



1-INTRODUÇÃO

1.1 – Abrindo um Arquivo Local no Navegador

Usando o Netscape Navigator:

Digite no campo *Go to* o endereço completo do arquivo que se deseja abrir precedido de file:/// . A figura 1.1 ilustra essa situação. Nesse caso será aberto o arquivo intro.html que se encontra no diretório Cursos\HomePage do drive c:\.

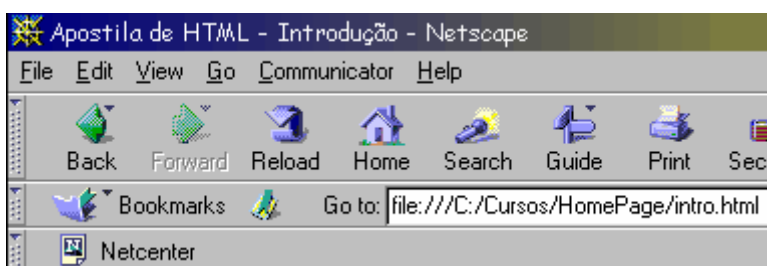


Figura 1.1 - Campo *Go to* do Netscape Navigator indicando um arquivo local

USANDO O MICROSOFT INTERNET EXPLORER

Digite no campo *Endereço* o endereço completo do arquivo que se deseja abrir. A figura 1.1 ilustra essa situação. Nesse caso será aberto o arquivo intro.html que se encontra no diretório Cursos\HomePage do drive c:\.

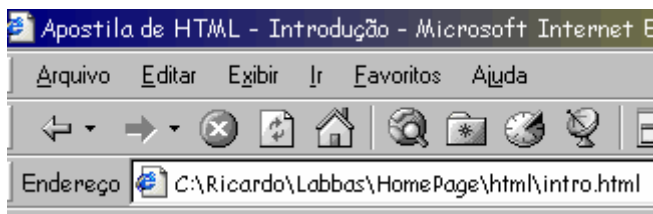


Figura 1.2 - Campo *Endereço* do Microsoft Internet Explorer indicando um arquivo local



1.2 - Acessando um Site na Internet

USANDO O NETSCAPE NAVIGATOR:

Digite no campo *Go to* o endereço completo do site que se deseja abrir, pode-se omitir o **http: //**. A figura 1.3 ilustra essa situação. Nesse caso estaremos acessando a página <http://www.labbas.eng.uerj.br/cursos/hp>.

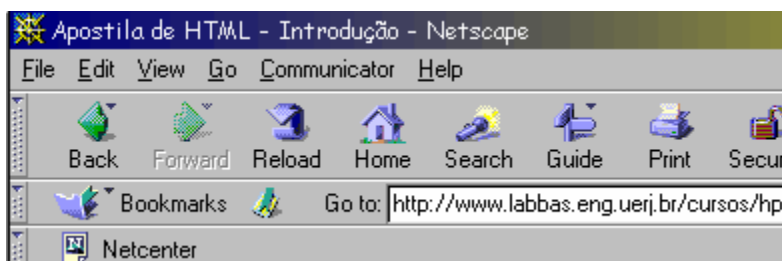


Figura 1.3 - Campo *Go to* do Netscape Navigator indicando um site na Internet

USANDO O MICROSOFT INTERNET EXPLORER:

Digite no campo *Endereço* o endereço completo do site que se deseja abrir, pode-se omitir o **http: //**. A figura 1.4 ilustra essa situação. Nesse caso estaremos acessando a página <http://www.labbas.eng.uerj.br/cursos/hp>.



Figura 1.4 - Campo *Endereço* do Microsoft Internet Explorer indicando um site na Internet

1.3 - Navegação Básica com Netscape e Internet Explorer

O botão Back:

Clique no botão back para voltar à página visitada anteriormente



O BOTÃO FORWARD:

Clique no botão Forward para voltar a última página que foi acessada antes de iniciar o clique do botão back.



Figura 1.5 - Botão Back



Figura 1.6 - Botão Forward

1.4 - Como Salvar uma Página da Internet

A maneira mais fácil de aprender HTML é analisando o código-fonte da página de outra pessoa. Para que possamos fazer essa análise e mudarmos o que for de nosso interesse devemos salvar a página em um arquivo no nosso computador. Para isso clicamos no menu Arquivo/Salvar Como... (File/Save As...)

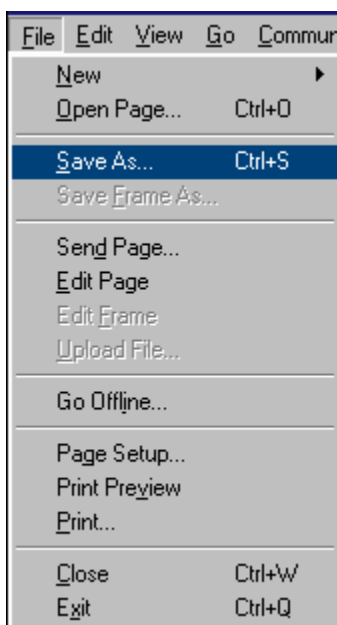


Figura 1.8

Com isso aparecerá a caixa de diálogo de gravação do windows (figura 1.9) e você deve informar o diretório e o nome do arquivo a ser gravado.

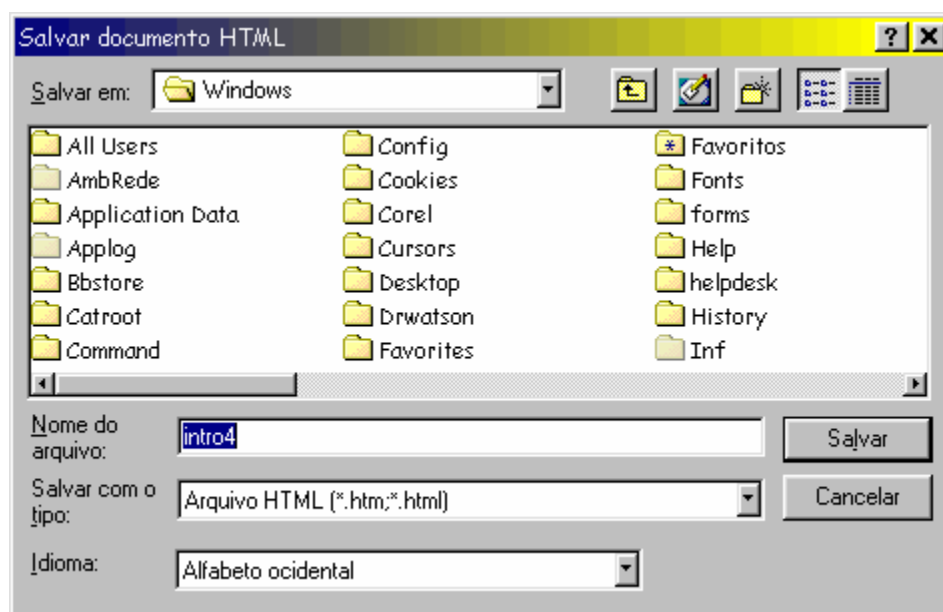


Figura 1.9 - Caixa de diálogo de gravação do Windows

1.5 - Como Salvar uma Figura da Internet

Salvar imagens da Internet pode ser, sem dúvida nenhuma, uma grande fonte de material para sua página, mas, lembre-se sempre que alguns sites estão protegidos por direitos autorais o que o impede de copiar as imagens ali contidas.

Para salvar a imagem, clique com o botão direito do mouse sobre a figura e selecione a opção *salvar figura como...*, O navegador mostrará a caixa de diálogo de gravação do windows (figura 1.9) onde você poderá escolher o diretório e o nome da imagem a serem gravados.

1.6 - Como Editar Html com o Bloco de Notas

O Bloco de Notas (Notepad) do windows é o melhor e mais simples editor de HTML, toda essa apostila foi construída nele. Para você iniciar o Bloco de Notas clique no botão iniciar da barra de tarefas, selecione Programas, Acessórios e finalmente Bloco de Notas ou Notepad.

Já no Bloco de Notas clique em Arquivo e selecione a opção Abrir. Surge uma caixa de diálogo, altere o campo *Arquivo do tipo* para *Todos os Arquivos (*.*)* e então selecione o arquivo.html a ser editado.

Com isso você estará tendo acesso ao código-fonte do seu HTML. A figura 1.10 mostra o bloco de notas com o código fonte dessa página da apostila.



1.7 - As Tags ou Comandos Html

Quando um navegador exibe uma página da Web, ele lê o texto da página e procura símbolos especiais denominados tags, que informam como o texto ou informação deve ser exibido. Um tag pode dizer que um determinado texto deve ser exibido em *itálico* e outro em **negrito** ou pode informar ainda a existência de um parágrafo, etc...

As tags normalmente são especificadas em pares, delimitando o início e o fim do texto que sofrerá determinada formatação, mas também existem tags individuais.

As tags são identificadas pelos sinais '< >' ou '</ >'.
</>

Entre os sinais '< >' são especificados os comandos e, no caso de tags que envolvem o texto sua finalização deve ser feita utilizando-se o sinal '</ >' indicando que a tag está finalizando a marcação de um texto.

O formato genérico de uma tag é '<Nome_da_tag parâmetros >' texto '</Nome_da_tag>'.
</>

Vamos ver agora como podemos criar um arquivo HTML. Já com o Bloco de Notas aberto, clique no menu Arquivo e selecione a opção Novo. Um documento em branco aparecerá. Digite agora o código abaixo:

```
<html>

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

<body>
  Essa é a menor página HTML possível
</body>

</html>
```

1.8 - Análise do Menor Html Possível

Na seção anterior vimos como pode ser construído o menor HTML possível. Agora vamos fazer a análise desse arquivo.

A primeira tag do arquivo é <html>, esse tag indica ao navegador que se trata de um arquivo do tipo HTML.

A segunda tag que aparece é a tag <head> que indica o início do cabeçalho, as informações contidas dentro do cabeçalho não são exibidas pelo navegador.



A terceira tag é <title> que informa o início do título da página, no nosso caso 'Minha Primeira Página'. O título é o que aparece na barra superior do navegador.

Agora temos </title> que encerra o título, </head> que encerra o cabeçalho.

A próxima tag é <body> que indica o início do corpo do documento, as informações contidas no corpo são aquelas que serão exibidas pelo navegador.

Temos então o texto que é exibido na tela e depois a tag </body> que finaliza o corpo do documento e por último </html> que encerra o documento.

1.9 - Principais Elementos de uma Página Html

Uma página HTML é composta basicamente de títulos, textos e links. Todos esses elementos são posicionados na página por meios de tags. Os próximos capítulos dessa apostila mostrarão gradativamente como deve ser usada cada tag para se criar uma página da Web que seja bastante confortável.

É importante lembrar que aqui só serão descritas técnicas básicas de criação de Homepages, é imprescindível que você salve páginas de outras pessoas e analise o código-fonte para poder fazer páginas que sejam realmente atrativas.

1.10 – Exercícios

1. Como salvar uma imagem da internet para o meu computador?
2. Como salvar uma página da Internet para o meu computador?
3. Quando uma página é salva, as figuras que aparecem nela são gravadas?

2 – FORMATAÇÃO

2.1 - Parágrafos e Quebras de Linha

Ao contrário dos editores de texto tradicionais como o Microsoft Word e o próprio Bloco de Notas, nos quais você insere uma quebra de linha sempre que pressiona a tecla Enter, em um arquivo HTML é necessário colocar uma tag específica para que o navegador entenda que você deseja mudar de linha ou começar um novo parágrafo. Se você colocar, no código fonte, várias quebras de linha no seu texto, perceberá que no navegador seu texto aparecerá contínuo. Para criar quebras de linha e parágrafos existem as tags <p> e
.

Inserir Parágrafo - A tag <p> :

Para forçar o início de um novo parágrafo, ou seja, avançar uma linha em branco e iniciar o texto na segunda linha em branco após o final do parágrafo anterior é preciso colocar no código-fonte uma tag <p>. Essa tag pode aparecer individualmente ou em um par <p> </p>.



Quando uma tag <p> é inserida em um documento HTML, o navegador vai sempre pular uma linha em branco e se posicionar no início segunda linha, independentemente do tamanho da janela, ou seja, a quebra de linha será sempre naquele ponto.

**Inserir Quebra de Linha - A tag
 :**

A tag <p> representa um parágrafo e isso faz com que o navegador insira uma linha em branco entre o texto anterior a tag e o seguinte. Já a tag
 representa uma quebra de linha, o efeito é parecido com o produzido no Bloco de Notas quando pressionamos a tecla Enter, ou seja, o texto passa para a linha de baixo, porém não é deixada nenhuma linha em branco.

Pegamos o código-fonte do exemplo anterior, sobre o uso da tag <p> e substituímos algumas tags <p> por tags
, *veja a diferença*. Ao contrário da tag <p>, a tag
 sempre aparece individualmente.

Recuo de parágrafos - A tag <dd> :

A tag <dd> tem por objetivo inserir um recuo do texto na posição onde foi especificado. Esse recurso é muito útil, visto que é a forma mais fácil de se produzir recuos em parágrafos.

Muito criadores de páginas para a web não utilizam essa tag, isso é considerado um erro muito grande de tipografia. Depois da invenção de navegadores gráficos, a Web deixou de ser simplesmente um veículo de comunicação onde pesquisadores e cientistas do mundo todo compartilhavam suas informações.

Hoje a Web é muito mais que isso e é importante que as páginas tenham um bom visual. Em uma boa tipografia, parágrafos não são separados por linhas em branco e sim por recuos na primeira linha.

Um parágrafo deve começar na linha imediatamente abaixo daquela onde terminou o parágrafo anterior e deve receber um recuo de tamanho igual a três ou cinco letras.

Infelizmente a HTML ainda não pode nos proporcionar o controle de qual recuo será aplicado, mas pelo menos já é possível recuarmos os parágrafos.

Linhas de Separação - A tag <hr> :

A tag <hr> tem por objetivo inserir uma linha separadora na posição onde foi especificado. Apesar desse recurso ser muito usado pelos criadores de Homepages, os maiores Designers não recomendam seu uso, pois uma barra horizontal é uma barreira que deve ser ultrapassada pelo leitor para que ele continue sua leitura e isso pode fazer com que o visitante saia da página.



As linhas de separação, em geral, são usadas em locais onde elas poderiam simplesmente não existir, pois na verdade não estão separando dois assuntos completamente distintos e sim dois tópicos relacionados ao mesmo assunto. Nesse caso, a inserção de um parágrafo é suficiente.

Caso seja necessário tratar de dois assuntos completamente diferentes é recomendado que se dê um espaçamento maior entre eles ao invés de se utilizar uma linha separadora ou então que se trate desses assuntos em páginas diferentes.

2.2 - Estilos de Texto

Assim como em editores de textos, você pode criar uma série de efeitos no texto, alterando a forma ou tipo de fonte. Todos os comandos que alteram o estilo de texto aparecem em pares que marcam o início e o fim do texto que sofrerá a formatação.

As principais modificações possíveis são:

Negrito - A Tag ` ... ` :

A tag `` é usada para marcar o início de um texto que deve receber o estilo negrito, a tag `` marca o final do mesmo.

Itálico - A Tag `<i> ... </i>` :

A tag `<i>` é usada para marcar o início de um texto que deve receber o estilo itálico, a tag `</i>` marca o final do mesmo.

Sublinhado - A Tag `<u> ... </u>` :

A tag `<u>` é usada para marcar o início de um texto que deve receber o estilo negrito, a tag `</u>` marca o final do mesmo.

Typewriter - A Tag `<tt> ... </tt>` :

A tag `<tt>` é usada para marcar o início de um texto que deve ter espaçamento regular, a tag `</tt>` marca seu final.

Sobrescrito - A Tag `^{...}` :

A tag `^{` é usada para marcar o início de um texto que deve receber sobrescrito, a tag `}` marca o final do mesmo.



Subscrito - A Tag _{...} :

A tag _{é usada para marcar o início de um texto que deve receber subscrito, a tag} marca o final do mesmo.

Pulsante - A Tag <blink> ... </blink> :

A tag <blink> é usada para marcar o início de um texto que deve ficar piscando, a tag </blink> marca o final do mesmo.

Apesar de ser bastante legal, nesse ponto do aprendizado, brincarmos com a tag <blink>, é importante lembrarmos que, por questões de design, seu uso em páginas da Web não é recomendado.

Cabeçalhos - As Tags <h_n> ... </h_n> :

A linguagem HTML possui seis tags, de <h1> até <h6>, que marcam um cabeçalho. A tag <h1> indica um cabeçalho de hierarquia superior ao da tag <h2> e assim por diante.

Essas tags aplicam negrito e modificam o tamanho dos textos que são envolvidos por elas. As tags <h_n> também inserem uma linha em branco antes e outra depois do texto envolvido por elas.

O uso de tags <h_n> é perigoso, pois cada navegador apresenta um tamanho diferente para o texto envolvido. Seu uso só é recomendado em situações onde seja mais necessário a representação hierárquica dos títulos que a visual como no caso de livros ou apostilas on-line.

Texto pré-formatado - A Tag <pre> ... </pre> :

A tag <pre> faz com que o texto que ela envolve preserve sua formatação original como marcas de tabulação, fim de linha gerado pela tecla Enter, etc. Com essa tag, o texto aparece na tela do navegador como está no Bloco de Notas.

A tag <pre> é muito útil para colocarmos músicas cifradas ou códigos de programação em nossas páginas, pois nesses casos seria muito trabalhoso fazermos o alinhamento do texto.

O texto envolvido pela tag <pre>, normalmente é apresentado em outra fonte. Essa fonte muitas vezes é a Courier New.

Centralização - A Tag <center> ... </center> :

Como padrão, os textos, cabeçalhos, figuras e tudo mais são alinhadas pela esquerda. A tag <center> pode ser utilizada para centralizar qualquer desses elementos na página.



Apesar de ser bastante útil, com as implementações de folhas de estilo e JavaScript essa tag está sendo aos poucos substituída pela tag <div> que veremos mais adiante. Por enquanto, use <center> para fazer com que elementos da página fiquem centralizados.

2.3 - MODIFICANDO AS CORES DO TEXTO E DE FUNDO

Na seção anterior falamos sobre as tags responsáveis por aplicar efeitos aos textos. Nessa seção iremos falar das tags que definem o tipo de fonte, seu tamanho e cor e do parâmetro responsável pela cor de fundo.

Alterando a fonte - A Tag ... :

A tag , bem como as vistas anteriormente, deve aparecer sempre em pares. A tag por si só não define nenhuma formatação ao texto, ela necessita de parâmetros para que produza alguma diferença na formatação.

Para que o navegador saiba qual o tipo, tamanho e cor de fonte ele deve apresentar, informamos esses valores dentro da tag em forma de parâmetros. Os principais parâmetros da tag são:

face - Define o tipo de fonte.
size - Define o tamanho da fonte.
color - Define a cor da fonte.

A síntese da tag é:

```
<font face='nome' size='tamanho' color='cor'>
```

Size

O parâmetro size='tamanho' especifica o tamanho da fonte utilizada. Os valores de size podem variar entre 1 (o menor) e 7 (o maior). O tamanho padrão da maioria dos usuários é size=3 e esse é o tamanho com o qual a fonte será mostrada caso esse parâmetro não seja definido.

Também podemos especificar o valor do parâmetro size por valores relativos. Nesse caso, informamos qual valor devemos adicionar ou subtrair do valor padrão. Um exemplo desse tipo de especificação é dado abaixo:

```
<font size='+1'>
```



No caso acima o tamanho da fonte será igual ao anterior adicionado de um, logo, se antes de definirmos `size=+1` o tamanho da fonte era 3, agora será 4.

Nota: O tamanho com que a fonte será visualizada depende da resolução de vídeo do usuário, ou seja, se um usuário utiliza uma definição de 1024 x 768 pixels ele verá uma fonte de tamanho 3 menor que um usuário que estiver 800 x 600 pixels de definição no seu vídeo.

Dessa forma fica realmente impossível saber qual tamanho da letra será visto pelo visitante. Certamente, se o visitante tem uma definição de 1024 x 768 ou maior, seu tamanho padrão de fonte não é 3, pois uma fonte de tamanho 3 fica pequena demais para se ter uma fácil leitura. Do mesmo modo que se o visitante utiliza uma tela de 640 x 480 pixels esse tamanho de fonte é muito grande.

Para evitarmos que as letras de nossa página fiquem ou muito grandes ou muito pequenas, procuramos sempre definir o parâmetro `size` de forma relativa, pois se o usuário tem como padrão um tamanho 4 ou 2, e queremos uma letra um pouco maior que as outras basta definirmos `size='+1'` e não precisamos nos preocupar com o padrão do usuário.

Face

O parâmetro `face` permite que seja escolhida uma fonte diferente para o texto. Pode ser especificada qualquer fonte, sendo que o texto somente aparecerá com aquele tipo de fonte caso o visitante a possua instalado no seu sistema, portanto se for especificado um valor "Comic Sans MS" para o parâmetro `face`, o texto somente será apresentado em Comic Sans MS se essa fonte estiver instalada no computador do usuário.

Dessa forma torna-se impossível garantir que um texto apareça com a fonte que especificamos. Para amenizar esse problema a HTML nos dá a possibilidade de especificarmos várias fontes no parâmetro `face`, de forma que, se o sistema do visitante não possuir a primeira fonte, a segunda será utilizada e se o sistema também não tiver a segunda, a terceira será a escolhida e assim por diante. Caso nenhuma das opções especificadas no parâmetro `face` esteja instalada no computador do visitante, o texto será mostrado na fonte padrão.

A sintaxe do parâmetro `face`, com várias fontes fica a seguinte:

```
<font face='Gill Sans Mt, Helvetica, Arial'>
```

Color

Esse parâmetro permite que seja escolhida uma cor para o texto. Essa cor pode ser especificada em RGB hexadecimal ou através de um nome que corresponde ao nome da cor em inglês.



A questão de cores em HTML por incrível que pareça é uma das mais complicadas, principalmente quando trabalhamos com figuras e pretendemos representar milhões de cores. Certamente, não precisamos de milhões de cores para colorir um texto, por isso deixaremos o problema dos milhões de cores mais para frente.

O sistema RGB consiste na forma de representar cores pela mistura das cores vermelha (Red), verde (Green) e azul (Blue). Nesse sistema, cada uma dessas três cores pode assumir valores de 0 a 255.

A forma RGB hexadecimal consiste no mesmo sistema de mistura de cores, porém representamos os valores de 0 a 255 em hexadecimal, de 00 a FF. Pode realmente parecer muito complicado, mas com o tempo começamos a compreender o sistema.

De início eu te garanto que você vai preferir definir as cores pelos nomes, mas quando uma cor intermediária for a pretendida, será necessário o uso da notação RGB hexadecimal. Abaixo temos uma pequena tabela com o nome de algumas cores.

chsia	Fu	ack	Bl	een	Gr	own	Br
ve	Oli	ver	Sil	hite	W	ue	Bl
an	Cy	ay	Gr	me	Li	vy	Na
d	Re	al	Te	llow	Ye	elBlue	Ste
nk	Pi	rkRed	Da	ange	Or	rkgreen	Da

Agora uma tabela com algumas cores em RGB hexadecimal

#F	#0	#3	#9
FFFFFF	00000	36699	96633
#0	#6	#F	#F
03366	60000	FFFCC	FCC99
#C	#F	#6	#F
CF FCC	F3399	6CC00	F00FF
#0	#0	#3	#6
00033	03300	30000	66666

A tag , com todos os seus parâmetros, fica da seguinte forma:

```
<font face='Gill Sans Mt, Helvetica, Arial' color='orange' size='+2'>
```

Cores de fundo - Os parâmetros da tag <body>



Agora que já vimos o que são e como podem ser utilizados os parâmetros, chegou a hora de saber que a tag `<body>` também tem seus parâmetros. Os parâmetros dessa tag são muitos, porém, nessa seção falaremos somente de 4 deles.

São eles `bgcolor`, `text`, `link`, `alink` e `vlink`.

- `bgcolor` - Define a cor de fundo;
- `text` - Define a cor dos textos da página;
- `link` - Define a cor dos links da página;
- `alink` - Define a cor dos links ativos;
- `vlink` - Define a cor dos links visitados.

Bgcolor

O parâmetro `bgcolor` da tag `<body>` e de tantas outras tags define a cor de fundo. Essa cor é especificada da maneira como foi explicada na seção anterior.

Text

Esse parâmetro define a cor do texto. tendo sido definido o parâmetro `text`, os textos serão apresentados daquela cor a menos que o texto esteja envolvido por uma tag `` na qual tenha sido especificada uma cor diferente.

Link

Apesar de ainda não termos estudados os links, você deve saber que eles têm uma cor diferente do texto e que essa cor geralmente é azul. O parâmetro `link` define exatamente essa cor.

Alink

Define a cor do link ativo. A cor definida nesse parâmetro aparece colorindo o link no momento em que se pressiona o botão do mouse.

Vlink

Esse parâmetro define a cor dos links que já foram visitados, o padrão é roxo.

Veremos agora como ficaria uma tag `<body>` que define preto como cor de fundo, verde para o texto, um amarelo esverdeado para link, laranja para links ativos e um vermelho claro para links visitados.



```
<body bgcolor='black' text='#00ff00' link='#ccff00' alink='orange'  
vlink='#ff6666'>
```

2.4 - Exercícios



1 Crie uma página igual à mostrada na figura mostrada abaixo.

Tente fazer sua página o mais parecido possível com esta imagem.



3 – LISTAS

Assim como cabeçalhos, que separam partes do texto, as listas são uma outra forma de estruturar um documento HTML. As listas devem ser usadas em situações onde temos que disponibilizar algum tipo de informação e que é essencial vincularmos essa informação a uma hierarquia, em caso contrário esqueça o uso de listas, sejam elas ordenadas ou não.

3.1 - Listas Ordenadas

Listas ordenadas são aquelas que numeram os itens da lista. Abaixo temos um exemplo desse tipo de lista.

1. Listas ordenadas são definidas pela tag `` . ``;
2. Cada item da lista é definido pela tag ``;
3. Você pode trocar os marcadores de uma lista ordenada;
4. Esses marcadores podem ser 1, 2, 3,...; A,B,C,...; a,b,c,...; I, II, III,... ou ainda i, ii, iii,...;
5. Tudo o que estiver entre `` `` recebe um recuo.

Para trocarmos os marcadores de um lista ordenada, utilizamos o parâmetro `type` que, para listas ordenadas pode receber os valores 1, A, a, I ou i. Podemos também definir qual será o primeiro número, ou letra, de nossa lista com o parâmetro `start` que deve ser igual ao primeiro número ou letra de nossa lista.

Abaixo temos o código de uma lista que serve de exemplo. Nesse caso temos uma lista ordenada em números romanos minúsculos começando em 'iv'.

```
<ol type='i' start='iv'>
```

```
<li> Primeiro item da lista, indicado pelo número iv;  
<li> Segundo item, indicador v;  
<li> Terceiro item, o indicador é vi;  
<li> mais um item e assim por diante...
```

```
</ol>
```

Faça um arquivo HTML que contenha o código acima e verifique o resultado.

3.2 - Listas Não Ordenadas

Listas não ordenadas não ordenam seus itens, apenas adicionam bullets que são marcadores que não tem nenhuma função dentro do texto e que não serão de maneira alguma utilizados por bons designers.



- Listas não ordenadas são definidas pela tag ` .. `;
- Cada item da lista é definido pela tag ``;
- Você pode trocar os marcadores de uma lista ordenada;
- Esses marcadores podem ser discos, círculos ou quadrados;
- Tudo o que estiver entre ` ` recebe um recuo.

Para trocarmos os marcadores de uma lista não ordenada, utilizamos o parâmetro `type` que, para listas não ordenadas pode receber os valores `disc`, `circle` ou `square`.

O uso de marcadores de listas não ordenadas pode ser substituído por um simples recuo no texto e isso já é suficiente. O usuário não precisa de bolinhas que giram para avisar a ele que aquilo se trata de uma lista. Uma boa página deve ter, no lugar de bullets, espaçamentos horizontais e verticais entre os itens de forma a deixar claro a divisão em itens.

Dica: Utilize as tags ` ... ` para produzir um recuo maior em um texto. Tags `<dd>` quando concatenadas produzem um só efeito enquanto que listas, ordenadas ou não, quando aninhadas produzem efeito acumulativo. Uma boa prática de HTML para se produzir o efeito de um recuo igual a dois `<dd>` é a sequência `<dd> texto `.

3.3 - Listas Alinhadas

É possível aninharmos várias listas, ou seja, colocarmos uma dentro da outra para criarmos estruturas hierárquicas. Como já foi dito anteriormente, o uso de listas só é recomendado quando a hierarquia do documento é algo fundamental e nem nessas situações é recomendado o uso de marcadores.

O "alinhamento" de listas ordenadas com listas não ordenadas também é possível. Abaixo temos dois exemplos, o primeiro demonstra duas listas aninhadas com a utilização da tag `` que coloca marcadores no início de cada item, já no segundo exemplo a tag `` foi omitida.

- Introdução
 - Abrindo um arquivo local no navegador
 - Acessando um site na internet
- Formatação
 - Parágrafos e quebras de linha
 - Estilos de textos

Agora o mesmo exemplo sem os marcadores.

Introdução

Abrindo um arquivo local no navegadorAcessando um site na internet

Formatação

Parágrafos e quebras de linhaEstilos de textos



A lista do segundo exemplo mantém a hierarquia, porém não tem marcadores. Uma lista sem marcadores é mais limpa e mais fácil de ser compreendida. Para que possamos construir uma lista sem os marcadores, basta eliminarmos as tags e colocarmos um
 no fim de cada item da lista.

3.4 – Exercícios

1. Modifique o código abaixo de forma a manter a estrutura hierárquica da página e eliminando todos os tipos de marcadores. Ache soluções para os problemas que irão surgir com a retirada dos marcadores.

```
<html>

<head>
  <title>
    Exercício 1 do capítulo 3
  </title>
</head>

<body bgcolor='black' text='white'>

<font face='verdana'>
<dd>
  Modifique essa página de forma a manter a
  hierarquia das listas aqui contidas,
  eliminando todos os marcadores.
<dd>
  A retirada dos marcadores deve provocar o
  colapso de alguns itens da lista.
  Descubra uma solução para esse
  problema.
<p>
<ul type='disc'>
<li> Introdução - Nesse capítulo
  foi feita uma revisão de como usar
  o navegador além de introduzirmos o
  aluno a linguagem HTML 4.0.
<li> Formatação - Aqui o aluno
  aprendeu como aplicar negrito, itálico
  e outros efeitos no texto como, por
  exemplo, trocar a cor da fonte e do
  fundo.
```



```
<li> Listas - Esse é o capítulo que  
    está sendo estudado nesse momento,  
    tentamos mostrar as listas ao aluno  
    mas o advertimos quanto ao seu uso  
    com os marcadores.  
<li> Links e Âncoras - Esse é o  
    próximo capítulo. Nele falaremos dos  
    links que são atalhos ou ligações entre  
    páginas ou entre pontos dentro da mesma  
    página e das âncoras que são os pontos  
    para onde um link da mesma página aponta.  
</ul>  
</font>  
  
</body>  
  
</html>
```

4 - LINKS E ÂNCORAS

Links são, com toda certeza, um dos elementos mais importantes de uma página da Web, pois com eles você pode criar documentos que façam referência a endereços do computador ou da Web.

O uso de links também facilita em muito a criação de documentos extensos, cujas seções e tópicos podem ser rapidamente acessadas por meio de links.

Vamos começar nossos estudos procurando compreender como funciona a identificação ou endereçamento de documentos na Internet.

4.1 - Url (Uniform Resource Locator)

A Internet usa uma nomenclatura específica para indicar o endereço de um documento. Essa nomenclatura chamada de URL inclui três componentes que são:

Protocolo do documento (http, gopher ou WAIS);

O endereço do servidor e

O nome do arquivo

Esses componentes são organizados da seguinte forma:

protocolo://servidor/NomeDoArquivo

No caso da nossa apostila, temos:

<http://www.labbas.eng.uerj.br/cursos/hp/index.html>



4.2 - Links Locais ou Relativos

Links locais são aqueles que fazem ligações entre documentos que se encontram no mesmo computador ou servidor. A tag `<a> ... ` é a responsável pelos links e pelas âncoras. No caso da construção de links locais a sintaxe básica é a seguinte:

```
<a href='Nome_do_Arquivo'>Arquivo</a>
```

Além do parâmetro `href`, existem outros como `target`, `tabindex` e `shape` que também podem ser especificados. Por enquanto vamos nos restringir aos parâmetros `href`, e `tabindex`.

O Parâmetro `href`, como você já deve ter percebido é responsável por indicar o endereço do arquivo "linkado". Já o `tabindex`, define a posição do link na ordem de tabulação.

Temos abaixo um exemplo de um link local com esses dois parâmetros definidos.

```
<a href='../exemplos/links.html' tabindex='2'>Links  
Interessantes</a>
```

Nota: Links locais ou externos não precisam apontar para um arquivo.html, qualquer link pode apontar para qualquer arquivo. No caso de se construir um link que aponte para arquivos do tipo JPEG ou GIF, as imagens serão abertas pelo navegador. Caso o usuário possua o plug-in para a visualização de ShockWave, links para arquivos .swf farão com que o navegador abra a animação.

Caso o link faça referência a um arquivo de um tipo não reconhecido pelo navegador, será feito o download do arquivo. links para .zip funcionam muito bem para essa finalidade.

4.3 - Links Absolutos

Os links absolutos fazem referência ao endereço completo do arquivo na Web, incluindo protocolo, nome do servidor e nome do arquivo. Links absolutos são bastantes úteis quando se pretende ligar dois documentos que não se encontram no mesmo servidor e, por isso, precisam ser referenciados pelos seus endereços completos.

O uso de links absolutos não é aconselhável para ligar dois documentos da mesma home page. Para isso utilizamos links locais que também podem ser chamados de links relativos.

Também não é aconselhável que se utilize links absolutos para fazer a ligação com uma página interna de outro site, pois, o dono desse outro site pode resolver mudar o nome de seu arquivo e seu link ficará inútil.



A grande aplicação dos links absolutos é fazer ligações com as páginas principais de outros sites. Abaixo temos um exemplo de link absoluto.

```
<a href='http://www.labbas.eng.uerj.br'>Página do LabBas</a>
```

O código acima gera um que link liga uma página a página do LabBas.

Dica: Prefira:

"Quem Sou Eu
Minhas Fotos".

Ao invés de:

"Para saber quem é o autor, Clique Aqui
Para ver minhas fotos, Clique Aqui"

4.4 – Âncoras

É possível interligarmos partes de uma mesma página, para isso precisamos criar âncoras as quais serão referidas pelos links. Uma âncora é criada quando definimos o parâmetro name da tag <a>.

Uma âncora é um ponto de referência ou endereço que será acessado por um link e é usada dentro de um documento para determinar o início de uma seção. O parâmetro name define o nome dessa âncora e esse nome será mencionado pelo link que a acessará.

A sintaxe da tag <a> quando usada como âncora é:

```
<a name='Nome_da_Âncora'>Texto opcional</a>
```

A sintaxe do link que acessa uma âncora é:

```
<a href='#Nome_da_Âncora'>link</a>
```

É importante observar que para fazermos referência a uma âncora é necessário colocarmos # antes do nome da âncora. Caso contrário o navegador entenderá que estamos fazendo referência a um arquivo.

4.5 – Exercícios

1. Crie uma página que contenha os seguintes links:

LabBas Lee Faculdade de Engenharia UERJ.

Procure fazer com que a página fique agradável, não coloque simplesmente links. Dê um título, faça um resumo explicando o que será encontrado em cada link, coloque tudo o que achar que vai contribuir para o visual da página.

2. Crie uma página que fale de vários assuntos de seu interesse. É importante que todos os assuntos sejam colocados dentro de um único documento. Crie, no topo da página e onde mais achar necessário um menu com links para âncoras de forma a facilitar a navegação pela sua página.



Não coloque linhas separadoras para separar os assuntos, utilize espaços verticais de forma que só seja possível a visualização de um assunto por tela. Evite também o uso de marcadores (bullets) no menu.

5 – IMAGENS

Nesse ponto já somos capazes de fazer bastante coisa com textos, porém você certamente está ansioso para saber como colocar imagens na sua página. E isso é exatamente o que aprenderemos nesse capítulo.

Páginas HTML suportam imagens nos padrões JPEG e GIF. Já existem plug-ins para os navegadores que os tornam capazes de ler imagens no formato PNG, mas pelo fato de poucos usuários possuírem esse plug-in não daremos ênfase a esse tipo de imagem.

5.1 - A Tag e seus Parâmetros

A tag é a responsável pela inserção de imagens dentro de uma página HTML. Seus parâmetros mais importantes são:

src	border
alt	hspace
width	vspace
height	ismap
align	

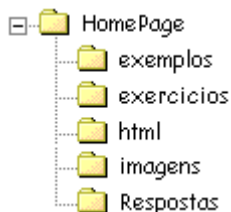
Por enquanto não estudaremos o parâmetro ismap, pois ele é utilizado para a construção de mapas de imagem, assunto tratado posteriormente.

O parâmetro SRC:

Esse parâmetro especifica o endereço da figura. As regras para endereçamento de figuras são as mesmas explicadas na seção 4.1, ou seja, esse parâmetro pode ser especificado como o endereço absoluto ou relativo da imagem.

Observe a seguinte estrutura de diretórios.

Para inserirmos uma imagem de nome "tochan.gif" que está no diretório imagens





Em um documento HTML que esta no diretório html, colocaremos o seguinte código:

```
<img src='../imagens/tochan.gif'>
```

Dica: É recomendado que sempre se especifique o endereço de uma imagem de forma relativa.

O parâmetro ALT:

O parâmetro alt especifica qual texto será apresentado enquanto a imagem não carrega, alguns navegadores também exibem esse texto quando se passa o mouse sobre a imagem.

Os Parâmetros WIDTH e HEIGHT:

Esses parâmetros definem, respectivamente, a largura e a altura em pixels da imagem. É possível redimensionar uma imagem com a alteração desses parâmetros, porém é recomendado que você redimensione a imagem no Photoshop ou em outro software gráfico.

Uma imagem, normalmente demora a carregar e, isso faz com que os textos apareçam primeiro na página. Caso os parâmetros width e height não tenham sido especificados, o navegador não irá saber o tamanho da imagem até que ela seja carregada e por isso irá colocar os textos em lugares errados e, quando sua imagem carregar irá deslocar todos os textos para os locais corretos.

Para evitar que os textos de sua página fiquem "andando" na página é fortemente recomendado que sempre seja especificado os valores de width e height.

A especificação de uma imagem com os quatro parâmetros estudados até agora:

```
<img src='Nome_da_Imagem' alt='Texto opcional' width='Largura  
em pixels' height='Altura em pixels'>
```

Dica: Para descobrir a largura e a altura de uma imagem utilize o Photoshop ou outro software gráfico. O ACDSee, por exemplo, informa a largura, a altura e a profundidade de cores ao abrir uma imagem.

Caso esteja usando o Photoshop, abra a imagem e selecione o menu "imagem" (image) e depois "tamanho da imagem" (image size).

O parâmetro ALIGN:

Esse parâmetro define o alinhamento de uma imagem. O parâmetro align pode receber valores left, right, top e middle.

No caso de definirmos align como left ou right, a imagem será apresentada em um dos lados da tela, enquanto o texto flui ao seu lado.



Essa imagem tem align=left. Isso faz com que a imagem seja apresentada a esquerda e o texto possa fluir ao lado da imagem. Para isso é fundamental que a tag anteceda o texto.



Essa imagem tem align=right. Isso faz com que a imagem seja apresentada a direita e o texto possa fluir ao lado da imagem. também é preciso que a tag anteceda o texto.

Se definirmos align como top ou middle, a imagem aparecerá como um "caractere" na linha do texto, ou seja, na posição em que for definida.

Nesse caso, a imagem tem align='top' pelo topo.



e isso causa um alinhamento

Nesse caso, a imagem tem align='middle' alinhamento centralizado



e isso causa um

No último caso, não definimos align alinhada por baixo.



isso faz com que a imagem seja

O parâmetro BORDER:

O parâmetro border define a largura da borda de uma imagem. O Netscape somente apresenta bordas nas imagens caso essas sejam links (veremos imagens como links mais à frente) já o Explorer apresenta as bordas de toda forma.



Por questões de design, é recomendado que sempre se defina `border='0'`. A única situação em que é aconselhável definirmos `border='1'`, é durante o desenvolvimento do site, para sabermos quais são os limites de nossa imagem.

Os parâmetros HSPACE e VSPACE:

Esses parâmetros definem quanto de espaçamento horizontal e vertical deve ser dado nos lados da imagem.

5.2 - Imagens de Fundo

Podemos utilizar imagens para compor o fundo de uma página, para isso basta definirmos o parâmetro `background` da tag `<body>` como sendo o endereço da imagem que desejamos colocar no fundo.

Quando definirmos o fundo de uma página como sendo uma determinada imagem, essa imagem será repetida várias vezes, tanto para o lado quanto para baixo, até que ocupe toda a tela.

Não utilize imagens que tenham uma grande variação de cores como fundo de página, pois isso causa dificuldade na leitura dos textos, prefira imagens com pouco contraste.

5.3 - Imagens como Links

Podemos definir uma imagem como sendo um link, para isso basta colocarmos a tag `` entre as tags `<a>`.

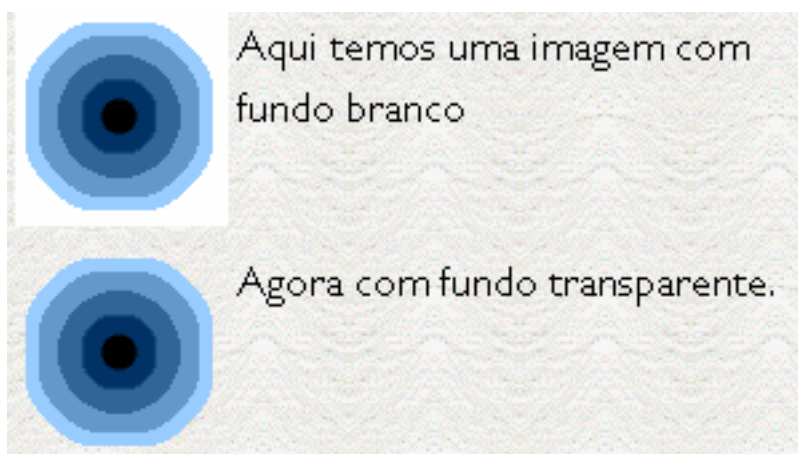
Quando definimos uma imagem como link, o navegador automaticamente coloca bordas na imagem, por isso, devemos sempre nos lembrar de colocar `border='0'` nessa situação.

5.4 - Imagens Transparentes e Entrelaçadas

Existem duas características das imagens gif que são muito importantes para a Web. Uma delas é a transparência e a outra o entrelaçamento.

A transparência dos GIF permite-nos construir imagens com o fundo transparente, de forma que, ao carregada a cor de fundo da imagem desaparece, permitindo a visualização do fundo da página.

Veja a diferença entre uma imagem transparente e outra não transparente abaixo.



Uma imagem é lida de cima para baixo, isso faz com que imagens grandes levem bastante tempo para aparecerem por completo na tela dos navegadores. Normalmente aparece a parte de cima da imagem enquanto o restante vai aparecendo aos poucos.

Em um gif entrelaçado, os dados são gravados de forma que ao ser carregada, a imagem apareça por completo em baixa resolução e vá aumentando sua resolução conforme é carregada.

5.5 - Criando Imagens Transparentes e Entrelaçadas

Para criar imagens transparentes e entrelaçadas podemos utilizar vários softwares gráficos entre eles o Photoshop e o PaintShop Pro 4. Vamos utilizar o Photoshop 5 como base de nossos estudos.

Para que o Photoshop seja capaz de exportar GIFs é necessário que se tenha o plug-in Gif98-a que é o responsável pela exportação gif.

Não temos como objetivo dessa apostila o ensino de Photoshop, por isso, vamos considerar que você já tenha obtido a imagem de alguma outra fonte.

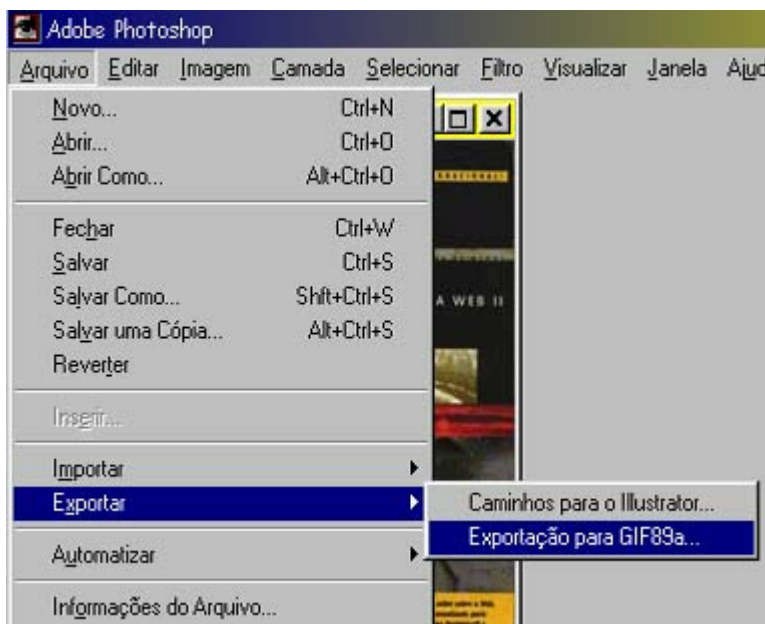
Abra a imagem no Photoshop, selecione a opção "imagem"(image) do menu principal e depois selecione "modo"(mode), verifique se esta marcada a opção "cores indexadas"(indexed colors).

Caso não seja esse o modo de trabalho atual, selecione-o como tal. A figura abaixo demonstra essa operação.

Achate as camadas e escolha o número de cores. É importante notar que quanto maior o número de cores, maior o tamanho do arquivo e pior a qualidade da imagem. Devemos procurar o número ideal de cores de forma a conseguirmos uma imagem de boa qualidade e um arquivo pequeno.



Selecione "Arquivo"(file),no menu principal, selecione então "Exportar"(Export) e depois "Exportação para GIF89a.."(GIF89a Export). A figura abaixo ilustra essa operação.



Caixa de opções para exportação GIF89a deve aparecer, marque entrelaçar para criar um gif entrelaçado e selecione as cores que deseja que sejam apresentadas como transparente. feito isso, clique em OK, a caixa salvar, padrão do windows deve aparecer, escolha o nome do arquivo, confirme e pronto.





5.6 - A Compressão Gif

O formato GIF usa um algoritmo de compressão denominado Lempel-Ziv-Welch ou LZW. Esse modelo de compressão é sem perdas, o que significa que a imagem comprimida tem a mesma aparência do original.

Em geral, é vantagem usarmos a compressão GIF para tudo aquilo que não for fotográfico ou fortemente sombreado. Fotos pequenas, de até 100 x 100 pixels também podem ser comprimidas com sucesso pelo GIF.

O modelo LZW utiliza técnicas de reconhecimento de padrões para conseguir a compressão. A essência do algoritmo é substituir extensões horizontais de um pixel da mesma cor por um número que indica a extensão da sequência.

Imagine uma imagem 90 x 90 pixels, 256 cores, onde a metade superior é um certo azul e a inferior é um certo amarelo. Se fossemos analisar seu tamanho sem compressão, teríamos $90 \times 90 \times 8 = 64800$ bits que resultariam em 7,91 Kbytes. Com a compressão GIF dessa imagem temos um arquivo de 1 Kbyte.

Isso acontece porque o arquivo sem compressão teria a definição da cor de cada pixel, ocupando 8 bits para a definição de cada cor, como a imagem tem 8100 (90×90) pixels teríamos um arquivo de 8100 bytes (7,91 Kbytes). Já com a compressão temos a definição do primeiro pixel e uma instrução para ele se repetir por 90 colunas e 45 linhas.

Então temos a definição do segundo pixel com outra instrução para que ele se repita por 90 colunas e 45 linhas.

Em ambos os casos, temos cada pixel definido por 256 cores. Isso demonstra que quanto maior a extensão de uma cor, menor fica nosso arquivo gif. Agora vamos reduzir as cores dos GIF.

Os pixels são agora definidos por duas cores e com isso temos arquivos de 187 bytes para cada figura.

Como podemos ver, se soubermos trabalhar com os gif teremos imagens com a mesma qualidade e infinitamente menores. Nesse caso, uma imagem de 24534 bytes foi comprimida em 187 bytes sem perda de informação.

Temos que nos atentar então a dois fatores quando da construção de imagens GIF. A quantidade de cores e a extensão de segmentos da mesma cor.

5.7 - A Compressão Jpeg

A compressão JPEG é a melhor maneira de se comprimir imagens fotográficas. Esse tipo de compressão não funciona bem para imagens pequenas ou sem grandes sombreamentos.

Em geral, esse tipo de compressão grava uma boa cópia da versão preto e branco da imagem, à qual nossos olhos são sensíveis e comprime grande parte das sutis diferenças de cores que não podemos distinguir.



O computador remonta a imagem após re-criar as cores a partir de uma notação abreviada gravada no arquivo. o JPEG não trabalha linha por linha como o GIF e sim em zonas de cores.

A compressão JPEG é um processo com perdas, mesmo que se comprima uma imagem usando as configurações de melhor qualidade, a imagem não será idêntica à versão original, isso se dá pelo fato do JPEG gravar informações de apenas alguns pixels, descartando os outros e interpolando para recompor a imagem.

Em geral, imagens nítidas geram JPEG maiores, pois é necessário que haja um menor espaçamento entre os pixels que serão gravados no JPEG. Imagens desfocadas geram JPEG menores, pois podemos aumentar o espaçamento e com isso diminuir o número de pixels gravados no JPEG

5.8 - Exercícios

1. Construa uma página de entrada para seu site pessoal.

Uma boa página de entrada deve carregar rapidamente, e deve criar uma expectativa no visitante.

Procure colocar imagens que estimulem a entrada do visitante no seu site. Uma das imagens de sua página de entrada deve ser um link para a página de núcleo de seu site. Deve ficar claro para o visitante que ele deve clicar ali para entrar, isso não quer dizer que você deva mandar ele clicar em algum lugar para entrar.

Abaixo temos alguns exemplos de algumas páginas de entrada.

