha a cidade que deseja visitar</p>
<select name="cidades">
<option>Escolha</option>
<option value="Porto_Alegre">Porto Alegre</option>
<option value="Florianopolis">Florianópolis</option>
<option value="Curitiba">Curitiba</option>
<option value="Sao_Paulo">São Paulo</option>
<option value="Rio_de_Janeiro">Rio de Janeiro</option>
<option value="salvador">Salvador</option>
<option value="fortaleza">Fortaleza</option>
</select>
</form>
```

Escolha a cidade que deseja visitar



É possível também criar uma lista de opções rolável. Ela possui exatamente a mesma estrutura das tags do menu de seleção única, a diferença é a presença do atributo **SIZE** na tag `<SELECT>...</SELECT>`, que define o número de linhas a serem exibidas simultaneamente no menu. Se o nº de opções for maior que o valor definido em no atributo size, o navegador ativa automaticamente, uma barra de rolagem.

10.6 Botões de Envio e Apagar

Quando o usuário clica sobre o botão "Enviar" (ou "Submit"), as repostas e texto inseridos no formulário são enviados para processamento. O atributo `action` do elemento `<form>` contém o endereço (URL) do recurso da Web que está encarregado de realizar esse processamento. É para lá que o conteúdo do formulário é enviado. Já o botão de "Apagar" (ou "Reset") limpa todos os campos do formulário para uma nova consulta ou inclusão de dados

```
<form name="dados" method="post" action="resposta.html">
```

Nome de usuário:

```
<input type="text" name="usuario"><br> <br>
```

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

```
<input type="reset" value="Apagar">
```

```
</form>
```

Visualizando no navegador:

Nome de usuário: