

# **O HTML NÃO MORRE NUNCA!**

**CURSO BÁSICO DE HTML**

**Marcos Elias Picão**

Edições .MEP  
2007

Marcos Elias Picão é produtor dos sites e projetos do Grupo Mep Produções. Entre os destaques estão o blog Explorando e Aprendendo ([www.explorando.cjb.net](http://www.explorando.cjb.net)), que traz semanalmente muitas dicas de informática para todos os níveis, programas, sites, downloads, etc., e seus programas para Windows. Ele desenvolve o Mep Texto, um editor de textos puro muito prático para o trabalho com HTML, com diversos recursos, mas sem deixar de ser um editor de textos puro.

O conteúdo desta apostila é um agrupamento levemente editado do mini-curso de HTML escrito por ele, adaptado para páginas. A versão on line do curso continua na Internet, no site [www.mepsites.cjb.net](http://www.mepsites.cjb.net).

## Prefácio

*A criação de sites antigamente se resumia à produção de páginas HTML, que deviam ficar juntas de todas as imagens de um mesmo site, numa pasta em um servidor web. Hoje este cenário é muito diferente, um site feito apenas com o HTML seria muito pobre. Recursos que aplicam efeitos ou permitem uma maior interação com o usuário (cliente), como JavaScript, Flash, controles Active-X são essenciais para um site de boa aparência e funcionalidade. Ao mesmo tempo, do lado das máquinas onde os sites são hospedados (servidores), o processamento e armazenagem de informações são muito maiores e mais acessíveis. Entram aí tecnologias que permitem construir páginas dinâmicas, como PHP, ASP, JSP, etc. Recentemente, com a web 2.0, têm surgido uma série de sites mais interativos e melhores, usando um conjunto de diversas tecnologias já existentes, permitindo carregar informações mais rapidamente, inclusive onde apenas a parte afetada da página é recarregada e atualizada. O exemplo mais popular disso é o serviço de webmail do Google, o GMail, usando Ajax.*

*Se por um lado o HTML não serve para fazer coisas sofisticadas, por outro ele é a base de tudo. O JavaScript sem HTML não é nada. Um site todo feito em Flash precisa ser carregado dentro de uma página, por meio de comandos HTML. Linguagens de script do lado do servidor, como o PHP ou ASP, processam as informações recebidas através das páginas, HTML, e devolvem o resultado pronto ao cliente, em... Advinhe o quê? HTML! Eis a essência do título desta pequena obra para iniciantes: HTML nunca morre!*

*Claro, pode ser uma chatice e uma enorme perda de tempo ficar criando as páginas digitando o código HTML, isso atrasaria a produção. Ambientes de desenvolvimento web, como o Dreamweaver (da Adobe/Macromedia) ou o FrontPage ou Expression Web (ambos da Microsoft), facilitam a criação, manutenção e modificação de páginas da web. Editores menos sofisticados, incluindo editores gratuitos como o Composer, que vinha com o Netscape e depois foi separado no Mozilla Firefox, podem ser uma mão na roda para páginas isoladas, pequenas. Mas o bom webmaster deve conhecer senão todos, pelo menos a maioria dos comandos HTML. Não basta conhecer, tem que saber aplicá-los, em situações reais onde você pode corrigir erros ou fazer aquilo que não está conseguindo com um editor gráfico.*

*Sendo assim, mesmo com todo o avanço das tendências e tecnologias, não deixe de saber manusear o HTML. Este pequeno livro veio justamente para isso!*

*O autor*

## Índice

Introdução.....	4
Os arquivos.....	5
Textos.....	8
Estilos.....	14
Cores Gerais.....	15
Listas.....	15
Imagens.....	17
Links.....	20
Tabelas.....	23
Scripts.....	27
Formação Continuada.....	29
Manual de Referência Rápida.....	30

## Introdução

As páginas para a Internet são feitas usando a linguagem HTML. Mas o que vem a ser HTML?

HTML é uma linguagem, com regras e palavras que foi 'universalizada' e tornou-se a base dos sites da **web** na Internet. Ela indica, com instruções escritas, como as páginas devem ser. Indica que imagem ficará em que local, como e onde devem aparecer os textos da página, as imagens, etc. Ao navegar na Internet, o navegador (browser, programa que visualiza as páginas da Internet, como o Internet Explorer ou Mozilla, por exemplo) recebe os comandos (códigos) escritos em HTML, interpreta-os (lê) e mostra na tela o que estava indicado para mostrar. Desse modo, HTML se resume a textos, marcações, anotações, que fazemos e o navegador interpreta.

HTML significa "*HyperText Markup Language*", em português: "Linguagem de Marcação de HiperTexto". É uma linguagem onde você marca instruções de hipertexto. Hipertexto é o termo utilizado para os links, característicos das páginas web: um texto que, se clicado, abre uma outra página.

HTML é escrito em **texto puro**. O melhor editor para se criar e editar esses arquivos é um editor de texto puro, sem formatação, tal como o Mep Texto ou o próprio Bloco de notas, que vem com o sistema Windows. Os sistemas Linux também geralmente incluem diversos editores de texto puro. Ao preparar as páginas, estaremos digitando comandos para serem interpretados pelo navegador; não podemos formatá-los - pois não é uma pessoa que vai ler, mas é o computador. As imagens, formatações, e tudo mais de uma página da web são escritas indicadas. Você deve saber que num editor de textos puro não se pode inserir imagens junto com os textos. Para que elas apareçam na página da web, colocaremos 'comandos', 'instruções' indicando isso.

Por exemplo, para que um texto apareça em negrito na tela, usaremos na preparação da página o comando de negrito (que é **<b>**). Veja como deveria ser programado:

```
<b>Hoje eu estou aqui.</b>
```

Para que, no navegador, fosse exibido o seguinte resultado:

**Hoje eu estou aqui.**

Neste caso, o navegador interpreta **<b>** como comando de negrito: tudo o que estiver entre **<b>** e **</b>** aparecerá em negrito. E dessa forma temos muitos outros comandos que você vai conhecer ao longo do curso e no dia-a-dia, tais como para colocar letras em diferentes tamanhos, cores, imagens, **links**, etc.

Por falar nisso, os links são muito importantes: são eles que permitem a navegação, ou seja, clicando em textos ou imagens, o **internauta** é levado a outra página. Os links também podem ser para um endereço de e-mail, para descarregar (fazer **download**) de arquivos e programas, etc. Geralmente são usados textos e imagens nos links, mas também é muito comum o uso de outras tecnologias, como o **flash**, por exemplo.

## Os arquivos

As páginas da Internet são um tipo de arquivo de computador, de **extensão** .HTM ou .HTML. Estes arquivos armazenam as informações de como será a página, e eles são carregados e lidos pelo navegador quando um site ou página é acessado. Basicamente há um arquivo HTML para cada **página da web**. Nesta etapa vamos mostrar como criar e editar esses arquivos, passo essencial para se treinar qualquer código HTML. Depois começaremos a criar as páginas propriamente ditas.

Você vai precisar apenas de dois programas básicos: um editor de textos puro e um navegador. A dupla ideal seria o Bloco de Notas e o Internet Explorer, do Windows. Fazemos no bloco de notas e visualizamos no Internet Explorer. Recomendamos utilizar o **Mep Texto**, se possível, no lugar do Bloco de notas, pois ele facilita bastante o trabalho com arquivos HTML. Baixe-o em <http://meptexto.mepsites.cjb.net>.

Nos exemplos deste curso, utilizamos o Bloco de notas, o Internet Explorer e estamos executando o Windows 98.

Esse processo consiste no seguinte: digitamos os códigos no Bloco de Notas, salvamos o arquivo com a extensão .htm e abrimos ele no Internet Explorer. Durante a edição, podemos trabalhar com as duas janelas abertas. É só ir salvando no Bloco de Notas e atualizando no Internet Explorer. Você não precisa estar conectado à Internet para fazer, nem para visualizar. Os arquivos HTML são como os outros, podem ser vistos e trabalhados no próprio computador. Você só precisará da conexão para publicá-los, e para acessar o site (veremos isso mais pra frente). Você pode copiar os arquivos para um disquete, por exemplo, e publicar em outro computador - não necessariamente onde se fez as páginas.

Obs.: para digitarmos os comandos, podemos usar letras maiúsculas ou minúsculas, o resultado é o mesmo. Por exemplo, <HTML>, <html> e <Html> são a mesma coisa, e isso não gera nenhum erro. Escreva como preferir, mas procure manter uma organização para facilitar a você mesmo: tudo em minúsculas, ou tudo em maiúsculas.

Toda página HTML tem uma estrutura, que deve ser respeitada. Essa estrutura identifica o arquivo como sendo uma página, e delimita as regiões da página. Essa estrutura é a seguinte:

```
<html>
<head>
<title>Título da Página</title>
</head>
<body>

Conteúdo da página

</body>
</html>
```

Como você pode perceber, a regra geral do HTML é que os comandos têm uma marcação (**tag**) de abertura e de fechamento. A tag <html> identifica que a

página é uma página HTML. Tudo relativo a essa página deve estar entre `<html>` e `</html>`.

O cabeçalho contém informações sobre a página, mas que não aparecem na área de visualização do navegador. A principal destas informações é o título, que geralmente é exibido na barra de título do navegador (a barra superior da janela, geralmente na cor azul) e utilizado na catalogação dos sistemas de busca. `<head>` determina o cabeçalho, e é encerrado com `</head>`. Repare que sempre ao encerrar a tag, utiliza-se uma barra (/).

O corpo da página, delimitado pelas tags `<body>` e `</body>` é a parte principal. Tudo o que for aparecer na página deve estar dentro da tag `<body>`, como os textos e imagens, por exemplo. A expressão "dentro da tag" se refere a estar "entre" a tag de abertura e de fechamento.

Encerramos a tag `<body>` e a página, com `</body></html>`.

Esse padrão deve ser seguido em todas as páginas, e em cada página que você montar deve antes de tudo digitar a estrutura básica do HTML, para depois inserir o conteúdo. Os códigos HTML podem ser digitados numa única linha, mas dar "Enter" e organizar o encadeamento facilita a edição, pois para modificar as páginas devemos alterar o seu **código fonte**.

Para começar, vamos construir uma página simples, que servirá de base para as demais. Abra o Bloco de Notas e digite o seguinte:

```
<html>
<head>
<title>Minha Primeira Página</title>
</head>

<body>

Oi! Esta é minha primeira página HTML...

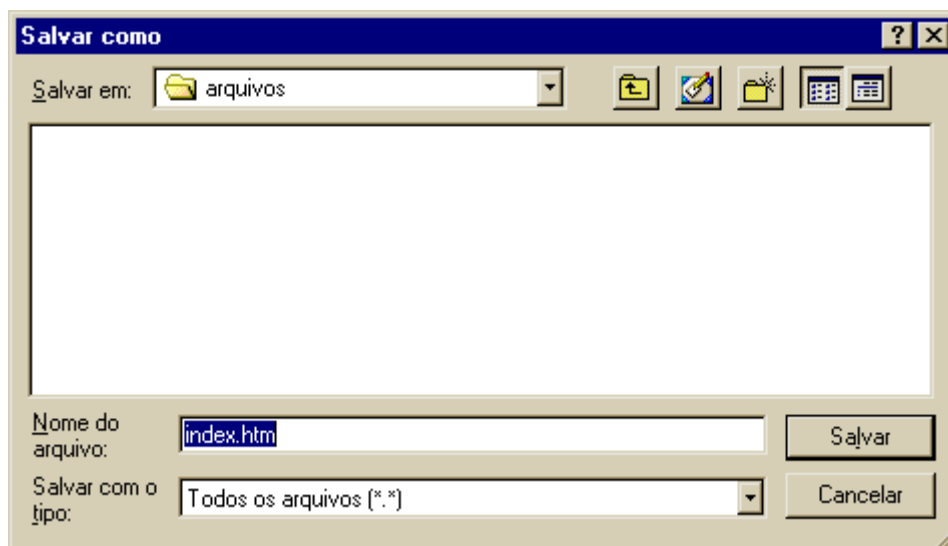
</body>
</html>
```

Para você não se confundir, tudo o que tiver que digitar estará escrito na fonte `Courier New`, e em **roxo**. Trabalhe com atenção para não se esquecer de fechar uma tag. Isso poderia gerar um erro na interpretação da página pelo navegador.

Vamos salvar o arquivo. Crie uma pasta no seu computador ou num disquete, para guardar o que você está fazendo - se você seguir esse curso, precisará das páginas já feitas, em especial ao estudar os links. A primeira página de um site, a que aparece logo ao entrar no endereço, tem um nome padrão: geralmente **index.htm**. Dependendo do servidor, pode ser **default.htm**. Ela é exibida primeiro, é a home page, a página de entrada. Nas outras páginas o visitante chega através dos links, ou então digitando o endereço completo dela. É bom saber que em html não podemos dar espaços nos nomes dos arquivos (na verdade até podemos, mas isso poderia trazer problemas; veja aqui porque é bom evitar...). Use apenas letras, números, - ou \_. Não use pontos, espaços nem caracteres especiais. É recomendável ainda dar os nomes dos arquivos tudo com

letras minúsculas. Por exemplo, meu\_site.htm, aventurasfantasticas.htm, dica01.htm, etc.

Clique no menu **Arquivo**, e escolha **Salvar como**. Só na primeira vez você escolhe salvar como; nas atualizações, depois que a página já tem um nome e está num local, é só clicar em Arquivo e escolher Salvar. Observe:

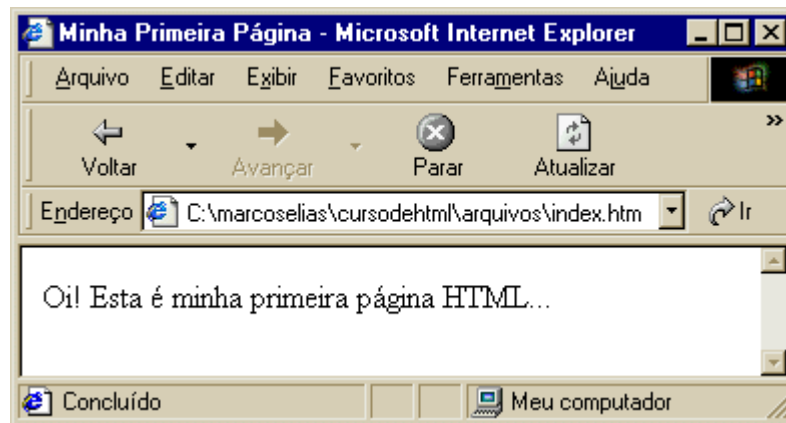


Na caixa de diálogo "Salvar como", dê o nome de **index.htm**, e no campo "Salvar com o tipo" escolha **Todos os arquivos**. Salve na sua pasta ou no seu disquete.

Observação: você pode salvar as páginas com a extensão ".htm" ou ".html", a que você preferir. Porém, nas páginas de um mesmo site, utilize apenas uma das extensões. Em diversos locais você deverá colocar referências a outras páginas, e deverá informar o nome completo, com a extensão. Neste curso, utilizamos nos exemplos apenas a extensão ".htm".

No Bloco de notas é necessário escolher a opção "Todos os arquivos", pois senão o bloco de notas salvaria como texto - o que na verdade é uma página da web. O Mep Texto oferece a possibilidade de salvar diretamente no formato .htm ou .html, sem precisar digitar a extensão.

Feito isso, vamos visualizar. Pode deixar o Bloco de Notas aberto, não precisa fechar. Abra a sua pasta ou o seu disquete, e dê um clique duplo (dois cliques sucessivos e rapidamente) sobre o ícone do arquivo index. No caso do Mep Texto, basta estar com o arquivo aberto e escolher a opção "Visualizar no navegador", no menu HTML. Ele será aberto no Internet Explorer (ou no navegador que você tiver instalado como padrão). Veja:



Observe o título, na barra azul, com o que você definiu entre as tags <title> e </title>. O texto aparece na fonte e tamanhos padrão definidos pelo usuário do navegador, geralmente a fonte é Times New Roman.

Isso é o básico da estrutura e criação dos arquivos HTML. Para cada página, lembre-se de dar a extensão .htm e salvar com o tipo "Todos os arquivos" (no Bloco de notas).

Veja agora como editar: No Bloco de notas, clique em Arquivo, escolha Abrir, localize o arquivo desejado e pronto. No Mep Texto existem diversas formas, a mais prática é clicar com o botão direito do mouse sobre o ícone da página a ser editada e escolher "Enviar para > Mep Texto". Aí é só modificar e clicar em Salvar. Só use a opção "salvar como" na primeira vez que você salvar um arquivo, ou depois, se quiser salvá-lo em outro local ou com outro nome.

## Textos

Com textos vamos aprender a definir fontes, tamanhos e cores, parágrafos e depois em formatação de estilos.

### Fontes

Definir fontes é muito fácil, mas as fontes escolhidas não são salvas com a página - o usuário tem diversas fontes instaladas no computador, e são com elas que ele visualiza as páginas da Internet. Desse modo, prefira utilizar as fontes mais comuns, como Arial, Verdana, Times New Roman, Courier New, Tahoma, Century, etc. Se você definir uma fonte, e o visitante não tê-la no computador, ele vai ver o texto com a fonte padrão do navegador - provavelmente a Times New Roman.

Usamos a tag <font>, e o fechamento </font>.

Dentro desta tag, na de abertura somente, inserimos os atributos, que servem para reforçar um comando. Eles são separados uns dos outros por um espaço, e seus valores devem estar entre aspas. Veja um exemplo:

```
<font face="arial" size=4 color="red">Olá pessoal!</font>
```

Isso geraria um efeito assim:

**Olá pessoal!**

Nesse caso, o atributo **face** determina o tipo, o nome da fonte. Esse nome deve estar entre aspas. Por exemplo, `face="Verdana"` ou `face="Courier New"`.

O atributo `size` determina o tamanho da fonte. Em HTML temos 7 tamanhos, que vão de 1 (o menor) a 7 (o maior). O tamanho padrão, ou *default*, é o 3. Isso quer dizer que se você não definir o tamanho, ela aparecerá automaticamente no tamanho número 3. Veja um exemplo com cada um para ver os tamanhos possíveis:

Fonte com `size=1`

Fonte com `size=2`

Fonte com `size=3`

Fonte com `size=4`

Fonte com `size=5`

Fonte com `size=6`

Fonte com `size=7`

O número deve ser colocado preferencialmente sem aspas, apenas use aspas nos atributos se forem textos ou expressões.

E o atributo **color** determina a cor. Esta deve ser escrita entre aspas; pode ser o nome em inglês para as cores mais comuns, ou o código hexadecimal dela. Cada cor tem um código utilizado na Internet, que a identifica e que é único. Ele tem o formato `#xxyyzz`, onde `xx` indica a coloração de vermelho, `yy` de verde, e `zz` de azul. Algumas cores e seus códigos hexadecimais:

Branco: `#FFFFFF`  
Preto: `#000000`  
Vermelho: `#FF0000`  
Verde: `#00FF00`  
Azul: `#0000FF`  
Amarelo: `#FFFF00`

No código hexadecimal, nunca se esqueça de escrever entre aspas e com o símbolo `#` (na maioria dos teclados, `Shift + 3`). Por exemplo, `color="#FF0000"`. Prefira utilizar os códigos em vez dos nomes em inglês; eles são aceitos por todos os navegadores, enquanto os nomes em inglês podem não ser válidos para todos.

Dica: para obter o código de alguma cor você pode consultar a tabela de cores, nos acessórios do site [www.mepsites.cjb.net](http://www.mepsites.cjb.net).

Abra o arquivo index (se você não o tem, consulte o item anterior e o faça - clique em "Os Arquivos", no menu ao lado), e editaremos algumas coisas. Lembre-se: Bloco de notas, Arquivo, Abrir.

Coloque os códigos destacados, antes e depois da frase, de modo que fique assim:

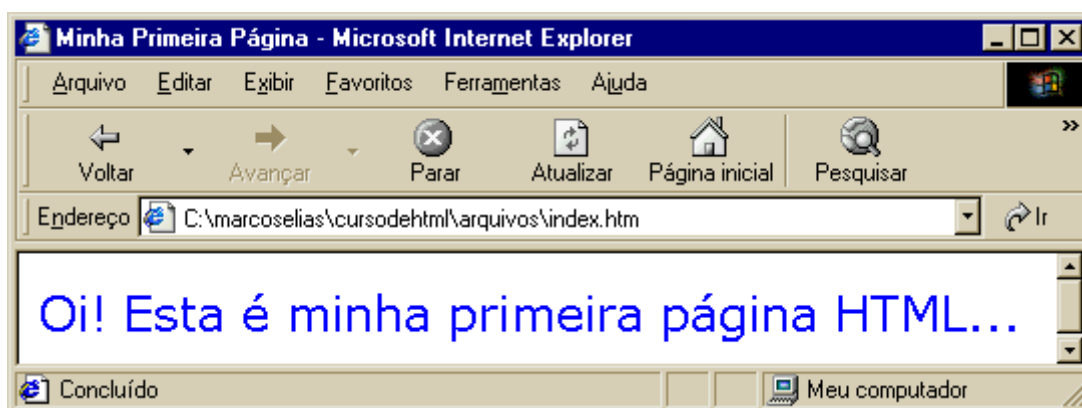
```
<html>
<head>
<title>Minha Primeira Página</title>
</head>

<body>

<font face="verdana" size=5 color="#0000FF">Oi! Esta é minha
primeira página HTML...</font>

</body>
</html>
```

Clique em Arquivo, escolha Salvar. Abra o arquivo index no Internet Explorer, e visualize: (se você ainda estiver com a página aberta no Explorer, clique no botão Atualizar, para recarregá-la sem ter que abri-la novamente)



Observe a diferença... Bastante, não?

Você pode usar a tag <font> com apenas 1 ou 2 atributos, não é necessário os três. Por exemplo, <font size=4> determinaria o tamanho do texto igual a 4, e a fonte e a cor ficariam com o padrão. Vale notar também que na tag do fechamento não se coloca os atributos.

### Parágrafos

Os parágrafos consistem em blocos de textos (com imagens ou não) agrupados por assunto. Em HTML ele é definido pela tag <p>, com fechamento </p>.

Se você tentou, deve ter percebido que ao dar "Enters" no código fonte da página, o resultado no navegador era contraditório: os textos não obedeciam as quebras de linha. Para isso devemos usar parágrafos (<p>) ou a tag de quebra de linha (<br>).

Veja um exemplo:

```
<p><font face="arial" size=3>Meu nome é Marcos. Gosto de internet,  
informática, ouvir música, jogar conversa fora, ouvir rádio...</font></p>  
<p><font face="arial" size=3>Montei meu site uma vez que ouvi no rádio  
um cara divulgando o dele, em um provedor gratuito - eu não sabia que era  
gratuito. Entrei, vi o link do hpG e não parei mais... E cada vez  
mais.</font></p>
```

Que geraria:

Meu nome é Marcos. Gosto de internet, informática, ouvir música, jogar conversa fora, ouvir rádio...

Montei meu site uma vez que ouvi no rádio um cara divulgando o dele, em um provedor gratuito - eu não sabia que era gratuito. Entrei, vi o link do hpG e não parei mais... E cada vez mais.

Nas páginas web, é mantido um espaço padrão entre os parágrafos. Se você não quer esse espaço padrão, não formate com parágrafos. Use a tag `<br>`, de quebra de linha. Ao inserir `<br>`, o texto seguinte ficará na linha de baixo, sem o espaço interparagrafar. Também pode ser usada em imagens, que você verá depois, para deixar uma embaixo da outra - e não na mesma linha. Note que a tag `<br>` é única: não tem as tag de fechamento; portanto, não use `</br>`. E para deixar mais linhas, dê mais tags `<br>` seguidas.

Usando a tag `<br>`:

```
<font face="arial" size=3>Meu nome é Marcos. Gosto de internet,  
informática, ouvir música, jogar conversa fora, ouvir rádio...<br>Montei  
meu site uma vez que ouvi no rádio um cara divulgando o dele, em um  
provedor gratuito - eu não sabia que era gratuito. Entrei, vi o link do hpG e  
não parei mais... E cada vez mais.</font>
```

Que geraria:

Meu nome é Marcos. Gosto de internet, informática, ouvir música, jogar conversa fora, ouvir rádio...

Montei meu site uma vez que ouvi no rádio um cara divulgando o dele, em um provedor gratuito - eu não sabia que era gratuito. Entrei, vi o link do hpG e não parei mais... E cada vez mais.

## Alinhamento

O alinhamento centralizado pode ser obtido com o uso da tag `<center> ... </center>`. Abra o bloco de Notas e crie um novo arquivo, dê o nome de `paragrafos.htm` (não se esqueça de escolher a opção 'Salvar com o tipo Todos os arquivos'). Salve na mesma pasta ou disquete que está o `index`. Digite o seguinte, e salve:

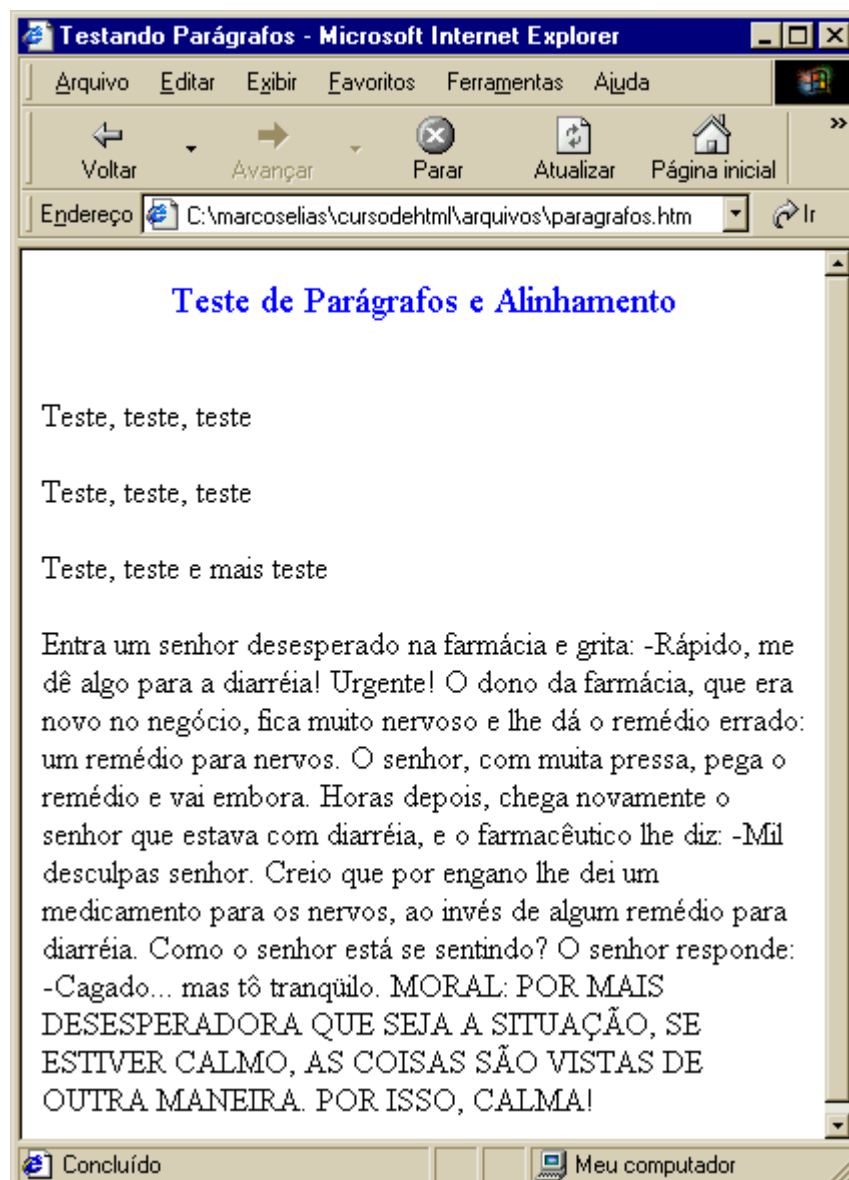
```
<html>  
<head>  
<title>Testando Parágrafos</title>  
</head>
```

```
<body>

<center><font face="verddana" size=4 color="#0000FF">Teste de
Parágrafos e Alinhamento</font></center>
<br><br>
<p>Teste, teste, teste</p>
<p>Teste, teste, teste</p>
<p>Teste, teste e mais teste</p>
<p>Último parágrafo... Coloque aqui um texto que dê algumas
linhas...</p>

</body>
</html>
```

Visualize o arquivo. No último parágrafo, coloque um texto qualquer - digite ou copie de algum lugar, de modo que no Internet Explorer ocupe umas 6 linhas ou mais.



O atributo que define o alinhamento nos parágrafos é **align**, e pode ter os valores **left** (esquerda), **right** (direita), **center** (centralizado) ou **justify** (justificado). Cuidado: é ali**GN**, não ali**NG**. Esse erro é comum, mas a visualização não entenderia.

Modifique os códigos da página que você fez, deixando assim:

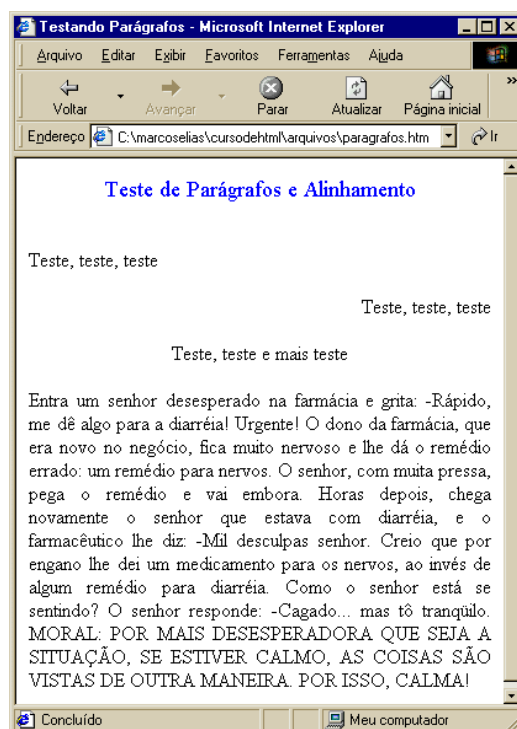
```
<html>
<head>
<title>Testando Parágrafos</title>
</head>

<body>

<center><font face="verddana" size=4 color="#0000FF">Teste de
Parágrafos e Alinhamento</font></center>
<br><br>
<p align="left">Teste, teste, teste</p>
<p align="right">Teste, teste, teste</p>
<p align="center">Teste, teste e mais teste</p>
<p align="justify">Último parágrafo... Coloque aqui um texto que
dê algumas linhas...</p>

</body>
</html>
```

Salve e atualize no Internet Explorer:



Observe os alinhamentos, em especial o do quarto parágrafo.

## Estilos

Os estilos são muito fáceis de aplicar e decorar. Você pode combiná-los, por exemplo, negrito e itálico. Só não se esqueça de fechar a tag, senão a formatação se estende para tudo o que estiver depois dela.

### Negrito

Use a tag `<b>` e `</b>`.

`<b>Exemplo</b>` geraria:

#### **Exemplo**

### Itálico

A tag é `<i>` e `</i>`.

`<i>Texto Inclinado</i>` geraria:

*Texto Inclinado*

### Sublinhado

A tag é `<u>`. Não é bom usar esse efeito, pois geralmente os links são textos sublinhados, e se você usar esse comando a pessoa pode tentar clicar num texto pensando que é um link.

`<u>Sites para a Internet</u>` geraria:

Sites para a Internet

### Sobrescrito

Esse efeito coloca o texto em forma sobrescrita, como nos expoentes. Usa-se o par `<sup>` e `</sup>`. Exemplo:

`Windows<sup>xp</sup>` geraria:

Windows<sup>xp</sup>

### Subscrito

Um efeito usado na notação de índices. Usa-se o par `<sub>` e `</sub>`. Exemplo:

`Índice<sub>1</sub>, índice<sub>2</sub>`. geraria:

Índice<sub>1</sub>, índice<sub>2</sub>.

É bom você dar uma treinada, formatando e desformatando textos e páginas.

## Cores Gerais

Você pode predefinir a formatação de cores das páginas na tag <body>, assim não precisa aplicar cores aos textos em todos os parágrafos.

Veja a tag <body> com alguns atributos:

```
<body text="#000000" bgcolor="#FFFF00" link="#0000FF"
vlink="#0000FF" alink="#FF0000">
```

**text** determina a cor dos textos na página. Se você quiser usar cores diferentes em frases, palavras ou parágrafos, use a tag <font>. A cor padrão geralmente é preto.

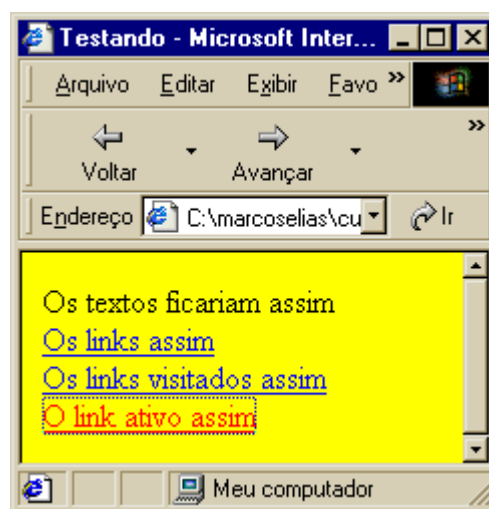
**bgcolor** determina a cor de fundo da página. Se você não definir, aparecerá a cor de fundo padrão do navegador - geralmente branco ou cinza. O ideal, para garantir que todos os visitantes verão a página com fundo branco, é definir bgcolor="#FFFFFF". bgcolor vem de background-color, ou cor de fundo.

**link** determina a cor dos links da página. O padrão é azul.

**vlink** determina a cor dos links visitados, ou já exibidos no navegador. Geralmente vem um tom roxo-acinzentado como padrão.

**alink** é para a cor do link ativo, isto é, a cor do link enquanto o internauta clica nele. O padrão é azul.

A tag <body> ilustrada acima geraria uma página assim:



## Listas

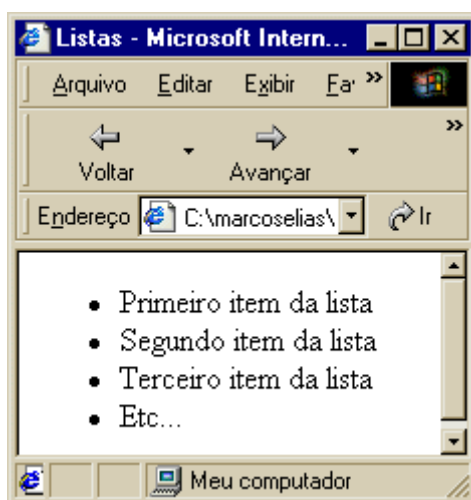
As listas servem para enumerar, listar um conjunto de itens. Em HTML temos três listas, a com marcadores, a numerada e a de definição. Nesse curso básico veremos as duas primeiras.

### Com Marcadores

A lista com marcadores é definida pela tag `<ul>`, e tem a seguinte estrutura:

```
<ul>  
<li>Primeiro item da lista  
<li>Segundo item da lista  
<li>Terceiro item da lista  
<li>Etc...  
</ul>
```

Repare que não se fecha a tag `<li>`, e a tag `<ul>` só é fechada ao encerrar a lista. Isso geraria um resultado assim:



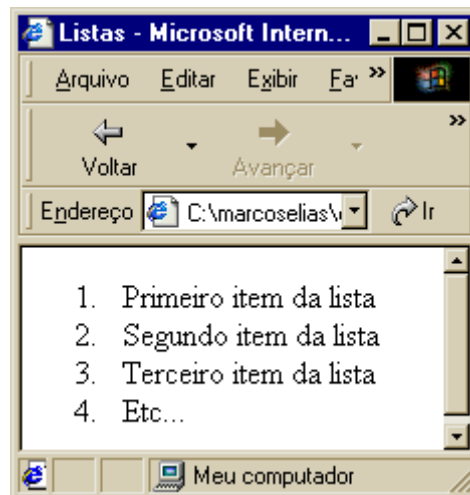
### Numerada

A vantagem de usar a lista numerada em vez de colocar os números um por um, à mão, é que ela é atualizada automaticamente sempre que você inserir ou remover algum item, em qualquer posição. Se fosse digitada à mão, você deveria alterar os números, caso fizesse modificações no meio da lista.

A lista numerada é definida pela tag `<ol>`, e é semelhante à com marcadores:

```
<ol>  
<li>Primeiro item da lista  
<li>Segundo item da lista  
<li>Terceiro item da lista  
<li>Etc...  
</ol>
```

Veja o resultado:



As listas também são muito usadas para colocar links, deixando a página mais organizada.

## Imagens

Colocar imagens em uma página é muito, muito fácil. Mas no início, sua maior dificuldade poderá ser ajustar a imagem, para que ela fique como você desejava. Ainda assim, no HTML temos muitas limitações no que se refere à apresentação visual, sem contar que uma mesma página pode ser exibida de maneiras um pouco diferente em diferentes navegadores. Para facilitar, procure utilizar imagens com o tamanho certo, para não ter que redimensioná-las.

Se você mexe com computador já algum tempo, deve saber que imagens em bitmap (.bmp) são muito grandes em tamanho de arquivo - ocupam bastante espaço. Assim sendo, demoram muito para carregar, e para a Internet estão descartadas. O ideal é utilizar **somente** imagens nos formato GIF ou JPG (o mesmo que JPEG). Alguns navegadores exibem outras imagens, mas para evitar erros e dores de cabeça, utilize esses formatos. Se suas imagens estão em bitmap, você precisa usar um programa que a converta em GIF ou JPG. O GIF é ideal para logotipos, imagens sem muita resolução e cores. O JPG é ideal para fotos bonitas ou grandes, e de qualidade, é claro. Em JPG a imagem pode ter muito mais cores e uma resolução e profundidade bem maiores - gerando uma imagem mais limpa e nítida, se ampliada, por exemplo.

Para colocar uma imagem numa página da web, esta deve estar na **mesma pasta** em que está a página que a conterá. Quando uma imagem está num servidor web, você pode usá-la colocando o endereço completo dela (URL), por exemplo, "http://home.meusite1.com.etc/imagens/logos/comp-net.gif". No caso de uma imagem local, use o nome dela completo, com a extensão. Por exemplo, "lasgo.gif", logo\_energia.jpg", "praia01425.jpeg".

No Bloco de Notas não dá para colar imagens. Para que ela apareça na Internet, vamos colocar um comando HTML que indique isso. Esse comando é: <img>. Veja seus principais atributos:

```

```

**src** determina o endereço, ou o nome da imagem. Se ela está na mesma pasta da página, basta colocar seu nome - com a extensão. Se ela está na web, pode-se colocar o endereço dela completo, com http://.

**border** determina uma borda em torno da imagem, em pixels. Um pixel é cada ponto de luz que seu monitor pode exibir. Geralmente, para a Internet, usa-se a resolução de 800 x 600, isso quer dizer que a tela está dividida em 800 pontos na horizontal por 600 na vertical. São esses pontos que formam todo o que aparece na tela, usando combinações de cores. Você pode usar border=1, border=2, etc. O ideal, para que apareça somente a imagem, é usar border=0.

**bordercolor** determina a cor da borda, desde que você tenha colocado um valor diferente de zero no atributo border. Se você colocou border=0, ou não colocou border, nem precisa colocar bordercolor.

**alt** - Antigamente muitos navegadores não exibiam imagens. O alt servia como um texto alternativo: nesses navegadores aparecia o texto, em vez da imagem. Hoje ele é utilizado como uma informação a mais sobre a imagem, e geralmente aparece enquanto ela está carregando ou quando deixamos o cursor do mouse sobre a imagem por um tempo. É opcional.

**width** - determina a largura da imagem, que pode ser em pontos (pixels) ou em porcentagem da janela atual. Só deve ser usado em casos que você queira redimensionar a imagem.

**height** - semelhante a width, determina a altura da imagem. Se você definir qualquer valor para width e height, a imagem poderá ficar muito distorcida. Você deve diminuir proporcionalmente. Por exemplo, pegar as dimensões originais da figura, dividir por 2 os dois números, e colocar a largura e altura de acordo com esses números. Mas, como foi dito, é bom você usar de início imagens no tamanho real, sem definir width nem height.

**align** - determina o alinhamento da imagem com relação aos textos em volta dela. Num curso inicial, o ideal é usar left (esquerda), right (direita) ou center (centralizado).

Nem todos precisam ser colocados. Para que a imagem apareça, basta colocar ``, onde imagem.gif é o nome da imagem, com sua extensão - desde que ela esteja na mesma página em que deverá aparecer. Observe também que a tag `<img>` não tem fechamento, portanto, não existe `</img>`.

Para acompanhar os procedimentos seguintes, você precisará de uma imagem não muito grande, de preferência em JPG ou GIF. Nos exemplos está sendo usada uma imagem de Nova York, nos EUA.

No Bloco de Notas crie uma nova página, e salve-a com o nome de imagens.htm. Digite o seguinte:

```
<html>
<head>
<title>Testando imagens</title>
</head>

<body bgcolor="#FFFFFF" text="#000000">
```

```

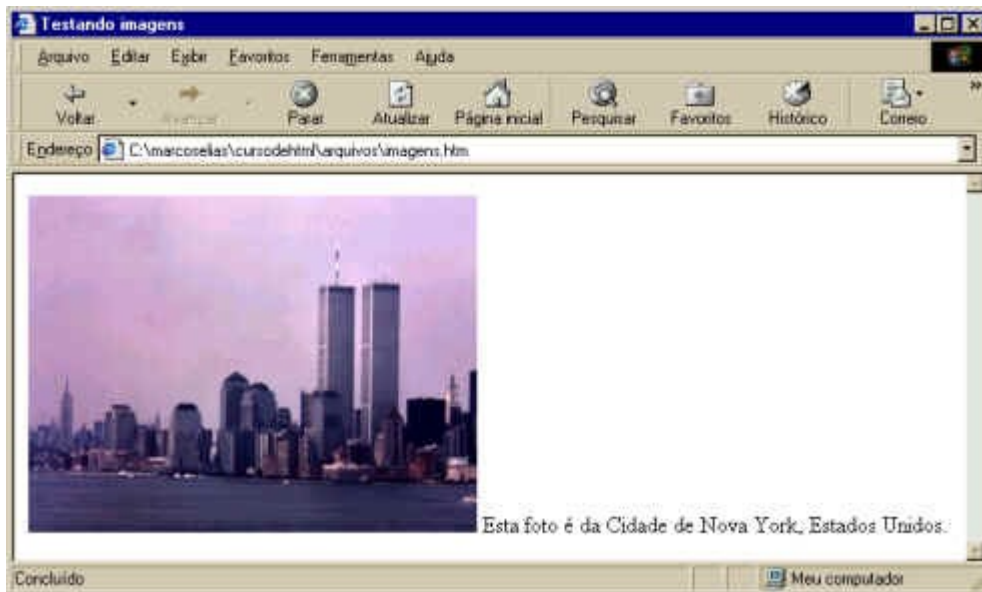
```

Esta foto é da Cidade de Nova York, Estados Unidos.

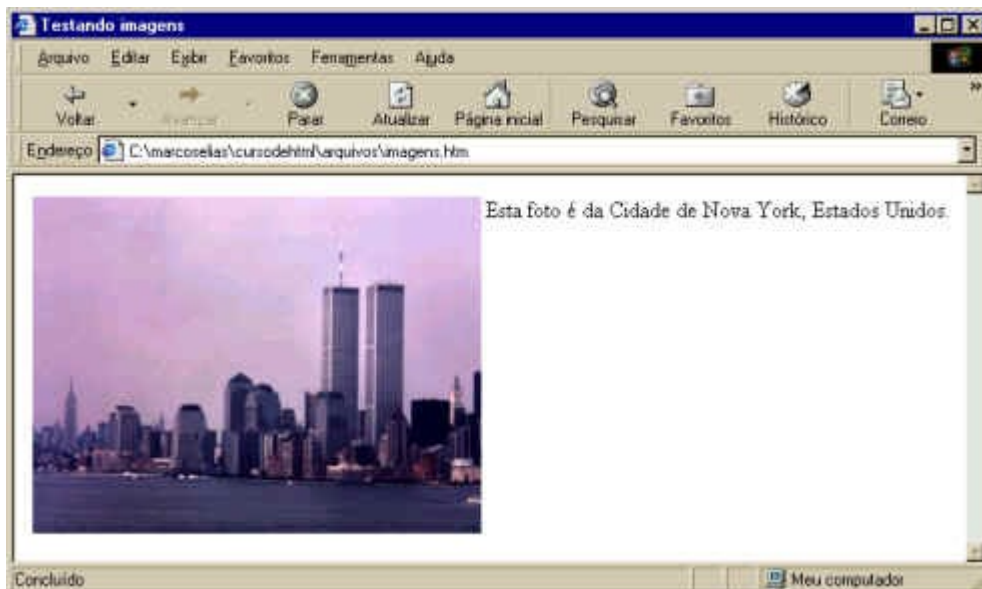
```
</body>
```

```
</html>
```

Salve e visualize:



Altere a tag img, insira align="left", salve e observe:



Experimente o right e center. Se quiser, pode deixá-la sem align, e colocar depois da imagem e antes do texto uma ou mais quebras de linha, com a tag `<br>`. Para centralizá-la, você pode usar a tag `<center>` antes da tag da imagem, e `</center>` depois dela.

Para publicar imagens nos blogs, prefira usar a ferramenta deles para isso. Em casos que não permitam, não dá para carregar as imagens para lá. Então deve-se publicá-la em algum outro local e inserir a tag com o endereço completo, exemplo:

```

```

Com o tempo você pega mais coisas. Agora que já tem o início, fica mais fácil.

## Links

Os links constituem a base da navegação. Quando uma pessoa acessar seu site, a página que ela vai ver é a index.htm. Na index (que seria *índice*, traduzido) você deve colocar links para as outras páginas. E nessas outras páginas, também, criando uma estrutura de navegação que facilite ao visitante ir para qualquer página, de qualquer página. Você pode pôr um link para a página inicial (index), a pessoa clica ali e escolhe outra página para ver.

Os links podem ligar páginas, imagens, arquivos, outros sites ou até mesmo o envio de uma mensagem para um endereço de e-mail - isso se o visitante tiver um programa de e-mail instalado e configurado corretamente, como o Outlook Express, por exemplo.

Um link se resume à seguinte sintaxe:

```
<a href="pagina.htm">Clique aqui</a>
```

O comando <a> vem de âncora, que também pode linkar para partes dentro da própria página. O atributo href determina a página ou arquivo que será aberto. Entre <a ...> e </a> fica o texto em que a pessoa clicará - ou uma imagem. No navegador, os links aparecerão na cor padrão (geralmente azul) e sublinhados. Se você definiu uma cor para os links na tag <body>, será essa cor que será usada.

Vamos modificar a página index, e colocar um link para as páginas paragrafos.htm e imagens.htm.

Abra o bloco de notas, clique em Arquivo, escolha Abrir e localize o arquivo index.

Insira o que está destacado:

```
<html>
<head>
<title>Minha Primeira Página</title>
</head>

<body>

<font face="verdana" size=5 color="#0000FF">Oi! Esta é minha
primeira página HTML...</font>

<a href="paragrafos.htm">Veja os testes de parágrafos</a>
<br>
```

```
<a href="imagens.htm">Veja os testes de imagens</a>

</body>
</html>
```

Visualize no navegador e clique nos links para testar.

Você pode, é claro, formatar os links de acordo com seu gosto, levando em consideração um jogo de cores lógico que fique bem apresentável, como em todo tipo de apresentação gráfica - não só no computador.

Edite o arquivo index novamente, para melhorar os links. Deixe-o assim:

```
<html>
<head>
<title>Minha Primeira Página</title>
</head>

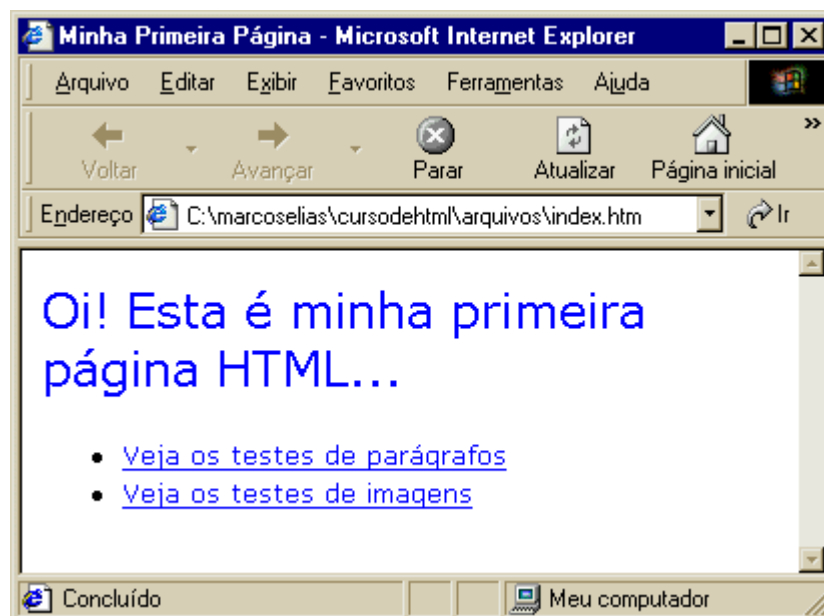
<body>

<font face="verdana" size=5 color="#0000FF">Oi! Esta é minha
primeira página HTML...</font>

<font face="verdana" size=2>
<ul>
<li><a href="paragrafos.htm">Veja os testes de parágrafos</a>
<li><a href="imagens.htm">Veja os testes de imagens</a>
</ul>
</font>

</body>
</html>
```

E visualize:



Para colocar um link para outro site, ou para um arquivo que se encontra em outro endereço na Internet, basta informar o caminho completo do arquivo (com `http://`). Por exemplo, um link para o site da Mep Sites:

```
<a href="http://www.mepsites.cjb.net">Mep Sites</a>
```

Para a pessoa visualizar melhor, pode-se colocar o texto do link como sendo o próprio endereço:

```
<a href="http://www.mepsites.cjb.net">www.mepsites.cjb.net</a>
```

Quando colocamos links para sites externos, que não a nossa página, é bom fazer com que o link se abra numa nova janela - assim a sua página ainda fica aberta. Basta definir `"_blank"` no atributo `target` da tag `<a>`. Veja:

```
<a href="http://mepsites.cjb.net" target="_blank">Mep Sites</a>
```

Nesse caso o site Mep Sites se abriria numa nova janela.

Para colocar um link para um endereço de e-mail:

```
<a href="mailto:email@provedor.etc">email@provedor.etc</a>
```

Você pode colocar o texto que quiser, ou uma imagem. Por exemplo, com imagens:

```
<a href="mailto:testes@com.br"></a>
```

Recorde, para não errar:

Página: `href="nomedapagina.htm"`

E-mail: `href="mailto:endereco@site.com.etc"`

Site: `href="http://www.com.br"`

É importante tomar cuidado, e não se esqueça de fechar as aspas e as tags (no caso, com `</a>`), pois o link poderia dar errado e estragar a página - é claro, pode ser editado a qualquer momento.

Geralmente os sites apresentam um **menu**, uma lista, com links para as principais páginas. Esse menu normalmente é repetido em todas as páginas do site, onde for conveniente. E se você modificar, acrescentar algum item, ou retirar, deve atualizar em todas as páginas que contenham aquele menu. Fica mais fácil copiando e colando, mas se são muitas páginas... Aí se recorre a outros recursos, como quadros - ou frames, por exemplo.

No caso, insira nas duas páginas que você fez um link para a index, para que a pessoa possa retornar ao ponto inicial. Como texto, pode-se colocar **Home** - Home é a primeira página de um site (seria "casa", de *home page*, página da casa).

## Tabelas

As tabelas servem de um modo geral para duas coisas: armazenar dados de forma organizada e estruturar o layout da página. Você já pôde perceber que em HTML não temos total controle sobre a disposição dos elementos na tela do navegador. Por exemplo, exibir um logotipo num canto específico, alinhar texto com várias imagens, etc. Utilizando tabelas, isso se torna mais fácil.

Uma tabela é definida pela tag **<table>**, com a do fechamento, **</table>**. Entre essas tags colocamos outras, que vão compor as linhas e células da tabela. Saiba que célula de uma tabela é cada um dos seus “quadros”, onde colocamos as informações.

Veja um exemplo de tabela simples:

```
<table border=3>
<tr>
<td>
Aqui fica o conteúdo...
</td>
</tr>
</table>
```

Este resultado geraria uma tabela de uma única linha. As linhas são definidas pelo par **<tr> </tr>**. Se quisermos uma tabela com 4 linhas, devemos colocar 4 tags **<tr> </tr>** na tabela. Dentro de uma linha, colocamos as células, que definem quantas colunas a tabela terá. **É importante sempre, antes de começar a construir a página HTML, fazer um rascunho num papel, de um esboço da tabela.** Trace um retângulo e divida-o em linhas e colunas, para depois partir para o HTML.

Acompanhe agora um exemplo prático, que criará uma tabela de 4 linhas e 3 colunas. Se você estiver usando o Bloco de Notas, pressione a tecla TAB, no teclado, para deixar as tabulações e ficar mais fácil de entender os códigos. Abra um novo documento e salve com o nome de **tabelas1.htm**, na sua pasta ou no seu disquete. Digite-o assim:

```
<html>
<head>
<title>Criando tabelas</title>
</head>

<body bgcolor="#ffffff">

<table border=3>

<tr>
<td>Linha 1 Coluna 1</td>
<td>Linha 1 Coluna 2</td>
<td>Linha 1 Coluna 3</td>
</tr>

<tr>
<td>Linha 2 Coluna 1</td>
<td>Linha 2 Coluna 2</td>
```

```
<td>Linha 2 Coluna 3</td>
</tr>
```

```
<tr>
<td>Linha 3 Coluna 1</td>
<td>Linha 3 Coluna 2</td>
<td>Linha 3 Coluna 3</td>
</tr>
```

```
<tr>
<td>Linha 4 Coluna 1</td>
<td>Linha 4 Coluna 2</td>
<td>Linha 4 Coluna 3</td>
</tr>
```

```
</table>
```

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

Para facilitar, colorimos cada linha com uma cor diferente. Veja que o número de células foi definido como 3 antes da criação da tabela, então todas as linhas devem ter 3 células, mesmo que não sejam utilizadas, senão a tabela perde a diagramação e fica deformada, torta, incorreta.

Vamos analisar o código. Na primeira linha da tabela, definimos a tabela usando **<table border=3>**. Esse border=3 determina que a borda da tabela seja de 3 pixels.

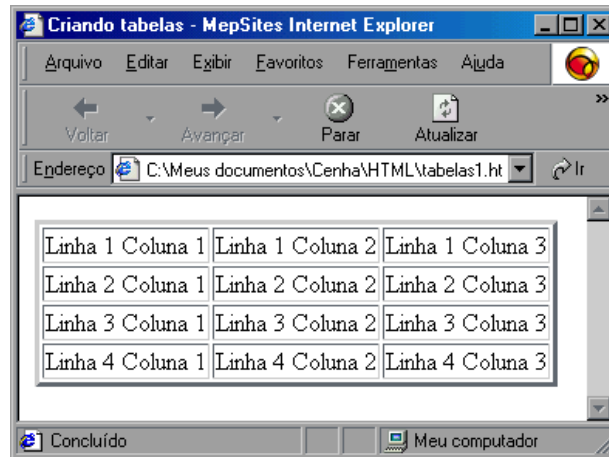
No indicado em verde, começamos uma linha com **<tr>**, e dentro dela colocamos três células, com três pares de **<td> ... </td>**. Ao final, fechamos a linha com **</tr>**.

Depois repetimos o processo para a segunda linha (em azul), também com três células. A mesma coisa foi feita para a terceira e quarta linhas.

Finalmente, fechamos a tabela com **</table>**.

Note que devemos sempre fechar. Os comandos da tabela são "encaixados", um dentro do outro, como numa hierarquia, ou seja, primeiro **TABLE**, depois **TR** e por fim **TD**.

Veja a tabela que criamos no navegador:



Vamos montar agora um outro exemplo, com textos e imagens. Arrume alguma imagem qualquer, de preferência pequena, para utilizar. Digite e salve com o nome **tabelas2.htm**, não se esqueça de escolher "Todos os arquivos" no Bloco de notas, ao salvar.

```
<html>
<head>
<title>Criando tabelas</title>
</head>

<body bgcolor="#ffffff">

<table border=3>

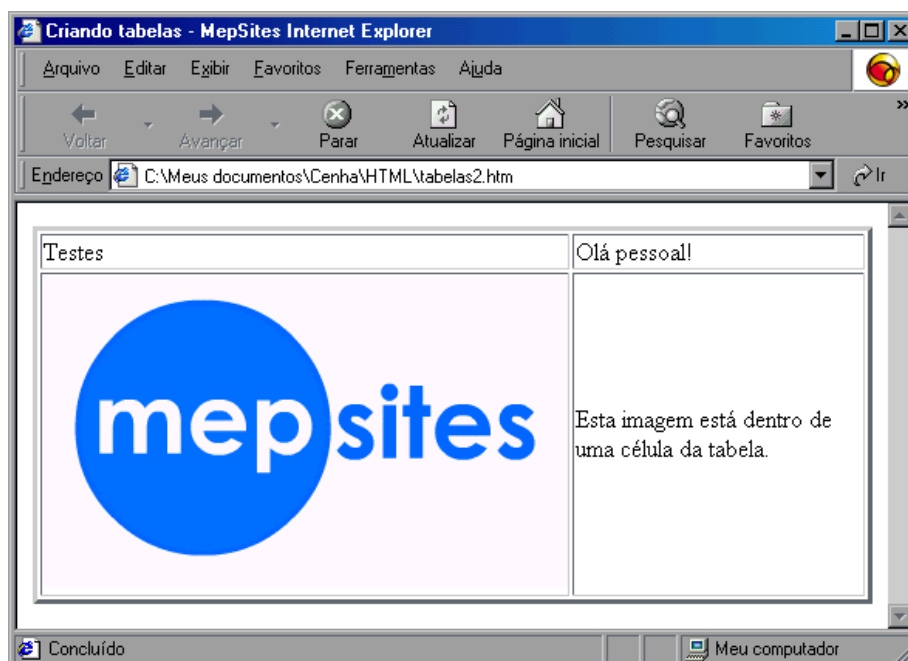
<tr>
<td> Testes </td>
<td> Olá pessoal! </td>
</tr>

<tr>
<td>  </td>
<td> Esta imagem está dentro de uma célula da tabela.</td>
</tr>


</table>

</body>
</html>
```

Salve e visualize:



Veja que é uma tabela de duas linhas, portanto temos no código dois grupos de `<tr> ... </tr>`. E como ela tem duas colunas, colocamos dois grupos de `<td> ... </td>` dentro das tags das linhas.

Se a janela do seu navegador estiver maximizada, clique no botão , para restaurá-la. Depois, clique nas bordas da janela e arraste, para redimensioná-la. Veja que o texto se ajusta automaticamente de modo que fique, na medida do possível, visível na tela.

Quando se trata de tabelas grandes, ou tabelas que estruturam a página, isso é indesejável.

Vamos então colocar um atributo no código da tabela que defina uma largura fixa. Lembre-se de que a resolução ideal para a Internet é a de 800 x 600, então não podemos definir uma largura como 810, ou 800 mesmo, senão não caberia de uma vez na tela. Geralmente, para expandir lado a lado, pode-se usar 770.

Modifique o código da página de nome tabela2.htm, inserindo **width=770** na tag **<table>**. Ela ficaria assim:

```
<table border=3 width=770>
```

Feito isso, clique em Arquivo, Salvar, e visualize no Internet Explorer.

Podemos também definir uma cor de fundo para a tabela toda, ou para alguma célula específica. Basta colocar **bgcolor="#XXYYZZ"** na tag **<table>** ou **<td>** correspondente. Veja um exemplo, salve com o nome de **tabela3.htm**:

```
<html>
<head>
<title>Criando tabelas</title>
</head>

<body bgcolor="#ffffff">
```

```
<table border=3 width=500>

<tr>
<td bgcolor="#FF0000"> ... </td>
<td bgcolor="#00FF00"> ... </td>

</tr>

<tr>
<td bgcolor="#FFF000"> ... </td>
<td bgcolor="#FFCC99"> ... </td>

</tr>

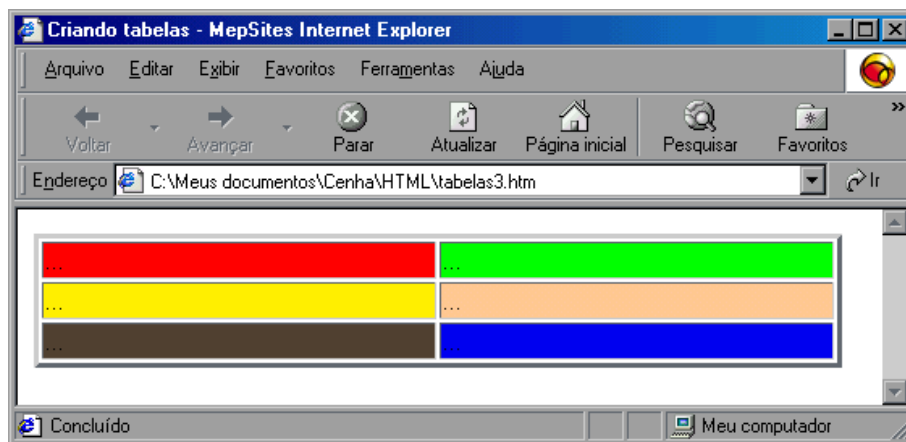
<tr>
<td bgcolor="#554432"> ... </td>
<td bgcolor="#0000F0"> ... </td>

</tr>

</table>

</body>
</html>
```

Salve e visualize no navegador:



Se quiser, altere essa página, e coloque **border=0**, para que não apareça borda alguma.

Também podemos definir uma altura fixa da tabela. Basta colocar **height=250**, por exemplo, na tag **<table>**. Lembre-se de que geralmente a altura da tela é de 600 pixels, descontando-se ainda as barras de ferramentas, a barra de tarefas... Você pode colocar algum valor, salvar e visualizar para testar o melhor.

Pode ir mais longe ainda, e personalizar a cor da borda, colocando na tag **<table>** o atributo **borderColor="#XXYYZZ"**. Talvez não funcione nos navegadores mais antigos. E, é claro, só terá efeito se você não definir a borda com o valor zero, pois assim ela não apareceria mesmo.

## Scripts

No HTML temos muitas restrições, em especial quanto a efeitos especiais, posição dos itens na tela, etc. O HTML serve mais para estruturar a página, e formatá-la.

Pensando nisso foi desenvolvida, já a algum tempo atrás, a tecnologia JavaScript. É uma linguagem - com base na linguagem de programação Java, mas sem manter nenhum vínculo com esta - de criação de *scripts* para páginas na Internet. Scripts podem ser considerados trechos de programas, que quando inseridos nas páginas da web oferecem recursos adicionais. O mais utilizado é o JavaScript, embora existam outras linguagens.

Os códigos em JavaScript não são interpretados por todos os navegadores. Geralmente os mais antigos não os reconhecem - pois estariam tentando ler alguma coisa sem conhecer a língua em que esta coisa foi escrita. Hoje praticamente todos reconhecem e interpretam JavaScript muito bem, em especial o Internet Explorer e o Mozilla Firefox, que veio caminhando e tomou o lugar do Netscape.

Nosso objetivo aqui não é ensiná-lo a programar em JavaScript, apenas a inserir scripts em uma página da web. Muitos sites fornecem códigos prontos, que geram efeitos especiais. Como por exemplo, janelas animadas, imagens que seguem o mouse, mensagens rolantes na tela, janelas pop-ups - aquelas pequenininhas que se abrem automaticamente, como de propaganda, etc.

Basta copiar o código e colar. O lugar onde colar?

Você deve colar na página em que quer que o efeito apareça, ou no template do seu blog. Prefira colar os códigos maiores no final de tudo, depois de `</html>`. Alguns precisam ser colocados dentro do cabeçalho, nesse caso basta colá-lo entre as tags `<head>` e `</head>`.

Veja um exemplo:

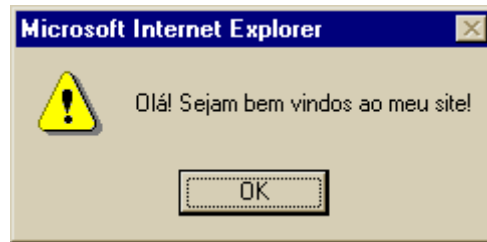
```
<html>
<head>
<title>Minha Primeira Página</title>
</head>

<body>
<script language="JavaScript">
alert('Olá! Sejam bem vindos ao meu site!');
</script>

<font face="verdana" size=5 color="#FF0000">Oi!</font>

</body>
</html>
```

No caso, esse script inserido geraria uma mensagem numa caixa de diálogo, assim que a página fosse aberta:



Para inserir scripts, o maior cuidado deve ser observado nas aspas pois eles são muito mais fáceis de apresentarem erros de configuração do que o HTML. Em JavaScript a estrutura básica é a seguinte:

```
<script language="javascript">
```

Códigos do script

```
</script>
```

No site [Mep Sites \(www.mepsites.cjb.net\)](http://www.mepsites.cjb.net) também há códigos, simples e avançados, sobre JavaScript. Um bom que recomendo também é o [www.codigofonte.net](http://www.codigofonte.net), e existem muitos, muitos outros. Sem querer ser completo nem perfeccionista, eis mais alguns que recomendo: [www.plugmasters.com.br](http://www.plugmasters.com.br), [www.imasters.com.br](http://www.imasters.com.br), [www.htmlstaff.org](http://www.htmlstaff.org). Todos em português ☺ Boa sorte!

## Formação continuada

Dicas de aprofundamento: se você pretende criar páginas para a Internet, este curso lhe forneceu os passos iniciais e básicos, que todos que trabalham com a Internet devem saber. No entanto, provavelmente você deverá estudar outras áreas e tecnologias, dependendo do que você for fazer. Por exemplo, sites de compras on line, galerias, sites com fóruns ou comunidades onde as pessoas publicam mensagens na hora... Isso tudo usa o que você aprendeu aqui sim, mas, além disso, usam recursos de áreas de programação para a web, que podem ser feitos em ASP, PHP, etc (ASP e PHP são duas tecnologias mais populares para criação de páginas da web dinâmicas). Ainda deverá aprender a trabalhar com alguns tipos de imagem, para que possa tratá-las antes da publicação... É na verdade uma formação continuada, que só depende de você e do que você pretende fazer.

Agradeço por você ter lido este mini-curso!

O autor

## Créditos

Copyright © 2004-2007 Marcos Elias Picão

Mep Sites - Mep Produções 2004-2007

Esse texto é próprio; qualquer semelhança com outras obras/sites/publicações é mera coincidência.

A modificação ou reprodução total ou parcial deste material, em qualquer tipo de mídia que não seja o arquivo original, é expressamente proibida. Você pode compartilhá-lo gratuitamente com amigos, colegas, na Internet, em CDs, etc., inclusive no seu site. Mas não pode modificá-lo nem revendê-lo.

<http://www.mepsites.cjb.net>

Microsoft Windows 98, Windows XP, Internet Explorer são marcas e/ou nomes comerciais registrados de Microsoft Corporation. Mep Texto é um software de propriedade do Grupo Mep Produções. Os outros nomes de produtos ou serviços reais podem ser marcas comerciais ou registradas de seus respectivos produtores, desenvolvedores ou mantenedores.

Imagens de software foram utilizadas para fins absolutamente didáticos.

© 2004-2007 Marcos Elias Picão - Todos direitos reservados.

## ***HTML: Referência Básica e Rápida***

**Nota Importante:** Para utilizar esta referência é necessário já ter os conhecimentos básicos de HTML sobre os temas abordados. Este material serve para consulta rápida e/ou revisão geral de conceitos, não necessariamente para estudo de HTML.

### **Editando os arquivos HTML no Bloco de Notas**

Digite os comandos HTML em maiúsculas ou minúsculas, como preferir. É melhor que se faça em minúsculas.

Dê o nome do arquivo sem espaços, caracteres especiais nem acentos. A extensão deve ser .htm ou .html (escolha a melhor para você e em todas as suas páginas a utilize; não misture num mesmo site páginas com .htm e com .html, pois você pode se confundir depois).

O nome do arquivo da página de abertura do site deve ser **index.html**, **index.htm**, **default.html** ou **default.htm**, depende do servidor em que o site for ser hospedado. O mais comum é **index.html**.

Na hora de salvar pela primeira vez, escolha **Salvar como**, e na caixa 'Salvar como o tipo' escolha a opção '**Todos os arquivos**'. É necessário utilizar essa opção para o correto funcionamento, senão o Bloco de Notas daria a extensão e tipo de arquivo de texto. Nas outras vezes que for salvar o mesmo arquivo, apenas clique em **Salvar**. Para editar uma página que está fechada, nunca use o 'Exibir código fonte'. Abra o Bloco de Notas, clique em **Arquivo** e escolha **Abrir**.

No navegador, para visualizar as alterações feitas, utilize o comando Atualizar (Recarregar, *Reload*). No Internet Explorer, basta pressionar a tecla F5, estando com a página já aberta.

**Atributos dos comandos:** Se forem palavras, textos ou expressões, sempre utilize aspas. Se forem números, nunca utilize aspas. Se for o código de uma cor, coloque sempre entre aspas e com o símbolo #.

### **Estrutura geral de toda página:**

```
<html>
<head>
<title>
Seu título aqui
</title>
</head>
<body>
```

Conteúdo da página aqui

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

**Cores gerais da página:** altere a tag <body>:

```
<body text="#cor" link="#cor" vlink="#cor" alink="#cor" bgcolor="#cor">
```

Onde: **text** determina a cor dos textos; **link** determina a cor dos links; **alink** determina a cor dos links enquanto são clicados (links ativos); **bgcolor** determina a cor de fundo da página. Esses itens são opcionais, não é preciso usar todos.

**Imagem de fundo:** Coloque **background="imagem.gif"** dentro da tag <body>. Substitua imagem.gif pelo nome correto da imagem, e ela deve estar na mesma pasta dos arquivos do site. Se for jpg, coloque a extensão jpg. (ou a que for realmente). Exemplo: <body background="fundo002.jpeg">.

**Textos e fontes:** <font face="NomeDaFonte" size=X color="#cor">Texto aqui</font>

Em **face** coloque o nome da fonte (ex.: Arial, Verdana, Courier, etc). Em **size** coloque o tamanho, que vai de 1 (o menor) a 7 (o maior). O padrão é o 3. Em **color**, coloque o código da cor que o texto deve aparecer. (Há uma tabela de cores no site [www.mepsites.cjb.net](http://www.mepsites.cjb.net)).

**Subtítulos:** <hX>Título</hX>, onde **X** é um número de 1 (o título maior) a 6 (o título menor). Não é recomendável usar esse comando; prefira colocar <font size=...>.

**Negrito:** <b>texto aqui</b> **Exemplo**

**Itálico:** <i>texto aqui</i> *Exemplo*

**Sublinhado:** <u>texto aqui</u> Exemplo Dica: nunca use sublinhado, pois o visitante pode pensar que é um link e tentar clicar.

**Texto riscado:** <s>texto aqui</s> *Exemplo*

**Texto sobrescrito:** <sup>Texto aqui</sup>. Exemplo: Windows<sup>xp</sup> geraria Windows<sup>xp</sup>

**Texto subscrito:** <sub>Texto aqui</sub>. Exemplo: x<sub>1</sub> geraria x<sub>1</sub>

**Texto em fonte tipo máquina de escrever:** <tt>Texto aqui</tt> *Exemplo*

**Texto tipo código de programação:** <code>Texto aqui</code> *Exemplo*

**Lista com marcadores:**

```
<ul>
<li>Item 1
<li>Item 2
<li>Item 3
<li>Etc
</ul>
```

**Lista numerada:**

```
<ol>
<li>Item 1
<li>Item 2
<li>Item 3
<li>Etc
</ol>
```

**Parágrafos:**

```
<p align="alinhamento">Texto aqui</p>
```

Coloque no lugar de alinhamento **left** (esquerdo), **right** (direito), **center** (centralizado), ou **justify** (justificado). Obs.: entre um parágrafo e outro, é deixado automaticamente uma linha em branco.

Cuidado: é **align**, não **aling**.

**Blocos de texto:**

```
<div align="alinhamento">Texto aqui</div>
```

Coloque em alinhamento **left**, **right**, **center** ou **justify** (como nos parágrafos). Este comando tem a vantagem de não deixar linhas em branco entre um bloco e outro.

**Alinhamento centralizado:** <center> Coisas centralizadas </center>

Pode ser usado com textos, imagens, tabelas ou outro item qualquer.

**Alinhamento justificado:** <blockquote> Texto aqui </blockquote>

**Imagem:**

```

```

Em **src** determine o nome da imagem (que deve estar na mesma pasta dos arquivos do site) ou o endereço completo (com http://). **border** determina uma borda, é sempre bom deixar o valor 0 (zero). **width** e **height** determinam, respectivamente, a largura e a altura da imagem, caso queira redimensioná-la. Não é necessário usar. Você deve colocar números proporcionais às dimensões reais da imagem, para não deformá-la. As medidas são dadas em pixels. Se não colocar esses atributos, ela aparece no tamanho original. **hspace** e **vspace** determinam, respectivamente, uma margem horizontal e vertical para a figura. É um espaçamento como uma margem, 'invisível', para distanciar a imagem do texto ou do que estiver ao seu redor (inclusive outras imagens). No lugar de alinhamento use **left** (para ela ficar a esquerda do texto que a precede), **right** (para ficar a direita do texto que a precede), ou **center** (para ficar centralizada). Você pode ainda determinar o atributo **alt="Texto alternativo aqui"**, que exibe um texto caso a imagem não seja localizada, ou se o navegador do usuário não interpretar imagens. Hoje em dia é comum aparecer uma mensagem na tela ao deixar o mouse parado sobre a figura por um instante. O texto que aparece é definido com esse atributo.

**Links:**

`<a href="pagina.html">Clique aqui</a>` (a página deve estar na mesma pasta dos arquivos do site)

Para e-mail: `<a href="mailto:usuario@provedor.com.etc">Deixe sua mensagem</a>`

Dica: coloque o próprio endereço entre `<a...>` e `</a>`, para aparecer o endereço do e-mail na tela.

Para outro site: `<a href="http://www.mepsites.cjb.net">Mep Sites</a>`

Para sites, use sempre o `http://`.

Dica: coloque o próprio endereço entre `<a...>` e `</a>`, para aparecer o endereço do site na tela.

Se quiser usar uma imagem como link, use a sintaxe `<a href="..."></a>`

**Âncoras - link na mesma página:**

Antes você deve nomear cada local. Para nomear: `<a name="Explicacao4">Explicação 4</a>`. No lugar que você vai colocar o link, utilize: `<a href="#Explicacao4">Veja a explicação 4</a>`. Altere os textos de acordo com o que quiser. Não dê nomes aos locais com espaços, caracteres especiais nem acentos.

**Comentários dentro do código:** `<!-- Seus comentários aqui -->`

Serve para você mesmo que prepara a página ou alguém que veja o código dela ler a mensagem, como dicas e explicações. Os comentários não aparecem no navegador, são só para quem abrir os códigos mesmo.

**Textos pré formatados:** `<pre>Texto aqui (pode dar enter e quebras de linhas)</pre>`

Neste comando, os espaços que você der no código, e quebras de linha, serão respeitados, não precisando então usar `<p>` nem `<br>`. Geralmente os navegadores mostrarão o texto numa fonte monoespaçada, como a *Courier*, por exemplo.

**Tabelas:**

Imagine a seguinte estrutura:

```
<tabela>
<linha1>
<quadro1>Texto aqui</quadro1>
<quadro2>Texto aqui</quadro2>
<quadro3>Texto aqui</quadro3>
</linha1>
<linha2>
<quadro1>Texto aqui</quadro1>
<quadro2>Texto aqui</quadro2>
<quadro3>Texto aqui</quadro3>
</linha2>
<linha3>
```

```

<quadro1>Texto aqui</quadro1>
<quadro2>Texto aqui</quadro2>
<quadro3>Texto aqui</quadro3>
</linha3>
</tabela>

```

Pois bem, basta usar **<table>** em vez de tabela, **<tr>** em vez de linha1, linha2, etc e **<td>** em vez de quadros (células). Ficaria:

```

<table>
<tr>
<td>Texto aqui</td>
<td>Texto aqui</td>
<td>Texto aqui</td>
</tr>
<tr>
<td>Texto aqui</td>
<td>Texto aqui</td>
<td>Texto aqui</td>
</tr>
<tr>
<td>Texto aqui</td>
<td>Texto aqui</td>
<td>Texto aqui</td>
</tr>
</table>

```

Colorimos cada linha para você perceber melhor. Este exemplo geraria uma tabela de 3 linhas, com 3 colunas. Pode-se usar os atributos **border** (espessura da borda), **borderColor** (cor da borda), **bgcolor** (cor de fundo), **background** (imagem de fundo), entre outros, na tag **<table>** (para formatar a tabela inteira), ou então separadamente, em cada linha (na tag **<tr>**), ou ainda em cada célula da tabela (na tag **<td>**). Este último é o mais usado.

Nota importante: Se você usar 3 células na primeira linha, com três pares de **<td>...</td>**, deverá colocar 3 também em todas as linhas, senão poderá deformar toda a tabela.

Na tag **<table>** ainda podemos ter **cellspacing=X** (um valor numérico, em pixels) que indica um espaçamento entre cada célula da tabela, e **cellpadding=X** (também um número), que indica um enchimento em cada célula. É bom usar valores pequenos, como 3 ou 5, por exemplo.

Para mesclar a célula atual com as que vêm a sua direita, utilize **colspan=X** (coloque no lugar de X o número 2, ou 3, 4, depende do número de células a mesclar). Se forem linhas, utilize **rowspan=X** (para meaclar X linhas, para baixo da célula atual).

## Frames (Quadros):

Na página index:

```

<frameset cols="200,*" noresize>
<frame name="esquerdo" src="pagina1.html">
<frame name="direito" src="pagina2.html">
</frameset>

```

Este dividirá a tela em duas partes, na vertical (em duas colunas). Se quiser na horizontal, utilize rows em vez de cols. Na primeira ficará a página **pagina1.html**, e na segunda a página **pagina2.html**. O item **noresize** indica que ele não pode ser redimensionado pelo visitante. No lugar de **valor1** você coloca, em pixels, a medida da largura do primeiro quadro. O asterisco no segundo, indica que o segundo ocupará o restante da tela.

Nos links você deve colocar `<a href="NomeDaPagina.html" target="NomeDoQuadro">Clique aqui</a>`.

Em **NomeDoQuadro** você dá o nome do quadro em que ela deverá ser aberta, corretamente, como você o definiu. No exemplo, seria **"direito"**. Ao clicar no link, se você não colocar o nome do frame em que ele deverá abrir, ele se abrirá então no mesmo quadro em que estiver o link.

Você pode colocar em **target** (destino) também **\_blank**, para o link se abrir numa nova janela, ou **\_top**, para ele se abrir na mesma tela onde estão os quadros, limpando assim todos os quadros. Um exemplo para abrir o link em nova janela: `<a href="pagina.html" target="_blank">Clique aqui</a>`