

CURSO BÁSICO DE HTML

Apostila desenvolvida para ser usada em um curso básico de HTML do comitê Fome-Zero Joinville.

Quaisquer dúvidas, sugestões, reclamações enviar e-mail para felipe_at_gallois_dot_us.

Felipe Gallois

2008

Este trabalho está licenciado sob uma Licença Creative Commons Atribuição-UsoNãoComercial-Compartilhamento pela mesma licença. Para ver uma cópia desta licença, visite <http://creativecommons.org/licenses/by-nc-sa/2.5/br/> ou envie uma carta para Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.



Atribuição-Uso Não-Comercial-Compartilhamento pela mesma licença 2.5 Brasil

Você pode:

- copiar, distribuir, exibir e executar a obra
- criar obras derivadas

Sob as seguintes condições:



Atribuição. Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.



Uso Não-Comercial. Você não pode utilizar esta obra com finalidades comerciais.




Compartilhamento pela mesma Licença. Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.

Qualquer direito de uso legítimo (ou "fair use") concedido por lei, ou qualquer outro direito protegido pela legislação local, não são em hipótese alguma afetados pelo disposto acima.

Este é um sumário para leigos da [Licença Jurídica \(na íntegra\)](#).

[Termo de exoneração de responsabilidade](#) 

Sumário

Introdução.....	4
Elementos.....	5
Começando sua página.....	6
Hyperlinks.....	11
Imagens.....	14
Listas.....	17
Tabelas.....	20
Frames.....	25
Forms.....	28
Considerações Finais.....	32
Lista de caracteres especiais.....	32
Lista de tags.....	33

Introdução

Uma página da web (webpage) é uma fonte de informações que é adequada à World Wide Web e que pode ser acessada por um navegador web. Isso quer dizer que ela é um documento que é feito de modo à atender requisitos da rede mundial de computadores e capaz de ser visualizada a partir de um programa específico para esse fim.

As páginas web estão localizadas ou em um computador local ou remoto e disponibilizadas através de um servidor web. O acesso a elas pode ser restringido à redes locais, apenas ao próprio computador ou ainda ser publicada na Internet. A requisição e o acesso a elas é feita por meio de um protocolo chamado HTTP (Hypertext Transfer Protocol, ou Protocolo de transferência de hipertexto).

Originalmente, as páginas web eram arquivos estáticos de texto guardados dentro de servidores web. Atualmente, entretanto, é comum encontrarmos servidores que geram dinamicamente os arquivos (x)HTML de acordo com a requisição do navegador. Existem ainda maneiras de obtermos comportamento dinâmico pelo lado do cliente (navegador) por meio de implementações de Ajax e JavaScript, por exemplo.

As webpages utilizam de HTML (Hypertext Markup Language) para fazer a formatação dos dados. É importante notar que HTML não é uma linguagem de programação, mas sim de formatação. Isso guarda diferenças fundamentais entre HTML e, por exemplo, JavaScript, uma vez que a primeira limita-se apenas a mudar a maneira como os dados são dispostos na tela.

Para melhorar a aparência das webpages é possível usar código CSS (Cascading Style Sheet) que pode ser escrito em um arquivo em separado ou dentro do próprio HTML. É preferível valer-se da primeira opção uma vez que isso torna seu código mais reutilizável e, além disso, o uso de CSS torna-se mais eficiente já que aproveita-se melhor da maneira como o HTTP trabalha, com o navegador baixando uma única vez as informações e guardando-as em cache.

As imagens são armazenadas também como arquivos separados em servidores web. O navegador faz o trabalho de baixar todas as imagens enquanto a página é processada, mantendo a conexão com o servidor até que todas as informações sejam baixadas e fornecidas. A grande maioria deles renderiza as imagens juntamente com os textos e outros materiais na página a ser exibida.

Elementos

Uma página web apresenta uma série de elementos que serão usados para a interação com o usuário. Eles dividem-se em duas grandes categorias, os que são mostrados e os que não são.

Entre os mostrados, temos as informações textuais, com diversas variações de renderização; as informações não-textuais, como imagens estáticas (gif, jpg, png, etc) ou animadas (animated gif, flash, shockwave), áudio (midi, wav) e vídeo (wmv, rm, mov, etc), e; informações interativas que são mais complexas, atreladas à interface. Nesse ponto, temos textos interativos, ilustrações interativas, botões, hyperlinks e formulários.

Como elementos não mostrados temos os comentários, meta dados(DTD), informações de diagramação e estilo (CSS) e scripts (JavaScript).

Nem todos esses elementos serão tratados no escopo do curso. Muitos deles requerem ferramentas adicionais e uma grande quantidade de recursos do computador, tanto para a criação quanto para a exibição e são usados apenas para tornar mais agradável a experiência de navegação do usuário.

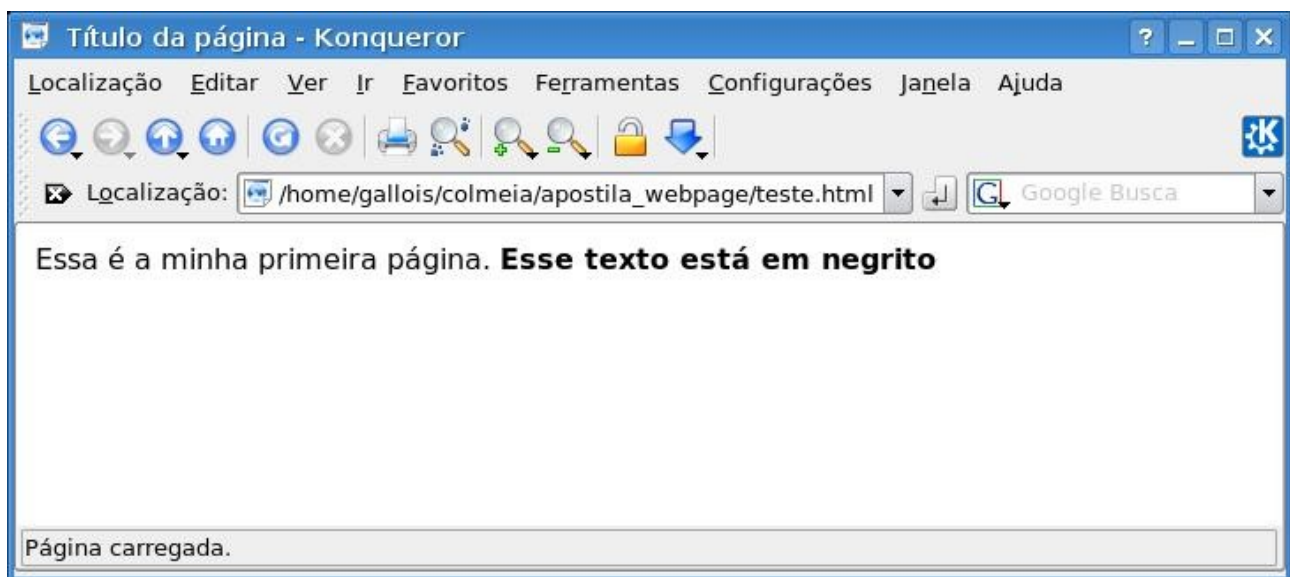
Começando sua página

Para criar suas próprias páginas não é necessário um vasto arsenal de ferramentas, apenas um editor de texto e um navegador já são suficientes. Para um iniciante é até melhor que não sejam usados recursos muito complexos, eles podem fazer com que muitos detalhes triviais, mas importantes, passem despercebidos. Para o nosso curso, usaremos um editor de texto simples de sua preferência (como o kwrite, o gedit ou o Mousepad). Recursos como o syntax highlighting (ou destaque de sintaxe) são realmente úteis durante a criação da sua página ao mesmo tempo que não "fazem o trabalho" por você, então é prudente que você deixe-o ligado. Qualquer navegador que tenha interface gráfica serve para o nosso curso, fique a vontade para escolher o de sua preferência.

Começaremos com algo bem simples, abra o editor de texto e insira o seguinte

```
<html>
<head>
<title>Título da página</title>
</head>
<body>
Essa é a minha primeira página. <b>Esse texto está em
negrito.</b>
</body>
</html>
```

Salve o arquivo como teste.html e abra com um navegador. Você verá o seguinte



Agora vamos explicar algumas coisas sobre o exercício acima. Primeiramente, tudo que estiver entre os sinais de menor que e maior que ('<' e '>'), incluindo eles são chamadas 'tags'. As tags são partes importantes em documentos HTML, elas são as responsáveis por toda a formatação do texto. Com elas você pode dividir o documento em área de cabeçalho e corpo, colocar títulos, criar tabelas, divisões, formulários e ainda formatar trechos específicos do texto, como a parte em negrito acima.

A primeira tag <html>. Ela diz ao navegador que esse ponto marca o início de um documento HTML, enquanto que a última tag </html> diz a ele exatamente o que é esperado, que esse é o fim do documento HTML. O texto que está compreendido entre as tags <head> e </head> é informação sobre a área de cabeçalho, como você pode perceber com <title> e </title> onde o texto que está entre essas tags aparecerá no título da janela do navegador.

Prosseguindo, temos as tags <body> e </body> que são responsáveis por guardar os dados referentes ao corpo do documento. Isso significa que o texto que estiver entre elas será exibido pelo navegador. Por fim, as tags e farão com que o texto entre elas seja exibido em negrito (em inglês, bold).

Você já deve ter percebido que as tags utilizadas sempre têm um "fechamento" como

por exemplo, <html> -> </html>, <head> -> </head> e <body> -> </body>. Isso realmente é válido para uma grande quantidade delas, mas não será onipresente, portanto fique atento.

Uma consideração a ser feita sobre as tags é que elas não são sensíveis à caixa, o que significa que é o mesmo que embora as convenções atuais incentivem o uso de todas as tags em letras minúsculas. Então, para seguir os mais novos padrões web você deve usar letras minúsculas. O padrão xHTML demanda necessariamente o uso delas.

Uma outra questão que pode surgir é sobre a extensão dos arquivos. Antigamente, muitos sistemas não permitiam salvar extensões que tivessem mais do que três caracteres e por isso era comum ver arquivos com nomes como 'index.htm' ou 'home.htm'. Hoje em dia é raro encontrarmos um sistema moderno que não permita extensões com um número de caracteres que vai muito além de três, logo é totalmente aceitável usar 'index.html' ou 'home.html' como é o caso do nosso último exemplo (teste.html).

Os espaços utilizados para separar uma palavra da outra aparecerão normalmente na sua visualização, porém, caso você use mais de um espaço, eles não serão considerados. Para que isso ocorra, é preciso usar uma tag especial.

Trataremos agora dos elementos HTML. Considerando o nosso último exemplo, 'Esse texto est´m; em negrito' é um elemento onde: 1) o elemento HTML inicia-se com a tag de início ''; 2) o conteúdo do elemento é 'Esse texto est´m; em negrito', e; 3) o elemento HTML termina com a tag de fim ''.

Também é um elemento HTML o seguinte trecho:

```
<body>
Essa &acutem; a minha primeira p&acutem;gina. <b>Esse texto est&acutem; em
negrito.</b>
</body>
```

O início do elemento é '<body>' e o fim '</body>'.

Um detalhe importante a ser mostrado aqui é o que diz respeito à acentuação. Você deve ter notado algumas coisas "estranhas" nos exemplos acima, como '´m;' ou '´m;'. Essas são referências para os caracteres especiais como 'é' ou 'á' no exemplo acima, mas que também são utilizados em outros casos. O uso deles se deve ao fato de que alguns caracteres são reservados, facilmente perceptível em '<' e '>' que servem para abrir e fechar tags. O que faríamos para inserir esses caracteres em um texto dentro do HTML? Usamos referências especiais para tal, seguindo o exemplo '<' e '>'. Falaremos mais sobre eles na hora devida.

Usaremos agora algumas tags para tornarmos nosso texto mais complexo. Abra o editor de textos e crie o arquivo 'teste2.html'. Escreva o código a seguir.

Cabeçalhos, parágrafos e quebras de linha

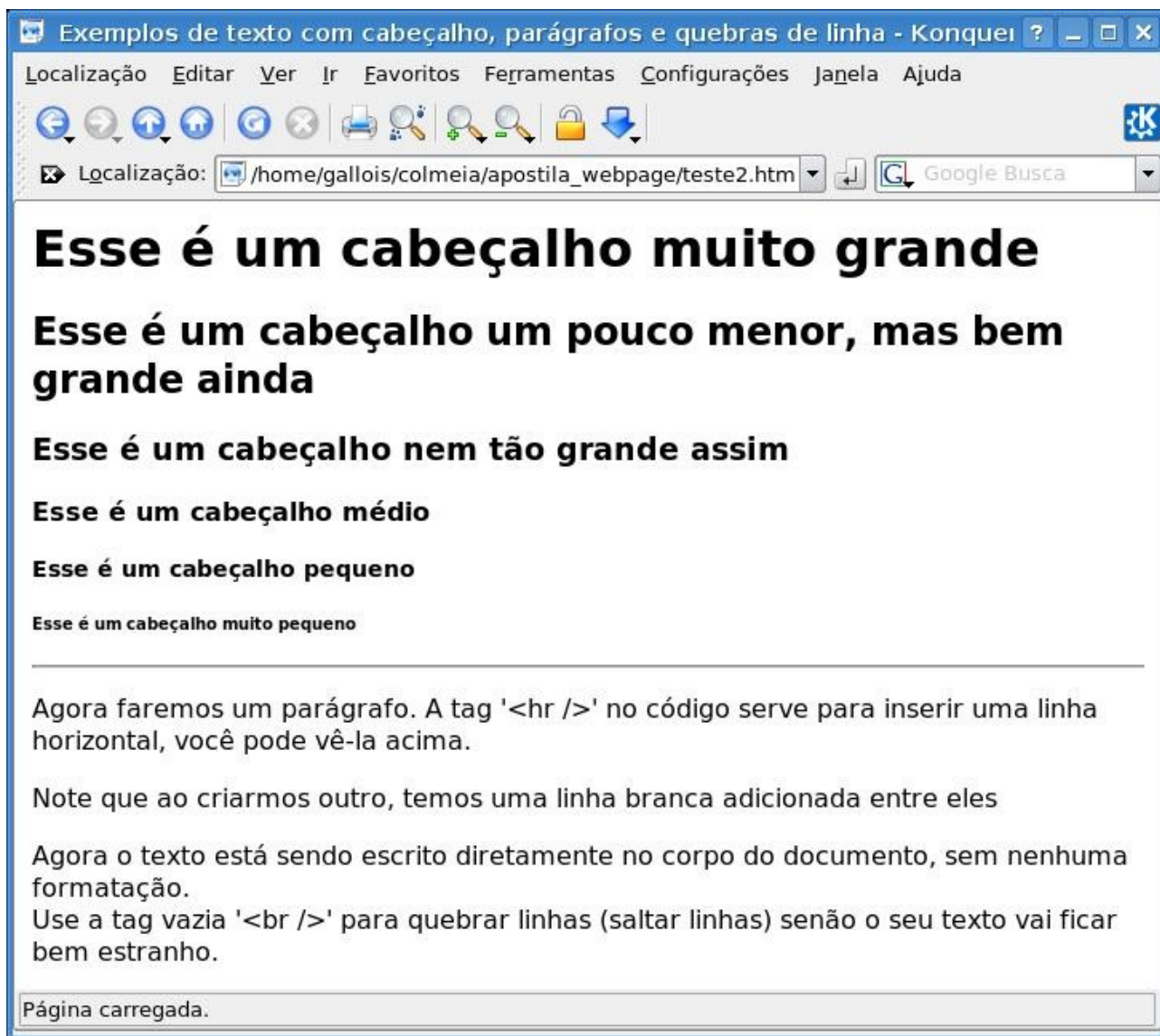
```
<html>
<head>
<title>Exemplos de texto com cabe&ccedil;alho, par&acutem;grafos e quebras
de linha</title>
</head>
<body>
<h1>Esse &acutem; um cabe&ccedil;alho muito grande</h1>
<h2>Esse &acutem; um cabe&ccedil;alho um pouco menor, mas bem grande
ainda</h2>
<h3>Esse &acutem; um cabe&ccedil;alho nem t&atilde;o grande assim</h3>
<h4>Esse &acutem; um cabe&ccedil;alho m&ecutem;dio</h4>
<h5>Esse &acutem; um cabe&ccedil;alho pequeno</h5>
<h6>Esse &acutem; um cabe&ccedil;alho muito pequeno</h6>
<hr />
<p>Agora faremos um par&acutem;grafo. A tag &apos;&lt;hr /&gt;&apos; no
c&ocutem;digito serve para inserir uma linha horizontal, voc&ecirc; pode v&ecirc;-
la acima.</p>
<p>Note que ao criarmos outro, temos uma linha branca adicionada entre
eles</p>
Agora o texto est&acutem; sendo escrito diretamente no corpo do documento,
sem nenhuma formata&ccedil;&atilde;o. <br />
Use a tag vazia &apos;&lt;br /&gt;&apos; para quebrar linhas (saltar
```

linhas) senão o seu texto vai ficar bem estranho.

<!-- Finalmente, esse é um comentário. O que você escrever aqui não será mostrado no documento exibido pelo navegador, mas pode ajudar a entender o que você quer fazer com o código. -->

</body>

</html>



The screenshot shows a web browser window titled "Exemplos de texto com cabeçalho, parágrafos e quebras de linha - Konqueror". The address bar shows the URL "/home/gallois/colmeia/apostila_webpage/teste2.htm". The page content displays several examples of heading tags:

- Esse é um cabeçalho muito grande** (H1)
- Esse é um cabeçalho um pouco menor, mas bem grande ainda** (H2)
- Esse é um cabeçalho nem tão grande assim** (H3)
- Esse é um cabeçalho médio** (H4)
- Esse é um cabeçalho pequeno** (H5)
- Esse é um cabeçalho muito pequeno** (H6)

Below the headings, there is a horizontal line (

). The text explains that the tag '<hr />' is used to insert a horizontal line. It notes that another line was added between paragraphs. It also explains that text is written directly in the body without formatting, and that the tag '
' is used to break lines. The status bar at the bottom indicates "Página carregada."

Agora vamos explicar um pouco mais sobre as tags usadas nesse exemplo. Preservamos a mesma estrutura de '<html><head><title></title></head><body></body></html>', mas acrescentamos diversas outras tags. As que se iniciam com o 'h' e são seguidas por um número de 1 a 6 ('<h1>' e '<h4>', por exemplo) servem para fazer um cabeçalho, um texto que se destacará do restante. O número determina o tamanho do mesmo, quanto menor o número, maior a tipografia. A tag '<hr />' serve para inserir uma linha horizontal ao longo de todo o documento enquanto a '
' apenas salta para a próxima linha. Uma observação sobre essas duas tags é que ambas podem ser escritas da seguinte maneira: '
' ou '<hr>'. A razão pela qual se coloca a barra no final dela é que elas são tags vazias, isto é, não precisam de uma tag para iniciá-las e outra para terminá-las. Os novos padrões da web determinam que toda e qualquer tag deve ser fechada, incluindo as tags vazias. Por isso colocamos a barra no final, para indicar que ela se inicia e se fecha nela mesma. Dessa maneira, garantimos que nosso código possibilite sua reutilização no futuro, quando as novas normas estiverem em vigor. Além dessas tags temos os parágrafos, representados por '<p>' e '</p>'. O texto compreendido entre elas será colocado como um parágrafo separado e ao término da tag uma nova linha será inserida.

Fora isso, vale ainda comentar no momento sobre os caracteres especiais utilizados. Temos vários deles nesse exemplo:

'´', "é"

'ç', "ç"

''', ""

'<', "<"

'>', ">"

'õ', "õ"

'ê', "ê"

são alguns deles. Ao final da apostila teremos uma tabela mostrando cada um deles.

Com base no que vimos até aqui, crie um arquivo chamado "seu_nome.html", não esqueça de colocar realmente o seu nome ;). Dentro dele, faça algo dizendo quem é você, seu nome, sua idade, o que faz da vida e qualquer outra coisa que quiser dizer. Não se esqueça de usar as tags que você viu até agora.

No final desse documento você verá alguns exemplos dos exercícios que serão passados aqui. Caso precise de alguma referência, não hesite em consultá-los.

Algumas tags apresentam atributos que podem ser modificados por você à medida que achar necessário. Veremos alguns exemplos abaixo, tomando como ponto de partida o exemplo de apresentação usado no exercício anterior.

```
<html>
<head>
<title>P&acute;gina do Gallois</title>
</head>
```

```
<body>
<h1>Ol&acute;, seja bem-vindo &agrave; minha webpage</h1>
<hr />
```

```
<p>Meu nome &acute; Felipe Gallois e sou estudante de Ci&ecirc;ncia da
Computa&ccedil;&atilde;o na UDESC. Eu sou bolsista do projeto Fome-Zero em
Joinville e respos&aacute;vel por fazer a apostila do curso de
cria&ccedil;&atilde;o de p&acute;ginas web.</p>
```

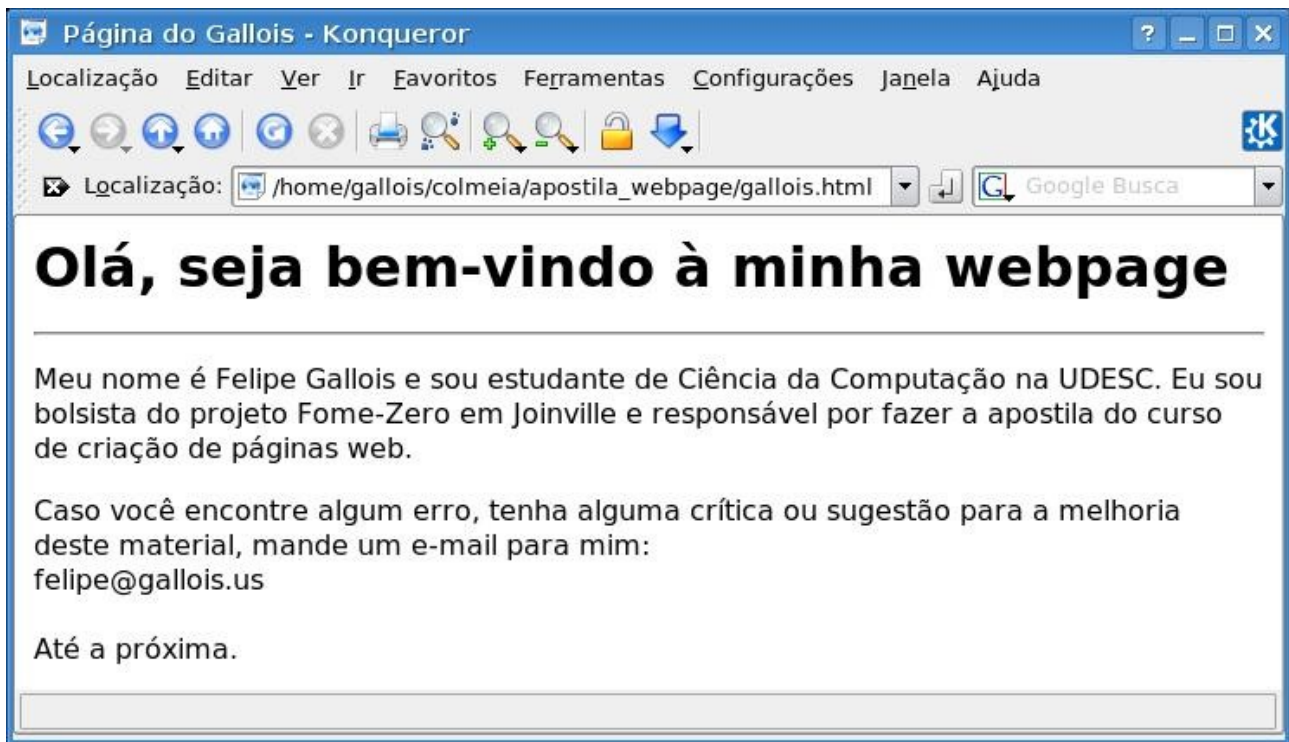
```
Caso voc&ecirc; encontre algum erro, tenha alguma cr&iacute;tica ou
sugest&atilde;o para a melhoria deste material, mande um e-mail para mim:<br />
```

```
felipe@gallois.us<br /><br />
```

```
At&acute; a pr&oacute;xima.
```

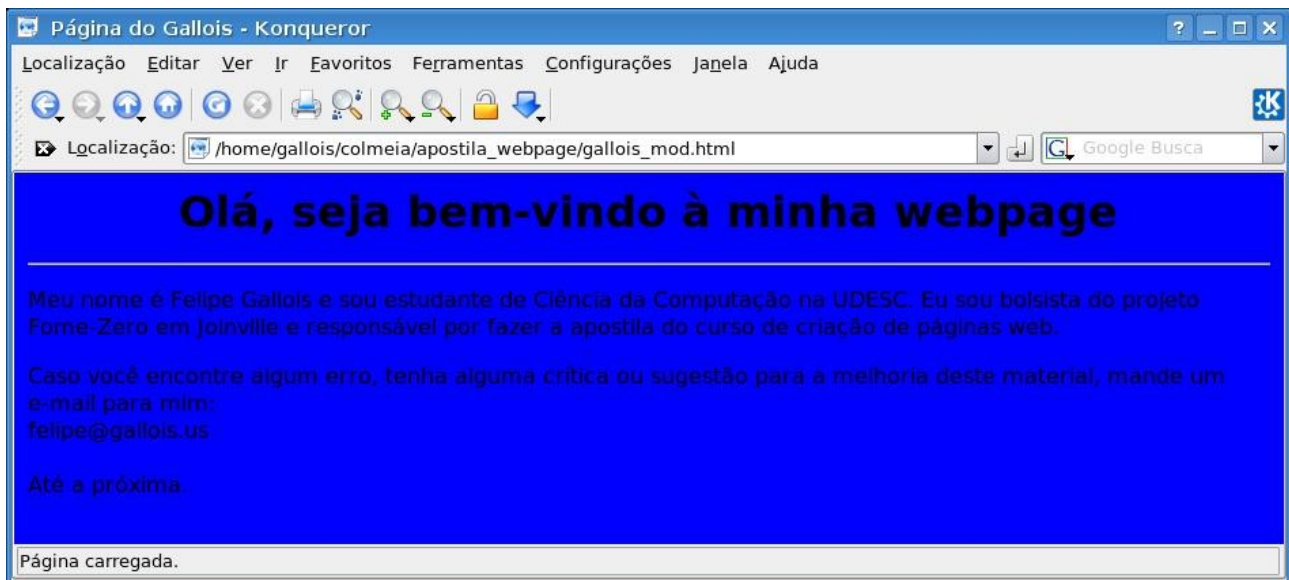
```
</body>
```

```
</html>
```



Vamos agora alterar alguns atributos, começando pelo cabeçalho. Altere

```
<h1>Olá, seja bem-vindo à minha webpage</h1>  
para  
<h1 align="center">Olá, seja bem-vindo à minha webpage</h1>  
Em seguida, altere  
<body> para <body bgcolor="blue">
```



Abra o arquivo e veja as modificações. O que ocorreu efetivamente? Ao inserir o atributo 'align="center"' na tag '<h1>' você solicitou que o texto fosse alinhado no centro da página. No caso do corpo do documento, o atributo 'bgcolor="blue"' quer dizer que você irá alterar a cor do plano de fundo (background color ou bgcolor) para azul (blue).

Note que os atributos preferencialmente devem ser escritos em letras minúsculas, novamente para atender às novas convenções. Os valores dos atributos devem sempre estar entre aspas, simples ou duplas, é indiferente. Fique atento apenas ao caso de você querer usar um valor de atributo que tenha aspas, onde você precisará alternar as duas para poder alcançar o resultado desejado, por ex. name='Edson Arantes "Pelé" do Nascimento'.

Hyperlinks

Um outro recurso amplamente utilizado em HTMLs é o hyperlink. Ele serve para ligar um documento à outro na Web. Basicamente, ele será composto pelo endereço de outro documento e, ao ser clicado, levará você ao endereço referenciado por ele. Existem duas maneiras de usar hyperlinks nos seus documentos, com textos ou com imagens. O texto aparecerá sublinhado e o cursor do mouse ficará diferente ao passar por cima dele. As imagens terão o mesmo efeito de mudança de cursor.

Primeiramente ensinaremos a criar links com texto. Eles podem ser criados com a tag '``', que deve ser fechada com '``'. O conteúdo entre a tag será o texto que deverá ser clicado para que o navegador redirecione. Pode ser qualquer coisa que você quiser, mas recomenda-se usar alguma coisa intuitiva. A Internet originalmente pretendia usar hyperlinks em abundância, quando seu conteúdo ainda era basicamente composto por textos científicos e documentos. O aumento da velocidade de transferência e o início do uso de imagens e meios multimídia na rede ofuscaram um pouco seu propósito original. Um exemplo para o uso de hyperlinks com texto seria:

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

Um recurso que pode ser útil se você quiser redirecionar a pessoa mas manter a sua própria página aberta é adicionar o atributo '`target="_blank"`' à sua tag '`<a>`'. O resultado é que ao clicar no link, uma nova janela do navegador abrirá com o endereço desejado. Note o exemplo abaixo:

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

Agora vamos mostrar como você cria um hyperlink usando uma imagem ao invés do texto. O processo é bem semelhante.

```
<a href="http://www.google.com"></a>
```

No lugar do texto inserimos uma imagem e ela automaticamente se transforma na âncora responsável por ligar o documento à outro.

Vamos aproveitar esse exemplo para darmos mais uma explicação, como inserir imagens. Você deve ter percebido que usamos uma tag diferente logo acima: '``'. Ela tem seus atributos que serão mostrados em breve. Ela serve exatamente para inserir imagens no documento HTML, servindo tanto para buscar imagens no computador local ou na rede. No nosso caso, usamos para buscar na Internet, mas isso é indiferente, o navegador trata qualquer um dos casos muito bem.

Para explicar como funciona o mecanismo de usar imagens que estão no computador local, devemos entender dois conceitos antes, o de caminho relativo e o de caminho absoluto. O caminho absoluto indica a localização do arquivo no computador a partir do diretório raiz. Por exemplo, um arquivo chamado 'foo.bar' que está dentro da pasta home do usuário Joao tem caminho absoluto '/home/Joao/foo.bar' ou o arquivo de configuração padrão de grande parte dos Xorg tem caminho absoluto '/etc/X11/xorg.conf'.

O conceito de caminho relativo distancia-se do de absoluto na medida em que não é possível determinar o caminho do arquivo sem saber a pasta em que se está no momento. Outra diferença é que torna-se mais complicado acessar um arquivo que esteja em uma pasta que está numa pasta que não descende da que você está na árvore de diretórios. Para dizermos o caminho relativo de um arquivo 'foo.bar' que está na mesma pasta em que estamos no momento, podemos dizer que é 'foo.bar'. Suponhamos agora a seguinte estrutura de pastas

```
arquivos
|-- fotos
|   |-- fotos_do_ano_novo
|   |   |-- ano_novo1.jpg
|   |   |-- ano_novo2.jpg
|   |   |-- ano_novo2.jpg
|   |-- fotos_da_pascoa
```

```

|   |   \-- pascoa1.jpg
|   |   \-- pascoa2.jpg
|   |   \-- pascoa3.jpg
|   |   \-- fotos_na_praia
|   |   |   \-- praia1.jpg
|   |   |   \-- praia2.jpg
|   |   |   \-- praia3.jpg
|   |   \-- foto1.jpg
|   |   \-- foto2.jpg
|   |   \-- foto3.jpg
|   \-- textos
|   |   \-- texto1.txt
|   |   \-- texto2.txt
|   |   \-- texto3.txt
|   \-- videos
|   |   \-- video1.mpg
|   |   \-- video2.mpg
|   |   \-- video3.mpg
|   \-- projetos
|   \-- projeto1
|   |   \-- tema1.tem
|   |   \-- tema2.tem
|   \-- projeto2
|   |   \-- tema3.tem
|   |   \-- tema4.tem
|   \-- prazos_dos_projetos.txt

```

Partindo da pasta arquivos, o caminho relativo do arquivo 'foto1.jpg' é 'fotos/foto1.jpg' enquanto o de 'texto2.txt' é '/textos/texto2.txt'. Dessa maneira, o arquivo 'tema1.tem' do projeto1 terá como caminho relativo 'projetos/projeto1/tema1.tem'. Até aí não existem muitas complicações, mas é necessário conhecer um pouco sobre como trabalhar com arquivos que estão em pastas que estão em níveis superiores. Isso exigirá o uso de '..'. Quando você estiver na pasta projeto1 e quiser saber o caminho relativo do arquivos 'prazos_dos_projetos.txt' deverá ter em mente que este arquivo encontra-se em um nível superior na árvore de diretórios com relação à qual você está no momento. Isso quer dizer que você deve retroceder um nível para só então poder chamar o arquivo desejado, e exatamente para isso será necessário o uso de '..'. Como solução, estando na pasta projeto1, o caminho relativo será '../prazos_dos_projetos.txt'.

Quando for necessário retroceder mais níveis ainda, você precisará usar mais '..'. Por exemplo, imagine-se agora no diretório 'projeto2' e deseja saber o caminho relativo para o arquivo 'praia3.jpg'. Ele será '../fotos/fotos_na_praia/praias3.jpg'. Os primeiros '..' te colocarão no diretório 'projetos' e o segundo no 'arquivos'. Deste ponto em diante você segue os passos de como se estivesse apenas buscando em sub-diretórios.

Dada essa longa explicação, prosseguiremos na parte de imagens no computador local. Fica bem simples se o princípio acima estiver entendido pelo aluno. Para adicionar uma imagem que esteja na mesma pasta do seu documento HTML apenas coloque

```

```

Caso ela esteja numa subpasta

```

```

E, finalmente, se ela estiver em pastas que não descendem da que você estiver

```

```

Você pode descer quantos níveis quiser até encontrar sua imagem, mas é recomendado, caso sua imagem esteja em lugares que não sejam o próprio caminho ou uma subpasta, que você utilize o caminho absoluto. Em geral, quando se está fazendo páginas de grande porte, é comum armazenar todas as imagens em subpastas do caminho onde se


encontra o documento HTML. Dessa maneira, o uso de caminhos relativos se torna muito mais ágil e facilita imensamente no caso de ser necessário uma mudança de servidor ou da pasta onde se encontram os arquivos, isso porque quando for completada a operação, nenhum caminho precisará ser consertado no seu código.

Testes com hyperlinks - Konqueror


Localização Editar Ver Ir Favoritos Ferramentas Configurações Janela Ajuda


Localização: /home/gallois/colmeia/apostila_webpage/teste3.html

Essa página terá alguns exemplos com [hyperlinks](#).
O mesmo link acima, só que agora ele vai abrir em uma nova janela e tem um nome diferente: [hyperlinks em outra janela](#).
Para criar os links você deve usar a tag ''. Não se esqueça de fechar a tag =)

Agora um exemplo de link com imagem tirada da Internet: 

As próximas imagens estão armazenadas localmente, olhe as duas maneiras de inseri-las no documento.

Link de carinha feliz com caminho relativo: 

Link de carinha feliz com caminho absoluto: 

Aparentemente as duas últimas linhas são iguais, mas o caminho das imagens é diferente, como você pode ver se procurar no código fonte da página. Para facilitar, olhe abaixo como ficou no meu computador:

Link de carinha feliz com caminho relativo: `<a>
`

Link de carinha feliz com caminho absoluto: ``

Página carregada.

Imagens

As imagens são elementos extremamente comuns nos documentos web atualmente. Elas fazem parte dos mais variados tipos de páginas e podem ser usadas tanto para decorar como para compor a informação de maneira a torná-la mais compreensível. É difícil navegarmos na Internet e entrarmos em uma página que não tenha pelo menos uma imagem. É importante notar que as imagens podem ter diversas aplicações nos documentos web, é possível usá-las como decoração para uma página, como plano de fundo, como explicação para alguma coisa (como usada nos livros), como ligação para outras páginas (hyperlinks), etc.

Os designers fazem muitas coisas com as imagens, principalmente gráficos vetoriais, para criar interfaces mais agradáveis no uso de páginas, paralelamente ao uso de CSS. O resultado traz páginas muito bonitas, mas que algumas vezes ficam bem pesadas para um computador lento ou uma internet de velocidade mais baixa. Cabe a você fazer o julgamento de equilibrar o documento e trazer uma experiência agradável ao usuário, tanto no ponto de vista de qualidade visual quanto em performance.

Como vimos anteriormente, o navegador interpreta as imagens a partir da leitura da tag '``'. Essa tag tem vários atributos e trataremos de alguns deles a seguir. Vamos supor que você queira inserir uma imagem que está no mesmo caminho do seu documento.

```

```

Essa tag deve ser suficiente para fazer com que a imagem apareça em seu navegador.

Você pode também querer pegar uma imagem que está na Internet, usando uma URL para encontrá-la.

```

```

Dessa maneira, você estará usando uma imagem encontrada na Internet, por exemplo.

O uso de caminho absoluto no computador local já foi descrito anteriormente e serve da mesma maneira para inserir uma imagem isolada no documento web.

O atributo 'alt' serve para dar um nome alternativo às imagens nas webpages.

```

```

Ao passar com o mouse em cima da imagem, você poderá notar que o texto mostrado será o que você colocou em 'alt'. Esse recurso também facilita muito a navegação para uma pessoa que estiver usando navegação em modo texto, uma vez que o nome da imagem que aparecerá na tela será o que estiver no valor do atributo 'alt'. O uso desse atributo é considerado uma boa prática e deve ser feito sempre que possível.

As imagens também podem ser usadas como plano de fundo para dar uma aparência mais sofisticada à sua página.

```
<body background="imagem_de_fundo.jpg">
```

Nesse exemplo em específico, não utilizamos a tag '``' pois o navegador já entenderá que ao usar o atributo 'background' da tag '`<body>`' chamaremos uma imagem para ser utilizada como plano de fundo.

Outro recurso que provavelmente será muito utilizado é o alinhamento das imagens com relação ao texto. Você pode basicamente colocá-la acima, no meio ou abaixo do texto. Seguem alguns exemplos

```
<p>  
Uma imagem  
  
no texto  
</p>
```

```
<p>
Uma imagem

no texto
</p>
```

```
<p>
Uma imagem

no texto
</p>
```

```
<p>
Uma imagem

no meio do texto
</p>
```

```
<p>

Uma imagem antes do texto
</p>
```

```
<p>
Uma imagem depois do texto

</p>
```

As linhas acima mostram diversas maneiras de se diagramar a imagem juntamente com o texto. Outra maneira de colocar as imagens na direita ou esquerda do seu texto pode ser da seguinte forma:

```
<p>

Um parágrafo com uma imagem. O atributo <code>align</code>
está configurado para <code>left</code> (esquerda). A imagem ficará
esquerda do texto;
</p>
```

```
<p>

Um parágrafo com uma imagem. O atributo <code>align</code>
está configurado para <code>right</code> (direita). A imagem ficará
direita do texto;
</p>
```

Outros atributos que serão usados com bastante frequência são os responsáveis por redimensionar as imagens. Você precisará de dois deles, um para altura (height) e outro para largura (width). Veja abaixo como utilizá-los.

```
<p>

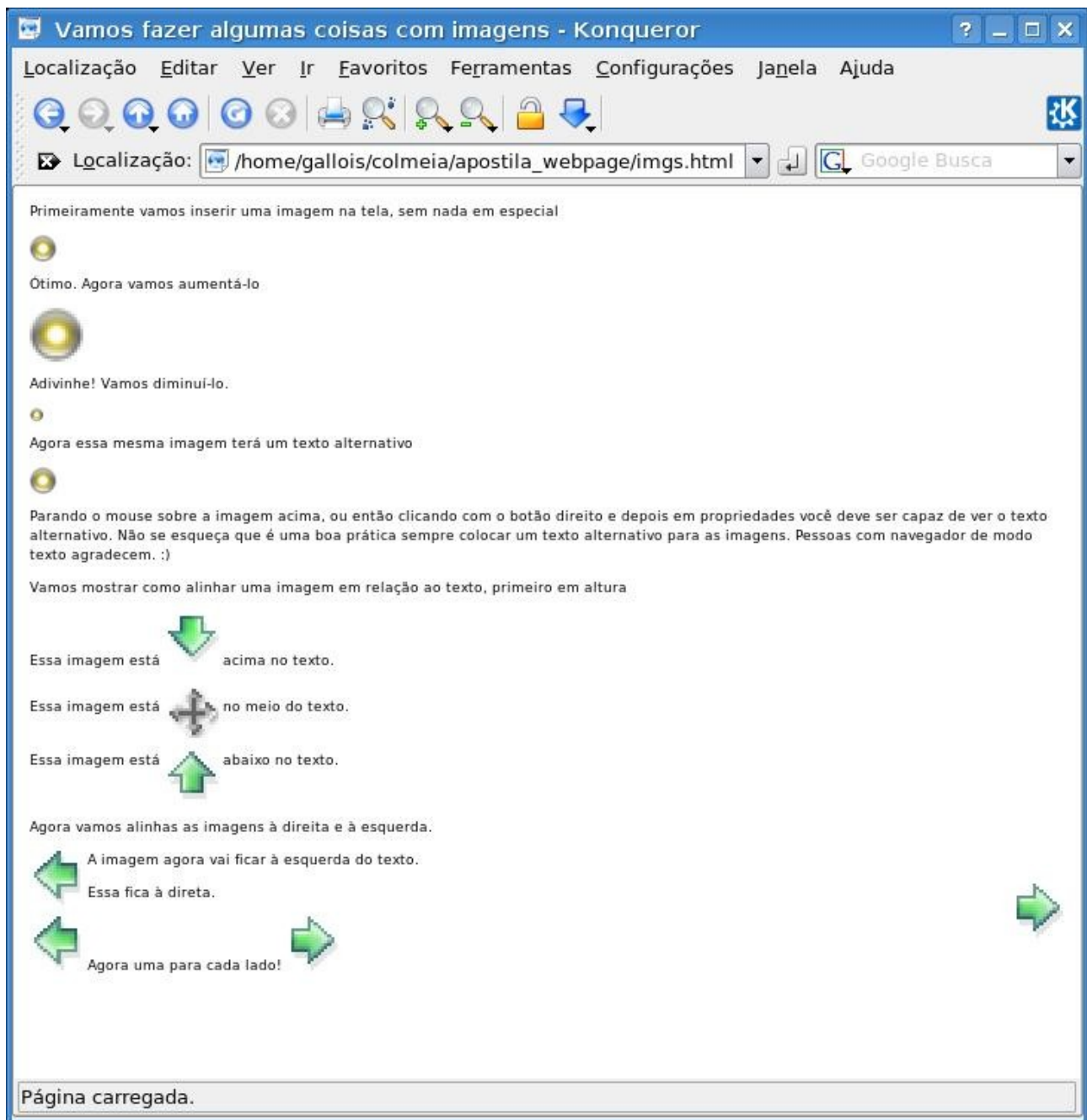
</p>
```

```
<p>

</p>
```

```
<p>
  
</p>

Mudar o valor dos atributos aumenta ou diminui a imagem conforme o que for indicado.



Vamos fazer algumas coisas com imagens - Konqueror

Localização Editar Ver Ir Favoritos Ferramentas Configurações Janela Ajuda

Localização: /home/gallois/colmeia/apostila\_webpage/imgs.html Google Busca

Primeiramente vamos inserir uma imagem na tela, sem nada em especial


Ótimo. Agora vamos aumentá-lo


Adivinhe! Vamos diminuí-lo.


Agora essa mesma imagem terá um texto alternativo

Parando o mouse sobre a imagem acima, ou então clicando com o botão direito e depois em propriedades você deve ser capaz de ver o texto alternativo. Não se esqueça que é uma boa prática sempre colocar um texto alternativo para as imagens. Pessoas com navegador de modo texto agradecem. :)

Vamos mostrar como alinhar uma imagem em relação ao texto, primeiro em altura

Essa imagem está  acima no texto.

Essa imagem está  no meio do texto.

Essa imagem está  abaixo no texto.

Agora vamos alinhar as imagens à direita e à esquerda.

A imagem agora vai ficar à esquerda do texto.

Essa fica à direita.

Agora uma para cada lado!

Página carregada.



# Listas

Nós temos três tipos de listas em HTML, listas ordenadas, listas não ordenadas e listas de definições. Uma lista não ordenada simplesmente coloca um marcador antes de cada um de seus elementos enquanto a lista ordenada numera-os. Além delas temos a lista de definições, que se difere por não ser uma lista de itens. Ela é uma lista de termos e definições. Vamos mostrar alguns exemplos para que você veja melhor. Crie um novo arquivo de texto como nome 'listas.html'.

```
<html>
<head>
<title>Alguns exemplos de listas</title>
</head>

<body>
<p>Veamos abaixo algumas listas e suas diferentes
características.</p>

<p>Primeiramente, uma lista não ordenada.</p>

<h4>Frutas</h4>

 Maçã
 Morango
 Laranja
 Limão
 Abacaxi

<p>Veja agora uma lista ordenada.</p>

<h4>Resultados finais do campeonato de Fórmula 1 de 1991.</h4>

 Senna
 Mansell
 Patrese
 Berger
 Prost
 Piquet

<p>Vamos definir algumas coisas abaixo.</p>
<h4>Bebidas</h4>
<dl>
<dt>Café</dt>
 <dd>Bebida quente e preta</dd>
<dt>Chá</dt>
 <dd>Tem de várias cores e pode ser servido tanto quente como
gelado</dd>
<dt>Água</dt>
 <dd>Sem cor, e com certeza gelada! ;)</dd>
</dl>

</body>
</html>
```

O exemplo é auto-explicativo com relação aos tipos de listas empregados, mas alguns detalhes serão deixados para você praticar. As listas possuem atributos também. Alguns deles

serão explicados aqui.

Em essência, você usa tags '<ul>' ou '<ol>' para iniciar suas listas (não ordenada e ordenada), e '<li>' para cada um dos itens listados. As tags devem sempre ser acompanhadas de fechamento. No caso da lista de definição você tem uma pequena alteração. Além da sua tag, que é '<dl>' cada um dos elementos dela devem ter duas tags diferentes, a '<dt>' para o item em si e '<dd>' para a definição do mesmo.

Para mudar um pouco suas listas ordenadas, tente coisas como:

```
<ol type="A">
```

Isso fará que ao invés de números, a lista seja ordenada com letras do alfabeto maiúsculas.

- A. Maçã
- B. Morango
- C. Laranja
- D. Limão
- E. Abacaxi

Usar

```
<ol type="a">
```

fará a mesma coisa, só que desta vez com letras minúsculas.

É ainda possível usar algarismos romanos para a numeração. Para isso, faça o seguinte.

```
<ol type="I">
```

O resultado será

- I. Maçã
- II. Morango
- III. Laranja
- IV. Limão
- V. Abacaxi

Quer algarismos romanos minúsculos?

```
<ol type="i">
```

Prontinho =)

Para mudar o tipo de marcadores usados nas listas não ordenadas, use:

```
<ul type="disc">
<ul type="circle">
<ul type="square">
```

O primeiro valor do atributo type fará com que os marcadores sejam discos preenchidos, o segundo discos vazios no meio (círculos) e o último mostrará quadrados. É bem simples, tente todos eles.

Outra idéia interessante de listas é que é possível aninhar várias delas. No seu arquivo 'listas.html', adicione o seguinte código.

```
<p>Uma lista aninhada para complicar um pouco</p>
<h4>Bebidas e alguns de seus tipos</h4>
<ul type="circle">
 Café;
 <ul type="square">
 com leite
 preto
 frappé;

```

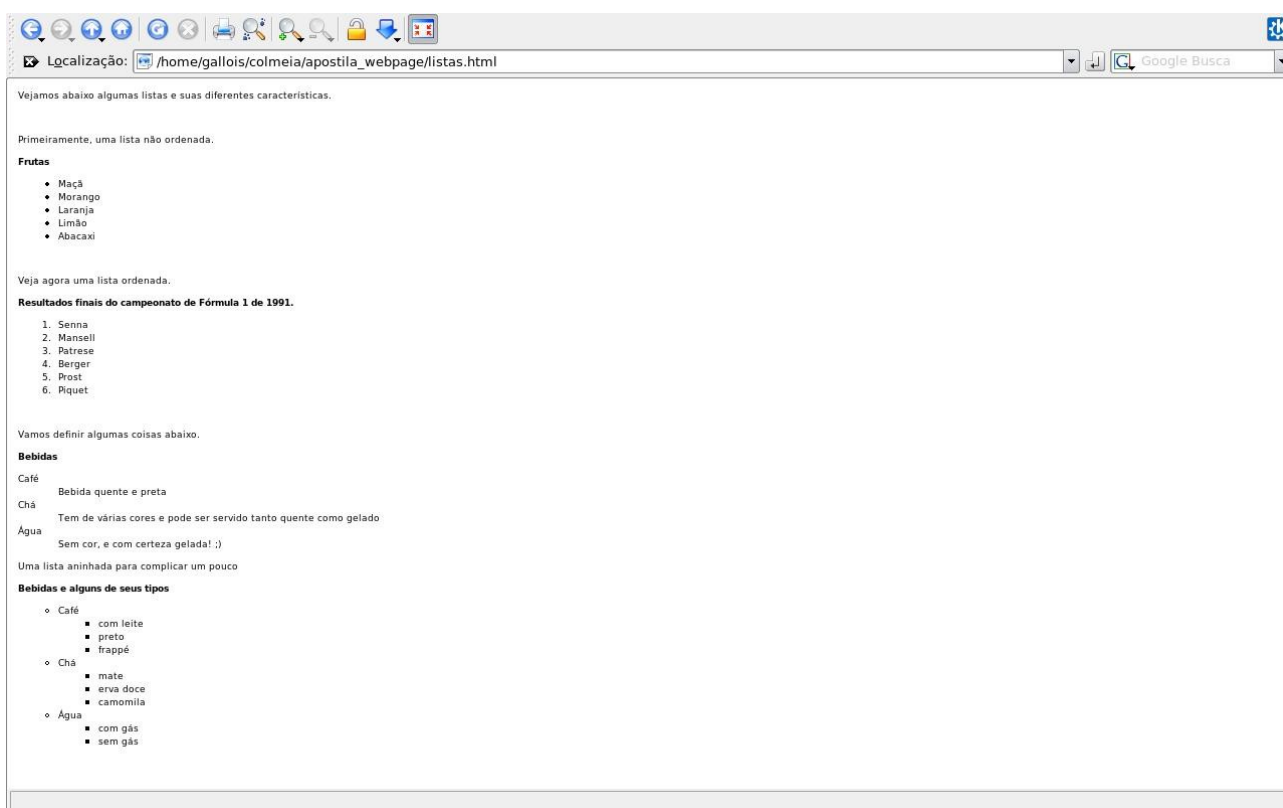
```

Ch&aaacute;
 <ul type="square">
 mate
 erva doce
 camomila

Água
 <ul type="square">
 com g&aaacute;s
 sem g&aaacute;s


```

Agora você tem uma lista aninhada dentro do seu documento e pode ver como ela funciona. É possível aninhar vários níveis, o que pode ser útil dependendo do tipo de lista que você pretende criar. Isso pode ser usado, por exemplo, para definir a árvore genealógica de uma família, embora provavelmente isso te dará muito trabalho! Use sua imaginação.



# Tabelas

As tabelas são em HTML exatamente o que são em qualquer outro lugar. Elas servem para organizar os dados de uma maneira que fique mais fácil para as pessoas entenderem ou tornar mais rápida a comunicação em determinadas ocasiões.

Vamos diretamente ao uso dessas tabelas para depois entrarmos em detalhes. Crie um arquivo chamado 'tabelas.html' e coloque o seguinte código.

```
<html>
<head>
<title>Uma página com tabelas</title>
</head>

<body>
<table border="1">
<tr>
 <th>Parte do computador</th>
 <th>O que ela faz</th>
</tr>
<tr>
 <td>Teclado</td>
 <td>Onde você digita</td>
</tr>
<tr>
 <td>Mouse</td>
 <td>Onde você mexe a seta</td>
</tr>
</table>
</body>
</html>
```

Você vai criar uma tabela com a tag '<table>'. A partir daí, você deve definir as linhas com a tag '<tr>' e as colunas com '<td>'. Cada um dos valores colocados entre as tags '<td>' é considerado estar dentro de uma célula. A tag '<th>' é usada para o cabeçalho da tabela, um texto destacado para explicar um pouco sobre a coluna em questão. O atributo 'border' da tag '<table>' determina a largura da borda da tabela, experimente usar um valor diferente para ver o resultado.

Agora insira dentro da tabela, depois dos outros itens as seguintes linhas:

```
<tr>
 <td>Adesivo no monitor</td>
 <td></td> <!-- não faz nada :)-->
</tr>
```

Recarregue o arquivo e você notará que a célula onde não há nada também ficará sem borda, o que pode dar uma aparência desagradável para o usuário, solucione o problema colocando o caracter especial '&nbsp;' nela.

```
<tr>
 <td>Adesivo no monitor</td>
 <td> </td> <!-- não faz nada :)-->
</tr>
```

Ele faz com que seja inserido um caracter em branco (non-breaking space), diferente de se colocar um espaço normal lá, que o navegador deixa de interpretar. Dessa forma, você engana a tabela como se tivesse realmente algo escrito lá, que de fato há, mas o usuário não verá isso.

É possível também fazer tabelas sem bordas, dependendo do seu objetivo:

```
<h4>A mesma tabela, sem bordas e com legenda</h4>
<table border="0">
```

```

 <caption align="top"><i>Minha legenda</i></caption>
 <caption align="bottom"><i>Minha outra legenda, abaixo da
tabela</i></caption>
 <tr>
 <th>Parte do computador</th>
 <th>0 que ela faz</th>
 </tr>
 <tr>
 <td>Teclado</td>
 <td>Onde você digita</td>
 </tr>
 <tr>
 <td>Mouse</td>
 <td>Onde você mexe a seta</td>
 </tr>
 <tr>
 <td>Adesivo no monitor</td>
 <td> </td> <!-- não faz nada :)-->
 </table>

```

Além da borda de tamanho zero (ou sem borda, se preferir), temos mais duas tags que até então não havíamos apresentado, uma delas se refere à tabela e a outra não. A tag '<caption>' serve para se colocar uma legenda na tabela e pode carregar o atributo 'align', que determinará se ela aparecerá em cima ou em baixo. A outra tag, '<i>', pode ser usada em qualquer lugar do texto e serve para deixar o texto que está entre seu início e fechamento em itálico, assim como a tag '<b>' faz ao passar para negrito.

Agora mostraremos alguns exemplos onde queremos que um determinado campo ocupe mais de uma célula.

```

<h4>Telefone de algumas pessoas que acabei de inventar</h4>
<table border="1">
<tr>
 <th>Nome</th>
 <th colspan="2">Telefone</th>
</tr>
<tr>
 <td>João Frango</td>
 <td>33 2548 3549</td>
 <td>33 2544 3578</td>
</tr>
<tr>
 <td>Valpano</td>
 <td>44 2578 3258</td>
 <td>44 2566 5799</td>
</tr>
</table>

```

```


<h4>Telefone das mesmas pessoas que acabei de inventar, formatando
diferente</h4>
<table border="1">
<tr>
 <th>Nome</th>
 <td>João Frango</td>
 <td>Valpano</td>
</tr>
<tr>
 <th rowspan="2">Telefone</th>
 <td>33 2548 3549</td>
 <td>33 2544 3578</td>
</tr>
<tr>

```

```

 <td>44 2578 3258</td>
 <td>44 2566 5799</td>
</tr>
</table>

```

O exemplo acima, além da utilidade proposta, mostra mais uma coisa: os cabeçalhos não precisam necessariamente ser o primeiro elemento da coluna. Na segunda tabela percebemos que ele está na verdade referindo-se à linha onde ele está. Voltando ao tema, usamos os atributos 'colspan' e 'rowspan' para estendermos uma célula, ambos estão disponíveis tanto para '<th>' quanto para '<td>'. O valor do atributo determinará quantas células o campo ocupará, 'colspan' para colunas e 'rowspan' para linhas.

Vale notar que é possível alinhar tabelas e colocar quaisquer outras tags dentro das células. É possível, por exemplo, fazer uma lista dentro de uma célula de uma tabela, inclusive aninhá-las. Só tome cuidado para não fazer algo tão complicado que você possa se perder ao desenvolver.

Outra possibilidade ao trabalhar com tabelas é alterar a distância entre o texto e as bordas da tabela. Esse parâmetro é controlado pelo atributo 'cellpadding' e pode ser observado no exemplo a seguir.

```

<h4>Tabela sem espa¸amento entre texto e bordas</h4>
<table border="1" cellpadding="0">
<tr>
 <td>1</td>
 <td>2</td>
</tr>
<tr>
 <td>3</td>
 <td>4</td>
</tr>
</table>

```

```


<h4>Tabela com espa¸amento entre texto e bordas</h4>
<table border="1" cellpadding="15">
<tr>
 <td>1</td>
 <td>2</td>
</tr>
<tr>
 <td>3</td>
 <td>4</td>
</tr>
</table>

```

Além de espaçamento entre texto e bordas também podemos deixar uma distância entre uma célula e outra com o atributo 'cellspacing', confira nesse outro exemplo.

```

<h4>Tabela sem espa¸amento entre células</h4>
<table border="1" cellspacing="0" bgcolor="yellow">
<tr>
 <td>1</td>
 <td>2</td>
</tr>
<tr>
 <td>3</td>
 <td>4</td>
</tr>
</table>

```

```


<h4>Tabela com espa¸amento entre células</h4>
<table border="1" cellspacing="15">

```

```

<tr>
 <td bgcolor="yellow">amarelo</td>
 <td bgcolor="red">vermelho</td>
</tr>
<tr>
 <td bgcolor="green">verde</td>
 <td bgcolor="blue">azul</td>
</tr>
</table>

```

Tanto a tabela toda como cada uma das células tratam do atributo background, assim como fizemos anteriormente com a tag '<body>'. Desse modo, é possível alterar a cor desses elementos, como você pode observar no exemplo acima.

Ainda podemos alinhar o texto dentro de uma célula pelo atributo 'align' da mesma maneira como fizemos em outros lugares, o exemplo abaixo é uma alteração daquela nossa "agenda de telefones".

```

<h4>Telefone das mesmas pessoas que acabei de inventar, formatando diferente</h4>

```

```

<table border="1">
 <tr>
 <th>Nome</th>
 <td align="center">João Frango</td>
 <td align="center">Valpano</td>
 </tr>
 <tr>
 <th rowspan="2">Telefone</th>
 <td>33 2548 3549</td>
 <td>33 2544 3578</td>
 </tr>
 <tr>
 <td>44 2578 3258</td>
 <td>44 2566 5799</td>
 </tr>
</table>

```

Note que o nome das pessoas agora estão centralizados nas células.

Existem outros atributos que não apresentaremos aqui para não alongarmos demais essa parte, com o conteúdo apresentando até aqui já é possível construir tabelas bem completas.

Parte do computador	O que ela faz
Teclado	Onde voce digita
Mouse	Onde voce mexe a seta
Adesivo no monitor	

A mesma tabela, sem bordas e com legenda

Minha legenda

Parte do computador	O que ela faz
Teclado	Onde voce digita
Mouse	Onde voce mexe a seta
Adesivo no monitor	

Telefones de algumas pessoas que soubei de inventar

Nome	Telefone
José Frango	22 2548 2549 22 2544 2578
Valjane	44 2578 2258 44 2588 5788

Telefones das mesmas pessoas que soubei de inventar, formataas de diferente

Nome	José Frango	Valjane
Telefone	22 2548 2549 22 2544 2578	44 2578 2258 44 2588 5788

Tabela sem espaçamento entre texto e bordas

1	2
3	4

Tabela com espaçamento entre texto e bordas

1	2
3	4

Tabela sem espaçamento entre células

1	2
3	4

Tabela com espaçamento entre células

amarelo	vermelho
verde	azul



# Frames

Com frames você pode inserir mais de um documento dentro da mesma tela do navegador. Você pode fazer frames horizontais ou verticais, que determinarão a maneira como os documentos serão exibidos. Cada documento HTML será chamado frame e cada frame é independente do outro.

Embora isso ajude bastante a organizar o conteúdo das páginas, algumas desvantagens são notáveis, como a maior necessidade de atenção do desenvolvedor para manter os documentos HTML e a dificuldade para a impressão de uma página inteira.

Basicamente, o frame faz com que o documento se divida todo e em cada uma dessas partes você possa criar uma página independente e diferente de todas as outras que você está utilizando. Imagine uma bandeja de refeitório toda dividida onde em cada lugar seja possível colocar um alimento diferente sem que ele se misture com o outro.

Para colocar frames em suas páginas, primeiramente você tem que utilizar a tag '<frameset>' que é o conjunto de frames que serão usados. Essa tag trará definições básicas, como se os frames estarão divididos em linhas ou colunas e o tamanho das mesmas. Veja abaixo um exemplo de utilização.

```
<frameset cols="30%, 70%">
 <frame src="uma_pagina.html" />
 <frame src="outra_pagina.html" />
</frameset>
```

Esse exemplo mostra a divisão dos frames com porcentagem de tamanho a ser ocupado na tela. Existe uma outra maneira de fazer isso, que é com o número de pixels que cada um utilizará.

```
<frameset cols="200, 300" />
```

Ou ainda você pode querer um frame use o espaço restante, como no exemplo abaixo:

```
<frameset cols="20%, 20%, *" />
```

Use os códigos abaixo para ter uma idéia do funcionamento dos frames.

frames.html

```
<html>
<head>
<title>Vamos usar uns frames</title>
</head>

<frameset cols="20%, 80%">
 <frame src="uma_pagina.html" />
 <frame src="outra_pagina.html" />
</frameset>

<body>
<noframes>0 seu navegador não suporta frames!</noframes>
</body>
</html>
```

uma\_pagina.html

```
<html>
<head>
<title>Essa é uma página</title>
</head>

<body>
Uma página qualquer! =)
```

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

outra\_pagina.html

```
<html>
<head>
<title>Essa é outra pàgina</title>
</head>
```

```
<frameset rows="100, *, 200">
 <frame src="oi.html" />
 <frame src="tudo_bem.html" />
 <frame src="tchau.html" />
</frameset>
```

```
<body>
<noframes>0 seu navegador não suporta frames!</noframes>
</body>
</html>
```

oi.html

```
<html>

<frameset cols="30%, 30%, *">
 <frame src="oi1.html" />
 <frame src="oi2.html" />
 <frame src="oi3.html" />
</frameset>
```

```
<body>
<noframes>0 seu navegador não suporta frames!</noframes>
</body>
</html>
```

oi1.html

```
<html>

<body bgcolor="red">
<i>oi um</i>
</body>

</html>
```

oi2.html

```
<html>

<body bgcolor="turquoise">
<i>oi dois</i>
</body>

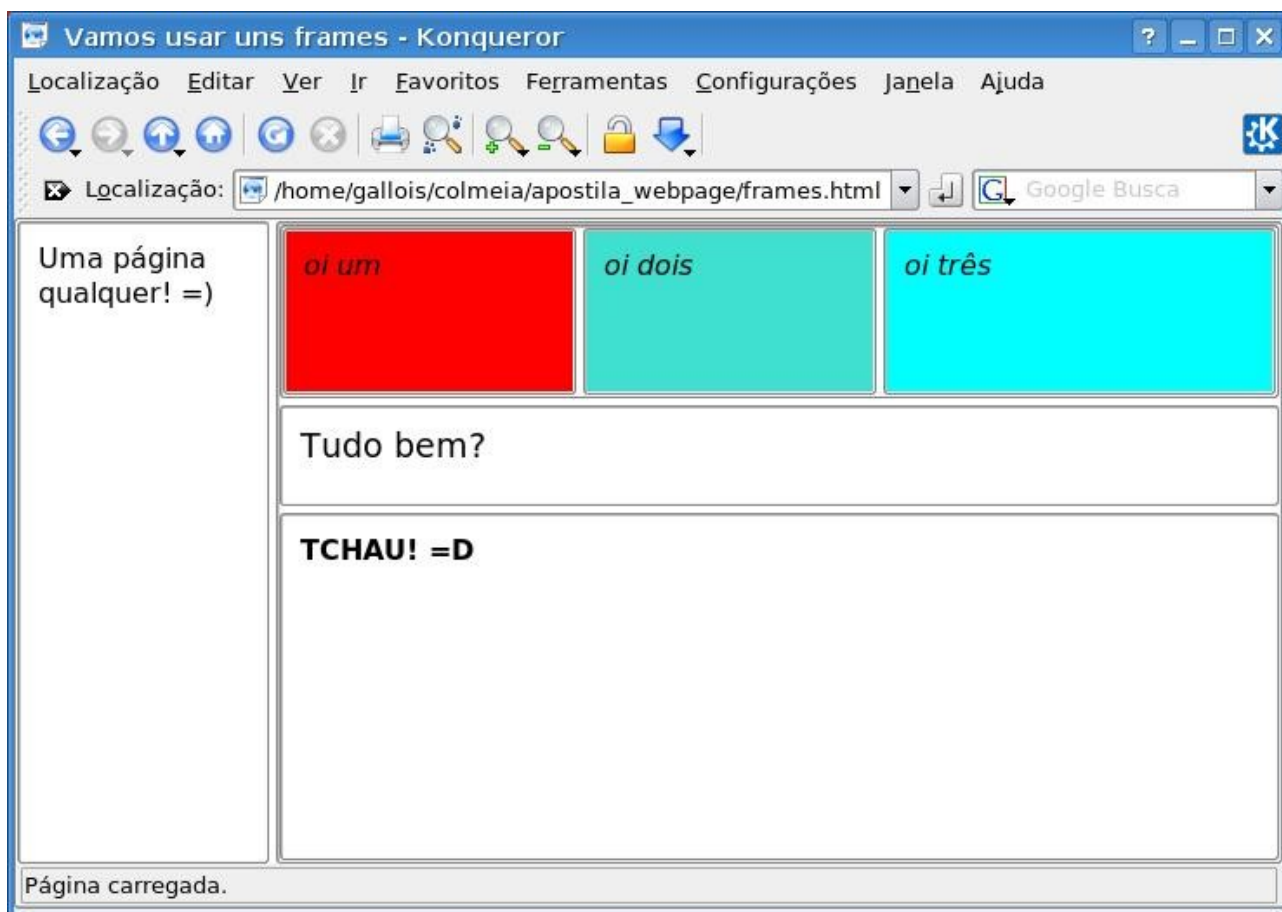
</html>
```

oi3.html

```
<html>

<body bgcolor="cyan">
<i>oi três</i>
```

```
</body>
</html>
tudo_bem.html
<html>
<big>Tudo bem?</big>
</html>
tchau.html
<html>
TCHAU! =D
</html>
```



Como vários documentos foram usados, organize-os de maneira a conseguir um resultado parecido com a imagem acima. Além dos frames tivemos algumas outras tags de formatação. '<strong>' faz basicamente o mesmo que a tag '<b>', isto é, deixa o texto em negrito, mas é na verdade um texto forte com ênfase. '<big>' deixa o texto em tamanho maior na tela. Outro detalhe a ser observado é que no caso dos frames que são compostos pelos documentos oi1, oi2 e oi3.html, cada um tem uma cor de fundo diferente. Isso foi usado apenas para mostrar que cada um dos frames é totalmente independente do outro, não compartilhando o mesmo plano de fundo.

Apenas para complementar, é aconselhável o uso da tag '<noframes>' nos documentos que contenham frames. Essa tag especifica o que deverá ser feito caso o navegador não seja capaz de trabalhar com frames.

# Forms

Um form é uma área que contém elementos de formulário em um documento HTML. Esses elementos são responsáveis por permitir que o usuário entre com informações (como campos de texto, áreas de texto, menus, etc.) em um formulário. São definidos pela tag '<form>'.  
Os formulários são constantes nas páginas encontradas na internet e basicamente todo cliente de webmail tem elementos de formulário, mais precisamente campo de texto e campo de senha para poder fazer a entrada de seu login e senha. Mesmo que você não se lembre deles agora, provavelmente verá que eles são familiares depois de alguns exemplos.

Faremos nosso primeiro exercício exatamente em cima destes dois exemplos. Os formulários são constantes nas páginas encontradas na internet e basicamente todo cliente de webmail tem elementos de formulário, mais precisamente campo de texto e campo de senha para poder fazer a entrada de seu login e senha. Mesmo que você não se lembre deles agora, provavelmente verá que eles são familiares depois de alguns exemplos.

Faremos nosso primeiro exercício exatamente em cima destes dois exemplos:

```
<html>
<head>
<title>Estamos quase acabando, paciência com os forms</title>
</head>

<body>
<p>Veja se isto te lembra algo em clientes de e-mail</p>
<form>
Login:<input type="text" name="login" />

Senha:<input type="password" name="password" />
</form>
</body>
</html>
```

Logo acima podemos ver algumas coisas a respeito dos forms. Uma delas é que eles não têm muita serventia se não tivermos outros elementos dentro deles, os '<input>' (Lembre-se de fechar a tag com um '/' no final dela. Como já disse, todas as tags devem ser fechadas!). São eles os responsáveis por mostrar no navegador os campos que serão responsáveis por possibilitar a entrada de algum tipo de informação por parte do usuário. O atributo 'type' determina o tipo do elemento, veja abaixo alguns deles:

text: Usado quando você quer inserir caracteres, números e afins no formulário.

password: Bem parecido com o 'text' mas sua entrada de dados não aparecerá explicitamente para o usuário. Usada principalmente, como o próprio nome sugere, para a entrada de senhas.

radio: São os radio buttons, ou botões de opção. Elas permitem que o usuário seja capaz de escolher um elemento entre um número limitado de eles.

checkbox: As check boxes, ou caixas de seleção, têm uma finalidade parecida com a dos botões, com a diferença de permitir que sejam selecionados mais de um elementos.

submit: O botão submit serve para enviar os dados.

reset: O botão reset serve para apagar todas as entradas do formulário.

Uma outra tag que é bem comum nas páginas é a '<select>' que geralmente também estão dentro de formulários. Ela terá sub-elementos que serão as tags '<option>' que serão valores que popularão o menu drop-down. O exemplo abaixo clarifica o seu uso

```
<select name="servico">
 <option value="email">E-mail</option>
 <option value="agenda">Agenda</option>
 <option value="noticias">Notícias</option>
 <option value="jogos">Jogos</option>
</select>
```

Podemos usar os botões de opção para fazer algo semelhante ao select

```



```

Os botões de opção precisam basicamente de dois atributos para funcionar corretamente. Um deles é o 'name', que diz o grupo ao qual ele pertence. Os botões devem estar agrupados para quando os dados forem recuperados na página que receber o formulário, as ações corretas sejam tomadas. O atributo 'value' é o valor do botão em si, dentro do grupo e sabendo qual o nome da opção, é possível tratar de maneira precisa o que deverá ocorrer com os dados do formulário.

O mesmo conceito de grupos e valores individuais também se aplica às caixas de seleção. Da mesma maneira elas devem ser agrupadas e depois receber um valor específico para cada uma delas. O exemplo abaixo mostra isso.

```

E-mail: <input type="checkbox" name="servico_cb" value="email" />

Agenda: <input type="checkbox" name="servico_cb" value="agenda" />

Notícias: <input type="checkbox" name="servico_cb" value="noticias"
/>

Jogos: <input type="checkbox" name="servico_cb" value="jogos" />


```

Como você pode perceber, elas possuem igualmente os campos 'name' e 'value' que funcionam da mesma maneira.

É possível colocar alguns botões também, eles podem ter várias utilidades, mas usando apenas HTML é pouco provável que você encontre alguma utilidade prática para eles. De qualquer maneira, veja abaixo como utilizá-los:

```
<input type="button" value="Oi, eu sou um botão">
```

Não há muito mistério quanto à sua sintaxe, o tipo deve ser 'button' e o atributo 'value' dirá o que aparecerá no texto do botão.

Vamos agora tratar o caso de dois botões bem comuns em formulários, 'submit' e 'reset'. Cada um deles é um tipo especial de elemento usados, respectivamente, para enviar o formulário e apagar tudo que foi entrado no mesmo. Para usá-los, o tipo (type) da tag '<input>' deve ser 'submit' ou 'reset'. Além disso, você pode dar um outro nome para o botão, alterando o valor do atributo 'value'.

```

<input type="submit" value="Enviar" />
<input type="reset" value="Apagar" />

```

As linhas acima mostram o exemplo de um botão 'submit' e um 'reset', os dois estão com seus 'value's alterados para poder exibir nomes em português. Coloque-os em um documento de exemplo, junto com mais campos de formulário e clique em ambos para entender o funcionamento deles. O botão reset tem uma característica mais explícita, você verá os campos que já estavam preenchidos serem todos esvaziados, e é exatamente isso que ele deve fazer. No caso do 'submit' os dados também serão apagados, mas algo mais acontece. Veja na barra de endereços do navegador que o URL será alterado de acordo com os dados que você tiver inserido. É exatamente dessa maneira que as páginas se comunicam.

É verdade que não faz muito sentido esse exercício, já que os dados inseridos não são utilizados para nada. O tratamento dessas informações é feita, de maneira mais básica, pelo CGI mas podem também ser usadas diversas linguagens de programação para os mais variados fins. Isso, porém, foge do escopo do presente material e portanto não será explicado. Entretanto, fica aqui mais um instrução referente à formulários, eles podem ser usados para redirecionar à alguma outra página com as informações que foram fornecidas. Isso é feito através do atributo 'action' da tag '<form>'.

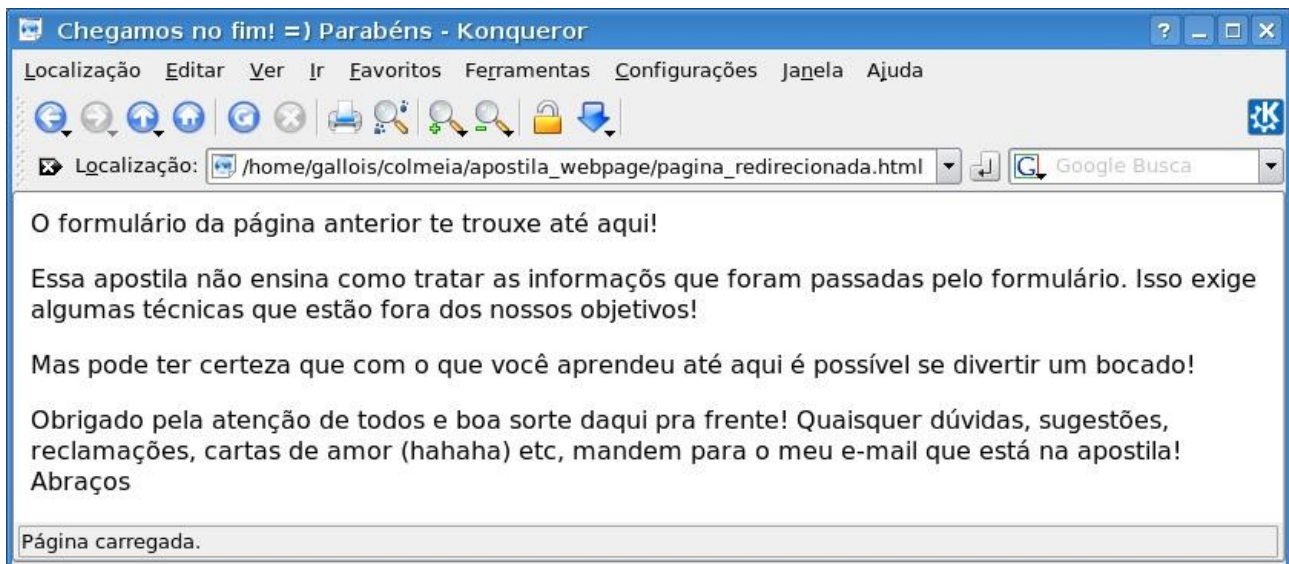
```
<form name="entrada" action="pagina_redirecionada.html">
```

Esse atributo está intimamente ligado ao botão 'submit', uma vez que esse endereço será acessado ao clicar nele. Resumindo, você coloca em um formulário a ação que deverá ser

desempenhada ao se clicar no botão 'submit'. O documento receberá as informações que estavam contidas no formulário e poderão, conseqüentemente, ser tratadas para algum fim específico. Faça o teste preenchendo um formulário e em seguida clicando no botão 'submit' que redirecione para uma página local válida, você poderá ver que no final do URL haverão os dados que estavam no formulário da página anterior.

The screenshot shows a Konqueror web browser window with the following content:

- Window Title:** Estamos quase acabando, paciência com os forms - Konqueror
- Menu Bar:** Localização, Editar, Ver, Ir, Favoritos, Ferramentas, Configurações, Janela, Ajuda
- Address Bar:** /home/gallois/colmeia/apostila\_webpage/forms.html
- Search Bar:** Google Busca
- Form Content:**
  - Greeting: Bem vindo à nossa página de serviços imaginária =D
  - Fields: Login: [text input], Senha: [password input]
  - Label: Qual serviço deseja utilizar?
  - Dropdown: E-mail
  - Two columns of radio buttons:
    - Column 1: Ou então, selecione por aqui. (E-mail, Agenda, Notícias, Jogos)
    - Column 2: Ou faça de conta que podemos acessar vários serviços ao mesmo tempo (E-mail, Agenda, Notícias, Jogos)
  - Button: Oi, eu sou um botão
  - Text: Terminando nosso formulário, veja os últimos botões da nossa apostila.
  - Buttons: Enviar, Apagar
- Status Bar:** Página carregada.



## Considerações Finais

Bom, com isso concluímos a apostila. Espero que seja útil como aprendizado e como referência rápida. Alguns tópicos interessantes para aprofundar o que foi mostrado aqui são:

JavaScript: linguagem de programação extremamente útil para lidar dinamicamente com os conteúdos.

CGI: Um protocolo usado para tratar requisições do cliente web e retornar conteúdo.

XML: Uma outra linguagem de formatação, pode ser usada para vários propósitos

CSS: Linguagem para descrever a apresentação de um documento HTML (entre outros).

Responsável por muitos dos layouts presentes nas páginas.

<div>: Tags de HTML que não foram abordadas aqui. Em conjunto com CSS, compõem uma grande parte das páginas atuais da internet.

Por fim, fique com uma lista de tags e códigos para caracteres especiais.

## Lista de caracteres especiais

&lt;	<	&gt;	>	
&amp;	&	&quot;	"	
&Aacute;	Á	&Agrave;	À	
&Eacute;	É	&Egrave;	È	
&Iacute;	Í	&Igrave;	Ì	
&Oacute;	Ó	&Ograve;	Ò	
&Uacute;	Ú	&Ugrave;	Ù	
&aacute;	á	&agrave;	à	
&eacute;	é	&egrave;	è	
&iacute;	í	&igrave;	ì	
&oacute;	ó	&ograve;	ò	
&uacute;	ú	&ugrave;	ù	
&Auml;	Ä	&Acirc;	Â	
&Euml;	Ë	&Ecirc;	Ê	
&Iuml;	Ï	&Icirc;	Î	
&Ouml;	Ö	&Ocirc;	Ô	
&Uuml;	Ü	&Ucirc;	Û	
&auml;	ä	&acirc;	â	
&euml;	ë	&ecirc;	ê	
&iuml;	ï	&icirc;	î	
&ouml;	ö	&ocirc;	ô	
&uuml;	ü	&ucirc;	û	
&Atilde;	Ã	&aring;	å	
&Ntilde;	Ñ	&Aring;	Å	
&Otilde;	Õ	&Ccedil;	Ç	
&atilde;	ã	&ccedil;	ç	
&ntilde;	ñ	&Yacute;	Ý	
&otilde;	õ	&yacute;	ý	
&Oslash;	Ø	&yuml;	ÿ	
&oslash;	ø	&THORN;	Þ	
&ETH;	Ð	&thorn;	þ	
&eth;	ð	&AElig;	Æ	
&szlig;	ß	&aelig;	æ	
&frac14;	$\frac{1}{4}$	&nbsp;		
&frac12;	$\frac{1}{2}$	&iexcl;	!	
&frac34;	$\frac{3}{4}$	&pound;	£	
&copy;	©	&yen;	¥	
&reg;	®	&sect;	§	
&ordf;	ª	&curren;	¤	
&sup2;	²	&brvbar;		
&sup3;	³	&laquo;	«	
&sup1;	¹	&not;	¬	
&macr;	-	&shy;	–	



&micro;	μ	&ordm;	º
&para;	¶	&acute;	´
&middot;	·	&uml;	¨
&deg;	°	&plusmn;	±
&cedil;	ç	&raquo;	»
&iquest;	¿		
&times;	×	&cent;	¢
&divide;	÷	&euro;	€
&#147;	“	&#153;	™
&#148;	”	&#137;	‰
&#140;	Œ	&#131;	ƒ
&#135;	‡	&#134;	†

## Lista de tags

tag	descrição
<!--...-->	define um comentário
<!DOCTYPE>	define o tipo do documento
<a>	define uma âncora
<abbr>	define uma abreviação
<acronym>	define um acrônimo
<address>	define um elemento de endereço
<area>	define uma área dentro de um mapa de imagem
<b>	define texto em negrito
<base>	define o URL base para todos os links da página
<bdo>	define a direção em que o texto será mostrado
<big>	define texto grande
<blockquote>	define uma longa citação
<body>	define o elemento de corpo
 	insere uma quebra de linha
<button>	define um botão
<caption>	define a legenda de uma tabela
<cite>	define uma citação
<code>	define um texto de código de computador
<col>	define atributos para as colunas de tabela
<colgroup>	define um grupo de colunas de tabela
<dd>	define a descrição de uma definição
<del>	define texto apagado
<div>	define uma seção no documento
<dfn>	define uma definição de termo
<dl>	define uma lista de definições
<dt>	define um termo de definição
<em>	define um texto enfatizado
<fieldset>	define um arranjo de campos
<form>	define um formulário
<frame>	define uma sub-janela (frame ou quadro)
<frameset>	define um arranjo de frames
<h1> to <h6>	define cabeçalhos 1 a 6
<head>	define informações sobre o documento
<hr>	define uma linha horizontal
<html>	define um documento HTML
<i>	define um texto em itálico
<iframe>	define um frame em linha
<img>	define uma imagem
<input>	define um campo de entrada
<ins>	define um texto inserido
<kbd>	define um texto de teclado
<label>	define um rótulo para controle de formulário
<legend>	define o título de um arranjo de campos
<li>	define um item de lista

<link>	define uma referência de recurso
<map>	define um mapa de imagem
<meta>	define meta-informação
<noframes>	define uma seção sem frames
<noscript>	define uma seção sem scripts
<object>	define um objeto embarcado
<ol>	define uma lista ordenada
<optgroup>	define um grupo de opções
<option>	define uma opção em uma lista drop-down
<p>	define um parágrafo
<param>	define um parâmetro para um objeto
<pre>	define texto pré-formatado (respeita espaços e parágrafos)
<q>	define uma citação curta
<samp>	define um exemplo de código de computador
<script>	define um script
<select>	define texto selecionável
<small>	define um texto pequeno
<span>	define uma seção em um documento
<strong>	define texto destacado
<style>	define uma definição de estilo
<sub>	define texto subscripto
<sup>	define texto sobrescrito
<table>	define uma tabela
<tbody>	define o corpo de uma tabela
<td>	define uma célula de uma tabela
<textarea>	define uma área de texto
<tfoot>	define o rodapé de uma tabela
<th>	define o cabeçalho de uma tabela
<thead>	define o cabeçalho de uma tabela
<title>	define o título de um documento
<tr>	define uma linha de tabela
<tt>	define texto monoespaçado
<ul>	define uma lista não ordenada
<var>	define uma variável