

# **Apostila de Introdução à Linguagem HTML**

## **- Disponibilização de Conteúdos na WEB -**

**Profa. Flávia Pereira de Carvalho**

**Março de 2007**

## Sumário

Página

<b>1 INTRODUÇÃO À LINGUAGEM HTML</b> .....	<b>3</b>
<b>2 EDIÇÃO DE DOCUMENTOS HTML</b> .....	<b>4</b>
<b>3 PUBLICAÇÃO DE DOCUMENTOS NA INTERNET (SITES)</b> .....	<b>4</b>
<b>4 DOCUMENTO HTML BÁSICO E SEUS COMPONENTES</b> .....	<b>5</b>
<b>5 SEÇÃO &lt;HEAD&gt;</b> .....	<b>5</b>
<b>6 SEÇÃO &lt;BODY&gt;</b> .....	<b>6</b>
<b>7 FONTES</b> .....	<b>7</b>
<u>CORES</u> .....	7
<u>TAMANHOS</u> .....	7
<u>TIPOS</u> .....	7
<b>8 CABEÇALHOS</b> .....	<b>8</b>
<u>ANINHAMENTO DE CABEÇALHOS</u> .....	9
<u>ALINHAMENTO DE CABEÇALHOS</u> .....	9
<b>9 SEPARADORES</b> .....	<b>10</b>
<u>QUEBRA DE LINHA</u> .....	10
<u>PARÁGRAFOS</u> .....	10
<b>10 LINHA HORIZONTAL</b> .....	<b>11</b>
<b>11 COMENTÁRIOS</b> .....	<b>12</b>
<b>12 LISTAS EM HTML</b> .....	<b>12</b>
<u>LISTAS DE DEFINIÇÃO</u> .....	12
<u>LISTAS COM MARCADORES</u> .....	13
<u>LISTAS NUMERADAS</u> .....	14
<u>LISTAS E SUB-LISTAS</u> .....	16
<b>13 FORMATAÇÃO DE TEXTOS E CARACTERES</b> .....	<b>17</b>
<b>14 FORMATAÇÃO DE FRASES</b> .....	<b>18</b>
<b>15 CARACTERES ESPECIAIS</b> .....	<b>19</b>
<b>16 MARQUEE</b> .....	<b>20</b>
<b>17 LIGAÇÕES (USO DE LINKS)</b> .....	<b>20</b>
<u>ATRIBUTOS</u> .....	21
<u>CAMINHOS</u> .....	21
<u>Caminho Relativo</u> .....	21
<u>Caminho Absoluto</u> .....	22
<u>INDICADORES</u> .....	22
<b>18 INSERÇÃO DE IMAGENS</b> .....	<b>23</b>
<u>ATRIBUTOS BÁSICOS DE IMAGEM</u> .....	23
<b>19 MOLDURAS DE IMAGEM</b> .....	<b>26</b>
<b>20 TABELAS</b> .....	<b>27</b>
<u>ELEMENTOS BÁSICOS DE TABELAS</u> .....	27
<u>TÍTULOS, LINHAS E ELEMENTOS</u> .....	27
<u>TÍTULOS COMPREENDENDO MAIS DE UMA COLUNA OU LINHA</u> .....	28
<u>TABELAS SEM BORDA</u> .....	29
<u>EXTENSÕES DE TABELAS</u> .....	29
<u>Cor de fundo</u> .....	29
<u>Cor de borda</u> .....	30
<b>21 FRAMES</b> .....	<b>30</b>

## 1 Introdução à Linguagem HTML

---

**HTML** (*HyperText Markup Language* - Linguagem de Formatação de Hipertexto) é fruto do "casamento" dos padrões HyTime e SGML.

**HyTime - Hypermedia/Time-based Document Structuring Language:** *HyTime* (ISO 10744:1992) - padrão para representação estruturada de hipermídia e informação baseada em tempo. Um documento é visto como um conjunto de eventos concorrentes dependentes de tempo (áudio, vídeo, etc.), conectados por *webs* ou *hyperlinks*. O padrão *HyTime* é independente dos padrões de processamento de texto em geral. Ele fornece a base para a construção de sistemas hipertextos padronizados, consistindo de documentos que aplicam os padrões de maneira particular.

**SGML - Standard Generalized Markup Language:** Padrão ISO 8879 de formatação de textos: não foi desenvolvido para hipertexto, mas torna-se conveniente para transformar documentos em hiper-objetos e para descrever as ligações. SGML não é padrão aplicado de maneira padronizada: todos os produtos SGML têm seu próprio sistema para traduzir as etiquetas (*tags*) para um particular formatador de texto.

**DTD - Document Type Definition:** Define as regras de formatação para uma dada classe de documentos. Um DTD ou uma referência para um DTD deve estar contido em qualquer documento conforme o padrão SGML.

**Portanto, HTML é definido segundo um DTD de SGML.**

Todo documento HTML apresenta elementos entre **parênteses angulares** (< e >). Esses elementos são as **etiquetas** (*tags*) de HTML, que são os comandos de formatação da linguagem. A maioria das etiquetas tem sua correspondente de fechamento, representada com uma "barra" (/):

<etiqueta>...</etiqueta>

Isso é necessário porque as etiquetas servem para definir a formatação de uma porção de texto, e assim marcamos onde começa e onde termina o texto com a formatação especificada por ela.

Alguns elementos são chamados "vazios", pois não marcam uma região de texto, apenas inserem alguma coisa no documento, não havendo a necessidade do fechamento:

<etiqueta>

Todos os elementos podem ter atributos:

<etiqueta **atributo1**=valor1 **atributo2**=valor2>...</etiqueta>

HTML é um recurso muito simples e acessível para a produção de documentos. Nesta apostila, será possível aprender grande parte de seus elementos.

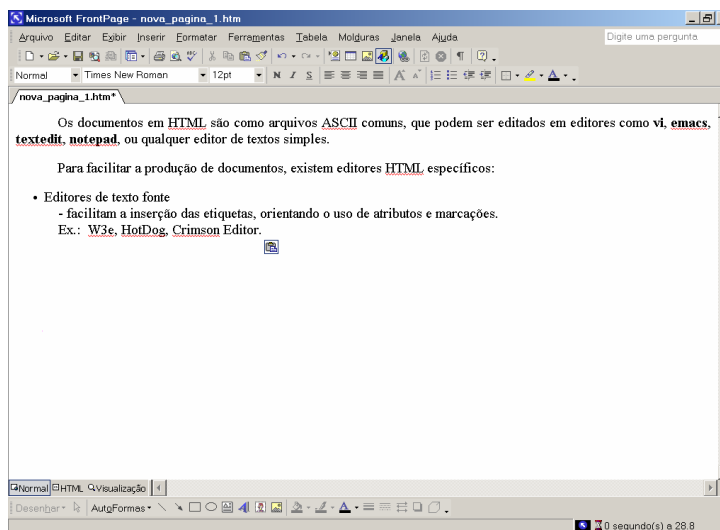


**Dica:** Não existem programas em HTML, pois HTML não é uma linguagem de programação, mas de formatação (marcação). Portanto, a rigor não existem "programadores" de HTML.

## 2 Edição de Documentos HTML

---

Existem Editores HTML chamados WYSIWYG (*what you see is what you get* - o que você vê é o que você tem). Eles oferecem ambiente de edição com um resultado final das marcações (pois o resultado final depende do *browser*<sup>1</sup> usado para visitar a página). Alguns bastante conhecidos são por exemplo: FrontPage e Dreamweaver.



**Figura 1 – Tela do FrontPage**

Além dos editores específicos para HTML, Editores de Textos bastante utilizados, como o Word, entre outros, permitem a exportação de seus documentos próprios para o formato HTML (menu Arquivo, Salvar como, Salvar\_como Tipo).

Um documento HTML, normalmente terá extensão **.html** ou **.htm**.

## 3 Publicação de Documentos na Internet (Sites)

---

Para que uma página esteja permanentemente disponível pela Web, ela precisa ter um endereço fixo, **alojada em um servidor**.

Existem vários provedores de espaço (*hosting*) gratuitos e também os provedores de acesso geralmente oferecem espaço para os *sites* de seus assinantes. *Sites* com fins lucrativos geralmente são hospedados em provedores de espaço pagos.

Definida a hospedagem, basta enviar para o provedor os arquivos de seu *site* (via FTP<sup>2</sup> ou por uma página de envio no próprio provedor de espaço) e suas páginas já estarão disponíveis para visitas no mundo todo.

---

<sup>1</sup> *Browser*: Navegador Internet. Exemplos: Netscape, Internet Explorer, Mozilla, etc.

<sup>2</sup> FTP - *File Transfer Protocol*: Protocolo de Transferência de Arquivos.

## 4 Documento HTML Básico e seus Componentes

---

A estrutura de um documento HTML apresenta os seguintes componentes:

```
<HTML>
<HEAD> <TITLE> Título do Documento </TITLE> </HEAD>
<BODY>
    texto,
    imagem,
    links, etc...
</BODY>
</HTML>
```

**Figura 2 – Componentes Básicos de um Documento HTML**

As etiquetas (*tags*) HTML **não são sensíveis à caixa**, ou seja, tanto faz escrever <HTML>, <Html>, <html>, <HtMl>, etc. Não tem diferença entre maiúsculas e minúsculas.

Os documentos se dividem em duas seções principais, que veremos a seguir.

## 5 Seção <HEAD>

---

<HEAD> contém informações sobre o documento. O elemento <TITLE>, por exemplo, define um título, que é mostrado no alto da janela do *browser*. Exemplo:

```
<HEAD> <TITLE> Página do Fulano de Tal </TITLE> </HEAD>
```

**Todo documento WWW deve ter um título**; esse título é referenciado em buscas pela rede, dando uma identidade ao documento. Ao adicionar uma página aos seus Favoritos (*Bookmarks*), o título da página se torna a âncora de atalho para ela. Por isso é sugerido que os títulos dos documentos sejam sugestivos, evitando-se títulos genéricos como "**Introdução**", por exemplo. O título também é bastante significativo para a listagem de uma página nos resultados de pesquisas nos *sites* de busca da Internet.

## 6 Seção <BODY>

---

Tudo que estiver contido em <BODY> será mostrado na janela principal do *browser*, sendo apresentado ao leitor. <BODY> pode conter cabeçalhos, parágrafos, listas, tabelas, *links* para outros documentos, imagens, formulários, animações, vídeos, sons e *scripts* embutidos.

### → Atributos de <BODY>

Através de atributos de <BODY>, podemos definir cores para os textos, *links* e para o fundo das páginas, bem como uma imagem de fundo (marca d'água):

```
<BODY BGCOLOR="#rrggbb" TEXT="#rrggbb" LINK="#rrggbb" ALINK="#rrggbb"
VLINK="#rrggbb" BACKGROUND="URL">
```

Onde:

**BGCOLOR:** *Background Color*, ou seja, cor de fundo da página. Quando não é indicada a cor de fundo, o *browser* irá mostrar uma cor padrão, geralmente cinza ou branco.

**TEXT:** Cor dos textos da página (padrão: preto).

**LINK:** Cor dos *links* (padrão: azul).

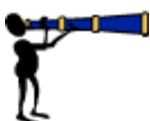
**ALINK:** Cor dos *links* quando acionados, clicados (padrão: vermelho).

**VLINK:** Cor dos *links* depois de visitados (padrão: azul escuro ou roxo).

Seus valores são dados em hexadecimal, equivalentes a cores no padrão RGB (Red, Green, Blue). Existem tabelas de cores com esses valores, mas grande parte dos editores já oferece uma interface bem amigável através da qual escolhemos as cores desejadas, sem nos preocuparmos com números esdrúxulos tais como #FF80A0.

*Browsers* que seguem a definição de HTML 3.2 em diante, também aceitam **16 nomes de cores**, tirados da paleta VGA do Windows - por exemplo, podemos escrever **BGCOLOR="BLUE"**. Porém, *browsers* mais antigos não apresentarão as cores indicadas.

**BACKGROUND:** Indica o URL<sup>3</sup> da imagem a ser replicada no fundo da página, como uma marca d'água. Para efeitos de design, é possível fixar a imagem de fundo, para que ela não se mova junto com o texto ao se rolar a página. Esse efeito *não é padrão* e funciona no Internet Explorer.



**Dica:** Os nomes das cores - os 16 nomes de cores aceitos desde a versão 3.2 da HTML são estes: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow.

---

<sup>3</sup> URL: O sistema de endereçamento da Web é baseado em uma sintaxe chamada URI (*Universal Resource Identifier* - Identificador Universal de Recursos). Os endereços que utilizamos atualmente são os URLs, que seguem essa sintaxe. URL significa *Uniform Resource Locator* - Localizador Uniforme de Recursos. Um exemplo de URL é: <http://www.faccat.br> Esse endereço identifica: **o protocolo de acesso** ao recurso desejado (http) e **a máquina** a ser acessada (www.faccat.br).

## 7 Fontes

---

Atributos das fontes:

### Cores

As cores são introduzidas através do elemento `<FONT>`, usando o sistema RGB para cores (da mesma forma que vimos para cores de documentos), como mostrado no exemplo a seguir:

```
<FONT COLOR="red"> Texto </FONT>
```

Produzirá na página: a palavra **Texto** escrita na cor **vermelha**.

Assim, um trecho de texto pode ter uma cor diferente da definição geral de cores, feita através dos atributos de `<BODY>`.

### Tamanhos

Veja a formatação a seguir: produzirá na página: a palavra **Texto** como o **tamaho 3**.

```
<FONT SIZE=3> Texto </FONT>
```

Este comando permite que o autor do documento altere o tamanho das letras em trechos específicos de texto. O tamanho básico dos textos é 3 (padrão). Podemos indicar tamanhos relativos a esse, por exemplo:

```
<FONT SIZE=+2> Letra maior </FONT>
```

Letra normal

```
<FONT SIZE=-2> Letra menor </FONT>
```

**Letra maior** Letra normal Letra menor

### Tipos

Uma evolução que permite a escolha da fonte para os textos (tipo de letra), é o atributo `FACE`:

```
<FONT FACE="Arial"> Texto </FONT>
```

Produzirá: a palavra Texto com a fonte (tipo de letra) Arial.

## **Mais exemplos:**

```
<FONT FACE="Verdana" COLOR="blue"> Fonte Verdana Azul </FONT>
```

Fonte Verdana Azul

```
<FONT FACE="Arial" COLOR="green"> Fonte Arial Verde </FONT>
```

Fonte Arial verde

```
<FONT FACE="Courier New" COLOR="red"> Fonte Courier New Vermelha  
</FONT>
```

Fonte Courier New Vermelha

## **8 Cabeçalhos**

---

Há seis níveis de cabeçalhos em HTML, de `<H1>` a `<H6>`:

```
<H1> Este é um cabeçalho de nível 1 </H1>
```

```
<H2> Este é um cabeçalho de nível 2 </H2>
```

```
<H3> Este é um cabeçalho de nível 3 </H3>
```

```
<H4> Este é um cabeçalho de nível 4 </H4>
```

```
<H5> Este é um cabeçalho de nível 5 </H5>
```

```
<H6> Este é um cabeçalho de nível 6 </H6>
```

→ Esses cabeçalhos são mostrados no *browser* da seguinte forma:

# Este é um cabeçalho de nível 1

## Este é um cabeçalho de nível 2

### Este é um cabeçalho de nível 3

#### Este é um cabeçalho de nível 4

##### Este é um cabeçalho de nível 5

###### Este é um cabeçalho de nível 6



## Aninhamento de cabeçalhos

Os cabeçalhos não podem ser aninhados, isto é, a formatação:

```
<H2>Este é <H1>um cabeçalho de nível 1</H1> dentro de um cabeçalho de nível 2</H2>
```

→ pode produzir algum resultado próximo ao desejado:

Este é  
um cabeçalho de nível 1  
dentro de um cabeçalho de nível 2

Mas o mais comum é que os *browsers* "entendam" essa formatação como sendo:

```
<H2>Este é</H2> <H1>um cabeçalho de nível 1</H1> dentro de um cabeçalho de nível 2</H2>
```

Ou seja, como se estivesse faltando uma etiqueta de fechamento de <H2> antes de <H1> e faltando uma abertura de <H2> depois do fechamento de <H1>, oferecendo o seguinte resultado:

Este é  
um cabeçalho de nível 1  
dentro de um cabeçalho de nível 2

Os editores WYSIWYG naturalmente não permitem o aninhamento de cabeçalhos.

## Alinhamento de cabeçalhos

Os cabeçalhos têm atributos de alinhamento:

```
<H2 ALIGN=CENTER>Cabeçalho centralizado</H2>
```

**Cabeçalho centralizado**

```
<H3 ALIGN=RIGHT>Cabeçalho alinhado à direita</H3>
```

**Cabeçalho alinhado à direita**

```
<H4 ALIGN=LEFT>Cabeçalho alinhado à esquerda (default=padrão)</H4>
```

**Cabeçalho alinhado à esquerda (default=padrão)**

## 9 Separadores

---

As quebras de linha do texto fonte não são significativas na apresentação de documentos em HTML. Para organizar os textos, precisamos de separadores, apresentados a seguir.

### Quebra de linha

Quando queremos mudar de linha, usamos o elemento `<BR>`. Isso só é necessário quando queremos uma quebra de linha em determinado ponto, pois os *browsers* já quebram as linhas automaticamente para apresentar os textos.

Com sucessivos `<BR>`, podemos inserir diversas linhas em branco nos documentos. Esse elemento tem um atributo especial, que será apresentado no item sobre inserção de imagens.

### Parágrafos

Para separar blocos de texto, usamos o elemento `<P>`, por exemplo:

Parágrafo 1; `<P>` Parágrafo 2.

→ que produz o seguinte:

**Parágrafo1;**

**Parágrafo2.**

→ Combinando parágrafos e quebras de linha, temos por exemplo:

Parágrafo 1;`<br>` linha 1 do parágrafo 1, `<br>`linha 2 do parágrafo 1.`<P>`Parágrafo 2;`<br>` linha 1 do parágrafo 2, `<br>`linha 2 do parágrafo 2.

→ O resultado da marcação acima é:

**Parágrafo 1;**  
**linha 1 do parágrafo 1,**  
**linha 2 do parágrafo 1.**

**Parágrafo 2;**  
**linha 1 do parágrafo 2,**  
**linha 2 do parágrafo 2.**

**<P>** tem atributo de alinhamento, semelhante aos cabeçalhos, como nos exemplos a seguir:

**<P ALIGN=CENTER>** Assim como os trens, as boas idéias às vezes chegam com atraso. **<BR>**  
(Giovani Guareschi) **</P>**

**Assim como os trens, as boas idéias às vezes chegam com atraso.  
(Giovani Guareschi)**

**<P ALIGN=RIGHT>**Como diz o provérbio chinês: “É melhor passar por ignorante uma vez do que permanecer ignorante para sempre”.**</P>**

**Como diz o provérbio chinês: “É melhor passar por ignorante uma vez do que permanecer ignorante para sempre”.**

**<P ALIGN=LEFT>**Este é o alinhamento padrão (*default*), e por isso não vou colocar nenhuma frase especial.**</P>**

**Este é o alinhamento padrão (default), e por isso não vou colocar nenhuma frase especial.**

## 10 Linha Horizontal

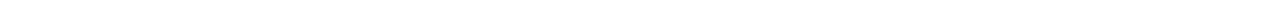
---

**<HR>** insere uma linha horizontal, como a apresentada abaixo:



Essa linha tem diversos atributos, oferecendo resultados diversos.

**<HR SIZE=7>** insere uma linha de largura 7 (pixels):



**<HR WIDTH=50%>** insere uma linha que ocupa 50% do espaço horizontal disponível:



**<HR WIDTH=30% ALIGN=RIGHT NOSHADE>** insere uma linha de comprimento 30% (do espaço horizontal disponível), alinhada à direita, sem efeito tridimensional:



`<HR SIZE=70 WIDTH=2 ALIGN=LEFT>` insere uma linha de largura 70 (pixels), comprimento 2 (pixels), alinhada à esquerda (o Netscape, aparentemente, não aceita esta formatação de `<HR>`):



**Dica: Divisões** - outros elementos usados para separar porções de texto são o `<DIV>` e o `<CENTER>`. `<CENTER>` centraliza os elementos (textos, imagens, tabelas) que estiverem dentro de sua marcação. `<DIV>` marca uma divisão lógica de um documento e é uma formatação bastante usada atualmente.

## 11 Comentários

---

Para fazer algum comentário no código HTML usa-se a seguinte notação:

```
<!-- blá blá blá -->
```

## 12 Listas em HTML

---

Há vários tipos de listas em HTML, sendo estas as mais usadas e corretamente apresentadas pelos *browsers*:

### Listas de Definição

Estas listas são chamadas também “Listas de Glossário”, uma vez que têm o seguinte formato:

```
<DL>  
<DT> termo a ser definido  
<DD> definição  
<DT> termo a ser definido  
<DD> definição  
</DL>
```

Que produz:

```
termo a ser definido  
definição  
termo a ser definido  
definição
```

Este tipo de lista é muito utilizado para diversos efeitos de organização de páginas, por permitir a tabulação do texto. Um exemplo é a lista composta abaixo:

```
<DL>
<DT> Imperadores do Brasil:
<DD> D. Pedro I
    <DL>
    <DD> Nome completo: Pedro de Alcântara Francisco Antônio João
    Carlos Xavier de Paula Miguel Rafael Joaquim José Gonzaga
    Pascoal Cipriano Serafim de Bragança e Bourbon
    </DL>
<DD> D. Pedro II
    <DL>
    <DD> Nome completo: Pedro de Alcântara João Carlos Leopoldo
    Salvador Bibiano Francisco Xavier de Paula Leocádio Miguel
    Gabriel Rafael Gonzaga
    </DL>
</DL>
```

### **Imperadores do Brasil:**

#### **D. Pedro I**

**Nome completo: Pedro de Alcântara Francisco Antônio João Carlos Xavier de Paula Miguel Rafael Joaquim José Gonzaga Pascoal Cipriano Serafim de Bragança e Bourbon**

#### **D. Pedro II**

**Nome completo: Pedro de Alcântara João Carlos Leopoldo Salvador Bibiano Francisco Xavier de Paula Leocádio Miguel Gabriel Rafael Gonzaga**

### **Listas com Marcadores**

São equivalentes às listas com marcadores do MS-Word, por exemplo:

```
<UL>
<LI> item de uma lista
<LI> item de uma lista, que pode ser tão grande quanto se queira,
sem que seja necessário se preocupar com a formatação das margens de
texto
<LI> item
</UL>
```

- **item de uma lista**
- **item de uma lista, que pode ser tão grande quanto se queira, sem que seja necessário se preocupar com a formatação das margens de texto**
- **item**

A diferença entre o resultado da marcação HTML e de um Editor de Textos está na mudança dos marcadores, assinalando os diversos níveis de listas compostas:

```
<UL>
<LI> Documentos básicos
<LI> Documentos avançados
  <UL>
  <LI> formulários
  <UL>
  <LI> CGI
  </UL>
  <LI> contadores
  <LI> relógios
  </UL>
<LI> Detalhes sobre imagens
</UL>
```

- **Documentos básicos**
- **Documentos avançados**
  - **formulários**
    - **CGI**
  - **contadores**
  - **relógios**
- **Detalhes sobre imagens**

### Listas Numeradas

```
<OL>
<LI> item de uma lista numerada
<LI> item de uma lista numerada, que pode ser tão grande quanto se
queira, sem que seja necessário se preocupar com a formatação das
margens de texto
<LI> item de lista numerada
</OL>
```

- 1. item de uma lista numerada**
- 2. item de uma lista numerada, que pode ser tão grande quanto se queira, sem que seja necessário se preocupar com a formatação das margens de texto**
- 3. item de lista numerada**

Estas listas não apresentam numeração em formato 1.1, 1.2, etc., quando compostas apresentam-se da seguinte forma:

- 1. Documentos básicos**
- 2. Documentos avançados**
  - 1. formulários**
    - 1. CGI**
    - 2. contadores**
    - 3. relógios**
  - 2. Detalhes sobre imagens**

Porém, através do atributo TYPE (HTML 3.2), pode-se lidar com a numeração dos itens:

```
<OL TYPE=I>
<LI> Documentos básicos
<LI> Documentos avançados
  <OL TYPE=a>
  <LI > formulários
  <OL TYPE=i>
  <LI>CGI
  </OL>
  <LI> contadores
  <LI> relógios
  </OL>
<LI> Detalhes sobre imagens
</OL>
```

- I. Documentos básicos**
- II. Documentos avançados**
  - a. formulários**
    - i. CGI**
  - b. contadores**
  - c. relógios**
- III. Detalhes sobre imagens**

Ainda segundo HTML 3.2, o atributo START pode indicar o início da numeração da lista:

```
<OL START=4 TYPE=A>
<LI> um item
<LI> outro item
<LI> mais um item
</OL>
```

- D. um item**
- E. outro item**
- F. mais um item**

## Listas e sub-listas

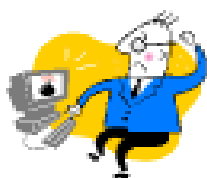
As listas podem ser aninhadas. Por exemplo:

```
<DL>
<DT> termo a ser definido
<DD> definição
<OL>
<LI> item de uma lista numerada
<LI> item de uma lista numerada
<UL>
<LI> item de uma lista
</UL>
<LI> item de uma lista numerada
</OL>
<DT> termo a ser definido
<DD> definição
</DL>
```

**termo a ser definido**  
**definição**

1. **item de uma lista numerada**
2. **item de uma lista numerada**
  - o **item de uma lista**
3. **item de uma lista numerada**

**termo a ser definido**  
**definição**



**Dica:** Se você não está vendo diferença alguma entre as listas comuns e as que têm atributos TYPE, isso se deve ao fato de seu *browser* não estar reconhecendo esses atributos como válidos. Trata-se de um *browser* de versão antiga. Isso deve ser pensado quando usamos atributos mais recentes: **nem todo usuário poderá ver o resultado das novas marcações.**



## 13 Formatação de Textos e Caracteres

---

### Blocos de texto

HTML oferece as seguintes formatações de blocos de texto:

#### **<PRE>**

Apresenta o texto na mesma maneira em que foi digitado, mantendo quebras de linha e tabulações:

```
<pre> uma linha aqui,  
outra ali,  
  etc. </pre>
```

Resulta em:

**uma linha aqui,  
outra ali,  
 etc.**

Uma vez que **<PRE>** mantém o texto original, não se deve forçar espaços com essa marcação dentro de outra marcação que já apresente tabulações e espaços específicos.

#### **<BLOCKQUOTE>**

É usado para citações longas:

```
<blockquote> A massa do Sol arqueia o espaço-tempo de tal maneira  
que, ainda que a Terra siga uma trajetória reta no espaço-tempo  
quadridimensional, parece-nos que se desloca em órbita circular no  
espaço tridimensional. </blockquote> (Stephen W. Hawking, "Uma Breve  
História do Tempo")
```

**A massa do Sol arqueia o espaço-tempo de tal maneira que, ainda que a Terra siga uma trajetória reta no espaço-tempo quadridimensional, parece-nos que se desloca em órbita circular no espaço tridimensional.**

**(Stephen W. Hawking, "Uma Breve História do Tempo")**

#### **<ADDRESS>**

Usado para formatar endereços de E-Mail e referências a autores de documentos, como por exemplo:

```
Envie críticas e sugestões para <address> fulano@xxx.com.br  
</address>
```

Envie críticas e sugestões para *fulano@xxx.com.br*

## 14 Formatação de Frases

---

<B>

Quando disponível no **browser**, é mostrado em **negrito** (em alguns *browsers*, pode aparecer sublinhado).

<I>

*Itálico*

<U>

Sublinhado. Deve ser usado com cuidado, pois confunde-se com a apresentação de *links*.

<STRIKE> ou <S>

Frase ~~riscada~~.

<BIG>

Fonte um pouco maior.

<SMALL>

Fonte um pouco menor.

<SUB>

Frase em estilo <sub>índice</sub>, como em H<sub>2</sub>O, por exemplo.

<SUP>

Frase em estilo <sup>expoente</sup>, como em Km<sup>2</sup>, por exemplo.

## 15 Caracteres Especiais

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

Três caracteres ASCII - **<**, **>**, e **&** têm significados especiais em HTML e são usados dentro de documentos seguindo a correspondência:

Entidade	Caractere
<code>&amp;lt;</code>	<b>&lt;</b>
<code>&amp;gt;</code>	<b>&gt;</b>
<code>&amp;amp;</code>	<b>&amp;</b>

Outras seqüências de *escape* suportam caracteres *ISO Latin1*. Aqui está uma tabela com os caracteres mais utilizados em Português:

Entidade	Caractere	Entidade	Caractere
<code>&amp;aacute;</code>	á	<code>&amp;Aacute;</code>	Á
<code>&amp;acirc;</code>	â	<code>&amp;Acirc;</code>	Â
<code>&amp;agrave;</code>	à	<code>&amp;Agrave;</code>	À
<code>&amp;atilde;</code>	ã	<code>&amp;Atilde;</code>	Ã
<code>&amp;ccedil;</code>	ç	<code>&amp;Ccedil;</code>	Ç
<code>&amp;eacute;</code>	é	<code>&amp;Eacute;</code>	É
<code>&amp;ecirc;</code>	ê	<code>&amp;Ecirc;</code>	Ê
<code>&amp;iacute;</code>	í	<code>&amp;Iacute;</code>	Í
<code>&amp;oacute;</code>	ó	<code>&amp;Oacute;</code>	Ó
<code>&amp;ocirc;</code>	ô	<code>&amp;Ocirc;</code>	Ô
<code>&amp;otilde;</code>	õ	<code>&amp;Otilde;</code>	Õ
<code>&amp;uacute;</code>	ú	<code>&amp;Uacute;</code>	Ú
<code>&amp;uuml;</code>	ü	<code>&amp;Uuml;</code>	Ü

Como vemos, as seqüências de *escape* são sensíveis à caixa. Os editores de HTML fazem essa tradução automaticamente.

Existem alguns símbolos que vêm sendo incorporados ao conjunto de caracteres reconhecidos em HTML. Por exemplo, `&copy;`, que é o símbolo ©, `&reg;` para ®, e `&sect;` para §.

Também pode-se usar seqüências com códigos ASCII, por exemplo:

`&#191;` Qué pasa, señor?

`¿`Qué pasa, señor?



**Dica: Por que usar essas formatações?** Por causa de um problema técnico relativo à *transmissão* dos caracteres especiais de várias línguas. Chama-se *conjunto de caracteres* uma representação digital de texto. O conjunto de representação amplamente utilizado é o ASCII. O ASCII padrão tem 7 bits, porém são usadas versões particulares desse padrão - o ASCII americano, ASCII português, etc., que utilizam 8 bits. Se geramos um texto em ASCII de 8 bits e, na transmissão, esse texto é tratado por um sistema que “compreende” apenas o

ASCII de 7 bits, os caracteres especiais serão transformados em caracteres diferentes, pela perda do oitavo bit. Os usuários da Internet já devem ter sofrido experiência semelhante com mensagens de e-mail, que chegam com caracteres estranhos, ou páginas da Web com textos em japonês ou russo.

## 16 Marquee

---

É possível obter o efeito de animação de texto, através da formatação <MARQUEE>.

```
<MARQUEE BEHAVIOR=efeito> Texto </MARQUEE>
```

Atributos de largura e direção do efeito permitem diversas apresentações diferentes, por exemplo:

```
<MARQUEE BEHAVIOR=SCROLL WIDTH=30%> Texto </MARQUEE>
```

```
<MARQUEE BEHAVIOR=SLIDE DIRECTION=RIGHT> Texto </MARQUEE>
```

```
<MARQUEE BEHAVIOR=SLIDE DIRECTION=LEFT> Texto </MARQUEE>
```

**Atenção:** O efeito Marquee só é executado no Internet Explorer e em versões recentes do Netscape - e de maneiras diferentes.

## 17 Ligações (uso de Links)

---

Com HTML é possível fazermos ligações de uma região de texto (ou imagem) a um outro documento (ou a outra parte do próprio documento). Você já deve ter visto em alguma página internet exemplos dessas ligações: o *browser* destaca essas regiões e imagens do texto, indicando que são ligações de hipertexto - também chamadas *hypertext links* ou *hyperlinks* ou simplesmente *links*, onde normalmente, o mouse vira uma “mãozinha” e ao clicar “chamamos” (abrimos) um outro documento, página web ou figura por exemplo.

Para inserir um link em um documento, utilizamos a etiqueta `<A>`, da seguinte forma:

```
<A HREF = "arquivo_destino"> âncora </A>
```

Onde:

**arquivo\_destino:** é o endereço do documento de destino, da página ou imagem a qual queremos abrir.

**âncora:** é o texto ou imagem que servirá de ligação hipertexto do documento sendo apresentado para o documento de destino.

### Atributos

`<A>` tem vários atributos que são utilizados de acordo com a ação associada ao link. Os mais usados são apresentados a seguir:

**HREF:** Indica o arquivo de destino da ligação de hipertexto.

**TARGET:** Indica o *frame* em que será carregado o `arquivo_destino`. Maiores detalhes na seção sobre *frames*.

**NAME:** Marca um **indicador**, isto é, uma região de um documento como destino de uma ligação. Maiores detalhes na seção sobre indicadores.

### Caminhos

Os links podem estar indicados como caminhos relativos ou absolutos.

#### **Caminho Relativo**

O caminho relativo pode ser usado sempre que queremos **fazer referência a um documento armazenado no mesmo servidor do documento atual**.

Através do campo de endereços do *browser*, é possível identificar se um documento (página) que está sendo visualizado está dentro de algum diretório (pasta). Como por exemplo se estivermos em um *browser* acessando a página da FACCAT e vemos o seguinte endereço: `http://www.faccat.br/apostilas/apostila_de_informatica.html`. O que podemos concluir é que o documento que está sendo visualizado no momento, chamado **apostila\_de\_informatica.html**, está localizado dentro de um diretório (pasta) chamado **apostilas** do servidor **www.faccat.br**.

Então para escrevermos por exemplo um link deste documento (apostila\_de\_informatica.html) para um documento chamado **doc.html** que está localizado no diretório **/apostilas/documentos/** do mesmo servidor [www.faccat.br](http://www.faccat.br), tudo que precisamos fazer é escrever:

```
<A HREF="documentos/doc.html"> Exemplo de Caminho Relativo </A>
```

Para usar *links* com caminhos relativos é preciso, portanto, conhecer a estrutura do diretório do servidor no qual estamos trabalhando, pois devemos indicar todo o caminho onde está o documento no qual estamos nos referindo no *link*. Quando há alguma dúvida, o melhor é usar o caminho absoluto.

## **Caminho Absoluto**

Utilizamos o caminho absoluto quando desejamos referenciar um documento que esteja em outro servidor, por exemplo:

```
<A HREF="http://www.faccat.br"> FACCAT - Faculdades de Taquara </A>
```

Que oferece um *link* **FACCAT - Faculdades de Taquara** que ao ser clicado com o mouse abrirá a página cujo endereço é <http://www.faccat.br>.

Com a mesma sintaxe, é possível escrever *links* para qualquer servidor de informações da Internet.

## **Indicadores**

O atributo NAME permite indicar um trecho de documento como **ponto de chegada** de uma ligação hipertexto.

A formatação: 

```
<A NAME="inicio"> Indicadores (uso de links) </A>
```

faz com que a âncora **Indicadores (uso de links)** seja o destino de um link.

Se escrevermos:

```
<A HREF="#inicio"> Topo do Documento </A>
```

teremos uma ligação hipertexto para um trecho deste mesmo documento:

## 18 Inserção de Imagens

---

O elemento **IMG** insere imagens que são apresentadas junto com os textos. Um atributo SRC deve estar presente, da seguinte forma:

```
<IMG SRC="URL_imagem">
```

Onde **URL\_imagem** é o URL do arquivo que contém a imagem que se quer inserir; pode ser referenciada uma imagem que esteja em um outro servidor (o que, logicamente, **não é conveniente**).

Assim, escrevendo:

```
<IMG SRC = "nome_da_figura.extensão">
```

ou, se ela estiver dentro de uma pasta no mesmo servidor:

```
<IMG SRC = "nome_da_pasta/nome_da_figura.extensão">
```

Por exemplo, se quisermos inserir na página uma imagem chamada *arvore.gif* localizada no mesmo servidor e na mesma pasta, a *tag* seria assim:

```
<IMG SRC = "arvore.gif">
```

As imagens usadas na Web, normalmente são armazenadas em arquivos com as seguintes extensões: **.gif**, **.jpg** (ou **.jpeg**), **.png**, **.bmp**.

### Atributos Básicos de Imagem

#### → ALT

Indica um texto alternativo, descrevendo brevemente a imagem, que é apresentado no lugar da imagem nos *browsers*. Ou quando se desabilita o carregamento de imagens em *browsers* gráficos. É recomendável que esteja sempre presente. Também aparecerá ao “passar o mouse sobre a imagem”.

```
<IMG SRC="URL_imagem" ALT="descrição_da_imagem">
```

Dessa forma:

```
<IMG SRC="newred.gif" ALT="Novo!!!">
```

É apresentado nos *browsers* gráficos assim: **New!** e, nos *browsers* texto, assim: [Novo!!!]. E ao passar o mouse sobre a figura também aparecerá a descrição Novo!!!

### → WIDTH e HEIGHT

Atributos de dimensão da imagem, em *pixels*<sup>4</sup>. Grande parte dos editores HTML coloca automaticamente os valores destes atributos, quando indicamos a inserção de uma imagem.

```
<IMG SRC="URL_imagem" ALT="descrição" WIDTH="largura"  
HEIGHT="altura">
```

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 a elas.

### → BORDER

Quando uma frase é marcada como âncora de um link, ela se apresenta sublinhada; quando uma imagem faz as 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.

\* Se quisermos uma borda mais larga:

```
<A HREF="URL"> <IMG SRC="imagem" ALT="descrição" BORDER=4> </A>
```

\* Se quisermos uma imagem sem borda:

```
<A HREF="URL"> <IMG SRC="imagem" ALT="descrição" BORDER=0>  
</A>
```

Essa borda pode ser apresentada também em imagens que não são âncora de *links*. Basta aplicar, por exemplo, a formatação:

```
<IMG SRC="figura1.gif" ALT="Minha Foto" BORDER=2>
```

---

<sup>4</sup> *Pixel*: O pixel, ou ponto, é a menor unidade que compõe uma imagem digital, ele contém os atributos de cor de cada ponto, a maior ou menor quantidade de *pixels* em uma área é que determinam a resolução da imagem, bem como seu maior ou menor tamanho em *bytes*. São pontos por centímetro quadrado. Quanto maior o número de *pixels* maior a qualidade da imagem e maior será o tamanho do arquivo também.



Assim, é possível dar mais destaque a uma imagem, sem ser necessário editá-la:



Foto Original



Foto com Borda gerada pelo HTML

## → ALIGN

```
<IMG SRC="imagem.extensão" ALT="descrição"  
ALIGN=alinhamento_desejado>
```

Existem também atributos de alinhamento, que produzem os seguintes resultados:

**ALIGN=TOP:** Alinha o texto adjacente com o topo da imagem, embora com linhas compridas o resultado não seja muito bom.

**ALIGN=MIDDLE:** Alinha o texto adjacente com o meio da imagem, embora com linhas compridas o resultado não seja muito bom.

**ALIGN=BOTTOM:** Alinha o texto adjacente com a parte de baixo da imagem (default).

**ALIGN=RIGHT:** Alinha imagem à direita, e tudo o que houver ao redor (texto, outras imagens) a partir do topo da imagem.

**ALIGN=LEFT:** Alinha imagem à esquerda, e tudo o que houver ao redor (texto, outras imagens) a partir do topo da imagem.

→ Para ter duas imagens, uma em cada margem, numa mesma linha, escreva:

```
<IMG ALIGN=LEFT SRC="imagem.extensão" ALT="descrição"> <IMG  
ALIGN=RIGHT SRC="imagem.extensão" ALT="descrição"> ...e pode-se  
escrever à vontade entre as imagens!
```

Isto resulta em:



...e pode-se escrever à vontade entre  
as imagens!



Um detalhe surgido com o alinhamento de imagens foi a necessidade de se liberar o texto desse alinhamento, ou seja:

Suponhamos um texto mais ou menos curto, que desejamos colocar com a imagem ilustrativa, mas gostaríamos que este trecho já estivesse abaixo da imagem! De acordo com o comprimento da primeira frase, não seria possível usar o alinhamento TOP.

Para conseguir isso, seria necessário incluir diversos <BR> consecutivos, inserindo linhas em branco; mesmo assim, o resultado final poderia ser bem pouco elegante. Surgiu, então, o atributo CLEAR para <BR>.

Com esse atributo, podemos, por exemplo ter um texto posicionado no ponto em que a margem direita fica livre, com <BR CLEAR=RIGHT> ou no ponto em que a margem esquerda fica livre, com <BR CLEAR=LEFT>

Dessa maneira, podemos controlar bem a posição relativa dos textos. Também pode-se posicionar o texto no ponto em que ambas as margens estão livres. Isso é conseguido com <BR CLEAR=ALL>

E, assim, vimos tudo sobre quebras de linha depois de imagens! ☺

## 19 Molduras de Imagem

---

Para melhorar ainda mais a apresentação das imagens junto com os textos, foram desenvolvidos atributos de moldura. Estes atributos definem o espaço - vertical e horizontal - deixado entre as imagens e os textos circundantes:

```
<IMG SRC="imagem.extensão" VSPACE=espaço_vertical>
```

```
<IMG SRC=" imagem.extensão " HSPACE=espaço_horizontal>
```

### Outros exemplos:

```
<IMG SRC="foto.gif" WIDTH="148" HEIGHT="95" ALIGN=left VSPACE="30">
```

```
<IMG SRC="foto.gif" WIDTH="160" HEIGHT="71" ALIGN=right HSPACE="30">
```

### Abaixo, temos um exemplo com os dois atributos, através da formatação:

```
<IMG SRC="foto.gif" ALIGN="LEFT" WIDTH="63" HEIGHT="68"  
HSPACE="20" VSPACE="20">
```

## 20 Tabelas

---

A formatação de tabelas foi adotada bem antes de sua inclusão na definição de HTML. A manipulação de tabelas, mesmo em editores, é trabalhosa. A maior diferença entre tabelas em HTML e em editores como o MS-Word, entretanto, é o fato das tabelas em HTML serem definidas apenas **em termos de linhas e não de colunas**. Mas isso será percebido no decorrer deste capítulo.

As tabelas foram uma grande conquista para os autores de documentos para a *Web*. Com elas é possível, por exemplo, termos páginas organizadas em colunas, sendo uma delas reservada aos *links* de navegação dentro de cada seção.

Tabelas implementam um conceito importante de *layout*: as “grades”, segundo as quais organizamos textos e ilustrações de maneira harmoniosa.

Como já foi possível perceber, as tabelas contêm textos, listas, parágrafos, imagens, formulários e várias outras formatações - inclusive outras tabelas. Novas versões de HTML e de *browsers* populares vêm acrescentando diversos atributos às tabelas, e nosso objetivo aqui é saber lidar com a maioria desses recursos disponíveis.

### Elementos básicos de tabelas

`<TABLE>...</TABLE>` delimita uma tabela. Um atributo básico é **BORDER**, que indica a apresentação da borda.

```
<TABLE BORDER="borda" >
...
</TABLE>
```

### Títulos, linhas e elementos

`<CAPTION>...</CAPTION>`  
define o título da tabela

`<TR>...</TR>`  
delimita uma linha

`<TH>...</TH>`  
define um cabeçalho para colunas ou linhas (dentro de `<TR>`)

`<TD>...</TD>`  
delimita um elemento ou célula (dentro de `<TR>`)

## Uma tabela simples:

```
<TABLE BORDER=4>
<CAPTION> Primeiro exemplo </CAPTION>
<TR> <TH> Coluna 1 </TH> <TH> Coluna 2 </TH> </TR>
<TR> <TD> linha1, coluna 1 </TD> <TD> linha 1, coluna 2 </TD>
</TR>
<TR> <TD> linha 2, coluna 1 </TD> <TD> linha 2, coluna 2 </TD>
</TR>
</TABLE>
```

Primeiro exemplo

Coluna 1	Coluna 2
linha1, coluna 1	linha 1, coluna 2
linha 2, coluna 1	linha 2, coluna 2

## Títulos compreendendo mais de uma coluna ou linha

É possível englobar colunas e linhas, através dos atributos **COLSPAN** (para colunas) e **ROWSPAN** (para linhas):

```
<TABLE BORDER=1>
<TR><TH COLSPAN=2>Colunas 1 e 2</TH></TR>
<TR><TD>linha1, coluna 1</TD><TD> linha 1, coluna 2</TD></TR>
<TR><TD>linha 2, coluna 1</TD><TD>linha 2, coluna 2</TD></TR>
<TR><TH ROWSPAN=3>3 linhas</TH><TD>uma linha</TD></TR>
<TR><TD>duas linhas</TD></TR>
<TR><TD>tres linhas</TD></TR>
</TABLE>
```

Colunas 1 e 2	
linha1, coluna 1	linha 1, coluna 2
linha 2, coluna 1	linha 2, coluna 2
<b>3 linhas</b>	uma linha
	duas linhas
	tres linhas

Neste exemplo, vemos que o cabeçalho **Colunas 1 e 2** compreende duas colunas (COLSPAN=2); o cabeçalho **3 linhas** compreende, por sua vez, 3 linhas (ROWSPAN=3).

## Tabelas sem borda

As páginas deste tutorial foram construídas com tabelas sem borda. Para tanto, foi empregada a seguinte declaração:

```
<TABLE BORDER="0">
...
</TABLE>
```



**Dica: Pior que os acentos...** A formatação de tabelas é complicada, e o texto fonte chega a ser quase ininteligível quando montamos tabelas complexas e fazemos uso de seus diversos atributos. A melhor opção, sem dúvida, é usar os editores WYSIWYG. Alguns editores de modo texto têm uma interface gráfica que ajuda na criação de tabelas, mas a edição de tabelas já existentes precisa ser feita à mão.

## Extensões de Tabelas

Diversas extensões de tabelas possibilitam a apresentação de efeitos muito bons nas páginas.

### Cor de fundo

```
<TABLE BORDER=5 CELLSPACING=5 CELLPADDING=10 BGCOLOR="#E1FFD9">
```

janeiro	fevereiro	março
abril	maio	junho

```
<TABLE BORDER=5 CELLSPACING=5 CELLPADDING=10>
<TR><TD BGCOLOR="#E1FFD9">janeiro</TD><TD>fevereiro</TD>
<TD BGCOLOR="#E1FFD9">março</TD></TR>
<TR><TD>abril</TD><TD BGCOLOR="#E1FFD9">maio</TD><TD>junho</TD></TR>
</TABLE>
```

janeiro	fevereiro	março
abril	maio	junho

## Cor de borda

```
<TABLE BORDER=5 CELLSPACING=5 CELLPADDING=10 BGCOLOR="#E1FFD9"
BORDERCOLOR="#00FF00">
```

janeiro	fevereiro	março
abril	maio	junho

```
<TABLE BORDER="1" CELLSPACING="0" CELLPADDING=10
BORDERCOLOR="#00FF00">
<TR>
<TD bgcolor="#E1FFD9">janeiro</TD><TD>fevereiro</TD>
<TD bgcolor="#E1FFD9">março</TD>
</TR>
<TR>
<TD>abril</TD><TD bgcolor="#E1FFD9">maio</TD><TD>junho</TD>
</TR>
</TABLE>
```

janeiro	fevereiro	março
abril	maio	junho

## 21 Frames

---

Os *frames* são divisões da tela do *browser* em diversas telas (ou “quadros”). Com isso, torna-se 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 (menus).

É muito fácil colocar *frames* em páginas, porém, nem todos os usuários gostam deles, pois nem sempre a navegação é fácil, além de problemas para a impressão e a marcação dos documentos interiores aos *frames* nos *bookmarks*. A alternativa natural para os *frames* são as tabelas.

Uma página com *frames* tem um texto fonte semelhante a:

```
<HTML>
<HEAD><TITLE>Assunto X</TITLE></HEAD>
<FRAMESET COLS="20%, 80%">
  <FRAME SRC="indice1.html">
  <FRAME SRC="apresenta.html" NAME="principal">
  <NOFRAME>
  <BODY>
  <H2>Bem-vindo à página do assunto X!</h2>
  <P>
  Blá blá blá blá blá
  blá blá blá blá blá
  </BODY>
  </NOFRAME>
</FRAMESET>
</HTML>
```

A parte **FRAMESET** define a divisão da página em “quadros”. Neste exemplo, a página será dividida em duas colunas, sendo a primeira com 20% do tamanho da tela, e a segunda coluna com os restantes 80% da tela.

Dentro da formatação de **FRAMESET**, temos os **FRAME SRC**, que são referências às páginas que serão mostradas nos *frames* definidos

Assim, no código acima vemos que a página **indice1.html** será mostrada na primeira coluna (que ocupará 20% da tela), e a página **apresenta.html** será mostrada na segunda (ocupando 80% da tela).

A formatação de *frames* inclui também uma parte **NOFRAME**, que é mostrada normalmente pelos *browsers* que não suportam sua apresentação.

Material completo com exemplos e detalhes sobre HTML que  
serviu de fonte para elaboração desta apostila em:  
<http://www.icmc.usp.br/ensino/material/html/intro.html>  
Autora: **Maria Alice Soares de Castro**