

eBook

HTML 4.01 – Curso Básico

Fábio Rodrigo Brandão

<HTML>

Versão do Documento: 1.0

As informações contidas neste documento estão sujeitas à alteração sem aviso prévio. Nenhuma parte deste documento pode ser reproduzida ou transmitida por qualquer meio, eletrônico ou mecânico, para qualquer propósito, sem permissão expressa, por escrito do autor.

Várias marcas registradas aparecem ao longo deste documento. Mais do que simplesmente listá-las e informar quem possui seus direitos de exploração ou então utilizar os símbolos “TM” ou “®” após seus nomes, o autor declara estar utilizando tais nomes apenas para fins editoriais, em benefício exclusivo do dono da marca registrada, sem intenção de infringir as regras e as leis de sua utilização.

Escreva para o autor: fabio@brainfonet.com

Elaborada por Fábio Rodrigo Brandão

Visite meu website: fabiobrandao.net.br

Requisitos de Hardware e Software

Antes de começar a estudar de forma prática o nosso material, verifique se seu equipamento atende aos requisitos mínimos relacionados:

Hardware

- Processador Pentium® 100Mhz;
- Memória RAM: 24 MB, recomendado 32 MB;
- Espaço livre em disco: 30 MB;
- Monitor SVGA.

Software

- Windows 95, 98, ME ou 2000;
- Internet Explorer 4.0 ou superior;
- Matrix Y2K ou Bloco de Notas;

Pré-requisitos

Para que você possa compreender bem todas as informações contidas neste material, é fundamental que você tenha feito um treinamento ou lido nosso material sobre Windows, Internet.

SUMÁRIO

Capítulo I — Introdução ao HTML.....	5
1.0 - Introdução.....	5
1.1 - Características Gerais do HTML	5
1.2 – Projetando	7
1.2.1 – Storyboarding.....	8
1.2.2 – Ferramentas e Linguagens	8
1.2.2 – Criando e Salvando seus documentos.....	8
1.3 - Escrevendo tags HTML	9
1.3 - Tags Básicas	10
1.3.1 – HTML	11
1.3.2 – HEAD	11
1.3.3 – TITLE	11
1.3.4 – BODY	11
1.4 – Inserindo comentários	12
1.4 - Tags de Formatação	13
1.4.1 - Criando Cabeçalhos	13
1.4.2 – Criando Parágrafos	13
1.4.3 - Quebras de linha.....	13
1.4.4 – Réguas Horizontais.....	14
1.5 – Listas.....	14
1.5.1 - Listas Não Ordenadas	14
1.5.2 - Listas Ordenadas.....	14
1.5.3 - Listas de Definição.....	15
1.6 - Texto Pré-Formatado	15
1.7 – Formatação de Estilo da Fonte e Informação.....	16
1.8 - FONT.....	16
1.9 - Caracteres Especiais.....	17
1.10 - Algumas Entidades Comuns	17
1.11 - O que é CSS.....	18
Capítulo II – Tabelas	19
2.0 - Definindo uma Tabela	19
2.0.1 - Criando Cabeçalho	20
2.3 – Criando Linhas e Colunas	20
2.3.1 - Personalizando as Linhas.....	21
2.3.1 - Personalizando as Células de Dados e Índices	21
Capítulo III - Interligando Documentos.....	23
3.1 - Criando Links Internos.....	23
3.2 - Criando Links Externos	24
4.0 - Elementos HTML para multimídia	26
4.1 - Arquivos MID.....	26
4.2 - Adicionando sons	26
4.3 - O que é Flash.....	27
Capítulo V - Trabalhando com Imagens	28
5.0 - GIF e JPEG	28
5.1 - Inserindo uma imagem.....	28
Capítulo VI - Formulários	30

6.1 - O que são os Formulários	30
6.1 - Definindo um Formulário	30
6.1.1 - Campo de Texto	32
6.1.2 - Campo Senha	33
6.1.3 - Entrada oculta	34
6.1.4 - Entrada de várias linhas de texto	34
6.1.5 - Menus com opções	35
6.1.6 - Botões sim ou não	37
6.1.7 - Botões de Radio	38
6.1.8 - Botões de submissão e limpeza	39
Capítulo VII – Trabalhando com Frames	41
7.0 - O que são Frames	41
7.1 - Criando o Arquivo de Frames Principal	41
7.2 - Links com Frames	43
7.3 - Atributos de Frames	43

Capítulo I — Introdução ao HTML

1.0 - Introdução

O objetivo deste material é ensinar-lhe, de forma simples e agradável, escrever códigos HTML para a criação de conteúdo para web. Contudo vale comentar que trabalhar com códigos não é nada confortável se comparado aos programas gráficos. Porém, um conhecedor dos códigos HTML terá total controle sobre o conteúdo desenvolvido, mesmo que ele seja criado utilizando-se uma outra ferramenta. Então mãos à obra.

HTML (**Hyper Text Mark-up Language**), é um conjunto de diretivas e métodos para controlar a apresentação e disposição gráfica do conteúdo de um documento. Ele se tornou a linguagem padrão para a criação de documentos publicados na web. Este documento pode ser composto por textos, imagens e outros objetos como sons. O resultado é o que chamamos de texto formatado ou documento formatado. Apesar da palavra linguagem de programação, gerar documentos HTML é muito similar ao trabalho que fazemos quando estamos usando nosso editor de textos preferido.

Os documentos HTML podem ser gerados a partir de qualquer editor de textos, como o Bloco de Notas e o Microsoft Word, desde que gravados com a extensão “.htm” ou “.html”. Isto é possível, devido ao formato da linguagem, o bom e velho “.txt” suportado praticamente por todos os editores de textos. Os documentos HTML também poderão ser gerados por editores HTML como o FrontPage, Dreamweaver e outros.

Os documentos HTML são a base para a maioria dos documentos disponíveis na web e também é suportada por outros aplicativos.

Atualmente a HTML está em sua quarta geração. Cada versão da HTML é um superconjunto de versões anteriores. A cada nova versão, novos recursos e funcionalidades são adicionados, retirando-se alguns dos recursos desnecessários nas versões anteriores.

As definições para HTML e também para várias outras tecnologias são determinadas pelo órgão W3C (www.w3.org). O atual diretor da W3C é Tim Berners-Lee que também é o inventor da World Wide Web.

1.1 - Características Gerais do HTML

Como foi mencionado, as instruções da linguagem HTML, também chamadas de marcações ou **TAGS** poderão ser escritas em qualquer editor de textos. Entretanto existem editores específicos para se trabalhar com HTML como o **FrontPage** da Microsoft e o **DreamWeaver** da Macromedia.

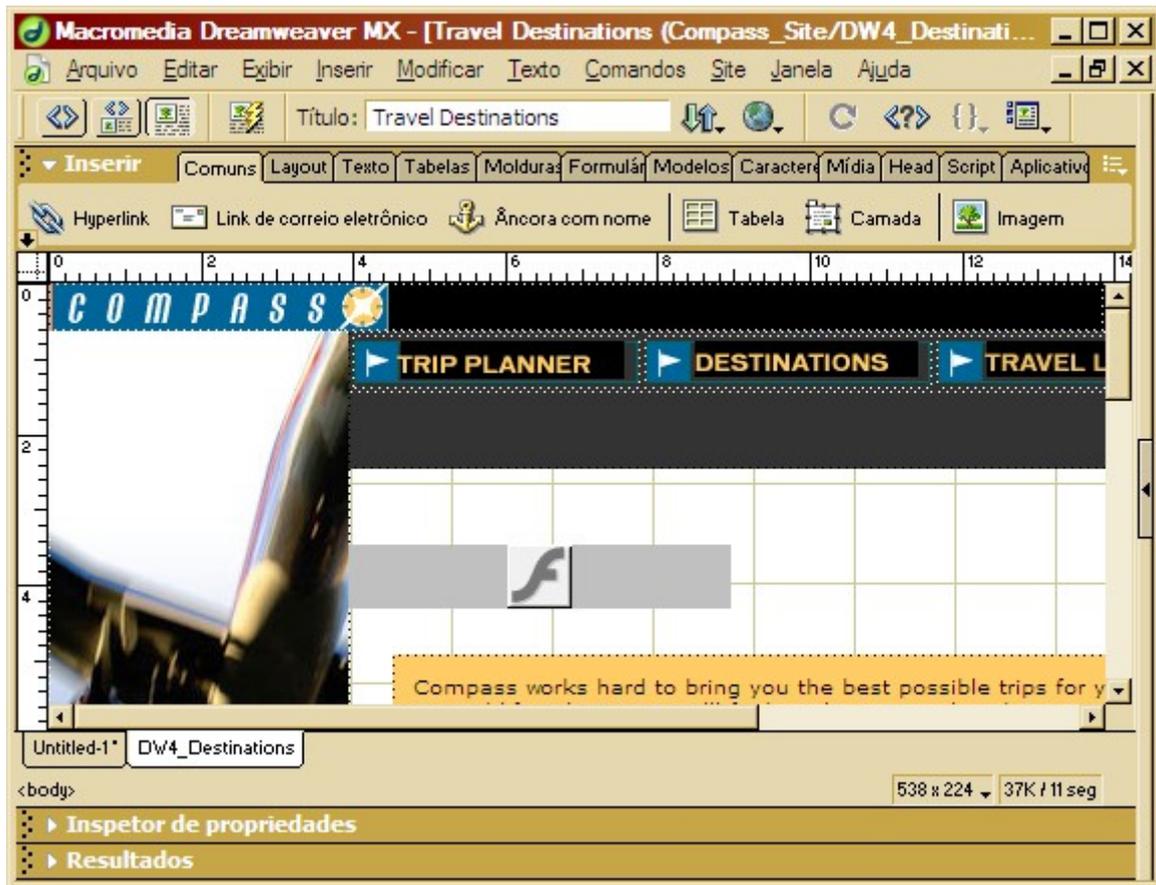


Figura 1 – Janela do DreamWeaver MX

Em nosso primeiro contato com a linguagem utilizaremos o Bloco de Notas para a edição das tags. Caso você não tenha o Bloco de Notas em seu sistema, poderá utilizar qualquer editor de textos.



Figura 2 – Janela do Bloco de Notas do Windows

O resultado de todo o trabalho deverá ser visualizado em um browser (programa navegador, capaz de recuperar páginas da Web para seu computador). Existem vários browsers disponíveis no mercado como o **Mosaic**, **Opera**, **Internet Explorer**, **Firefox**, **Opera** e **Netscape**. Os browsers também podem ser chamados de Clientes Web.

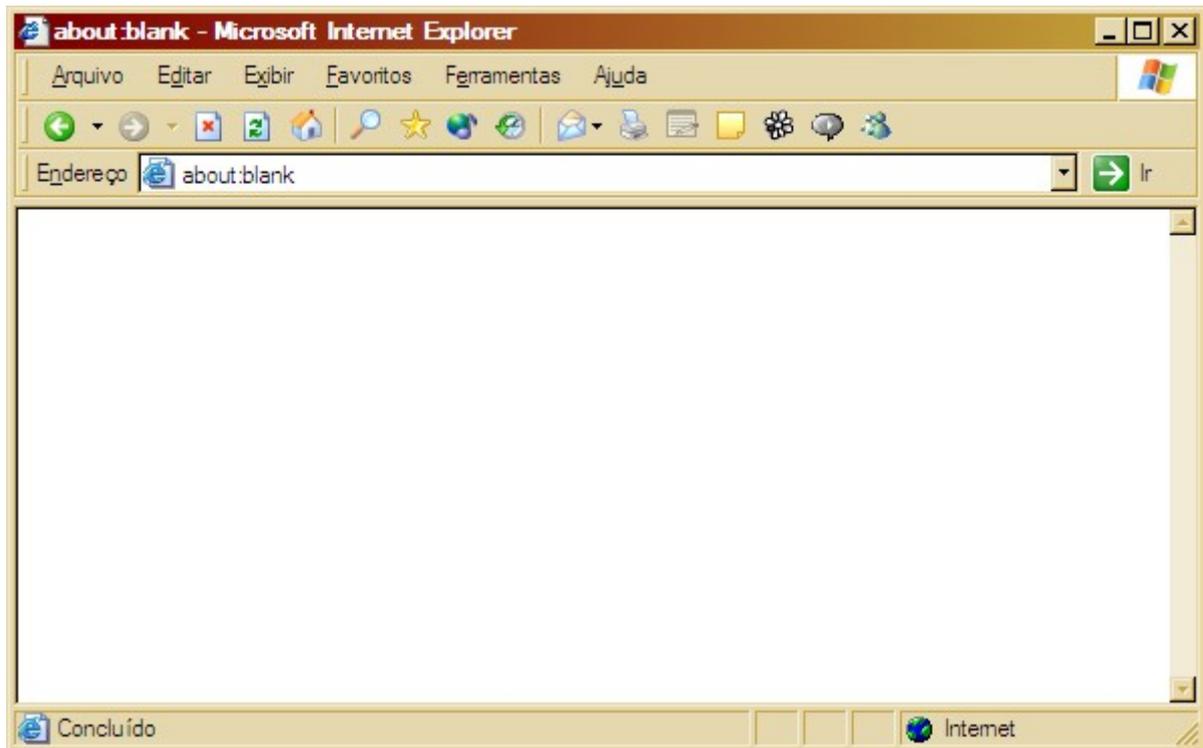


Figura 3 – Janela do Internet Explorer 6

Há diferenças entre os diversos clientes Web, de forma que nem todas as marcações e seus correspondentes recursos são suportados por todos eles. Quando um cliente não entende uma determinada marcação, ele simplesmente a ignora. Ao criar um documento, é importante testá-lo com vários clientes web para ver se o resultado esperado é o mesmo para navegadores diferentes. Use, sempre que possível, recursos que possam funcionar na maioria dos navegadores atuais.

**Dica:**

Utilize sempre uma versão anterior a versão mais recente dos navegadores para testar suas páginas. Lembre-se que os usuários demoram um pouco para atualizar seus navegadores.

A HTML utiliza marcações específicas e distintas para dizer ao browser como exibir o documento. Estas marcações poderão ser escritas em letras minúsculas ou maiúsculas. Então a notação **<title>** é equivalente a **<TITLE>** ou **<TiTIE>**.

1.2 – Projetando

Antes mesmo de começar a digitar suas tags HTML é necessário alguns conceitos básicos sobre como projetar seus documentos. Vamos a eles:

1.2.1 – Storyboarding

Reúna-se com a sua equipe de desenvolvimento, coloquem no papel todas as idéias sobre o material a ser desenvolvido. Não se esqueça de abordar os seguintes tópicos:

- Objetivo
- Público Alvo

É importante manter seu objetivo bem alinhado e desenvolver o conteúdo pensando sempre o público que irá acessá-lo. Lembre-se que o conteúdo é para os visitantes e não para você.

Primeiramente decida quais informações você quer transmitir aos seus leitores e depois projete quais os caminhos que gostaria que eles percorressem. Esta primeira etapa pode demorar um pouco, mas você ganhará tempo no final.



Dica:

Caso o conteúdo do seu projeto já exista na web, visite e veja qual foi o modelo de estrutura utilizado no projeto.

Faça um esboço de suas páginas e lembre-se que a primeira impressão é a que fica. Use as seguintes regras básicas na hora de fazer o rascunho de suas páginas:

- **Conteúdo** - O conteúdo deve ser direto. Deixe claro que a página é de um indivíduo ou de uma empresa e qual o seu objetivo.
- **Clareza** – Apresente suas informações com clareza. Não coloque texto demais nas páginas que você deseja chamar a atenção. Quando houve muito texto coloque um link para as pessoas optarem pela leitura dele. Use espaços em branco e separadores à vontade.
- **Cor** – Tenha bom senso em escolher as cores para suas páginas. Não use azul para textos com fontes pequenas. Sabe-se que o olho humano tem uma menor sensibilidade ao azul.

1.2.2 – Ferramentas e Linguagens

Após um esboço de todo o projeto, determine junto com a sua equipe quais as tecnologias e linguagens serão utilizadas no desenvolvimento. Veja se cada tecnologia sugerida é realmente necessária. Não se esqueça de levar em consideração o meio onde os documentos serão visualizados. Sabemos que outros dispositivos, além do microcomputador estão sendo utilizados para acessar a web como celulares e pdas. Uma boa dica seria desenvolver um projeto que poderia ser facilmente portabilizado para as necessidades do cliente caso ele mude de idéia.

1.2.2 – Criando e Salvando seus documentos

Agora é mão na massa. Determinem quantos e quais documentos serão criados para o projeto. Este material abordará somente a linguagem HTML e o que poderá ser feito com ela, mas outras linguagens e tecnologias poderão fazer parte do projeto, exigindo uma estruturação mais aprimorada e detalhada. É importante ressaltar que este material adota uma visão simplificada sobre o projeto e o

desenvolvimento. Crie seus documentos HTML usando o Bloco de Notas ou seu editor preferido, e ao salvá-los, utilize uma extensão padrão para estes documentos que poderá ser “.htm” ou “.html”.

Outra informação importante, é que o primeiro documento de um conjunto de documentos HTML deverá se chamar **INDEX.HTM** ou **INDEX.HTML**. Devemos utilizar estes nomes, pois assim estará configurado no Servidor Web (computador que hospeda e serve documentos para web). Caso você não vá hospedar seus documentos em um Servidor Web o nome não terá importância.

Para facilitar a organização dos documentos e demais arquivos que irá compor seus documentos como imagens, vídeos e sons, você poderá também, dividi-los em estrutura de pastas.

1.3 - Escrevendo tags HTML

As marcações (tags) do HTML devem ser colocadas entre os sinais de (<), "menor que" e (>), "maior que".

Veja um exemplo:

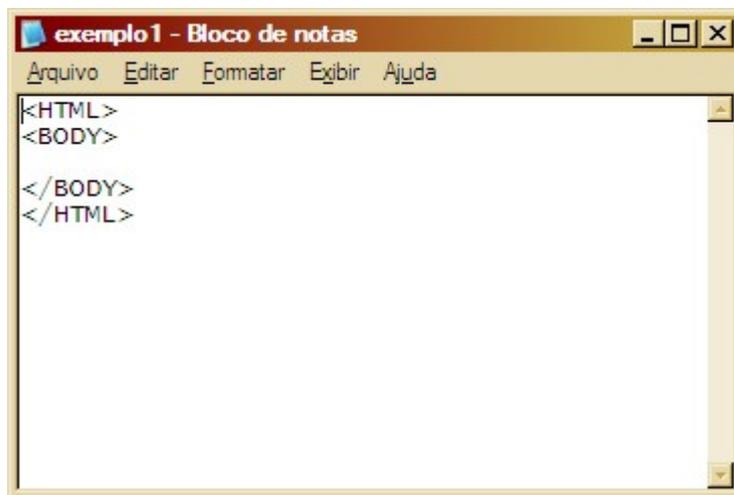


Figura 4 – Exemplo 1 (EXEMPLO1.HTML)

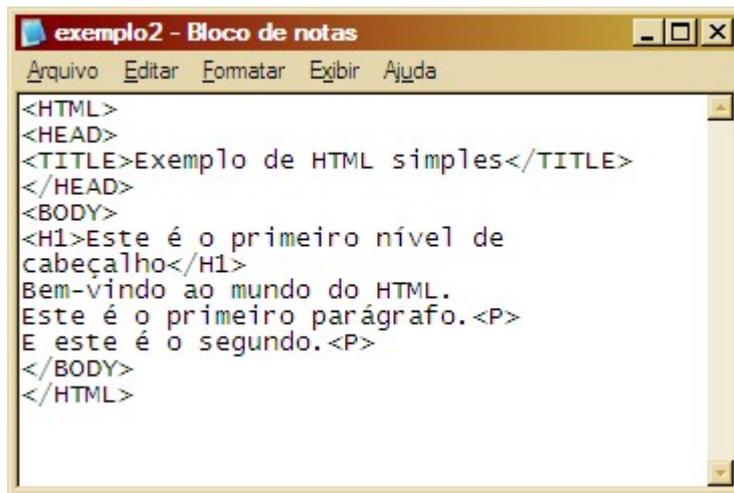
Se você estiver utilizando o material em CD-Rom, poderá visualizar os exemplos que estão na pasta exemplo.

De um modo geral, as tags aparecem em pares, por exemplo, <H1> Cabeçalho</H1>. O símbolo que termina uma determinada marcação é igual àquele que a inicia, antecedido por uma barra “/” e precedido pelo texto referente. No exemplo 1, veja que existem duas tags abertas e uma tag de fechamento para cada uma delas indicando que a tag foi encerrada.

Há exceções a esse funcionamento em pares das marcações. Por exemplo, a que indica um final de parágrafo: <P>. Não necessita de uma correspondente: </P>. A marcação que indica quebra de linha -
 - também não precisa de uma correspondente, e outras tais como <HR> e . Falaremos mais adiante sobre estas tags.

1.3 - Tags Básicas

Todo documento deve ser identificado como HTML com as tags `<HTML>` e `</HTML>`, ter uma área de cabeçalho `<HEAD>` e `</HEAD>` com o título do documento `<TITLE>` e `</TITLE>`, e uma área definida como corpo `<BODY>` e `</BODY>` onde será colocado o conteúdo do documento. Veja a figura 5 onde cada tag deve ser colocada:



```
<HTML>
<HEAD>
<TITLE>Exemplo de HTML simples</TITLE>
</HEAD>
<BODY>
<H1>Este é o primeiro nível de
cabeçalho</H1>
Bem-vindo ao mundo do HTML.
Este é o primeiro parágrafo.<P>
E este é o segundo.<P>
</BODY>
</HTML>
```

Figura 5 – Exemplo 2 (EXEMPLO2.HTML)

Todo documento em HTML deve possuir um título. De um modo geral o título aparece em lugar separado da página no alto da tela no Navegador, e é utilizado para identificar o documento. É interessante que o título possa sugerir claramente o conteúdo do documento. Veja como ficaria o resultado:

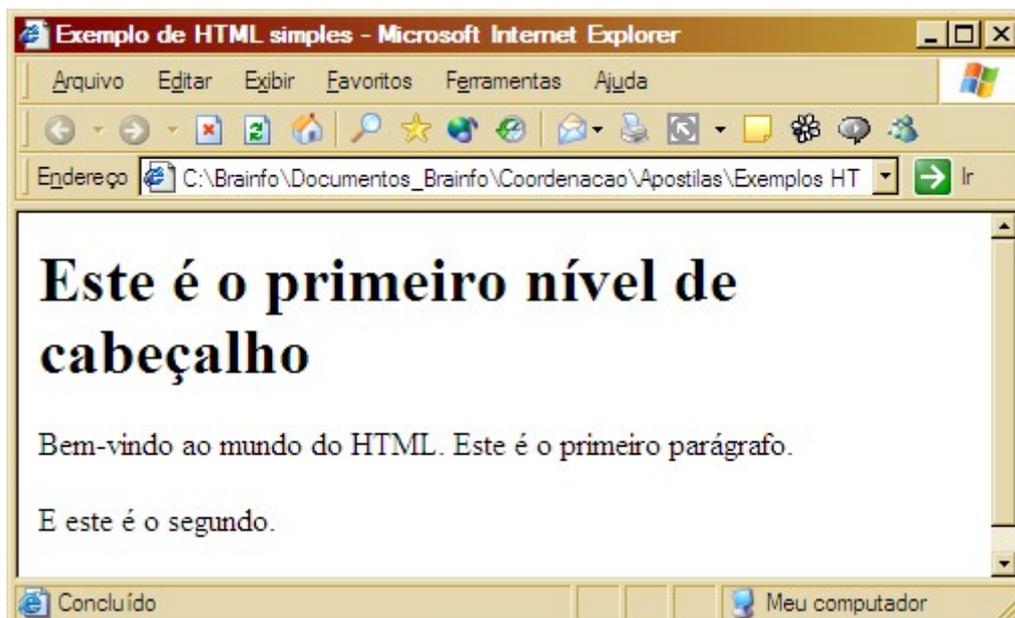


Figura 6 – Resultado do Código HTML

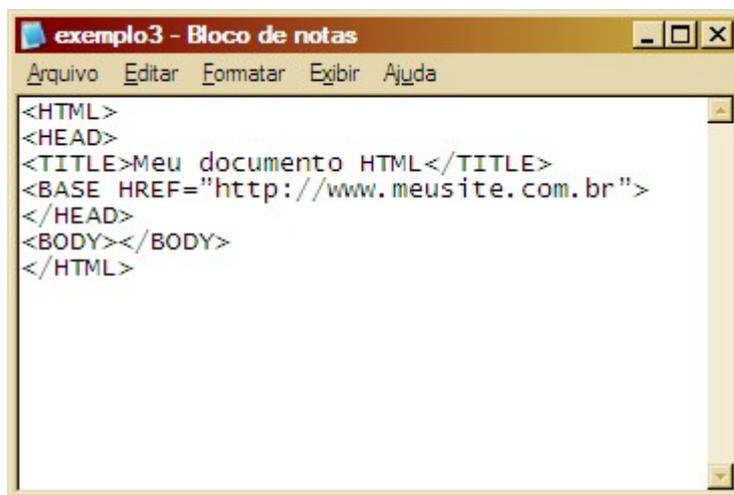
1.3.1 – HTML

Esta tag indica o início e o fim do documento HTML. Necessita de tag de fechamento.

1.3.2 – HEAD

Esta tag delimita a seção de cabeçalho do documento HTML. Na seção de cabeçalho, também é possível determinar algumas opções para o documento que não aparecem na exibição como:

BASE – Especifica o endereço base do documento que é utilizado para determinar o endereço completo dos (URLs) relativos que aparecem no documento.

A screenshot of a Notepad window titled "exemplo3 - Bloco de notas". The window has a menu bar with "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". The text area contains the following HTML code:

```
<HTML>
<HEAD>
<TITLE>Meu documento HTML</TITLE>
<BASE HREF="http://www.meusite.com.br">
</HEAD>
<BODY></BODY>
</HTML>
```

Figura 7 – Exemplo3 (EXEMPLO3.HTML)

1.3.3 – TITLE

Esta tag identifica o título do documento. O título pode ser um texto qualquer e deve ser colocado entre a tag de abertura e fechamento. Esta tag deve ser colocada entre as tags de cabeçalho. Se esta informação for omitida, será utilizado o nome do arquivo como título do documento.

1.3.4 – BODY

Identifica o corpo do documento. Esta tag deve estar posicionada após a tag </HEAD> que limita o cabeçalho. Entre as tags <BODY> e </BODY> será colocado todo o conteúdo a ser apresentado. A tag <BODY> possui alguns parâmetros:

BGCOLOR – Configura a cor de fundo da página, podendo ser o nome da cor em inglês ou um valor em hexadecimal como é o padrão.

Exemplo: <BODY BGCOLOR="White"> ou <BODY BGCOLOR="#FFFFFF">

TEXT – Configura a cor padrão do texto da página.

Exemplo: <BODY TEXT="Black"> ou <BODY TEXT="#000000">

BACKGROUND – Configura uma imagem GIF ou JPG usada para preencher o fundo da tela.

Exemplo: `<BODY BACKGROUND="//c:/imagens/fundo.gif">`

Tome cuidado ao usar este tipo de recurso. Imagens muito pesadas podem tornar o documento muito lento ao carregar. Veja o capítulo que fala sobre como inserir imagens.

BGPROPERTIES – Configura o plano de fundo como um painel fixo. Seu único valor é `fixed`. Esta tag é uma particularidade do Internet Explorer.

Exemplo: `<BODY BACKGROUND="//c:/imagens/fundo.gif" BGPROPERTIES="Fixed">`

LINK, ALINK E VLINK – Configura os valores de cores para os estados dos links de um documento.

LINK – Cor padrão do link.

ALINK – Cor do link quando ativo.

VLINK – Cor do link quando já visitado.

Exemplo: `<BODY LINK="#000080" ALINK="000000" VLINK="#FFFFFF">`

LEFTMARGIN E RIGHTMARGIN – Configura a margem esquerda e direita respectivamente do documento em pixels. Seu valor deve ser um número inteiro.

Exemplo: `<BODY LEFTMARGIN="5" RIGHTMARGIN="5">`

1.4 – Inserindo comentários

São úteis para documentação e são ignorados pelo browser. Os comentários devem ser escritos da seguinte forma:

```
<!-- Texto de comentário -->
```

Exemplo:

```
<HTML>
<HEAD>
<TITLE>Meu documento HTML</TITLE>
<BASE HREF="http://www.meusite.com.br">
</HEAD>
<!-- Aqui começa o corpo do documento -->
<BODY></BODY>
<!-- Aqui termina o corpo do documento -->
</HTML>
```



Dica:

Use comentários para indicar determinadas seções de seu documento facilitando a sua localização no momento da manutenção dos arquivos.

1.4 - Tags de Formatação

1.4.1 - Criando Cabeçalhos

Cabeçalhos normalmente são usados para títulos e sub-títulos de uma página. Existem seis níveis de cabeçalhos, numerados de 1 a 6, sendo o número 1 o de maior destaque. Cabeçalhos são exibidos em letras maiores e em negrito e podem aparecer de formas diferentes em cada navegador. O primeiro cabeçalho em cada documento deve estar marcado como <H1>.



Importante:

Ao definir o tamanho de um cabeçalho, você não está definindo o tamanho da letra (fonte 10, fonte 14). Você apenas define que ele aparecerá com maior tamanho e destaque que o resto do texto.

As tags relativas a cabeçalhos são:

<H y >Texto do cabeçalho </H y >, onde y é um número entre 1 e 6 especificando o nível do cabeçalho.

Exemplo: <H1>Este é o nível 1</H1>

Veja seu parâmetro:

ALIGN – Determina a posição horizontal do texto, em relação às margens. Pode ser:

- Left – Alinhado com a margem esquerda.
- Center – Centralizado.
- Right – Alinhado com a margem direita.
- Justify – Alinhado a ambas as margens.

Exemplo: <H1 ALIGN="center">Este é o nível 1</H1>

1.4.2 – Criando Parágrafos

A marcação <P> é utilizada para definir o início de um novo parágrafo, deixando uma linha em branco entre cada parágrafo. A HTML não reconhece o caractere de quebra de linha dos editores de texto. Mesmo que exista uma linha em branco, os clientes Web só reconhecem o início de um novo parágrafo mediante a marcação apropriada. Esta tag também possui o parâmetro de alinhamento com os mesmos valores.

Exemplo: <P ALIGN="center">Este parágrafo está centralizado</P>

1.4.3 - Quebras de linha

A marcação
 faz uma quebra de linha sem acrescentar espaço extra entre as linhas.

Exemplo: Primeira linha
 Esta linha ficará abaixo da linha anterior.

Note que esta tag não possui comando de fechamento.

1.4.4 – Réguas Horizontais

São barras horizontais 3D da largura da janela do browser, utilizadas para separar visualmente as seções de um documento. É limitado pela tag <HR>.

Além do parâmetro de alinhamento, esta tag possui outros parâmetros em particular:

NOSHADE – Não usar sombreamento de efeito 3D.

SIZE – Espessura, em pixels, da barra.

COLOR – Determinar a cor com que a barra deve ser apresentada. Particularidade do Internet Explorer V 3.0 e posteriores.

WIDTH – Configura a largura da linha. Seu valor pode ser em pixels ou porcentagem da janela.

Exemplo: <HR NOSHADE SIZE="3" COLOR="#000000" WIDTH="50%">

1.5 – Listas

A HTML possui três categorias de listas, com opções diferentes para cada uma delas:

- Listas não ordenadas
- Listas ordenadas
- Listas de definição

1.5.1 - Listas Não Ordenadas

São listas com marcadores representada pela maioria dos browsers pela marca bullet, um ponto grosso ao lado dos itens. A tag e delimitam a lista, quanto as tags e delimitam cada item da lista.

Exemplo:

```
<UL>
<LI>Chocolate</LI>
<LI>Bombom</LI>
<LI>Pirulito</LI>
</UL>
```

1.5.2 - Listas Ordenadas

São listas onde se utiliza um caractere para representar a ordem dos itens que poderá ser através de letras ou números. Estas listas são delimitadas pela tag que poderá ter o seguinte parâmetro:

TYPE – Determina o índice de sequência (1 – Numérico, a – Alfabético com letras minúsculas, A – Alfabético com letras maiúsculas, i – Algarismos romanos com letras minúsculas, I – Algarismos romanos com letras maiúsculas).

START – Estabelece o valor inicial da seqüência.

Exemplo:

```
<OL START="3" TYPE="A">
<LI>Chocolate</LI>
<LI>Bombom</LI>
<LI>Pirulito</LI>
</OL>
```

1.5.3 - Listas de Definição

Utilizada quando se deseja explicar um determinado termo. Os browsers normalmente mostram a definição em linha diferente do termo a ser definido. O termo a ser definido é delimitado pelas tags **<DT>** e **</DT>**, enquanto sua definição fica entre as tags **<DD>** e **</DD>**. A lista deverá ser colocada entre as tags **<DL>** e **</DL>**.

Exemplo:

```
<DL>
<DT>HTML</DT>
<DD>Linguagem de Marcação de texto.</DD>
</DL>
```

1.6 - Texto Pré-Formatado

Este elemento é usado para apresentar texto sem formatação, respeitando a preexistente. Neste caso serão apresentadas as mudanças de linhas e os espaçamentos.

Exemplo:

```
<PRE>
Código Qbasic
    'Calculo Simples
    INPUT A
    INPUT B
    C = A + B
    PRINT C
END
</PRE>
```

1.7 – Formatação de Estilo da Fonte e Informação

Veja na tabela abaixo, alguns elementos usados para a formatação do estilo da fonte e tipo de informação:

Marcação	Descrição
...	Coloca o texto em negrito
<I>...</I>	Coloca o texto em itálico
<U>..</U>	Coloca o texto sublinhado
<S>...</S>	Coloca o texto riscado
<BIG>...</BIG>	Aumenta o texto em 1 ponto
<SMALL>...</SMALL>	Diminui o texto em 1 ponto
_{...}	Coloca o texto subscrito
<SUB>...</SUP>	Coloca o texto superscrito
<CITE>...</CITE>	Apresentação de citações
<CODE>...</CODE>	Apresentação de código fonte
...	Texto enfatizado
<KBD>...</KBD>	Texto a ser digitado
<SAMP>...</SAMP>	Texto de amostra
...	Texto reforçado
<VAR>...</VAR>	Argumentos em código fonte

Exemplo:

```
<B><I>Este texto está em negrito e itálico</B></I>
```

1.8 - FONT

Este comando é utilizado para alterar as características das fontes dos caracteres. Possui as seguintes opções:

FACE – Determina o tipo de fonte utilizada pelos caracteres na ordem de preferência. Não esqueça de usar vírgula para separar cada tipo e de utilizar preferencialmente fontes padrão.

```
<FONT FACE="Arial, Verdana">Aqui será exibido em Arial, caso o usuário não a tenha em seu PC será apresentado com Verdana.</FONT>
```

SIZE – Determina o tamanho utilizado pelos caracteres através de um número inteiro entre 1 e 7 ou um número relativo ao tamanho corrente.

Exemplo:

```
<FONT SIZE="3">Este texto usa o tamanho 3</FONT>
<FONT SIZE="-1">Já este texto usa o tamanho 2</FONT>
```

COLOR – Cor a ser aplicada aos caracteres.

Exemplo:

```
<FONT SIZE="3" COLOR="#000000">Este texto usa o tamanho 3 e cor preto.</FONT>
```

1.9 - Caracteres Especiais

Alguns caracteres possuem significados especiais em HTML e para browsers Web. Esses caracteres incluem os símbolos de maior que (>) e menor que (<), que você usa na hora de digitar suas tags HTML. O mesmo ocorre com aspas (“”), e o símbolo E comercial (&). Estes símbolos são chamados de entidades. As entidades possuem o formato “&xxxx” onde xxxx é uma string de caracteres que identifica a entidade em particular.

Um exemplo de entidade muito usada é o espaço sem quebra que é representado da seguinte forma:

** **

Observe que as notações especiais sempre se iniciam por & (e comercial) e encerram-se com ponto e vírgula (;).

Quando esta entidade é adicionada ao documento HTML, os browsers que lerem sabem que devem exibir um caractere de espaço na tela, mas que o caractere não é qualificado como espaço em branco, de modo que os browsers não quebraram a linha neste ponto.

1.10 - Algumas Entidades Comuns

Entidade	Caractere	Descrição
<	<	Caractere menor que
>	>	Caractere maior que
&	&	“E” comercial
"	“	Aspas
þ	þ	“Thorn” islandês
Þ	Þ	Thorn” maiúsculo
©	©	Símbolo de copyright

®	®	Símbolo de marca registrada
ñ	ñ	n minúsculo com til

1.11 - O que é CSS

Embora este material não utilize esta linguagem, ela pode lhe ser muito útil para a formatação de seus documentos. A CSS (Cascading Style Sheets) permite que você realize tarefas que não são possíveis apenas com HTML valorizando a apresentação gráfica de seus documentos. Com a utilização de CSS podemos por exemplo separar as marcações de formatação dos textos dos documentos.

Capítulo II – Tabelas

2.0 - Definindo uma Tabela

Sem dúvida este é um dos recursos mais interessantes desta linguagem. Com o uso de tabelas poderemos organizar melhor as informações de nossas páginas. É bem simples a criação de uma tabela porém deve-se ter muita atenção ao criá-las. Uma tabela HTML fica delimitada pelas tags **<TABLE>** e **</TABLE>**. Esta tag possui os seguintes parâmetros:

ALIGN – Determina o alinhamento da tabela em relação as margens laterais podendo ser left, center ou right.

BGCOLOR – Determina a cor de fundo dos elementos da tabela.

BORDER – Espessura, em pixels, do fio do enquadramento. Como padrão possui o valor 0.

CELLPADDING – Determina o posicionamento vertical e horizontal do conteúdo da célula e seus limites. Seu valor é determinado em pixels ou em percentual.

CELLSPACING – Determina o posicionamento horizontal e vertical das células em relação aos limites da tabela e entre si. Seu valor é determinado em pixels ou em percentual.

FRAME – Determina os lados do quadro que envolve a tabela, que serão apresentados. Pode ter os seguintes valores:

- **Void** – Nenhum. Este é o valor presumido.
- **Above** – Apenas a parte superior.
- **Below** – Apenas o lado inferior.
- **Hsides** – Os lados horizontais.
- **Vsides** – Os lados verticais.
- **Lhs** – O lado esquerdo.
- **Rhs** – O lado direito.
- **Box** – Todos os quatro lados.

WIDTH – Largura, em pixels ou porcentagem, com que será exibida a tabela.

BACKGROUND – Imagem para ser usada como plano de fundo.

BORDERCOLOR – Determina a cor do fio em torno da tabela.

BORDERCOLORDARK – Permite dar um efeito 3D ao fio em torno da tabela aplicando uma cor diferente na parte inferior e direito do fio.

HEIGHT – Determina a altura padrão da tabela. Seu valor pode ser em pixels ou em percentual.

Exemplo:

```
<TABLE WIDTH="765" ALIGN="LEFT" BACKGROUND="#FFFFFF" BORDER="2" CELLPADDING="0" CELLSPACING="0">  
</TABLE>
```

2.0.1 - Criando Cabeçalho

Para definir uma legenda ou título para a tabela, utilizamos a tag **<CAPTION>** e **</CAPTION>** introduzindo o texto entre as duas tags. Por exemplo:

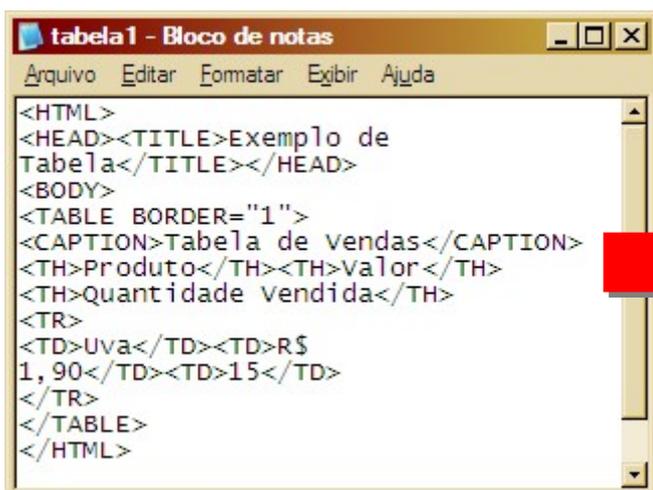
```
<TABLE>
<CAPTION>Tabela de Vendas</CAPTION>
</TABLE>
```

Para criar as células de título para cada coluna, também chamadas de células de cabeçalho, utilize as tags **<TH>** e **</TH>**. Veja o exemplo:

```
<TABLE>
<CAPTION>Tabela de Vendas</CAPTION>
<TH>Produto</TH><TH>Valor</TH><TH>Quantidade Vendida</TH>
</TABLE>
```

2.3 – Criando Linhas e Colunas

É claro que sua tabela utilizará linhas e colunas. Para a criação de linhas utilizamos as marcações **<TR>** e **</TR>**, enquanto para as células de dados utilizamos **<TD>** e **</TD>**. Veja um exemplo:



```
tabela1 - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
<HTML>
<HEAD><TITLE>Exemplo de
Tabela</TITLE></HEAD>
<BODY>
<TABLE BORDER="1">
<CAPTION>Tabela de vendas</CAPTION>
<TH>Produto</TH><TH>Valor</TH>
<TH>Quantidade vendida</TH>
<TR>
<TD>Uva</TD><TD>RS
1,90</TD><TD>15</TD>
</TR>
</TABLE>
</HTML>
```

Figura 8 – Tabela (TABELA1.HTML)

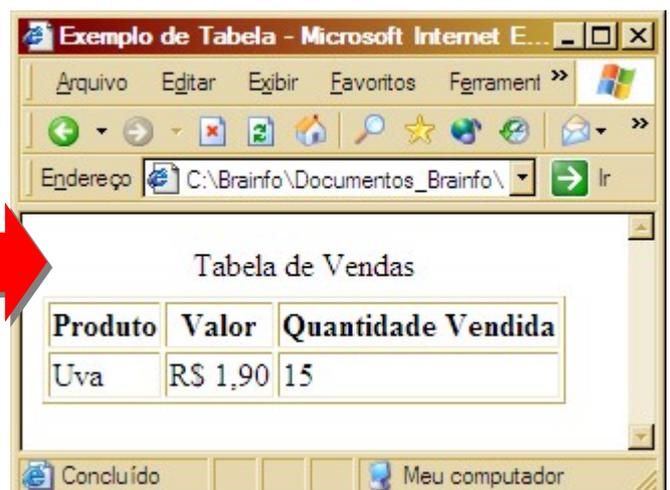
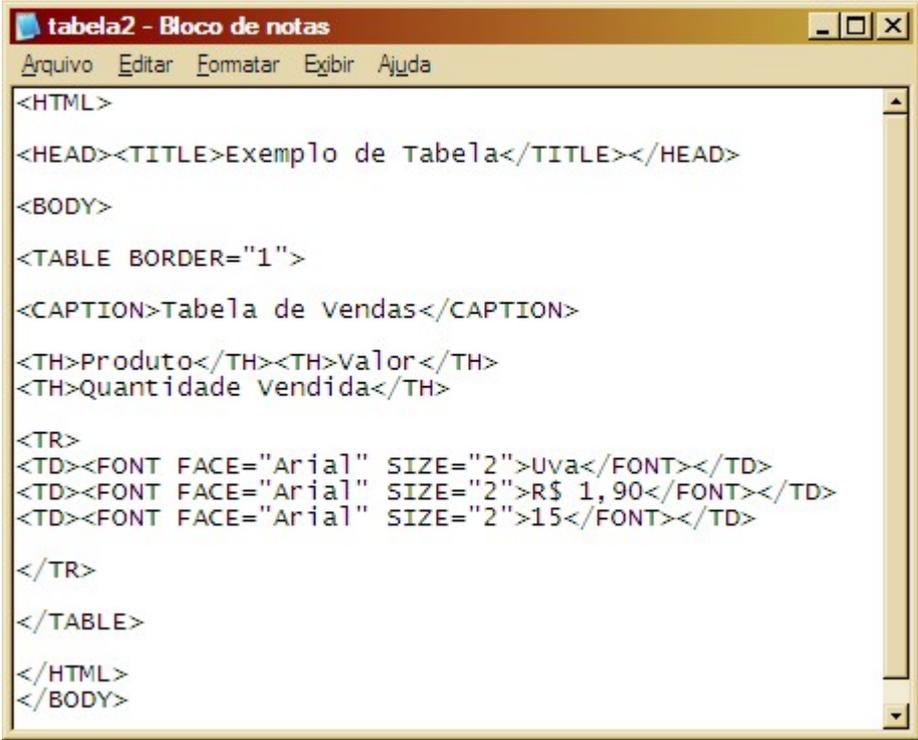


Figura 9 – Resultado da Tabela

O texto poderá ser formatado dentro de uma tabela colocando-se as tags de formatação dentro das células de dados. Imagens e outros objetos também podem fazer parte de uma tabela.

Veja um exemplo com o texto formatado:



```

<HTML>
<HEAD><TITLE>Exemplo de Tabela</TITLE></HEAD>
<BODY>
<TABLE BORDER="1">
<CAPTION>Tabela de Vendas</CAPTION>
<TH>Produto</TH><TH>Valor</TH>
<TH>Quantidade Vendida</TH>
<TR>
<TD><FONT FACE="Arial" SIZE="2">Uva</FONT></TD>
<TD><FONT FACE="Arial" SIZE="2">R$ 1,90</FONT></TD>
<TD><FONT FACE="Arial" SIZE="2">15</FONT></TD>
</TR>
</TABLE>
</HTML>
</BODY>

```

Figura 10 – Tabela com Formatação (TABELA2.HTML)

Observe que no nosso exemplo anterior, vários espaços foram dados entre as tags. Isto facilita a manutenção das mesmas porém não é obrigatório seu uso.

As tags **<TH>**, **<TR>** e **<TD>** permitem receber ajustes finos através de alguns parâmetros.

2.3.1 - Personalizando as Linhas

Utilize os seguintes parâmetros para personalizar suas linhas:

ALIGN – Alinha o conteúdo dentro da célula podendo ser *left* (esquerda), *center* (centro), *right* (direita) ou *justify* (justificado).

VALIGN – Determina o alinhamento vertical do conteúdo das células, podendo ser *top* (no alto), *middle* (no meio), *bottom* (embaixo) e *baseline* (linha base).

BGCOLOR – Determina a cor de preenchimento das células.

BORDERCOLOR – Determina a cor do fio em torno das células desta linha.

2.3.1 - Personalizando as Células de Dados e Índices

Utilize os seguintes parâmetros para personalizar suas células de dados e índices:

ALIGN – Alinha o conteúdo dentro da célula podendo ser *left* (esquerda), *center* (centro), *right* (direita) ou *justify* (justificado).

VALIGN – Determina o alinhamento vertical do conteúdo das células, podendo ser *top* (no alto), *middle* (no meio), *bottom* (embaixo) e *baseline* (linha base).

WIDTH – Determina, em pixels ou em percentual, a largura da célula de dados.

HEIGHT – Determina a altura da célula em pixels. A altura da linha é determinada pelo maior valor das alturas determinadas naquela linha.

BGCOLOR – Determina a cor de preenchimento das células.

COLSPAN – Número de colunas abrangidas pela célula. Utilizando-se o valor “0”, determina-se a utilização de todo o espaço da coluna.

ROWSPAN – Número de linhas abrangidas pela célula.

Capítulo III - Interligando Documentos

O principal poder do HTML vem da sua capacidade de interligar partes de um texto (e também imagens) a outros documentos. Os clientes exibem em destaque estas áreas ou pontos chaves (normalmente com cores diferentes ou sublinhado) para indicar que se trata de um link, ou interligação, no hipertexto.

A marcação **<A>**, que define o ponto de partida para os links, é denominada de **âncora**. Existem tipos diferentes de links:

- **Links Internos** – Quando o link aponta para uma outra seção dentro do mesmo documento.
- **Links Externos** – Quando o link direciona para outro documento podendo ser uma figura, outra página HTML ou em outro formato e até mesmo um arquivo para download.

3.1 - Criando Links Internos

Este tipo de link permite que você interligue diferentes seções de seus documentos. Para este tipo de link você precisa de um ponto de partida e um ponto de chegada. Suponhamos que você tenha no início de um documento HTML um pequeno índice. Ao clicar em um dos itens do índice o usuário seria levado para a seção correspondente dentro do mesmo documento. Neste caso o texto definido como link no índice será o ponto de partida, enquanto que a seção correspondente seria o ponto de chegada. Veja o exemplo:



Figura 11 – Links Internos (LINK1.HTML)

No exemplo, utilizamos a marcação **A** com o parâmetro **HREF=""** indicando o endereço do ponto de chegada e definimos o ponto de chegada usando o parâmetro **NAME**. Tanto em **HREF** quanto em

NAME deve-se colocar o mesmo nome, porém no **HREF**, antes do nome, deve-se utilizar o símbolo "#".

Entre as Tags **<A>** e **** coloque o texto que será utilizado como link.

3.2 - Criando Links Externos

Este tipo de ligação permite interligar documentos externos, como outras páginas **HTML** ou arquivos para download.



Você sabia?

Download é o processo de transferência de arquivos de um computador da rede para sua máquina. Upload é o processo de transferência de arquivos de seu computador para outro na rede.

Especifique o documento a ser interligado colocando como valor do parâmetro **HREF** o endereço correspondente ao documento desejado. Veja um exemplo:

```

link2 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
<HTML>
<HEAD><TITLE>Exemplo de Link Interno</TITLE></HEAD>
<BODY>

<A HREF="index.html">volte a página principal</A>

</BODY>
</HTML>
  
```

Figura 12 – Link Externo (LINK2.HTML)

Também podemos especificar **URLs** (Uniform Resource Locators) para especificar a localização de arquivos em servidores. Uma **URL** inclui o tipo de recurso acessado (p.e, gopher, WAIS), o endereço do servidor, e a localização do arquivo.

Escreve-se de acordo com o esquema abaixo:

Protocolo://servidor[:port]/jpath/filename

Onde o protocolo é uma das seguintes definições:

File	Um arquivo no seu sistema local, ou servidor de ftp
Http	Um arquivo em um servidor World Wide Web
Gopher	Um arquivo em um servidor Gopher
WAIS	Um arquivo em um servidor WAIS
News	Um Usenet newsgroup
Telnet	Uma conexão Telnet

Mailto	Um endereço de correio eletrônico
---------------	-----------------------------------

Veja um exemplo:



```
link2 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
<HTML>
<HEAD><TITLE>Exemplo de Link Interno</TITLE></HEAD>
<BODY>
<A HREF="http://www.brainfo.net.com">Visite a página da Brainfo</A>
</BODY>
</HTML>
```

Figura 13 – Link com URL (LINK3.HTML)

Os tipos de documentos não suportados pelo navegador deverão ser transferidos para seu computador para posterior visualização no processo de download.

Capítulo IV - Trabalhando com Objetos Multimídia

4.0 - Elementos HTML para multimídia

Podemos dizer que estamos utilizando recursos multimídia ao empregarmos 3 tipos diferentes de mídia em nossos documentos: texto, som e imagem.

O elemento básico usado para incluir elementos multimídia é o **<EMBED>**. Este elemento foi introduzido pela Netscape e depois adotado também parcialmente pela Microsoft. Como alternativa a este elemento, a Microsoft criou o elemento **<OBJECT>** que tem a mesma finalidade utilizando a tecnologia ActiveX, ao invés a tecnologia de plug-ins. Como o comando **<EMBED>** é interpretado pelos dois browsers, iremos nos focar nele. Os parâmetros básicos deste elemento são:

- **SRC** – Indica o nome (endereço) do arquivo a ser exibido.
- **HEIGHT** – Indica a altura da área a ser ocupada pelo objeto.
- **WIDTH** – Determina a largura de apresentação do objeto.



Você sabia?

Plug-ins são programas que adicionam novos recursos aos programas já existentes em nosso computador.

4.1 - Arquivos MID

Midi é um formato bastante compacto de música e por isso é transferido mais rapidamente para o computador dos visitantes de sua página. Os browsers atuais já têm plug-ins para arquivos MID, de forma que você não terá problemas em se fazer “ouvir”. Você poderá obter mais detalhes técnicos sobre MID em nosso curso de Multimídia.

4.2 - Adicionando sons

Abaixo você verá exemplos de como inserir sons dentro de seu documento HTML usando as duas tags citadas anteriormente:

Para browsers Netscape:

```
<EMBED SRC="nome_som.mid">
```

Para browsers Internet Explorer:

```
<BGSOUND SRC=" nome_som.mid">
```

Repare que no exemplo para navegadores Internet Explorer, usamos a tag **BGSOUND** para inserir o som ao documento.

Porém, essas formatações não farão efeito algum quando o browser não estiver configurado para “tocar” música, ou se o computador que receber a página não tiver um sistema de som configurado.

Como o Internet Explorer é o browser que estamos utilizando vamos ver alguns dos parâmetros do comando BGSOUND.

- **DELAY** – Número inteiro, representando o tempo de espera, em segundos, que deve ser observado antes de dar início a apresentação do elemento sonoro. Este parâmetro foi implementado apenas a partir da versão 3 do Mosaic.
- **LOOP** – Número inteiro, determinando o número de vezes que o elemento deve ser apresentado. O valor infinite determina que o som seja repetido indefinidamente.
- Este comando é um suporte do Internet Explorer v 2.0 e Mosaic v 3.0.
- **SRC** – URL apontando para a localização do arquivo que contém o elemento sonoro.
- **VOLUME** – Inteiro, de -10000 a 0, especificando a intensidade da reprodução.
- **BALANCE** – Inteiro de -10000 e 10000, determinando o balanceamento proporcional dos canais estéreo.

4.3 - O que é Flash

O Flash é um programa fácil de se utilizar e de alto impacto na produção de animações gráficas e interativas. Ele utiliza vetores e bitmaps e por causa disso, consegue produzir arquivos de tamanho reduzido, mas visualmente fantásticos. Você pode conhecer esta incrível ferramenta em nosso curso de Webdesign.

Capítulo V - Trabalhando com Imagens

5.0 - GIF e JPEG

As imagens usadas na Web devem ser relativamente leves, normalmente utilizamos **GIF** para figuras ou ilustrações pequenas e **JPEG** para fotos ou imagens que necessitam de maior definição ou qualidade.

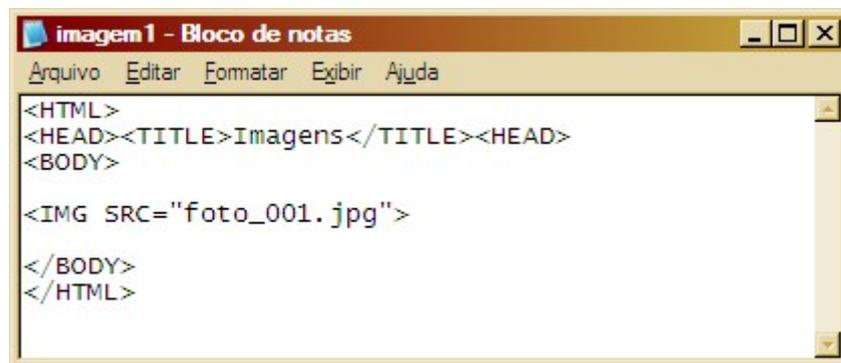
De maneira geral, JPEG é mais bem aplicado a imagens com variações de cor – fotografias digitalizadas, por exemplo – coloridas ou com mínimo 16 níveis em tons de cinza. Quanto mais complexa a imagem, melhor o resultado com o JPEG.

GIF é melhor para imagens com cores lisas (bordas, áreas com cores “puras”) realizando, nesses casos, uma compressão muito melhor do que JPEG faria. Nestes casos, JPEG produz imagens desfocadas. Por isso, imagens com preto-e-branco puros (sem tons de cinza) não devem ser processados em JPEG.

5.1 - Inserindo uma imagem

Para inserir imagens dentro de seus documentos HTML, utilize o comando **IMG** seguida do atributo **SRC** que deve estar presente para especificar o URL do arquivo que contém a imagem. A URL poderá ser até de um outro servidor que não o mesmo onde você hospeda seus arquivos, entretanto, isto não é conveniente.

Exemplo:



```
imagem1 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
<HTML>
<HEAD><TITLE>Imagens</TITLE></HEAD>
<BODY>

<IMG SRC="foto_001.jpg">

</BODY>
</HTML>
```

Figura 14 – Inserindo Imagem (IMAGEM1.HTML)

Veja agora alguns atributos que podem ser utilizados em suas imagens para obter um melhor acabamento:

ALT – Indica um texto alternativo, descrevendo brevemente a imagem, que é apresentado no lugar da imagem nos browsers texto, ou quando se desabilita o carregamento de imagens em browsers gráficos. Também é apresentado após o carregamento da imagem quando o ponteiro do mouse repousa sobre a mesma. É recomendável que esteja sempre presente.

WIDTH e HEIGHT – São atributos de dimensão da imagem, podendo receber valores em pixels ou em percentual. Grande parte dos editores HTML coloca automaticamente os valores destes atributos, quando indicamos a inserção de uma imagem.

Uma das vantagens de se usar esses atributos é que o browser pode montar mais rapidamente as páginas, por saber de antemão o espaço que deverá ser reservado para elas.

BORDER – Quando uma frase é marcada como âncora de um link, ela se apresenta sublinhada, quando uma imagem faz às vezes de âncora, ganha uma borda que indica sua condição de link. Porém, por questões de apresentação, nem sempre interessa termos essa borda ao redor da imagem. Assim, com o atributo **BORDER**, podemos controlar esse detalhe informando valores em pixels. Coloque o valor “0” para este parâmetro se quiser eliminar as bordas.

ALIGN – Determina o alinhamento da figura em relação ao texto podendo ser:

- **TOP** – Alinha o texto adjacente com o topo da imagem, embora com linhas compridas o resultado não seja muito bom.
- **MIDDLE** – Alinha o texto adjacente com o meio da imagem, embora com linhas compridas o resultado não seja muito bom.
- **BOTTOM** – Alinha o texto adjacente com a parte de baixo da imagem (default)
- **RIGHT** – Alinha imagem á direita, e tudo o que houver ao redor (texto, outras imagens) a partir do topo da imagem.
- **LEFT** – Alinha imagem á esquerda, e tudo o que houver ao redor (texto, outras imagens) a partir do topo da imagem.

Outros alinhamentos procuram posicionar as imagens com maior precisão com relação ao texto circundante: **TOPTXT**, **ABSMIDDLE** e **ABSBOTTOM**.

Exemplo:



Figura 15 – Imagem no Documento (IMAGEM2.HTML)

Capítulo VI - Formulários

6.1 - O que são os Formulários

Os Formulários permitem que o usuário que esteja navegando por um site possa submeter informações para o provedor do serviço. Normalmente os websites disponibilizam formulários de contato ou para inscrição em algum tipo de serviço.

Existem vários tipos de campos de entrada que podem ser utilizados, como:

- Campos de entrada de texto.
- Menus de múltipla escolha ou escolha única.
- Botões sim ou não.
- Botões para submissão ou limpeza de formulário, etc.

Cada um destes campos tem funcionalidade própria e você vai aprender em que situação aplica-los.

Provavelmente seu servidor utiliza um script para processar formulários. Você precisará das informações sobre este script para que seu formulário funcione corretamente. Normalmente estes scripts são escritos em linguagens como **CGI, PHP, ASP** e outras. Essas e outras informações sobre formulários podem ser obtidas com o suporte da empresa que lhe oferece o serviço de hospedagem. Caso seu plano de hospedagem não tenha um processador de formulários poderá usar um serviço gratuito ou processa-lo usando um cliente de e-mail, o que não é aconselhável pois nem todos os usuários possuem um configurado.

6.1 - Definindo um Formulário

Um Formulário é indicado pelas tags `<FORM>` e `</FORM>` como mostra o exemplo abaixo e deve ser colocado na seção de corpo de seu documento:

A tag de abertura do formulário deverá receber alguns parâmetros:

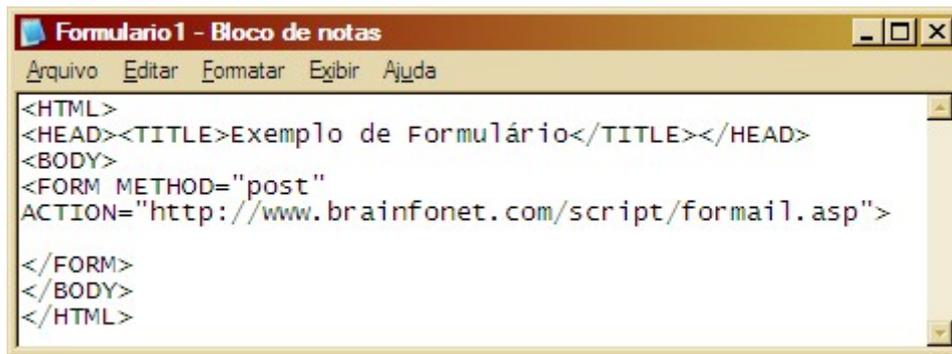
METHOD – Define o método de envio dos dados podendo ser:

- **GET** – A maioria dos documentos HTML são recuperados a partir da requisição de uma única URL ao servidor. Assim, um formulário que utilize esse método, envia toda sua informação ao final da URL ativada.

POST – Este método transmite toda a informação fornecida, via formulário, imediatamente após a URL ativada. Ou seja, quando o servidor recebe uma ativação de um formulário utilizando POST, ele sabe que precisa continuar “ouvindo” para obter a informação. Este é o melhor método.

O formulário também precisa saber para onde enviar as informações. Se você for usar um script para o processamento deverá informar a URL deste script, entretanto poderá enviar as informações diretamente para o e-mail do destinatário.

Veja um exemplo usando um script:



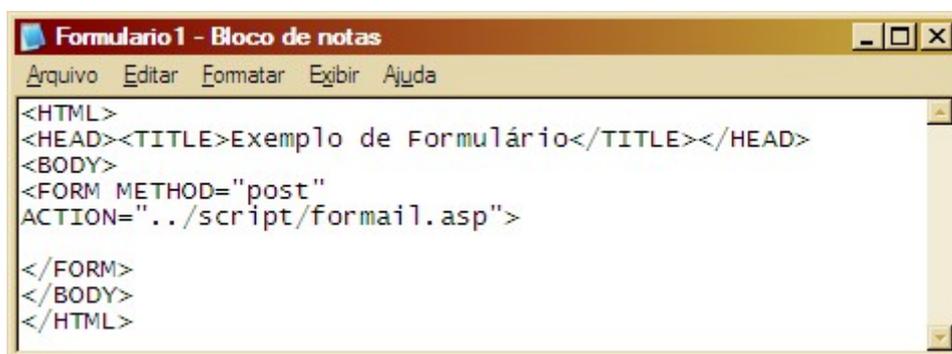
```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="http://www.brainfonet.com/script/formail.asp">

</FORM>
</BODY>
</HTML>
```

Figura 16 – Definindo um Formulário (FORMULARIO1.HTML)

Lembre-se que se você está referenciando um script que reside no mesmo servidor do formulário, você não precisa incluir a URL completa.

Veja um exemplo:

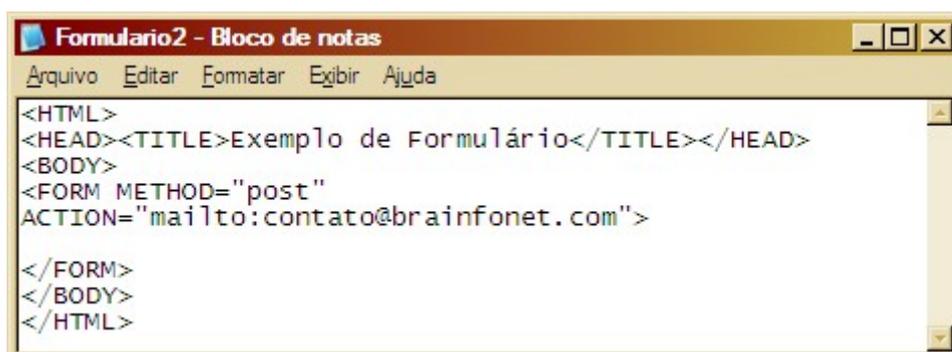


```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="../script/formail.asp">

</FORM>
</BODY>
</HTML>
```

Figura 17 – Definindo um Formulário (FORMULARIO2.HTML)

Caso queira que o processamento seja feito utilizando-se um cliente de e-mail, use o esquema apresentado no exemplo abaixo:

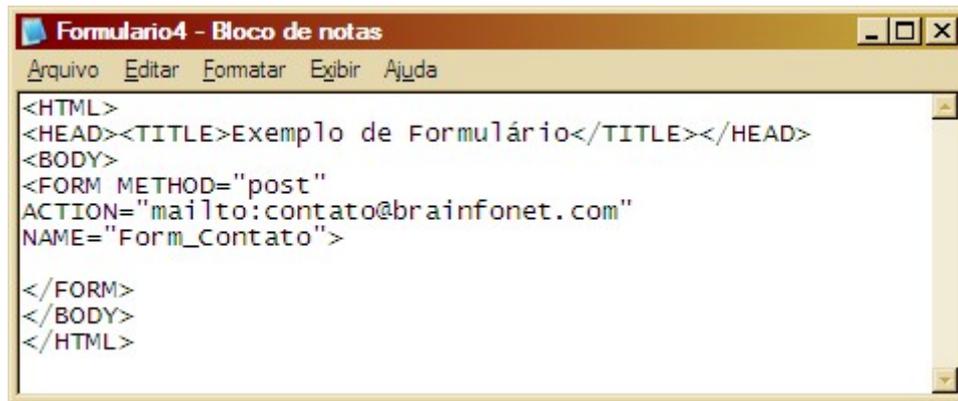


```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com">

</FORM>
</BODY>
</HTML>
```

Figura 18 – Usando um cliente de e-mail (FORMULARIO3.HTML)

É importante definir um nome para o formulário que poderá ser utilizado por um script posteriormente como **JavaScript**. Para isso utilize o parâmetro NAME. Veja um exemplo:



```

<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

</FORM>
</BODY>
</HTML>

```

Figura 19 – (FORMULARIO4.HTML)

**Você sabia?**

JavaScript é uma linguagem de programação da empresa Netscape, com algumas características da linguagem Java, porém bem menos complexa e de fácil aprendizado, que não precisa ser compilada.

Cada marcação de entrada em um formulário também deve possuir uma opção NAME associada, de tal forma que o script saiba qual o nome, isto é, como chamar cada valor digitado. Certamente você pode definir mais de um campo de entrada textual ou menu dentro de um formulário, mas certifique-se de que cada um possua um nome diferente.

6.1.1 - Campo de Texto

Este campo, permite a digitação de uma única palavra ou linha de texto, e possui uma largura padrão de 20 caracteres.

Deve ser escrito da seguinte maneira:

```
<INPUT TYPE="TEXT">
```

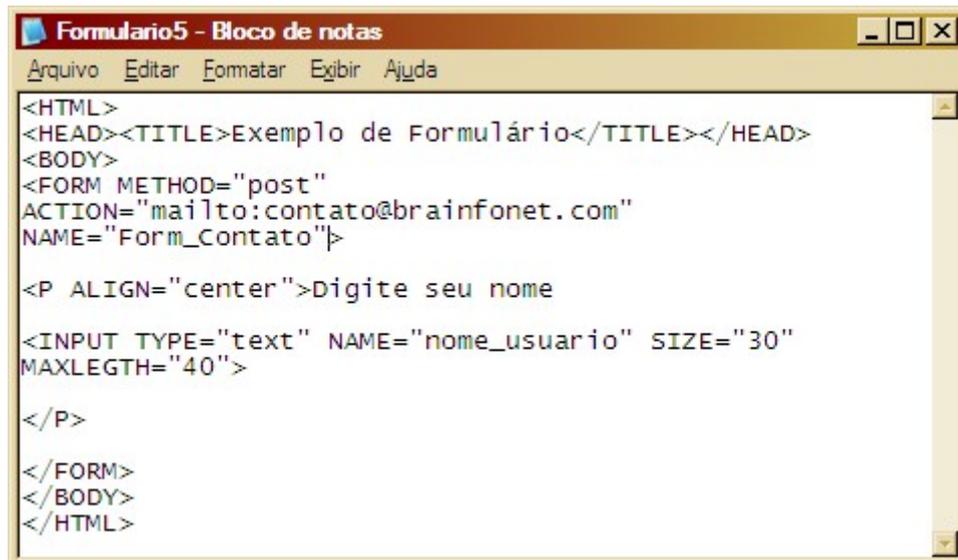
Parâmetros:

VALUE - Utilizando a marcação VALUE você especifica que texto aparecerá no campo quando o formulário for exibido. Este parâmetro é opcional.

SIZE - Esta marcação altera o tamanho deste campo exibido na tela. O usuário sempre poderá digitar mais caracteres do que o tamanho do campo na tela, pois o texto irá se deslocar para a esquerda dentro do campo.

MAXLENGTH - Limita o número de caracteres que o usuário pode digitar. O formulário irá emitir um bip de erro, se o usuário tentar digitar além do permitido.

Exemplo:



```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P ALIGN="center">Digite seu nome

<INPUT TYPE="text" NAME="nome_usuario" SIZE="30"
MAXLEGT="40">

</P>

</FORM>
</BODY>
</HTML>
```

Figura 20 – (FORMULARIO5.HTML)

Confira o resultado:

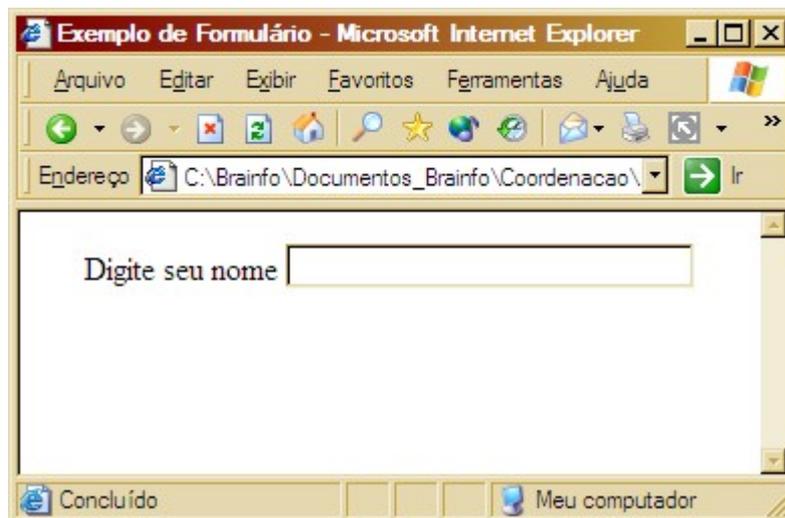


Figura 21 – Campo de Texto

Importante: O valor informado em NAME deve utilizar sublinhado, em vez de espaços em branco. Inserir espaços em branco nestes nomes, pode causar problemas no servidor na decodificação dos valores informados.

6.1.2 - Campo Senha

Marcações de entrada, do tipo **PASSWORD**, são idênticas aos campos do tipo texto, exceto pelo fato de todos os caracteres serem exibidos como asteriscos (*).

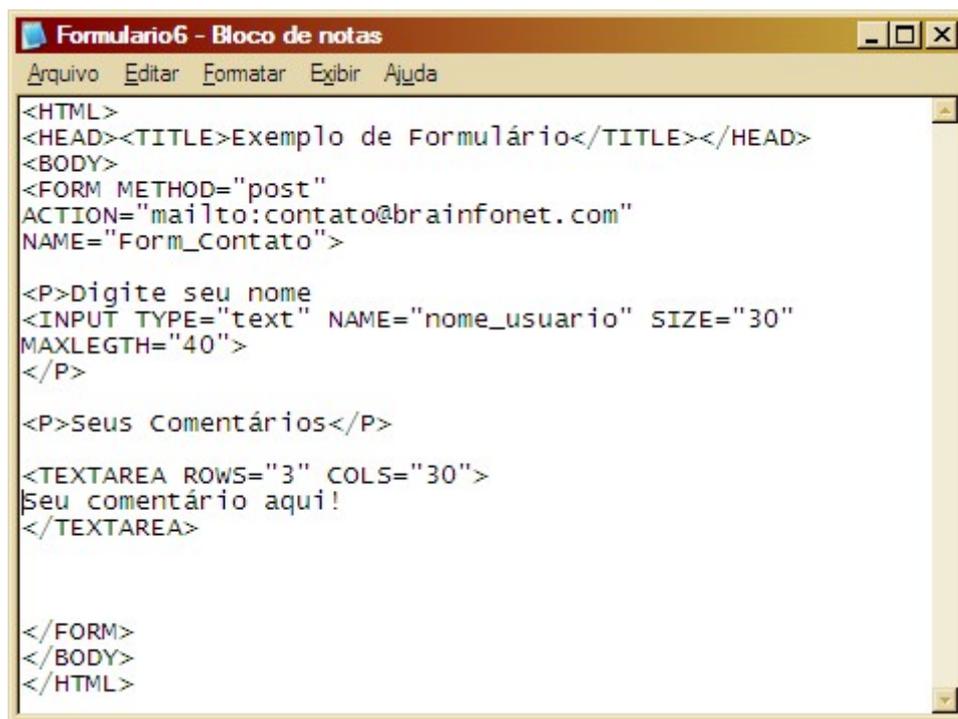
6.1.3 - Entrada oculta

É possível “esconder” uma informação passada dentro de blocos FORM com uma marcação `HIDDEN`. Esta informação é recebida pelo script de decodificação no servidor, mas não é diretamente exibida ao usuário no formulário.

6.1.4 - Entrada de várias linhas de texto

A marcação `TEXTAREA` não utiliza o formato convencional `INPUT TYPE` dos exemplos anteriores.

Veja um exemplo de sua utilização:



```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P>Digite seu nome
<INPUT TYPE="text" NAME="nome_usuario" SIZE="30"
MAXLENGH="40">
</P>

<P>Seus Comentários</P>

<TEXTAREA ROWS="3" COLS="30">
Seu comentário aqui!
</TEXTAREA>

</FORM>
</BODY>
</HTML>
```

Figura 22 – (FORMULARIO6.HTML)

Parâmetros:

ROWS – Especifica o número de linhas que será utilizado na exibição.

COLS – Especifica o número de colunas que será utilizado para exibição.

Se você deseja que um texto seja exibido no campo textual ao abrir o formulário, simplesmente coloque o texto entre as marcações de início e fim da `TEXTAREA`.

Veja o resultado do exemplo anterior:

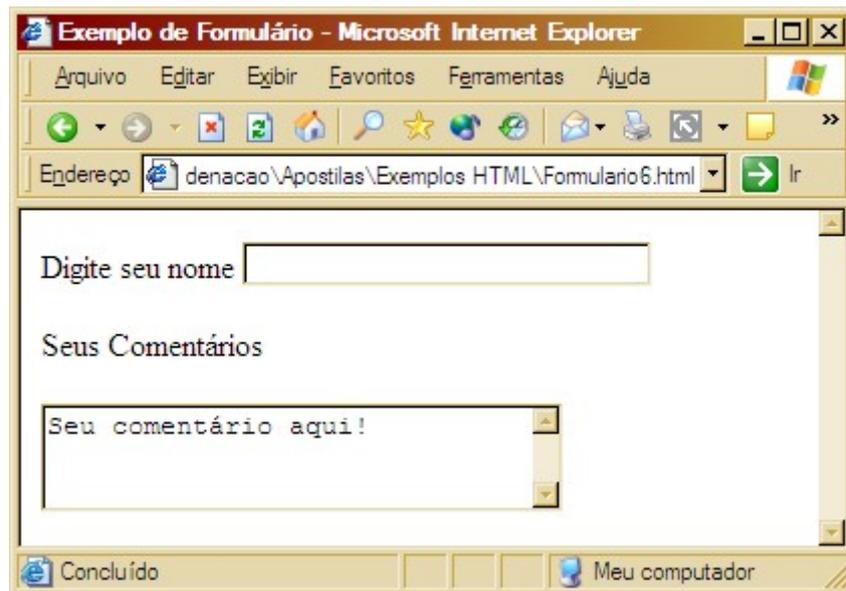


Figura 23 – Entrada de várias linhas

6.1.5 - Menus com opções

A marcação SELECT segue a mesma convenção de TEXTAREA. Ou seja, as opções de menu ficam entre a marcação de início **<SELECT>** e a de fim **</SELECT>**.

Veja um exemplo:

```

Formulário7 - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">
<P>selecione seu tipo de acesso:</P>
<SELECT NAME="Tipo_Acesso">
<OPTION VALUE="nulo" SELECTED>--- selecione aqui
----</OPTION>
<OPTION VALUE="Adsl">ADSL</OPTION>
<OPTION VALUE="Cable">Cable Modem</OPTION>
<OPTION VALUE="Linha">Linha Discada</OPTION>
</SELECT>
</FORM>
</BODY>

```

Figura 24 – (FORMULARIO7.HTML)

Observe que usamos o parâmetro NAME para definir o nome do menu de opções. Cada opção deve ser marcada com o comando <OPTION> seguido do parâmetro VALUE que define o seu valor e que será processado pelo script. Veja também que no primeiro OPTION usamos o parâmetro **SELECTED** para informar que esta é a opção que deverá vir selecionada. Caso esta marcação não for mencionada a primeira opção será usada como padrão.

Veja o resultado do exemplo:

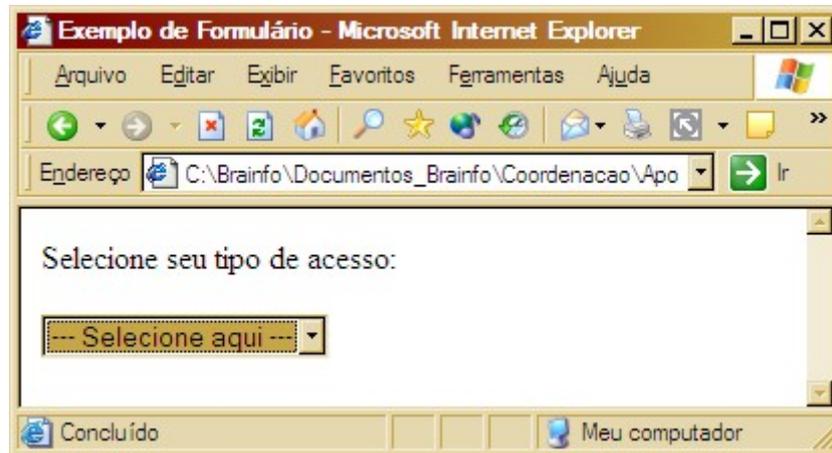


Figura 25 – Menu de Seleção

Você também pode obter um menu paginável utilizando a opção **SIZE** que recebe o valor do número de opções que deverão ser exibidas.

Veja o exemplo:

```

Formulario8 - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">
<P>Selecione seu tipo de acesso:</P>
<SELECT NAME="Tipo_Acesso" SELECTED SIZE="3">
<OPTION VALUE="nulo">--- selecione aqui ---</OPTION>
<OPTION VALUE="Adsl">ADSL</OPTION>
<OPTION VALUE="Cable">Cable Modem</OPTION>
<OPTION VALUE="Linha">Linha Discada</OPTION>
</SELECT>
</FORM>
</BODY>
</HTML>

```

Figura 26 – (FORMULARIO8.HTML)

Veja o resultado:

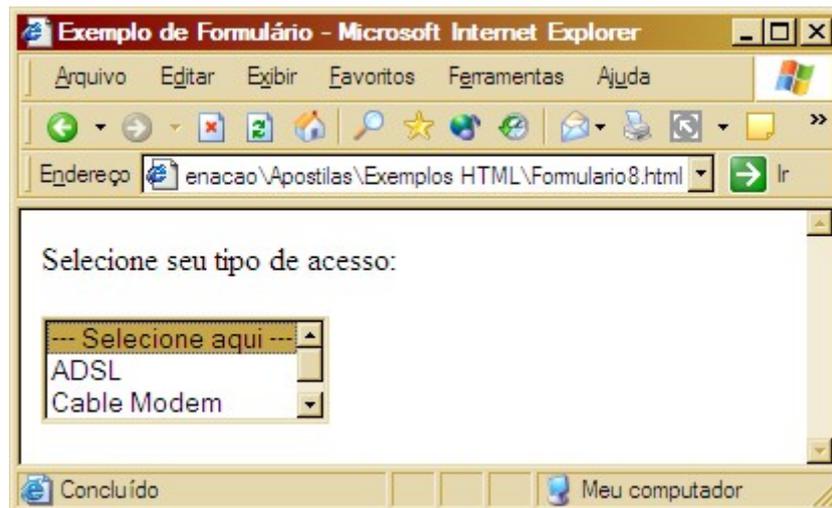


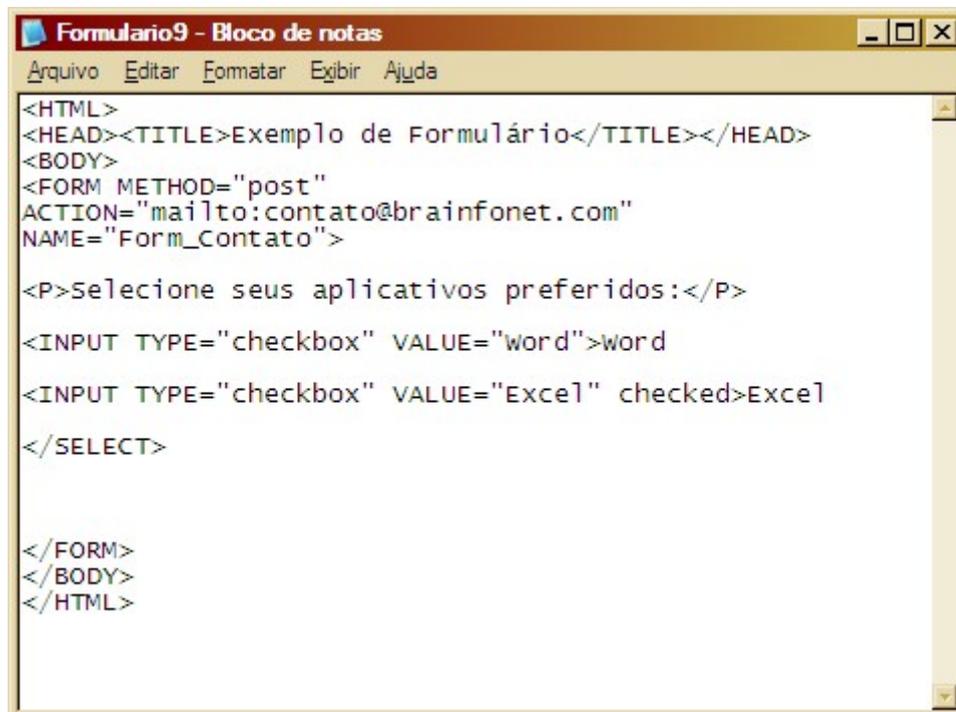
Figura 27 – Escolhendo várias opções

Para permitir que o usuário selecione múltiplas opções acrescente a opção **MULTIPLE**. A seleção múltipla deve ser feita utilizando-se a tecla **CTRL**.

6.1.6 - Botões sim ou não

Retornando ao formato de `INPUT TYPE=""`, a marcação `CHECKBOXES` é perfeita para escolher entre duas opções.

Observe um exemplo:



```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P>Selecione seus aplicativos preferidos:</P>

<INPUT TYPE="checkbox" VALUE="word">word
<INPUT TYPE="checkbox" VALUE="Excel" checked>Excel

</SELECT>

</FORM>
</BODY>
</HTML>
```

Figura 28 – (FORMULARIO9.HTML)

Use a opção **VALUE** para definir o valor processado pelo script. Caso o valor não for mencionado o valor "on" será enviado ao script.

Use a opção **CHECKED** caso queira marcar a opção como selecionada.

Veja o resultado do exemplo anterior:



Figura 29 – Botões Sim ou Não

6.1.7 - Botões de Radio

RADIOBUTTONS utilizam uma notação similar ao de CHECKBOXES, contudo, apenas uma opção pode ser escolhida.

Uma vez que uma seleção precisa ser feita, é melhor prover uma opção pré-selecionada com a opção **CHECKED**.

Veja nosso exemplo:

```

Formulario 10 - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P>Na sua opinião, qual o melhor sistema operacional:</P>

<INPUT TYPE="radio" VALUE="Dos">Ms-Dos
<INPUT TYPE="radio" VALUE="XP" CHECKED>windows XP
<INPUT TYPE="radio" VALUE="Linux">Linux

</SELECT>

</FORM>
</BODY>
</HTML>

```

Figura 30 – (FORMULARIO10.HTML)

Veja o resultado do código:

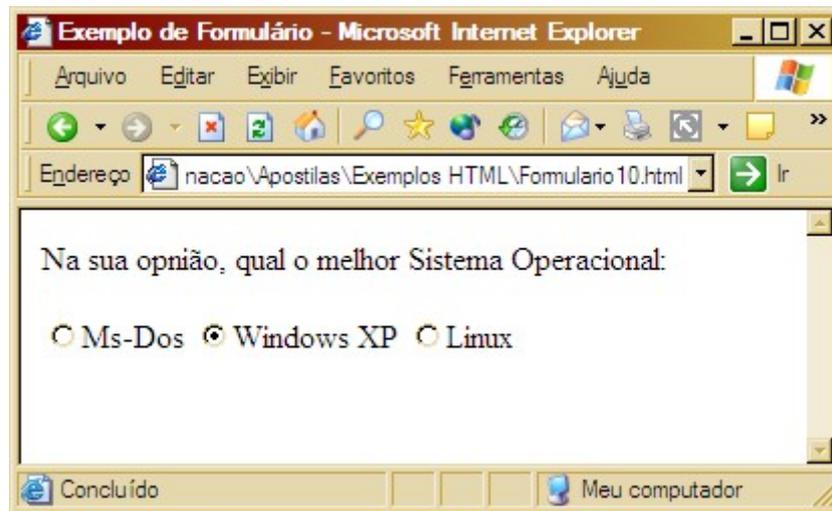


Figura 31 – Botões de Rádio

6.1.8 - Botões de submissão e limpeza

Em vez do usuário corrigir cada campo, um botão **RESET** pode ser utilizado para restaurar todos os campos a seus valores padrões, como se nenhuma informação houvesse sido digitada pelo usuário.

Observe o exemplo:

 A screenshot of a Notepad window titled "Formulario 11 - Bloco de notas". The window contains the following HTML code:


```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P ALIGN="center">Digite seu nome

<INPUT TYPE="text" NAME="nome_usuario" SIZE="30"
MAXLENGTH="40">

</P>

<INPUT TYPE="reset" VALUE="Limpar Formulário"
NAME="Limpa">

</FORM>
</BODY>
</HTML>
```

Figura 32 – (FORMULARIO11.HTML)

No exemplo usamos o comando **INPUT** onde a opção **TYPE** recebe o valor **RESET**. A opção **VALUE** define o texto exibido no botão e **NAME** especifica seu nome.

E finalmente, o formulário precisa de uma opção para enviar toda a informação digitada para o servidor, uma vez que o usuário terminou de preencher todos os campos de entrada. O botão SUBMIT transfere toda a informação para a URL especificada no elemento ACTION.

O botão de envio recebe na opção **TYPE** o valor **SUBMIT**.

Veja um exemplo já com os dois botões:



```
<HTML>
<HEAD><TITLE>Exemplo de Formulário</TITLE></HEAD>
<BODY>
<FORM METHOD="post"
ACTION="mailto:contato@brainfonet.com"
NAME="Form_Contato">

<P ALIGN="center">Digite seu nome

<INPUT TYPE="text" NAME="nome_usuario" SIZE="30"
MAXLENGH="40">

</P>

<INPUT TYPE="reset" VALUE="Limpar Formulário"
NAME="Limpa">

<INPUT TYPE="submit" VALUE="Enviar Dados" NAME="Envia">

</FORM>
</BODY>
</HTML>
```

Figura 33 – (FORMULARIO12.HTML)

Veja o resultado do código:



Figura 34 – Botões de Limpeza e Envio

Capítulo VII – Trabalhando com Frames

7.0 - O que são Frames

Os **Frames** são divisões da tela do browser em diversas telas (ou quadros) utilizados para tornar possível apresentar mais de uma página por vez: por exemplo, um índice principal em uma parte pequena da tela, e os textos relacionados ao índice em outra parte.

É muito fácil colocar frames em páginas, porém, nem todos os usuários gostam deles. Por isso deve-se avaliar a real necessidade da aplicação do recurso. Outro fator fundamental é saber que alguns navegadores antigos não suportam este tipo de recurso.

Para trabalhar com frames você precisa entender o seguinte:

- Deverá ser criado um arquivo que contenha os comandos referentes à criação dos frames. Este arquivo não utiliza o comando **BODY**.
- Deverão ser criados arquivos para cada um dos quadros criados e estes deverão ter uma referência no arquivo de frames principal que irá chamá-los.

Ao criar links, deverá ser indicado qual o quadro de destino.

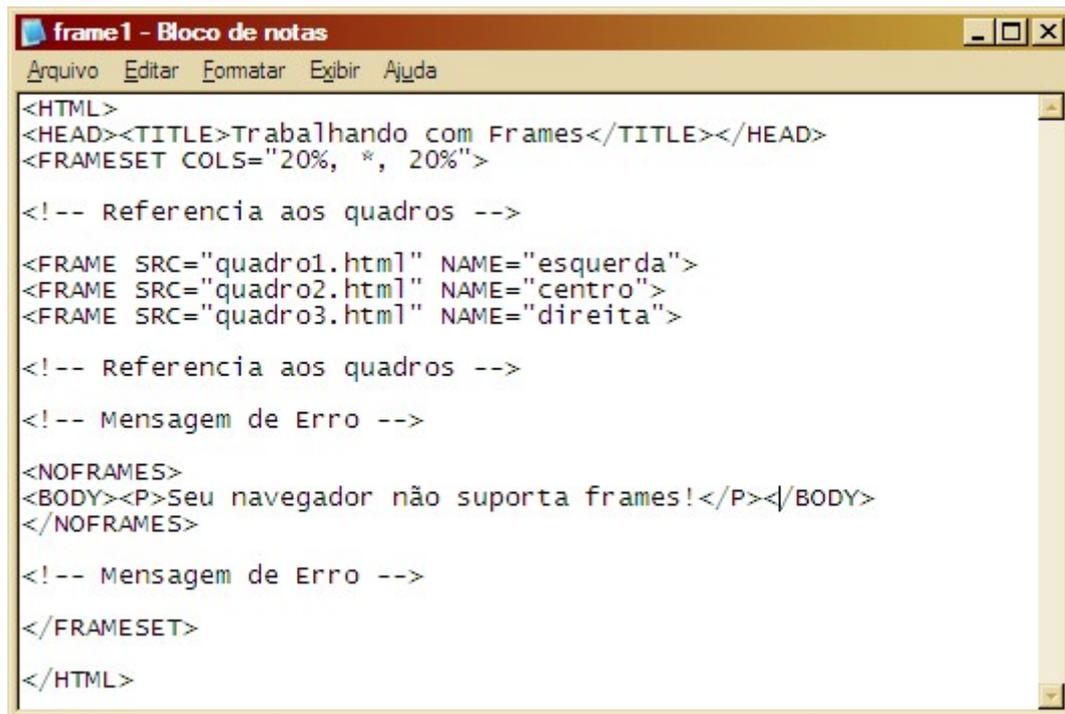
7.1 - Criando o Arquivo de Frames Principal

Você deve utilizar o comando **FRAME** e deverá determinar se utilizará uma divisão em linhas (ROWS) ou colunas (COLS) e quanto de espaço cada região irá utilizar. A porção de cada região poderá ser determinada em pixels ou em percentual. Utilize o símbolo "*" para determinar que uma determinada região deve ficar com o percentual restante.

A referência para cada um dos documentos referente aos quadros deverá ser feita na ordem em que foram feitas as divisões utilizando-se o comando **FRAMESET**. Este comando deve conter as opções **SRC** que recebe a **URL** do documento e **name** que identifica o nome do quadro que será utilizado posteriormente para determinar o destino dos links.

Caso queira apresentar um documento contendo uma mensagem para os usuários que não possuem navegador com suporte para frames inclua um comando **<NOFRAMES>** e **</NOFRAMES>** inserindo seu código HTML entre eles.

Veja um exemplo:



```

frame 1 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
<HTML>
<HEAD><TITLE>Trabalhando com Frames</TITLE></HEAD>
<FRAMESET COLS="20%, *, 20%">

<!-- Referencia aos quadros -->

<FRAME SRC="quadro1.html" NAME="esquerda">
<FRAME SRC="quadro2.html" NAME="centro">
<FRAME SRC="quadro3.html" NAME="direita">

<!-- Referencia aos quadros -->
<!-- Mensagem de Erro -->
<NOFRAMES>
<BODY><P>Seu navegador não suporta frames!</P></BODY>
</NOFRAMES>

<!-- Mensagem de Erro -->

</FRAMESET>
</HTML>

```

Figura 35 – (FRAME1.HTML)

Agora veja o resultado dos códigos:



Figura 36 – Documento dividido em 3 quadros

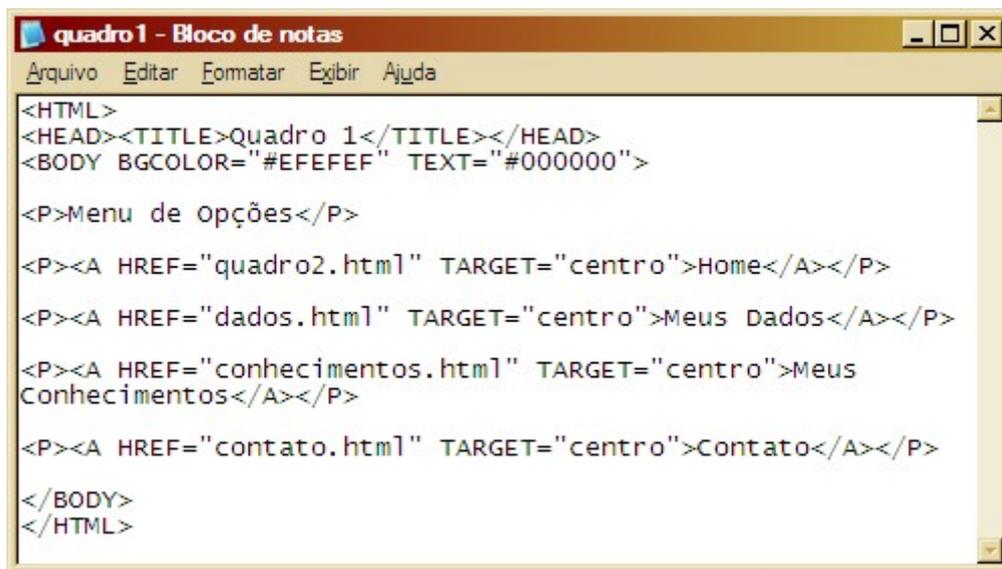
7.2 - Links com Frames

Sempre que aciona um link dentro de uma página, o padrão é que a página referente a esse link seja carregada na mesma janela da página anterior. O mesmo ocorrerá com os frames se você não especificar o quadro onde o novo documento deverá ser apresentado.

Para definir o nome do quadro de destino, você deve incluir a opção **NAME** contendo como valor o nome do quadro determinado com a opção **TARGET** no documento principal.

Veja o código fonte acima que o frame associado a apresenta.html tem um atributo **NAME**. Isso faz com que se possa “direcionar” o frame em que queremos mostrar determinada página ao acionarmos um link.

Veja um exemplo:



```

<HTML>
<HEAD><TITLE>Quadro 1</TITLE></HEAD>
<BODY BGCOLOR="#EFEFEF" TEXT="#000000">

<P>Menu de Opções</P>

<P><A HREF="quadro2.html" TARGET="centro">Home</A></P>

<P><A HREF="dados.html" TARGET="centro">Meus Dados</A></P>

<P><A HREF="conhecimentos.html" TARGET="centro">Meus
Conhecimentos</A></P>

<P><A HREF="contato.html" TARGET="centro">Contato</A></P>

</BODY>
</HTML>

```

Figura 37 – (QUADRO1.HTML)

7.3 - Atributos de Frames

Até este ponto, vimos os atributos **COLS** e **ROWS**, para **FRAMESET**, **SRC** e **NAME**, para **FRAME**.

Outros atributos permitem um maior controle sobre a apresentação dos quadros:

FRAMEBORDER – Determina a exibição ou não das bordas que compõem os quadros. Deve receber o valor “YES” ou “NO”. Esta opção deve ser utilizada junto do comando **FRAMESET**.

BORDER – Determina o espaço entre as bordas dos quadros. Para eliminar a borda use o valor “0”.

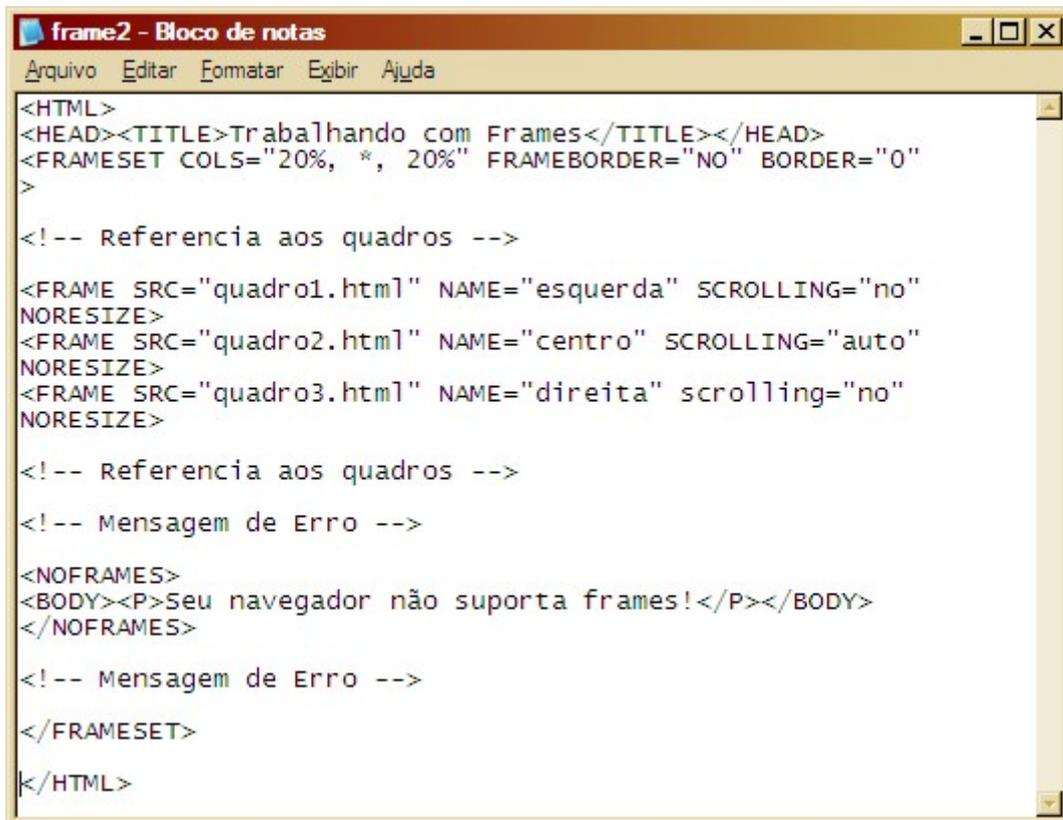
SCROLLING – Determina a condição da barra de rolagem de um quadro, podendo ser:

- Desativada – “NO”
- Ativada – “YES”
- Automática – “AUTO”

Esta opção deve ser usada com o comando **FRAME**.

Por padrão, os quadros podem ser redimensionados pelo usuário. Se quiser bloquear esta opção, inclua uma opção NORESIZE para o quadro desejado.

Veja um exemplo:



```
<HTML>
<HEAD><TITLE>Trabalhando com Frames</TITLE></HEAD>
<FRAMESET COLS="20%, *, 20%" FRAMEBORDER="NO" BORDER="0"
>
<!-- Referencia aos quadros -->
<FRAME SRC="quadro1.html" NAME="esquerda" SCROLLING="no"
NORESIZE>
<FRAME SRC="quadro2.html" NAME="centro" SCROLLING="auto"
NORESIZE>
<FRAME SRC="quadro3.html" NAME="direita" scrolling="no"
NORESIZE>
<!-- Referencia aos quadros -->
<!-- Mensagem de Erro -->
<NOFRAMES>
<BODY><P>Seu navegador não suporta frames!</P></BODY>
</NOFRAMES>
<!-- Mensagem de Erro -->
</FRAMESET>
</HTML>
```

Figura 38 – (FRAME2.HTML)

Capítulo VIII - Tabela de Cores

White rgb=#FFFFFF	Red rgb=#FF0000	Green rgb=#00FF00
Blue rgb=#0000FF	Magenta rgb=#FF00FF	Cyan rgb=#00FFFF
Yellow rgb=#FFFF00	Black rgb=#000000	Aquamarine rgb=#70DB93
Baker's Chocolate rgb=#5C3317	Blue Violet rgb=#9F5F9F	Brass rgb=#B5A642
Bright Gold rgb=#D9D919	Brown rgb=#A62A2A	Bronze rgb=#8C7853
Bronze II rgb=#A67D3D	Cadet Blue rgb=#5F9F9F	Cool Copper rgb=#D98719
Copper rgb=#B87333	Coral rgb=#FF7F00	Corn Flower Blue rgb=#42426F
Dark Brown rgb=#5C4033	Dark Green rgb=#2F4F2F	Dark Green Copper rgb=#4A766E
Dark Olive Green rgb=#4F4F2F	Dark Orchid rgb=#9932CD	Dark Purple rgb=#871F78
Dark Slate Blue rgb=#6B238E	Dark Slate Grey rgb=#2F4F4F	Dark Tan rgb=#97694F
Dark Turquoise rgb=#7093DB	Dark Wood rgb=#855E42	Dim Grey rgb=#545454
Dusty Rose rgb=#856363	Feldspar rgb=#D19275	Firebrick rgb=#8E2323
Forest Green rgb=#238E23	Gold rgb=#CD7F32	Goldenrod rgb=#DDB700
Grey rgb=#C0C0C0	Green Copper rgb=#527F76	Green Yellow rgb=#93DB70
Hunter Green rgb=#215E21	Indian Red rgb=#4E2F2F	Khaki rgb=#9F9F5F
Light Blue rgb=#C0D9D9	Light Grey rgb=#A8A8A8	Light Steel Blue rgb=#8F8FBD
Light Wood rgb=#E9C2A6	Lime Green rgb=#32CD32	Mandarian Orange rgb=#E47833
Maroon rgb=#8E236B	Medium Aquamarine rgb=#32CD99	Medium Blue rgb=#3232CD
Medium Forest Green rgb=#6B8E23	Medium Goldenrod rgb=#EAEAAE	Medium Orchid rgb=#9370DB
Medium Sea Green rgb=#426F42	Medium Slate Blue rgb=#7F00FF	Medium Spring Green rgb=#7FFF00
Medium Turquoise rgb=#70DBDB	Medium Violet Red rgb=#DB7093	Medium Wood rgb=#A68064
Midnight Blue rgb=#2F2F4F	Navy Blue rgb=#23238E	Neon Blue rgb=#4D4DFF
Neon Pink rgb=#FF6EC7	New Midnight Blue rgb=#00009C	New Tan rgb=#EBC79E
Old Gold rgb=#CFB53B	Orange rgb=#FF7F00	Orange Red rgb=#FF2400
Orchid rgb=#DB70DB	Pale Green rgb=#8FBC8F	Pink rgb=#BC8F8F
Plum rgb=#EAADEA	Quartz rgb=#D9D9F3	Rich Blue rgb=#5959AB

Salmon rgb=#6F4242	Scarlet rgb=#8C1717	Sea Green rgb=#238E68
Semi-Sweet Chocolate rgb=#6B4226	Sienna rgb=#8E6B23	Silver rgb=#E6E8FA
Sky Blue rgb=#3299CC	Slate Blue rgb=#007FFF	Spicy Pink rgb=#FF1CAE
Spring Green rgb=#00FF7F	Steel Blue rgb=#236B8E	Summer Sky rgb=#38B0DE
Tan rgb=#DB9370	Thistle rgb=#D8BFD8	Turquoise rgb=#ADEAEA
Very Dark Brown rgb=#5C4033	Very Light Grey rgb=#CDCDCD	Violet rgb=#4F2F4F
Violet Red rgb=#CC3299	Wheat rgb=#D8D8BF	