

**Apostila**

**<HTML>**

**Aprendendo a Criar Páginas para Web**

# Sobre a Apostila

Esta apostila foi desenvolvida para o ensino da criação de páginas web. Como a HTML é a linguagem básica de criação de páginas, esta apostila enfoca o ensino da HTML.

Hoje em dia, há uma grande variedade de editores que facilitam a criação de páginas e que não exigem do usuário um grande conhecimento de HTML. No entanto, esta apostila não usará nenhum destes editores, pois o objetivo é que a pessoa aprenda a criar páginas codificando os comandos HTML, usando para isso até mesmo um editor como o Bloco de Notas.

Esta apostila foi planejada para servir como um material de estudo para autoditadas, como também para ministrar cursos em empresas ou escolas e até mesmo como um guia de referência para desenvolvedores que já conhecem a HTML.

## **Público Alvo**

- Qualquer pessoa que deseja aprender HTML;
- Desenvolvedores que já trabalham com HTML;
- Professores ou instrutores que desejem usar esta apostila como material didático.

## **Pré-requisitos**

Não é necessário ter conhecimento prévio de HTML, mas é importante conhecer o Browser (navegador) e um editor de texto de sua preferência, como por exemplo, o Bloco de Notas.

## **Nível**

Esta apostila aborda tópicos que vão do nível BÁSICO ao INTERMEDIÁRIO.

# Introdução

Esta apostila foi criada para ser um recurso de aprendizado e consulta à linguagem HTML, para qualquer usuário de informática que deseja criar páginas para a internet.

Para facilitar o estudo e pesquisa de informação, ela foi dividida em tópicos, os quais foram apresentados em forma de capítulos. Os capítulos foram estruturados da seguinte forma:

1. **Conhecendo a HTML:** Os conceitos básicos e estrutura da linguagem HTML são apresentados neste capítulo.
2. **Formatação de Textos:** Aqui é explicada a formatação de textos através de comandos (tags) html específicos para esse fim. Algumas das formatações possíveis são: negrito, sublinhado, itálico, tipo, tamanho e cor da fonte, listas, entre outras.
3. **Links:** A funcionalidade mais importante na HTML é a possibilidade de criar elos (links) entre documentos e neste capítulo é mostrado como usar este grande recurso.
4. **Imagens:** O objetivo deste capítulo é ensinar como usar imagens nos documentos HTML. É visto os tipos de imagens mais usadas, os formatos GIF e JPEG, e um pouco de imagens mapeadas. Como criar imagens com boa qualidade para documentos HTML é um assunto extenso e por isso não será abordado.
5. **Tabelas:** Um grande recurso para se produzir documentos com uma boa diagramação e apresentação é o uso de tabelas. Este é um dos principais capítulos e é justamente aqui que está a transição entre o nível básico e o nível intermediário na HTML.
6. **Formulários:** Recurso poderoso da HTML que permite gerar interação entre o usuário e o documento. Depois que este recurso foi criado a HTML se difundiu e a internet passou a ser usada também para fins comerciais. Neste capítulo, este recurso será ensinado e exemplificado.
7. **Frames:** Este recurso permite exibir mais de um documento por tela. Para isso, a tela é dividida em partes chamadas frames. Este recurso é apresentado em detalhes neste capítulo.
8. **Outros Comandos:** Neste capítulo são apresentados outros comandos interessantes da HTML, como exibir áudio e vídeo, atualização automática de documento, etc.
9. **Caracteres Especiais:** Uma tabela de códigos de caracteres para serem usados em documentos HTML.
10. **Tabela de Cores:** Tabela com o código e nome de cores para usar nos documentos HTML.
11. **Referências:** Fontes de informação e pesquisa utilizadas para apoio ao desenvolvimento desta apostila.



# Sumário

<b>1. CONHECENDO A HTML.....</b>	<b>6</b>
1.1. COMANDOS HTML.....	6
1.2. ESTRUTURA BÁSICA DE UM DOCUMENTO.....	7
<b>2. FORMATAÇÃO DE TEXTOS.....</b>	<b>8</b>
2.1. TÍTULOS E SUBTÍTULOS.....	8
2.2. PARÁGRAFOS E QUEBRA DE LINHA.....	8
2.3. ESTILOS DE TEXTO.....	11
2.4. DEFININDO FONTES.....	11
2.5. LINHAS HORIZONTAIS E ALINHAMENTO DE ELEMENTOS.....	12
2.6. LISTAS.....	15
2.6.1. LISTAS NÃO ORDENADAS.....	16
2.6.2. LISTAS ORDENADAS.....	17
2.6.3. LISTAS DESCRITIVAS.....	18
<b>3. LINKS.....</b>	<b>20</b>
3.1. LINKS PARA ARQUIVOS LOCAIS.....	20
3.2. LINKS PARA ARQUIVOS EM OUTROS DIRETÓRIOS.....	21
3.3. LINKS PARA ARQUIVOS DE OUTROS SERVIDORES.....	23
3.4. LINKS PARA E-MAIL.....	23
3.5. LINKS PARA A MESMA PÁGINA - ÂNCORA.....	23
<b>4. IMAGENS.....</b>	<b>26</b>
4.1. IMAGENS COMO LINK.....	27
4.2. IMAGENS COMO PADRÃO DE FUNDO.....	28
4.3. IMAGENS TRANSPARENTES.....	28
4.4. IMAGENS MAPEADAS.....	28
4.5. IMAGENS ANIMADAS.....	29
<b>5. TABELAS.....</b>	<b>31</b>
5.1. TABELA SIMPLES.....	31
5.2. MODIFICANDO A APARÊNCIA DA TABELA - ATRIBUTOS.....	32
5.3. ATRIBUTOS DE LINHAS E CÉLULAS.....	33
5.4. MESCLANDO CÉLULAS.....	35
<b>6. FORMULÁRIOS.....</b>	<b>37</b>
6.1. CRIANDO UM FORMULÁRIO.....	37
6.2. TAGs DE FORMULÁRIO.....	38
<b>7. FRAMES.....</b>	<b>43</b>
7.1. CRIANDO FRAMES.....	43
7.2. LINKS COM FRAMES.....	46
<b>8. OUTROS COMANDOS.....</b>	<b>48</b>
8.1. REFRESH PAGE.....	48
8.2. AUDIO E VÍDEO.....	48
8.3. TEXTO ROLANTE.....	49
<b>9. CARACTERES ESPECIAIS.....</b>	<b>52</b>
<b>10. TABELA DE CORES.....</b>	<b>53</b>
<b>11. REFERÊNCIAS.....</b>	<b>54</b>

# 1. CONHECENDO A HTML

HTML significa **HyperText Markup Language** e é a linguagem de descrição de documentos usada na Web. Sua principal característica está definida em seu nome. A Html através de tags, posiciona e coloca hipertexto no documento.

Por sua vez, o hipertexto é a base da informação que, além da função comum de texto, permite ligações (links) para outros locais de um documento e também para documentos remotos (páginas, figuras, frames e tudo mais que um navegador pode ler ou suportar).

Periodicamente a Html ganha uma nova versão com mais recursos. Quem cria as novidades desta linguagem são as pessoas e empresas envolvidas, principalmente os desenvolvedores de softwares para navegação (Netscape, Microsoft, etc).

Todo este processo sofre a colaboração e a administração do **W3C - World Wide Web Consortium**.

## 1.1. COMANDOS HTML

Os comandos HTML são representados no texto através de **rótulos** (ou *tags*). O visualizador (Browser) interpreta como rótulos todos os elementos HTML *válidos* que estejam entre os sinais “menor que” (<) e “maior que” (>).

A maioria das TAGs HTML possuem um **rótulo inicial** e um **rótulo final**, envolvendo o texto que é marcado por eles. A sintaxe básica

**<TAG>** Texto marcado pela Tag **</TAG>**

Observe que o **rótulo de fechamento** tem o mesmo nome que o **rótulo inicial**, mas é precedido por uma barra ("/"). Algumas TAGs podem ter um ou mais **atributos**, que definem alguma característica especial. Os atributos, quando presentes, aparecem no rótulo inicial separados por espaços, logo após o nome da TAG.

Exemplos de tags HTML com atributos:

⇒ Define uma imagem de fundo para a página:

- **<body background="fundo.gif">... corpo do documento ...</body>**

⇒ Define uma borda para a tabela:

- **<table border> ... conteúdo e rótulos de tabela ... </table>**

⇒ Define uma linha horizontal sem sombra, com largura de 50% da página, 10 pixels de espessura e alinhada pela esquerda:

- **<hr noshadow width=50% size=10 align=left>**

---

## 1.2. ESTRUTURA BÁSICA DE UM DOCUMENTO

A estrutura básica de um documento HTML é a seguinte:

```
<HTML>
<HEAD>
  <TITLE> Título do documento </TITLE>
</HEAD>
<BODY>
  Corpo da página.
</BODY>
</HTML>
```

Vamos as explicações de cada um desses comandos:

- ⇒ **<HTML>...</HTML>** : Indicam o início e o fim do documento, todo o resto deve estar entre esses dois comandos.
- ⇒ **<HEAD> </HEAD>** : Delimitam a seção de cabeçalho do documento e nelas estão contidas as informações de configuração da página.
- ⇒ **<TITLE> </TITLE>** : Indicam o título do documento, que será apresentado na barra superior do Browser. Estes comandos devem estar sempre entre o comando acima.
- ⇒ **<BODY>...</BODY>** : Indicam o corpo do documento. Entre esses comandos estará a sua página em si. O texto, imagens e tudo que será apresentado pelo browser.

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

**Sintaxe:**

```
<BODY BGCOLOR="#rrggbb" TEXT="#rrggbb" LINK="#rrggbb" ALINK="#rrggbb" VLINK="#rrggbb"
BACKGROUND="URL" MARGINHEIGHT="0" MARGINWIDTH="0" LEFTMARGIN="0" TOPMARGIN="0">
```

**Onde:**

- ⇒ **BGCOLOR** - cor de fundo (padrão: cinza ou branco)
- ⇒ **TEXT** - cor dos textos da página (padrão: preto)
- ⇒ **LINK** - a cor dos links (padrão: azul)
- ⇒ **ALINK** - cor dos links, quando acionados (padrão: vermelho)
- ⇒ **VLINK** - cor dos links, depois de visitados (padrão: azul escuro ou roxo)
- ⇒ **BACKGROUND** - indica o URL da imagem a ser replicada no fundo da página, como uma marca d'água.
- ⇒ **LEFTMARGIN** - margem esquerda da página(MS Internet Explorer).
- ⇒ **TOPMARGIN** - margem superior da página(MS Internet Explorer).
- ⇒ **MARGINHEIGHT** - margem superior(Netscape).
- ⇒ **MARGINWIDTH** - margem esquerda(Netscape).

## 2. FORMATAÇÃO DE TEXTOS

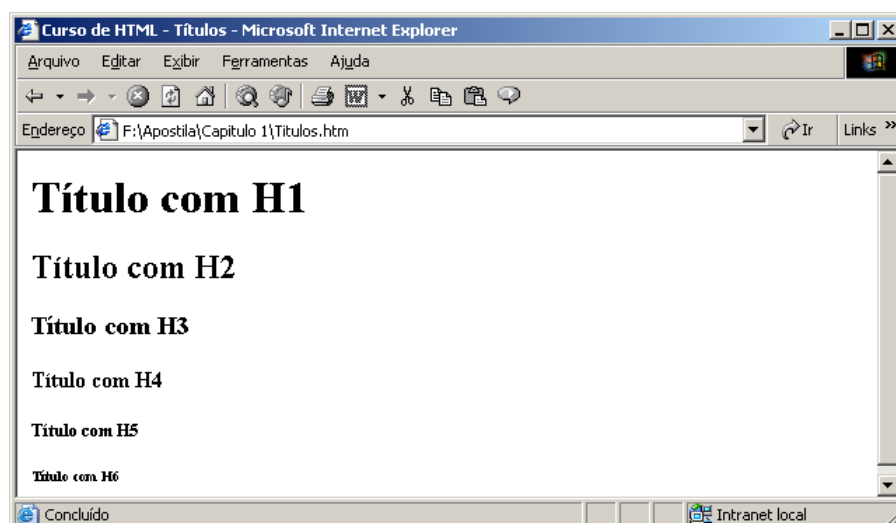
### 2.1. TÍTULOS E SUBTÍTULOS

Para demarcar títulos e subtítulos, use os TAGs de HEADER (H1 a H6).

Exemplo: Titulos.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Titulos</TITLE>
</HEAD>
<BODY>
  <H1>Título com H1</H1>
  <H2>Título com H2</H2>
  <H3>Título com H3</H3>
  <H4>Título com H4</H4>
  <H5>Título com H5</H5>
  <H6>Título com H6</H6>
</BODY>
</HTML>
```

Como fica no Browser:



### 2.2. PARÁGRAFOS E QUEBRA DE LINHA

Uma página HTML usa comandos especiais para definir a quebra de um parágrafo ou de uma linha. Vejamos alguns:



## TAG <P>

É responsável pela quebra de parágrafos, inserindo automaticamente uma linha em branco. O uso da tag </P> é opcional. Também é possível alinhar os parágrafos através desta tag.

### Sintaxe:

<P ALIGN="posição">Texto do parágrafo</P>

### Onde:

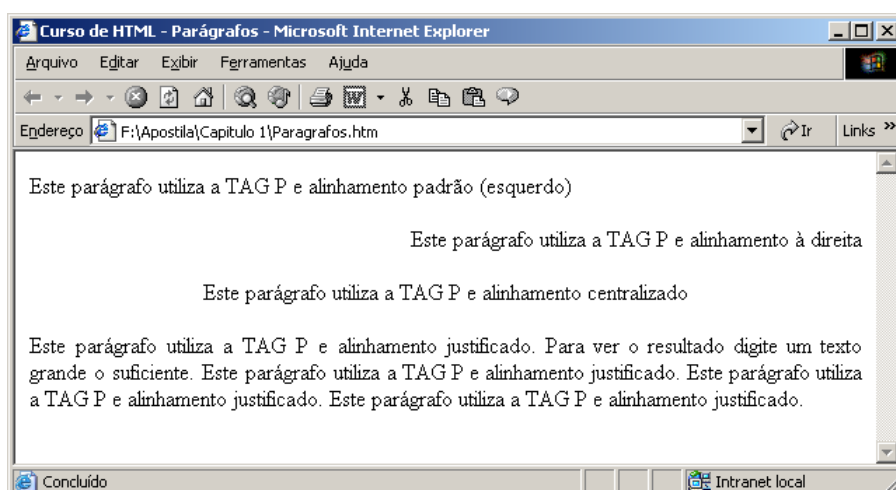
⇒ posição pode ser:

- **left**: alinhamento à esquerda;
- **center**: centralizado;
- **right**: alinhamento à direita.
- **justify**: justificado.

### Exemplo: Paragrafos.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Parágrafos</TITLE>
</HEAD>
<BODY>
  <P>Este parágrafo utiliza a TAG P e alinhamento padrão (esquerdo)
  <P ALIGN="Right">Este parágrafo utiliza a TAG P e alinhamento à direita
  <P ALIGN="Center">Este parágrafo utiliza a TAG P e alinhamento centralizado
  <P ALIGN="Justify">Este parágrafo utiliza a TAG P e alinhamento justificado.
    Para ver o resultado digite um texto grande o suficiente.
    Este parágrafo utiliza a TAG P e alinhamento justificado.
    Este parágrafo utiliza a TAG P e alinhamento justificado.
    Este parágrafo utiliza a TAG P e alinhamento justificado.
</BODY>
</HTML>
```

### Como fica no Browser:



## TAG <BR>

É responsável pela quebra de linha. Ela finaliza a linha atual e passa para a próxima. Esta TAG não necessita da TAG </BR>.

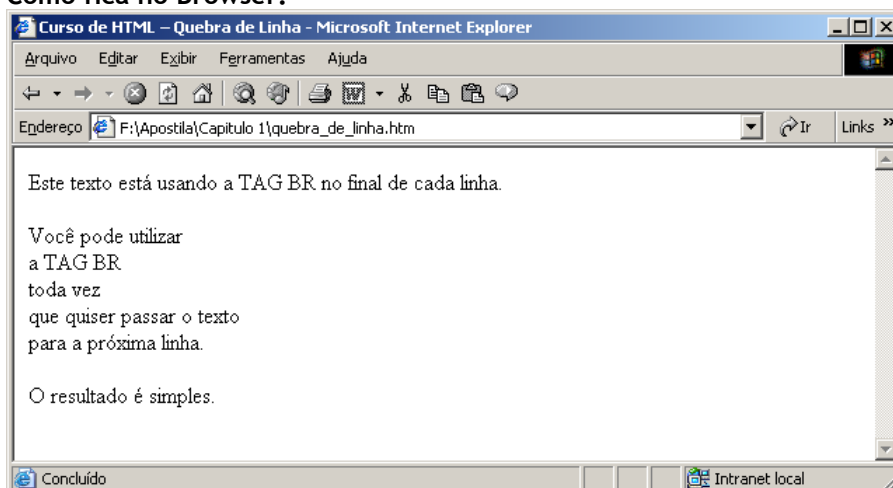
### Sintaxe:

Texto da linha<BR>

### Exemplo: Quebra\_de\_Linha.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Quebra de Linha</TITLE>
</HEAD>
<BODY>
  Este texto está usando a TAG BR no final de cada linha.<BR><BR>
  Você pode utilizar<BR> a TAG BR<BR> toda vez<BR> que quiser passar o
  texto<BR> para a próxima linha.<BR><BR>
  O resultado é simples.
</BODY>
</HTML>
```

### Como fica no Browser:



## TAG <PRE>

Permite que se insira em uma página um parágrafo pré-formatado, isto é, conforme foi digitado no editor.

### Sintaxe:

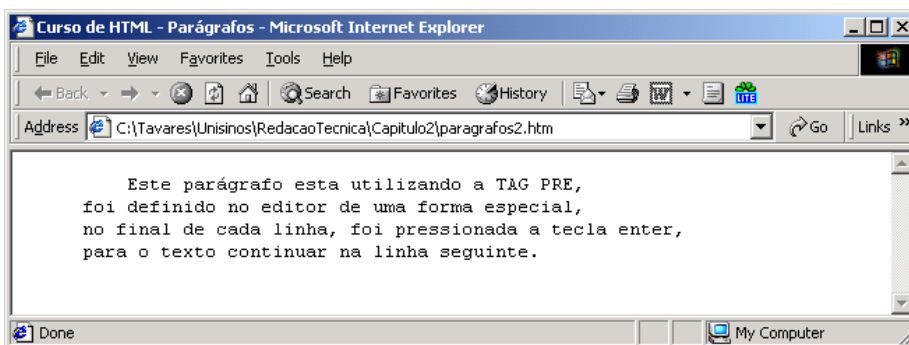
<PRE>Texto do parágrafo</PRE>

### Exemplo: Paragrafos2.htm

```
<HTML>
<HEAD>
```

```
<TITLE>Curso de HTML - Parágrafos</TITLE>
</HEAD>
<BODY>
  <PRE>
    Este parágrafo esta utilizando a TAG PRE,
    foi definido no editor de uma forma especial,
    no final de cada linha, foi pressionada a tecla enter,
    para o texto continuar na linha seguinte.
  </PRE>
</BODY>
</HTML>
```

Como fica no Browser:



## 2.3. ESTILOS DE TEXTO

Como em editores de texto, o HTML também permite alterar o estilo do texto. Vejamos alguns estilos:

- ⇒ **<B>...</B>** - Aplica o estilo **negrito**.
- ⇒ **<I>...</I>** - Aplica o estilo *itálico*.
- ⇒ **<U>...</U>** - Aplica o estilo sublinhado (nem todos os browser o reconhecem).
- ⇒ **<SUP>...</SUP>** - Faz com que o texto fique <sup>sobrescrito</sup>.
- ⇒ **<SUB>...</SUB>** - Faz com que o texto fique <sub>subscrito</sub>.
- ⇒ **<BIG>...</BIG>** - Aumenta a fonte e atribui negrito
- ⇒ **<SMALL>...</SMALL>** - Reduz e altera a fonte
- ⇒ **<TT>...</TT>** - Aplica um espaçamento regular ao texto.

## 2.4. DEFININDO FONTES

### TAG <FONT>

Permite alterar o tipo de fonte, o tamanho e a cor.

Sintaxe:

**<FONT SIZE="n" FACE="nome" COLOR="cor">Texto</FONT>**

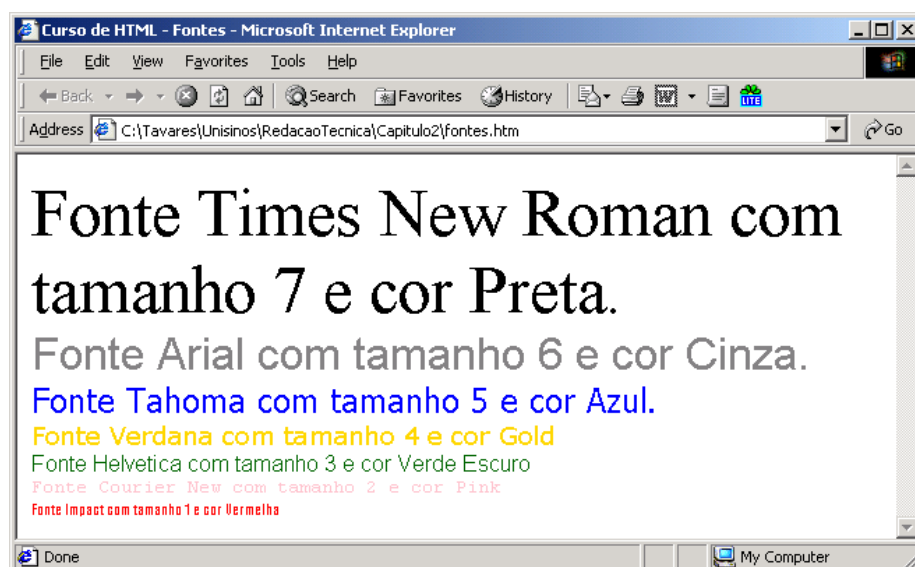
Onde:

- ⇒ **n**: o n varia de 1 a 7 e o valor 3 corresponde ao valor padrão;
- ⇒ **nome**: é o nome da fonte a ser utilizada;
- ⇒ **cor**: é a cor da fonte, definida em hexadecimal, ou através de um nome predefinido de cores.

### Exemplo: Fontes.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Fontes</TITLE>
</HEAD>
<BODY>
  <FONT SIZE="7" FACE="Times New Roman" COLOR="#FF0000">
    Fonte Times New Roman com tamanho 7 e cor Vermelha.</FONT><BR>
  <FONT SIZE="6" FACE="Arial" COLOR="#BC8F8F">
    Fonte Arial com tamanho 6 e cor Pink.</FONT><BR>
  <FONT SIZE="5" FACE="Tahoma" COLOR="Blue">
    Fonte Tahoma com tamanho 5 e cor Azul.</FONT><BR>
  <FONT SIZE="4" FACE="Verdana" COLOR="Gold">
    Fonte Verdana com tamanho 4 e cor Gold</FONT><BR>
  <FONT SIZE="3" FACE="Helvetica" COLOR="DarkGreen">
    Fonte Helvetica com tamanho 3 e cor Verde Escuro</FONT><BR>
  <FONT SIZE="2" FACE="Courier New" COLOR="Gray">
    Fonte Courier New com tamanho 2 e cor Cinza</FONT><BR>
  <FONT SIZE="1" FACE="Impact" COLOR="#000000">
    Fonte Impact com tamanho 1 e cor Preta</FONT><BR>
</BODY>
</HTML>
```

### Como fica no Browser:



## 2.5. LINHAS HORIZONTAIS E ALINHAMENTO DE ELEMENTOS

## TAG <DIV>

Ela permite o alinhamento horizontal de qualquer elemento em uma página. Esta TAG é muito utilizada em DHTML pela propriedade de agrupar elementos de página.

### Sintaxe:

`<DIV ALIGN="posição">Elementos a serem alinhados</DIV>`

### Onde:

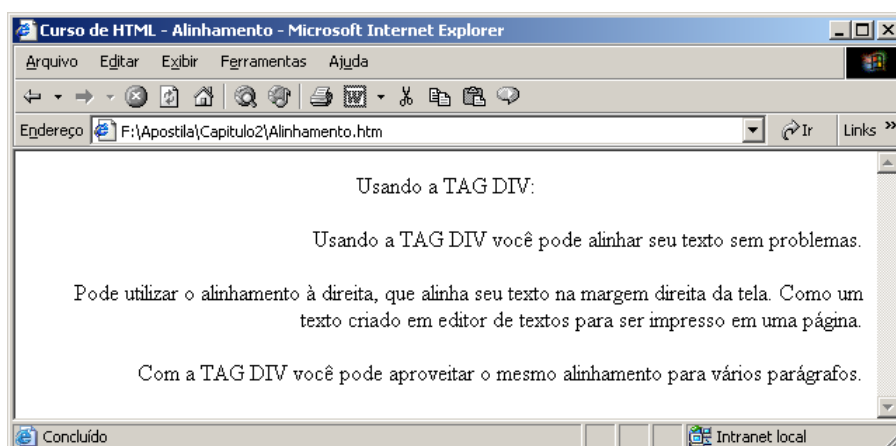
⇒ posição pode ser:

- **left**: alinhamento à esquerda;
- **center**: centralizado;
- **right**: alinhamento à direita.
- **justify**: justificado. Somente para textos.

### Exemplo: Alinhamento\_div.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Alinhamento</TITLE>
</HEAD>
<BODY>
  <DIV ALIGN="Center">Usando a TAG DIV:</DIV>
  <BR>
  <DIV ALIGN="Right">
    <P>Usando a TAG DIV você pode alinhar seu texto sem problemas.
    <P>Pode utilizar o alinhamento à direita, que alinha seu texto na margem
    direita
    da tela. Como um texto criado em editor de textos para ser impresso em uma
    página.
    <P>Com a TAG DIV você pode aproveitar o mesmo alinhamento para vários
    parágrafos.
  </DIV>
</BODY>
</HTML>
```

### Como fica no Browser:



## TAG <CENTER>

Ela permite a centralização horizontal de qualquer elemento em uma página. Tem o mesmo efeito que a TAG <DIV ALIGN="Center">.

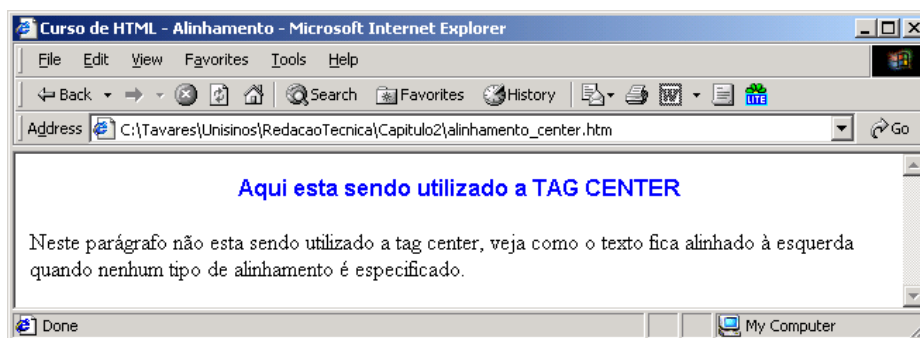
### Sintaxe:

<CENTER>Elementos a serem centralizados</CENTER>

### Exemplo: Alinhamento\_center.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Alinhamento</TITLE>
</HEAD>
<BODY>
<CENTER>
<FONT FACE="Arial" COLOR="Blue"><H4>Aqui esta sendo utilizado a TAG
CENTER</H4></FONT>
</CENTER>
Neste parágrafo não esta sendo utilizado a tag center, veja como o
texto fica alinhado à esquerda quando nenhum tipo de alinhamento é
especificado.
</BODY>
</HTML>
```

### Como fica no Browser:



## TAG <HR>

Insere uma linha horizontal no Browser.

### Sintaxe:

<HR WIDTH="n%" ALIGN="posição" SIZE="n" NOSHADE COLOR="#RRGGBB">

### Onde:

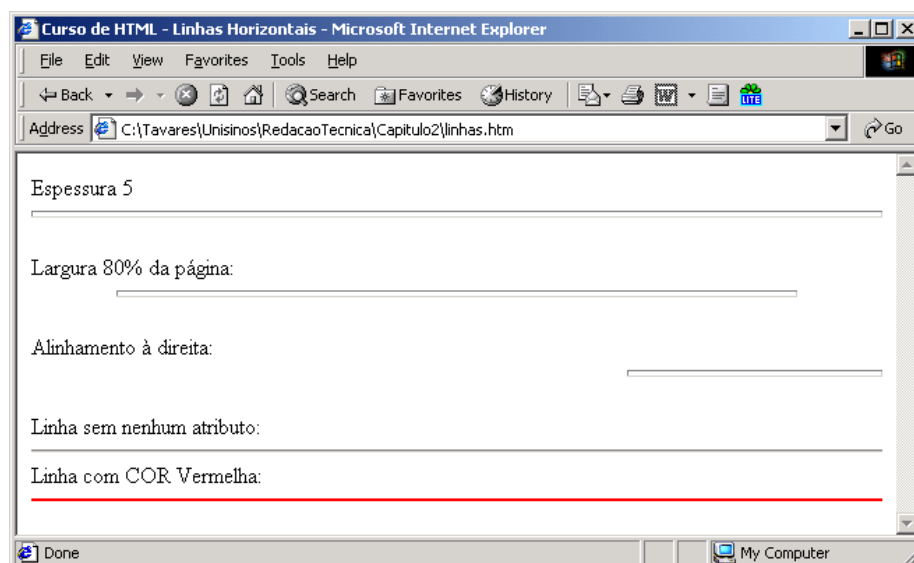
⇒ ALIGN="posição": pode ser left, center e right;

- ⇒ **WIDTH="n%"**: define a largura da linha, pode ser definida em pixels ou em porcentagem do tamanho horizontal da tela;
- ⇒ **SIZE="n"**: define a espessura da barra, em pixels;
- ⇒ **NOSHADE**: define que a barra não deve ser com efeito 3D;
- ⇒ **COLOR="#RRGGBB"**: define a cor da barra. (MS Internet Explorer).

### Exemplo: Linhas.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Linhas Horizontais</TITLE>
</HEAD>
<BODY>
Espessura 5<HR SIZE="05">
<BR>
Largura 80% da página:<HR SIZE="05" WIDTH="80%">
<BR>
Alinhamento à direita:<HR SIZE="05" ALIGN="Right" WIDTH="30%">
<BR>
Linha sem nenhum atributo:<HR>
Linha com COR Vermelha:<HR NOSHADE COLOR="Red">
</BODY>
</HTML>
```

### Como fica no Browser:



## 2.6. LISTAS

Utilizar listas é uma boa forma de organizar as páginas Web. As listas podem servir como resumos ou índices de todo o site, contendo links para outras páginas.

Basicamente existem três tipos de listas:

- Listas Não Ordenadas;
- Listas Ordenadas ou Numeradas;
- Listas Descritivas.

## 2.6.1. LISTAS NÃO ORDENADAS

### TAG <UL>

Permite inserir na página um texto de tópicos com marcadores de tipos diferentes.

Sintaxe:

```
<UL TYPE="formato">  
  <LI>Tópico 1</LI>  
  <LI>Tópico 2</LI>  
</UL>
```

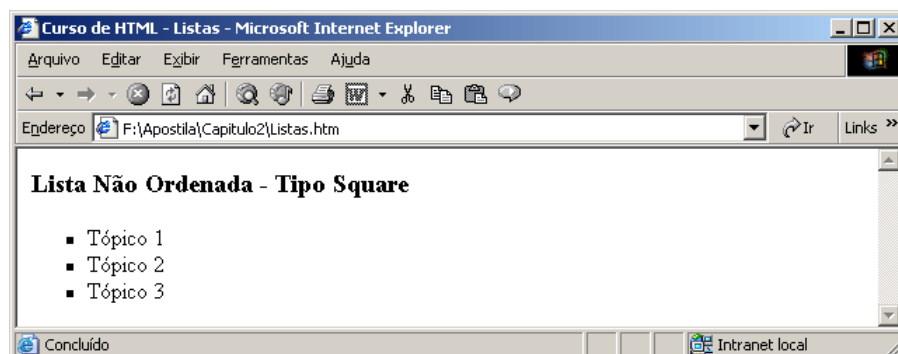
Onde:

- ⇒ <UL>: Início da Lista Não Ordenada.
- ⇒ TYPE="formato": Indica o formato do marcador. Pode ser:
  - disc: O marcador é um ponto. Este é o padrão;
  - square: O marcador é um quadrado;
  - circle: O marcador é um círculo.
- ⇒ <LI>: Tópicos da lista. Não há limites e não é necessário a TAG </LI>.
- ⇒ </UL>: Fim da Lista Não Ordenada.

Exemplo: Listas\_n\_ordenada.htm

```
<HTML>  
<HEAD>  
<TITLE>Curso de HTML - Listas</TITLE>  
</HEAD>  
<BODY>  
<H3>Lista Não Ordenada - Tipo Square</H3>  
<UL TYPE="square">  
  <LI>Tópico 1</LI>  
  <LI>Tópico 2</LI>  
  <LI>Tópico 3</LI>  
</UL>  
</BODY>  
</HTML>
```

Como fica no Browser:





## 2.6.2. LISTAS ORDENADAS

### TAG <OL>

Permite inserir na página uma lista numérica ou alfabética, com valor predefinido ou não.

Sintaxe:

```
<OL TYPE="formato" START="n">  
  <LI>Tópico 1</LI>  
  <LI>Tópico 2</LI>  
</OL>
```

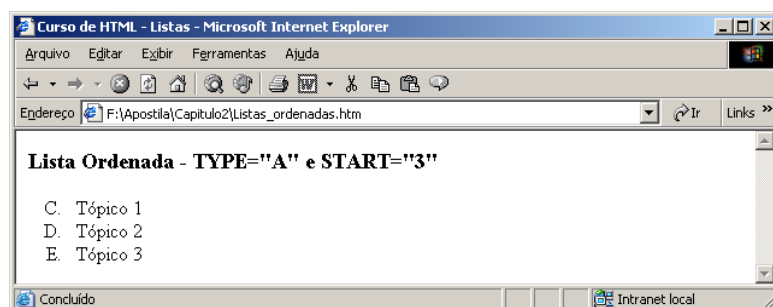
Onde:

- ⇒ **<OL>**: Início da Lista Ordenada.
- ⇒ **TYPE="formato"**: Indica o formato da numeração. Pode ser:
  - **1**: lista numérica padrão;
  - **A**: lista alfabética com letras maiúsculas;
  - **a**: lista alfabética com letras minúsculas;
  - **I**: lista numérica com números romanos maiúsculos;
  - **i**: lista numérica com números romanos minúsculos.
- ⇒ **START="n"**: O valor inicial da lista numerada.
- ⇒ **<LI>**: Tópicos da lista. Não há limites e não é necessário a TAG </LI>.
- ⇒ **</OL>**: Fim da Lista Ordenada.

Exemplo: Listas\_ordenadas.htm

```
<HTML>  
<HEAD>  
<TITLE>Curso de HTML - Listas</TITLE>  
</HEAD>  
<BODY>  
<H3>Lista Ordenada - TYPE="A" e START="3"</H3>  
<OL TYPE="A" START="3">  
  <LI>Tópico 1</LI>  
  <LI>Tópico 2</LI>  
  <LI>Tópico 3</LI>  
</OL>  
</BODY>  
</HTML>
```

Como fica no Browser:



### 2.6.3. LISTAS DESCRITIVAS

As listas descritivas são diferentes das outras listas. Cada item de uma lista descritiva possui dois componentes: um tópico e uma descrição do tópico.

#### TAG <DL>

Permite inserir na página uma lista descritiva.

##### Sintaxe:

```
<DL>
  <DT>Tópico 1
  <DD>Descrição do Tópico 1
  <DT>Tópico 2
  <DD>Descrição do Tópico 2
</DL>
```

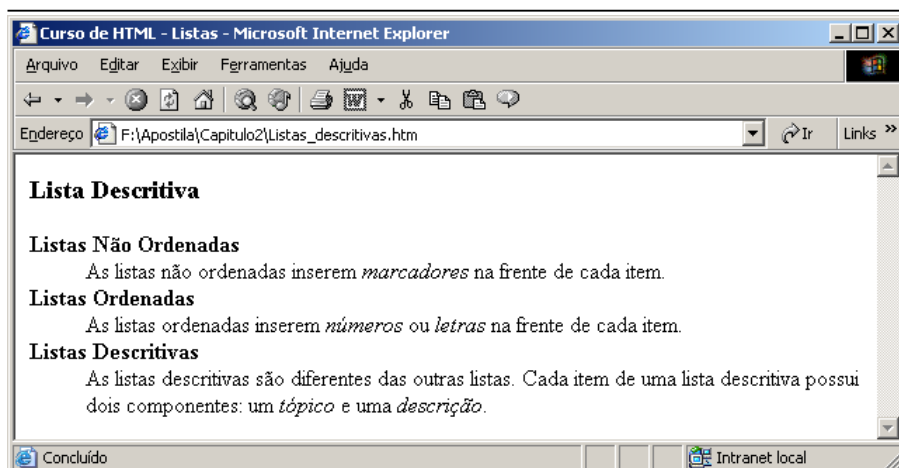
##### Onde:

- ⇒ <DL>: Início da Lista Descritiva.
- ⇒ <DT>: Tópico da Lista Descritiva. Não há limite e não é necessário a TAG </DT>.
- ⇒ <DD>: Descrição da lista. Para cada TAG <DT> existe uma TAG <DD>.
- ⇒ </DL>: Fim da Lista Descritiva.

##### Exemplo: Listas\_descritivas.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Listas</TITLE>
</HEAD>
<BODY>
<H3>Lista Descritiva</H3>
<DL>
  <DT><B>Listas Não Ordenadas</B>
  <DD>As listas não ordenadas inserem <I>marcadores</I> na frente de cada
item.
  <DT><B>Listas Ordenadas</B>
  <DD>As listas ordenadas inserem <I>números</I> ou <I>letras</I> na frente
de
    cada item.
  <DT><B>Listas Descritivas</B>
  <DD>As listas descritivas são diferentes das outras listas. Cada item de
uma
    lista descritiva possui dois componentes: um <I>tópico</I> e uma
    <I>descrição</I>.
</DL>
</BODY>
</HTML>
```

##### Como fica no Browser:



## 3. LINKS

A principal atração da Internet é sem dúvida a utilização de hipertexto. Uma página com ligação com outras páginas ou arquivos localizados em qualquer parte do mundo.

A aplicação de hipertexto não se restringe somente a ligações com outras páginas. É possível fazer ligações com trechos dentro da própria página. Isto é muito utilizado em páginas com textos longos.

### 3.1. LINKS PARA ARQUIVOS LOCAIS

Um link para uma página local é quando o arquivo que será linkado está no mesmo diretório da página que esta linkando. Neste caso é preciso informar somente o nome do arquivo.

**Sintaxe:**

`<A HREF="arquivo.extensão">Texto ou Imagem</A>`

**Onde:**

- ⇒ **"arquivo.extensão"**: Deve ser definido o nome do arquivo que será acessado e sua extensão.
- ⇒ **Texto ou Imagem**: É o texto ou imagem que servirá como link.

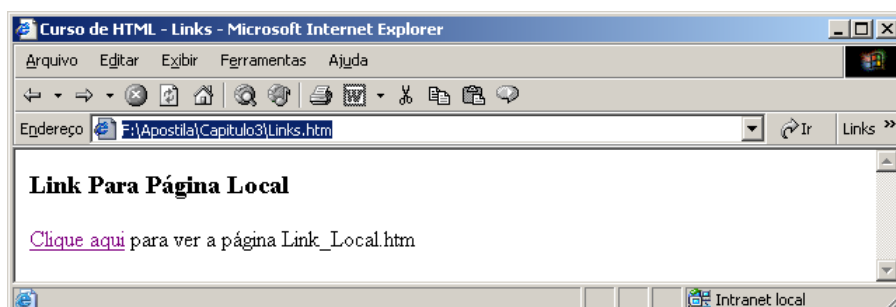
**Exemplo: Links.htm**

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links</TITLE>
</HEAD>
<BODY>
<H3>Link Para Página Local</H3>

<A HREF="Link_Local.htm">Clique aqui</A> para ver a página Link_Local.htm

</BODY>
</HTML>
```

**Como fica no Browser:**

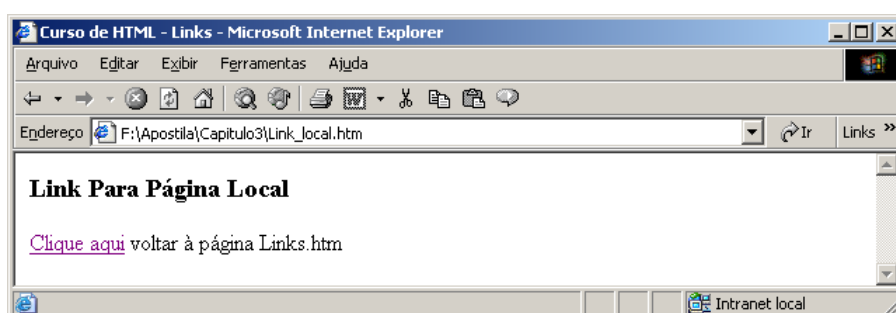


**Exemplo: Link\_Local.htm**

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links</TITLE>
</HEAD>
<BODY>
<H3>Link Para Página Local</H3>

<A HREF="Links.htm">Clique aqui</A> voltar à página Links.htm

</BODY>
</HTML>
```

**Como fica no Browser:**

## 3.2. LINKS PARA ARQUIVOS EM OUTROS DIRETÓRIOS

Para criar um link para um arquivo localizado em outro diretório (pasta) é necessário indicar o caminho completo para o arquivo

Regras de diretórios na Web:

- ⇒ O Ponto de partida para localizar um arquivo é o diretório atual;
- ⇒ A barra utilizada para separar diretórios é a barra convencional (/);
- ⇒ Para subir um nível, você deve utilizar os sinais "../".

**Sintaxe:**

**<A HREF="diretório/arquivo.extensão">Texto ou Imagem</A>**

Ou

**<A HREF="../diretório/arquivo.extensão">Texto ou Imagem</A>**

**Exemplo:**

Imagine a seguinte estrutura de diretórios:

- ⇒ C:\HTML\Aulas\Links.htm

- ⇒ C:\HTML\Link\_Home.htm
- ⇒ C:\HTML\Aulas\Distante\Link\_Distante.htm

De dentro da página Links.htm para acessar a página Link\_Home.htm o comando correto é:  
<A HREF=" ../Link\_Home.htm">Texto ou Imagem</A>

Para acessar a página Link\_Distante.htm o comando correto é:  
<A HREF="Distante/Link\_Distante.htm">Texto ou Imagem</A>

### Exemplo: Links.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links</TITLE>
</HEAD>
<BODY>
<H3>Link Para Página Local</H3>
<A HREF="Link_Local.htm">Clique aqui</A> para ver a página Link_Local.htm
<BR>
<H3>Link Para Página Que está um nível acima</H3>
<A HREF=" ../Link_Home.htm">Clique aqui</A> para ver a página ../Link_Home.htm
<BR>
<H3>Link Para Página Que está um nível abaixo</H3>
<A HREF="Distante/Link_Distante.htm">Clique aqui</A> para ver a página
Distante/Link_Distante.htm
</BODY>
</HTML>
```

### Exemplo: Pagina\_Home.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links</TITLE>
</HEAD>
<BODY>
<H3>Página em Home</H3>
Esta página está um nível de diretório acima que a página onde foi criado o
link.
<BR>
<A HREF="Aulas/Links.htm">Clique aqui</A> para voltar a página Links.htm
</BODY>
</HTML>
```

### Exemplo: Pagina\_Distante.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links</TITLE>
</HEAD>
<BODY>
<H3>Página Distante</H3>
```

```
Esta página está um nível abaixo do diretório da página onde foi criado o link.  
<BR>  
<A HREF="../Links.htm">Clique aqui</A> para voltar a página ../Links.htm  
</BODY>  
</HTML>
```

Teste estes exemplos no Browser, navegando entre as páginas através dos links.

### 3.3. LINKS PARA ARQUIVOS DE OUTROS SERVIDORES

Para criar um link para arquivos localizados em qualquer parte do mundo é necessário indicar o URL do arquivo.

Sintaxe:

`<A HREF="protocolo://servidor/arquivo">Texto ou Imagem</A>`

Onde:

- ⇒ **protocolo:** é o tipo de servidor que está sendo acessado. HTTP, FTP, etc;
- ⇒ **servidor:** é o computador que contém o arquivo;
- ⇒ **arquivo:** é o diretório e o nome do arquivo.

Exemplos:

```
<A HREF="ftp://www.empresa.com.br/docs/curriculo.doc">Curriculo</A>  
<A HREF="http://www.empresa.com.br/index.htm">Empresa X</A>
```

### 3.4. LINKS PARA E-MAIL

Para criar um link para um e-mail é preciso informar o endereço eletrônico na tag de link.

Exemplo:

```
<A HREF="mailto:aleckssandro@hotmail.com">aleckssandro@hotmail.com</A>
```

### 3.5. LINKS PARA A MESMA PÁGINA - ÂNCORA

Para criar um link para a mesma página é preciso definir seções na página. Cada seção terá um nome que servirá de referência para o link.

Para isso, é preciso criar uma âncora no início de cada seção que será acessada.

Uma âncora é um ponto de referência ou endereço. Ela é utilizada para marcar o início de cada seção. O nome da âncora será utilizado pelo link que a acessará.

Para Criar Âncora:

---

**<A NAME="Nome da Seção">Texto</A>**

Onde:

- ⇒ **NAME="Nome da Seção"**: Cria o nome da seção, marcando o início da mesma.
- ⇒ **Texto**: Não é necessário definir nada.

Para Linkar Âncora:

**<A HREF="#Nome da Seção">Texto</A>**

Onde:

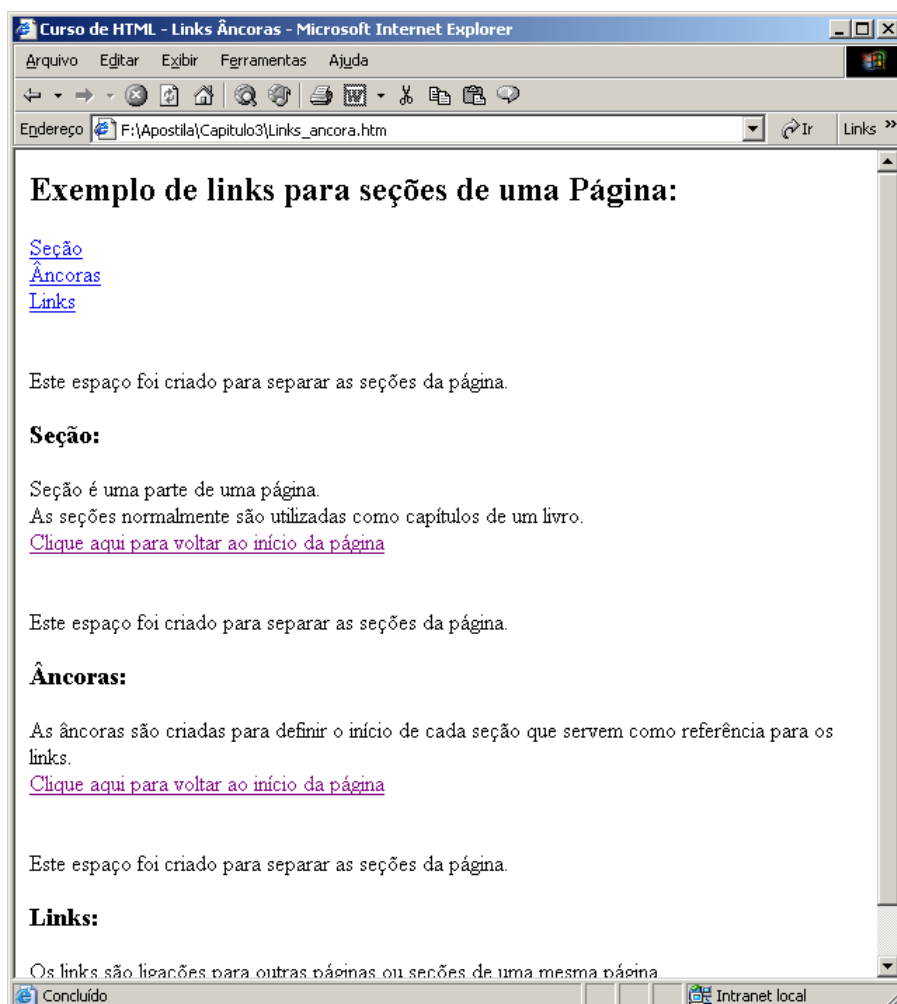
- ⇒ **#Nome da Seção**: Indica o nome da seção que deve ser acessada.
- ⇒ **Texto**: É o texto que servirá como link.

Exemplo: Links\_Ancoras.htm

```
<HTML>
<HEAD>
<TITLE>Curso de HTML - Links Âncoras</TITLE>
</HEAD>
<BODY>
<A NAME="Inicio"></A>
<H2>Exemplo de links para seções de uma Página:</H2>
<!-- Aqui estão os links -->
<A HREF="#Seção1">Seção</A><BR>
<A HREF="#Seção2">Âncoras</A><BR>
<A HREF="#Seção3">Links</A><BR>
<BR><BR>
Este espaço foi criado para separar as seções da página.
<BR><BR>
<!-- Aqui estão as seções -->
<A NAME="Seção1"></A>
<H3>Seção:</H3>
Seção é uma parte de uma página.<BR>
As seções normalmente são utilizadas como capítulos de um livro.<BR>
<A HREF="#Inicio">Clique aqui para voltar ao início da página</A><BR>
<BR><BR>
Este espaço foi criado para separar as seções da página.
<BR><BR>
<A NAME="Seção2"></A>
<H3>Âncoras:</H3>
As âncoras são criadas para definir o início de cada seção que servem como
referência para os links.<BR>
<A HREF="#Inicio">Clique aqui para voltar ao início da página</A><BR>
<BR><BR>
Este espaço foi criado para separar as seções da página.
<BR><BR>
<A NAME="Seção3"></A>
<H3>Links:</H3>
Os links são ligações para outras páginas ou seções de uma mesma página.<BR>
<A HREF="#Inicio">Clique aqui para voltar ao início da página</A><BR>
</BODY>
</HTML>
```



Como fica no Browser:



## 4. IMAGENS

Um ótimo recurso para tornar uma página web mais atraente é a utilização de imagens. Os formatos de arquivos utilizados para web são o GIF e o JPG ou JPEG.

Para editar imagens para web você pode usar o editor de sua preferência. Os mais poderosos e utilizados por Designers Web são o Corel Draw e o Photo Shop, porém o mais utilizado por programadores em geral é o Paint Shop Pro, que é o que vamos utilizar no curso para criarmos imagens, porém o propósito não é ensinar a utilização do mesmo e sim usá-lo como uma ferramenta.

O formato GIF (Graphics Interchange Format) suporta até 256 cores e gera arquivos em geral bem menores, com extensão .gif. Ele é muito utilizado para desenhos e gráficos web.

O formato JPEG (Joint Photographic Engineering Group) suporta até 16 milhões de cores, podendo reduzir o tamanho de uma imagem em até 10 vezes. Ele gera arquivos com extensão .jpg ou .jpeg, sendo muito utilizado para fotos por ter melhor resolução de cores.

### TAG <IMG>

Permite inserir na página uma imagem.

#### Sintaxe:

**<IMG SRC="arquivo" WIDTH="n" HEIGHT="n" BORDER="n" ALIGN="posição" ALT="texto">**

#### Onde:

- ⇒ **SRC="arquivo"**: Indica a URL(caminho) do arquivo de imagem. Digite o nome do arquivo e a extensão.
- ⇒ **WIDTH="n"**: Define a largura da imagem em pixels. Se não for informado o padrão é o tamanho original da imagem.
- ⇒ **HEIGHT="n"**: Indica a altura da imagem. Se não for informado o padrão é o original da imagem.
- ⇒ **BORDER="n"**: É a espessura da borda em pixels.
- ⇒ **ALIGN="posição"**: Indica a posição da imagem em relação ao texto. Posição pode ser:
  - **top**: alinha o texto pelo topo da imagem;
  - **bottom**: alinha o texto pela base da imagem(padrão);
  - **middle**: alinha o texto pelo centro da imagem.
- ⇒ **ALT="texto"**: É um texto alternativo para a imagem. Quando o Browser não exibe a imagem, a mensagem é exibida no lugar.

#### Exemplo: Imagens.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Imagens</TITLE>
</HEAD>
<BODY BGCOLOR="White">
```

```
<IMG SRC="Imagens/Estrela.gif" WIDTH="200" HEIGHT="200" BORDER="5"  
ALT="Estrela" ALIGN="Middle">  
Teste o alinhamento deste texto alterando o atributo ALIGN para Top, depois  
Bottom e  
depois Middle.  
</BODY>  
</HTML>
```

Teste no Browser.

## 4.1. IMAGENS COMO LINK

Para utilizar uma imagem como link acrescente a tag IMG entre as tag <A></A>. Quando se usa uma imagem como link, é preciso definir o atributo BORDER=0, pois alguns browsers exibem uma borda azul ao redor da imagem, prejudicando a aparência da página.

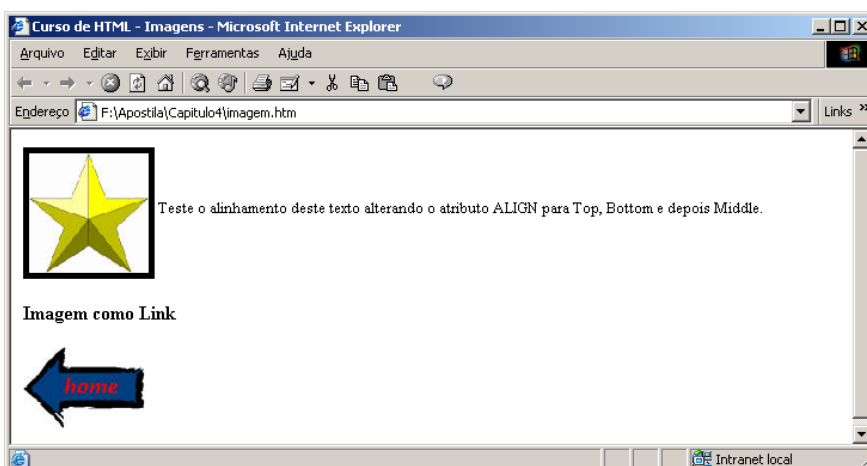
Sintaxe:

```
<A HREF="URL"><IMG SRC="Botao.gif"></A>
```

Exemplo: Imagens.htm

```
<HTML>  
<HEAD>  
  <TITLE>Curso de HTML - Imagens</TITLE>  
</HEAD>  
<BODY BGCOLOR="White">  
<IMG SRC="Imagens/Estrela.gif" WIDTH="200" HEIGHT="200" BORDER="5"  
ALT="Estrela" ALIGN="Middle">  
Teste o alinhamento deste texto alterando o atributo ALIGN para Top, Bottom e  
depois Middle.  
<H3>Imagem como Link</H3>  
<A HREF="Imagens.htm"><IMG SRC="Imagens/Botao.gif" BORDER="0"></A>  
</BODY>  
</HTML>
```

Veja como fica no Browser:



## 4.2. IMAGENS COMO PADRÃO DE FUNDO

É possível utilizar uma imagem como padrão de fundo em uma página. Para isso basta acrescentar o atributo BACKGROUND dentro da tag BODY. Tome cuidado para não utilizar arquivos muito grandes como padrão de fundo, pois tornariam a página mais lenta para carregar.

**Sintaxe:**

**<BODY BACKGROUND="imagem"></BODY>**

**Exemplo: Imagens.htm**

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Imagens</TITLE>
</HEAD>
<BODY BGCOLOR="White" BACKGROUND="Imagens/Estrela.gif">
<br><br><br>
Veja como a imagem da estrela fica como fundo na página.
</BODY>
</HTML>
```

**Teste no Browser.**

## 4.3. IMAGENS TRANSPARENTES

É possível tornar o fundo de uma imagem transparente, de forma que, ao ser carregada, a cor de fundo da imagem desaparece. Este recurso é muito útil para utilizar a mesma imagem em várias páginas de cores de fundo diferentes, sem que precise criar uma imagem para cada página.

Transparência em imagens só é suportada pelo formato GIF.

## 4.4. IMAGENS MAPEADAS

Uma imagem mapeada nada mais é do que uma imagem dividida em várias regiões, onde cada região é um link para uma página.

Existem dois tipos de imagens mapeadas:

- ⇒ **Server/Side:** As informações ficam em um arquivo residente no servidor onde está hospedada a página. Quando o usuário dá um clique na imagem, é enviado ao servidor as coordenadas da imagem que recebeu o clique e o nome do arquivo com as informações que deve ser utilizado.
- ⇒ **Cliente/Side:** As informações necessárias são armazenadas na própria página onde está a imagem mapeada. Por isso é mais rápido.

Neste curso vamos criar um mapa do tipo Client/Side. Para isso existem diversos programas que podem ser usados para mapear a imagem. Vamos utilizar o MapEdit.

Para criar uma imagem mapeada é preciso:

- ⇒ Criar uma imagem em seu editor preferido. Vamos utilizar a imagem Estrela\_Transparente.gif;
- ⇒ Criar uma página HTML com a imagem. Vamos criar a página Imagem\_Mapeada.htm;
- ⇒ Criar o mapa: definir as regiões e associar as URLs.

Veja o arquivo salvo por um editor de imagem mapeada:

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Imagem Mapeada</TITLE>
</HEAD>
<BODY BGCOLOR="White">
<IMG SRC="Imagens/Estrela_Transparente.gif" usemap="#Estrela_Transparente"
border="0">
<map name="Estrela_Transparente">
<area shape="circle" coords="49,55,17" href="Imagem_Mapeada.htm" title="">
<area shape="poly" coords="37,33,47,2,59,36,37,34" href="Imagem_Mapeada.htm"
title="">
<area shape="default" nohref>
</map>
</BODY>
</HTML>
```

Sintaxe:

```
<IMG SRC="arquivo" USEMAP="mapa">
<MAP NAME="nome">
  <AREA SHAPE="formato" COORDS="coordenadas" HREF="pagina">
</MAP>
```

Estrutura de um “Mapa”:

- ⇒ **<MAP></MAP>**: Indica o início de uma descrição de mapa processado no cliente. Atributo:
  - **NAME="nome"**: indica o nome do mapa;
- ⇒ **<AREA>**: Indica as regiões sensíveis que o mapa apresentará. Atributos:
  - **SHAPE="formato"**: indica o formato da região. Pode ser Circle, Poly, Rect, Default;
  - **COORDS="coordenadas"**: indica as coordenadas que formam a região;
  - **HREF="pagina"**: indica a página associada à região;
- ⇒ **<IMG>**: Indica a imagem mapeada. Atributo:
  - **SRC="arquivo"**: indica o nome do arquivo de imagem;
  - **USEMAP="mapa"**: indica o mapa que será utilizado;

## 4.5. IMAGENS ANIMADAS

Este é um dos recursos mais utilizados para atrair a atenção dos usuários. É possível criar animações complexas com Applets Java, objetos ActiveX e Flash. Porém a forma mais simples de criar animações é utilizar imagens GIF.

Para criar um gif animado é preciso utilizar um programa próprio para criação de gifs animados. Um dos programas mais utilizados e completos é o Animation Shop que vem com o Paint Shop Pro ([www.jasc.com](http://www.jasc.com)).

---

Nesta apostila não chegaremos a aprender como se faz uma imagem animada, mas para testar, baixe alguma da internet e teste. A tag **img** para uma imagem animada não muda em relação a uma imagem estática.

## 5. TABELAS

As tabelas são perfeitas para organizar e apresentar dados de forma prática. É possível inserir texto, imagens ou links em tabelas, distribuindo melhor as informações em uma página.

Este é um dos tópicos mais importantes, porque é imprescindível dominar bem o uso de tabelas para criar um bom layout html. Praticamente todas as páginas web utilizam recursos de tabela para diagramar seus elementos. Porém a utilização de estilos (CSS) para posicionamento de elementos na página está crescendo cada vez mais, e já existem páginas com layout muito bom sem o uso de tabelas, usando apenas CSS.

### 5.1. TABELA SIMPLES

Como padrão uma tabela não possui bordas. Cada linha da tabela é definida separadamente e é possível modificar a aparência da tabela, utilizando alguns atributos que veremos mais adiante.

O número de TR define o número de linhas na tabela e o número de TD (dentro da TAG TR) define o número de colunas da tabela.

**Sintaxe:**

```
<TABLE>
<TR>
  <TH>Cabeçalho da Tabela</TH>
  <TD>Dados da Tabela</TD>
</TR>
</TABLE>
```

**Onde:**

- ⇒ **<TABLE></TABLE>**: TAGs de início e fim de tabela.
- ⇒ **<TR></TR>**: define cada linha da tabela;
- ⇒ **<TH></TH>**: define o cabeçalho da tabela. É uma célula da tabela;
- ⇒ **<TD></TD>**: define os dados da tabela. É uma célula na tabela;
- ⇒ **Dados da Tabela**: podem ser texto, imagens e links;
- ⇒ **Cabeçalhos da Tabela**: cabeçalhos com texto, imagens ou links.

**Exemplo: Tabelas.htm**

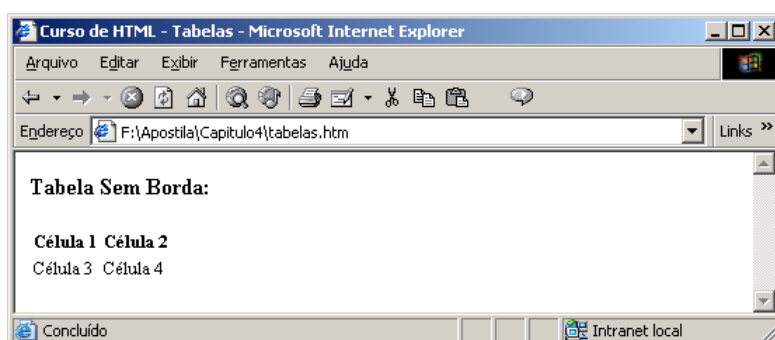
```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Tabelas</TITLE>
</HEAD>
<BODY>
<H3>Tabela Sem Borda:</H3>
<TABLE>
  <TR>
    <TH>Célula 1</TH>
```

```

    <TH>Célula 2</TH>
  </TR>
  <TR>
    <TD>Célula 3</TD>
    <TD>Célula 4</TD>
  </TR>
</TABLE>
</BODY>
</HTML>

```

Como fica no Browser:



## 5.2. MODIFICANDO A APARÊNCIA DA TABELA - ATRIBUTOS

É possível definir alguns atributos para tabelas, como bordas, alinhamento do conteúdo, espaçamento entre as células, espaçamento entre os dados e a próxima célula, largura da tabela, etc...

Vejamos alguns desses atributos:

Sintaxe:

```

<TABLE BORDER="n" WIDTH="n%" HEIGHT="n%" CELSPACING="n" CELLPADDING="n"
BGCOLOR="#RRGGBB">
</TABLE>

```

Onde:

- ⇒ **BORDER="n"**: espessura da borda em pixels.
- ⇒ **WIDTH="n%"**: largura da tabela em relação ao browser, em pixels ou %;
- ⇒ **HEIGHT="n%"**: altura da tabela em relação ao browser, em pixels ou %;
- ⇒ **CELLSPACING="n"**: espaço entre uma célula e outra, em pixels;
- ⇒ **CELLPADDING="n"**: espaço entre os dados e a borda da tabela, em pixels;
- ⇒ **BGCOLOR="#RRGGBB"**: cor de fundo das células da tabela.

Exemplo: Tabelas.htm

```

<HTML>
<HEAD>
  <TITLE>Curso de HTML - Tabelas</TITLE>

```

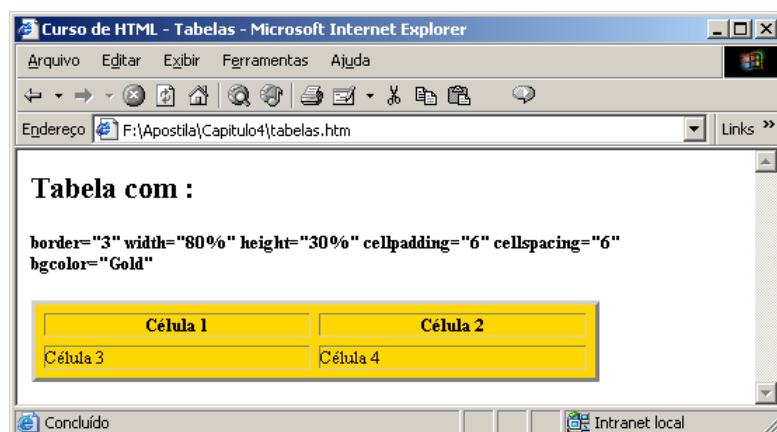


```

</HEAD>
<BODY>
<H2>Tabela com :</H2>
<H4>border="3" width="80%" height="30%" cellpadding="6" cellspacing="6"
bgcolor="Gold"</H4>
<TABLE border="3" width="80%" height="30%" cellspacing="6" cellpadding="0"
bgcolor="Gold">
  <TR>
    <TH>Célula 1</TH>
    <TH>Célula 2</TH>
  </TR>
  <TR>
    <TD>Célula 3</TD>
    <TD>Célula 4</TD>
  </TR>
</TABLE>
</BODY>
</HTML>

```

Como fica no Browser:



### 5.3. ATRIBUTOS DE LINHAS E CÉLULAS

Também é possível definir algumas propriedades de linhas e células individuais da tabela.

Sintaxe:

```

<TR ALIGN="posição" BGCOLOR="#RRGGBB" VALIGN="posição">
  <TD ALIGN="posição" VALIGN="posição" BGCOLOR="#RRGGBB" WIDTH="n%" NOWRAP></TD>
</TR>

```

Onde:

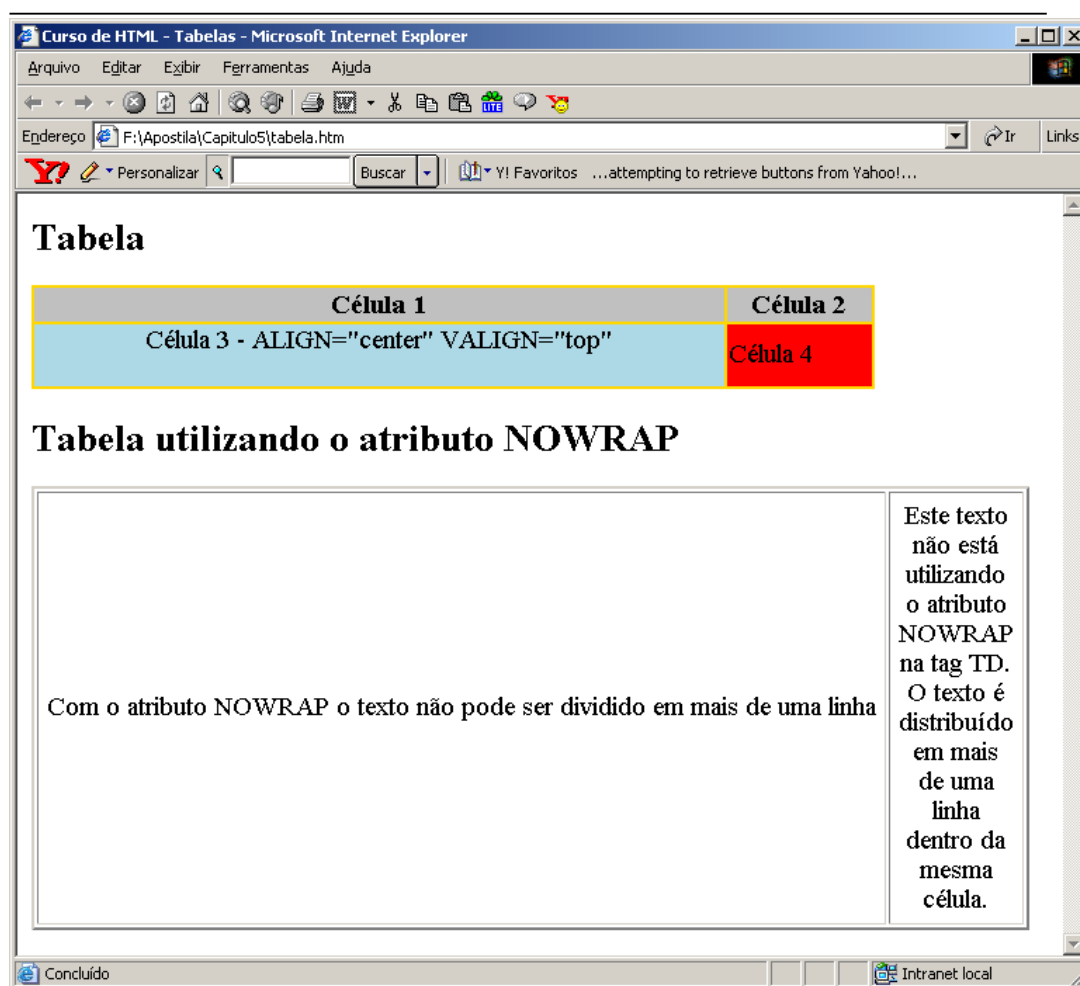
- ⇒ **ALIGN="posição"**: alinhamento horizontal do conteúdo da célula ou das células da linha. Pode ser left, right e center;
- ⇒ **BGCOLOR="#RRGGBB"**: define a cor de fundo das células da linha ou de uma célula individual;
- ⇒ **VALIGN="posição"**: alinhamento vertical de uma célula ou de células de uma linha;

- ⇒ **WIDTH="n%":** largura de uma célula em pixels ou %;
- ⇒ **NOWRAP:** indica ao browser que o texto da célula não pode ser dividido em mais de uma linha. A coluna inteira terá o tamanho do texto escrito naquela célula;

### Exemplo: Tabelas.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Tabelas</TITLE>
</HEAD>
<BODY>
<H2>Tabela</H2>
<TABLE BORDER="0" BGCOLOR="Gold" WIDTH="600" CELLPADDING="3">
  <TR BGCOLOR="Silver">
    <TH>Célula 1</TH>
    <TH>Célula 2</TH>
  </TR>
  <TR>
    <TD BGCOLOR="LightBlue" ALIGN="Center" VALIGN="top">
      Célula 3 - ALIGN="center" VALIGN="top"<BR><BR>
    </TD>
    <TD BGCOLOR="Red">Célula 4</TD>
  </TR>
</TABLE>
<H2>Tabela utilizando o atributo NOWRAP</H2>
<TABLE BORDER="2" WIDTH="80%" CELLPADDING="6">
  <TR ALIGN="Center">
    <TD NOWRAP>Com o atributo NOWRAP o texto não pode ser dividido em mais de
uma linha</TD>
    <TD>Este texto não está utilizando o atributo NOWRAP na tag TD. O texto é
distribuído
    em mais de uma linha dentro da mesma célula.
    </TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

Como fica no browser:



## 5.4. MESCLANDO CÉLULAS

É possível através de atributos na tag TD mesclar células em uma mesma linha ou em uma mesma coluna. Para mesclar células em uma mesma linha usar o atributo COLSPAN na tag TD e para mesclar células em uma mesma coluna usar o atributo ROWSPAN na tag TD.

Sintaxe:

```
<TD COLSPAN="n">Dados da Tabela</TD>
<TD ROWSPAN="n">Dados da Tabela</TD>
```

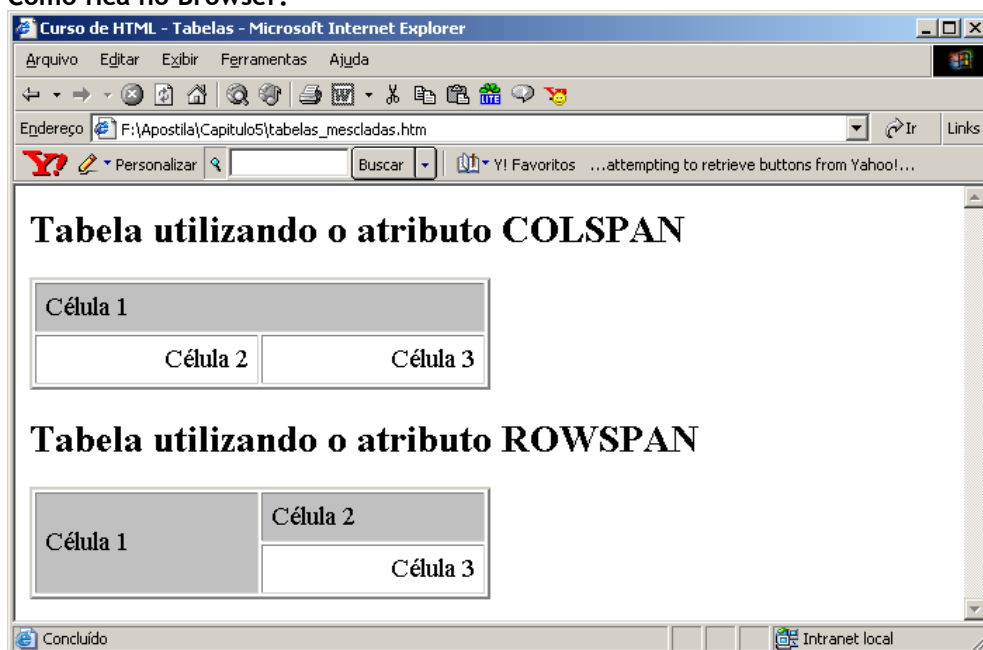
Onde:

- ⇒ COLSPAN="n": é o número de colunas que a célula ocupará;
- ⇒ ROWSPAN="n": é o número de linhas que a célula ocupará;

Exemplo: Tabelas\_Mescladas.gif

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Tabelas</TITLE>
</HEAD>
<BODY>
<H2>Tabela utilizando o atributo COLSPAN</H2>
<TABLE BORDER="2" WIDTH="50%" CELLPADDING="6">
  <TR BGCOLOR="Silver">
    <TD COLSPAN="2">Célula 1</TD>
  </TR>
  <TR ALIGN="Right">
    <TD>Célula 2</TD>
    <TD>Célula 3</TD>
  </TR>
</TABLE>
<H2>Tabela utilizando o atributo ROWSPAN</H2>
<TABLE BORDER="2" WIDTH="50%" CELLPADDING="6">
  <TR BGCOLOR="Silver">
    <TD ROWSPAN="2">Célula 1</TD>
    <TD>Célula 2</TD>
  </TR>
  <TR ALIGN="Right">
    <TD>Célula 3</TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

### Como fica no Browser:



## 6. FORMULÁRIOS

Através de um formulário o usuário pode interagir com o servidor, enviando dados que serão processados no servidor e devolvidos ao usuário.

Os formulários enviam os dados para um servidor http onde um programa CGI, geralmente em “PERL” ou “C”, processa os dados. O script CGI pode retornar, por exemplo, uma página ao usuário agradecendo o uso do formulário e/ou enviar um email para o criador da página.

Um script CGI (Common Gateway Interface), é uma interface que executa programas externos suportados por servidores de informações. No caso da internet, são os servidores http que dão suporte aos CGI.

Alguns exemplos de uso de scripts CGI:

- ⇒ tratar as requisições e dados enviados pelo usuário através de formulários;
- ⇒ servir de interface entre HTML e banco de dados SQL, fazendo a conversão da transação HTML para SQL e vice-versa;
- ⇒ converter dados do sistema em HTML gerando respostas para o cliente;

Os scripts são escritos em linguagens compatíveis com a plataforma sob a qual o servidor está rodando e devem produzir arquivos executáveis. As linguagens mais utilizadas são:

- ⇒ PERL
- ⇒ TCL
- ⇒ C Shell
- ⇒ Bourne Shell
- ⇒ C/C++

Um formulário também pode ser processado por Servidores de Páginas Ativas como ASP, JSP e PHP que estão muito difundidos na internet.

### 6.1. CRIANDO UM FORMULÁRIO

Cada elemento de um formulário é definido separadamente. Na definição do formulário, você informa o local do programa (CGI, ASP, PHP) que controlará o formulário e a forma ou método como os dados serão enviados para o servidor.

**Sintaxe:**

```
<FORM NAME="nome" METHOD="método" ACTION="URL" >  
  Elementos de Formulário  
</FORM>
```

**Onde:**

- ⇒ **METHOD:** Define o método utilizado pelo servidor para receber os dados do formulário. O método pode ser:

- **POST**: método mais utilizado, transmite toda a informação do formulário imediatamente após a URL;
- **GET**: anexa o conteúdo do formulário ao endereço da URL, por isso tem limitação de tamanho das informações;
- ⇒ **ACTION**: especifica o programa ou página do servidor que processará os dados do formulário;
- ⇒ **NAME**: define o nome do formulário.

## 6.2. TAGS DE FORMULÁRIO

### TAG <INPUT>

Define um campo de entrada de dados. Cada campo de um formulário atribui o seu conteúdo a uma variável que possui nome e tipo específico.

Sintaxe:

**<INPUT TYPE="tipo" SRC="imagem" NAME="nome" VALUE="valor" SIZE="tamanho" MAXLENGTH="comprimento" CHECKED>**

Onde:

- ⇒ **TYPE="tipo"** Define o tipo de variável que pode ser:
  - **TEXT**: Área de texto (usa os atributos: type, name, value, size, maxlength);
  - **PASSWORD**: Texto protegido por senha (usa os atributos: type, name, value, size e maxlength);
  - **HIDDEN**: Entrada oculta (atributos: type, name, value);
  - **RADIO**: Botão com uma opção (atributos: type, name, value, checked);
  - **CHECKBOX**: Botão com mais opções (type, name, value, checked);
  - **RESET**: Limpa os campos (type, name, value);
  - **SUBMIT**: Envia os dados do formulário (type, name, value);
  - **IMAGE**: Envia os dados do formulário (type, name, value, src);
  - **BUTTON**: Botão (atributos: type, name, value);
  - **FILE**: Abre uma janela para selecionar arquivo (type, name, value);
- ⇒ **NAME="nome"**: nome do campo ou variável;
- ⇒ **SRC="imagem"**: define a imagem que será usada no input do tipo image;
- ⇒ **VALUE="valor"**: define o valor do campo;
- ⇒ **CHECKED**: indica que o campo deve estar marcado (é o padrão);
- ⇒ **SIZE="tamanho"**: define o tamanho do campo, ou seja, a quantidade de caracteres que será exibido em campos do tipo Text e Password;
- ⇒ **MAXLENGTH="comprimento"**: define a quantidade de caracteres que o campo pode receber em campos do tipo Text e Password.

Exemplo: Formulario.htm

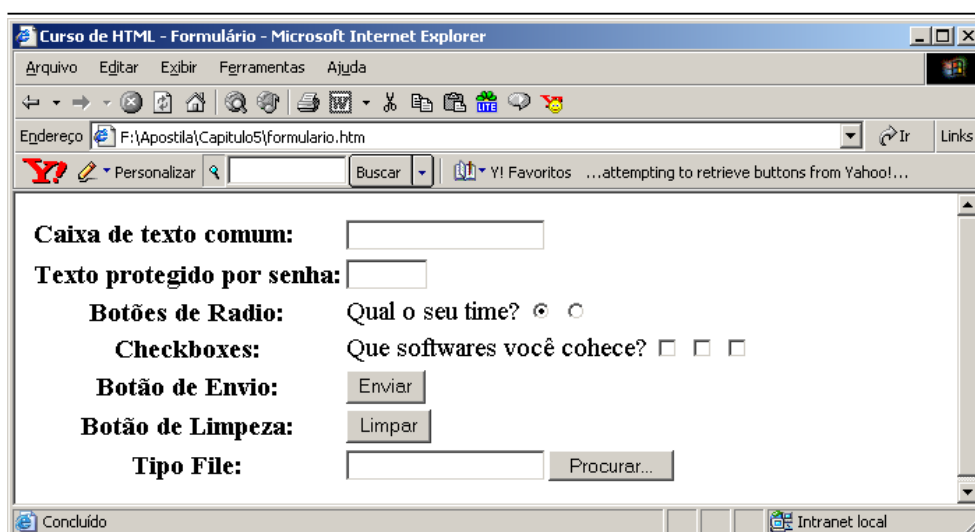
```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Formulário</TITLE>
</HEAD>
<BODY>
  <FORM ACTION="processa.asp" METHOD="POST">
    <TABLE>
      <TR>
```

```

<TH ALIGN="left" VALIGN="top">Caixa de texto comum:</TH>
<TD><INPUT TYPE="Text" NAME="txtTexto" SIZE="20" MAXLENGTH="25"></TD>
</TR>
<TR>
<TH>Texto protegido por senha:</TH>
<TD><INPUT TYPE="Password" NAME="txtTexto" SIZE="6" MAXLENGTH="6"></TD>
</TR>
<TR>
<TH>Botões de Radio:</TH>
<TD>
Qual o seu time?
<INPUT TYPE="Radio" NAME="radOpcao" VALUE="share" CHECKED>
<INPUT TYPE="Radio" NAME="radOpcao" VALUE="free">
</TD>
</TR>
<TR>
<TH>Checkboxes:</TH>
<TD>
Que softwares você cohece?
<INPUT TYPE="Checkbox" NAME="book" VALUE="Sim">
<INPUT TYPE="Checkbox" NAME="share" VALUE="Sim">
<INPUT TYPE="Checkbox" NAME="soft" VALUE="Sim">
</TD>
</TR>
<TR>
<TH>Botão de Envio:</TH>
<TD><INPUT TYPE="Submit" NAME="cmdEnvio" VALUE="Enviar"></TD>
</TR>
<TR>
<TH>Botão de Limpeza:</TH>
<TD><INPUT TYPE="Reset" NAME="cmdLimpar" VALUE="Limpar"></TD>
</TR>
<TR>
<TH>Tipo File:</TH>
<TD><INPUT TYPE="FILE" NAME="filArquivo"></TD>
</TR>
</TABLE>
</FORM>
</BODY>
</HTML>

```

Como fica no Browser:



## TAG <SELECT>

Define uma lista de itens que podem ser selecionados pelo usuário.

Sintaxe:

```
<SELECT NAME="nome" SIZE="tamanho" MULTIPLE LISTBOX>
  <OPTION SELECTED>Opção</OPTION>
  <OPTION>Opção</OPTION>
</SELECT>
```

Onde:

- ⇒ **NAME="nome"**: define o nome do campo;
- ⇒ **SIZE="tamanho"**: define a quantidade de linhas da lista;
- ⇒ **MULTIPLE**: define que vários itens da lista podem ser selecionados;
- ⇒ **LISTBOX**: define uma caixa drop-down para lista (deve ser utilizado sem os atributos SIZE e MULTIPLE);
- ⇒ **OPTION**: define cada opção da lista, as opções da lista devem ser definidas entre as tags <OPTION></OPTION>;
- ⇒ **SELECTED**: indica que a opção estará selecionada como default;

Exemplo: Formulario.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Formulário</TITLE>
</HEAD>
<BODY>
  <FORM ACTION="processa.asp" METHOD="POST">
    <TABLE BORDER="0" CELLPADDING="2">
      <TR>
        <TH ALIGN="Right">Hobbie:</TH>
        <TD>
          <SELECT NAME="Hobbie" SIZE="3" MULTIPLE>
            <OPTION>Carro</OPTION>
            <OPTION SELECTED>Futebol</OPTION>
```







## 7. 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. processados no servidor e devolvidos ao usuário.

O corpo de uma página Frame é diferente de uma página comum, e nesse arquivo só tem as dimensões das divisões e as páginas que iram aparecer em cada divisão.

A estrutura básica do corpo de uma página com frame:

```
<HTML>
<HEAD>
<TITLE>Titulo da página</TITLE>
</HEAD>
<FRAMESET COLS="15%,85%">
  <FRAME SRC="pagina1.html">
  <FRAME SRC="pagina2.html">
</FRAMESET>
</HTML>
```

### 7.1. CRIANDO FRAMES

Para utilizar o recurso de frames, é preciso utilizar um conjunto de tags. Cada uma com sua função:

- ⇒ **<FRAMESET>** é utilizada para criar um conjunto de frames e seus elementos. É a tag principal para a criação de frames e precisa ser encerrada com **</FRAMESET>**;
- ⇒ **<FRAME>** é utilizada para definir um frame (quadro) dentro de um conjunto de frames. Não necessita de tag de encerramento;

Sintaxe:

```
<FRAMESET COLS="colunas,largura" ROWS="linhas,altura" FRAMEBORDER="1/0" FRAMESPACING="n"
BORDER="n">
  <FRAME NAME="nome" SRC="URL" ALIGN="posição" FRAMEBORDER="1/0" MARGINHEIGHT="altura"
MARGINWIDTH="largura" SCROLLING="yes/no" NORESIZE>
</FRAMESET>
```

Onde:

- ⇒ **FRAMESET**: tag de abertura de um conjunto de frames;
- ⇒ **COLS="colunas,largura"**: define o número de colunas, o valor pode ser em pixels ou porcentagem. É utilizado para definir o número de quadros (frames) que terá o conjunto de frames;
- ⇒ **ROWS="linhas,altura"**: define o número de linhas, o valor pode ser em pixels ou em porcentagem. É utilizado para definir o número de quadros (frames) que terá o conjunto de frames;
- ⇒ **FRAMEBORDER="1/0"**: o valor 1 exibe uma borda 3D no frame, o valor 0 não exibe a borda;

- ⇒ **BORDER="n"**: define o tamanho da borda;
- ⇒ **FRAMESPACING="n"**: define um espaço em pixels entre os frames, nem todos os browsers reconhecem este atributo;
  
- ⇒ **FRAME**: define cada quadro (frame) separadamente;
- ⇒ **NAME="nome"**: define o nome do frame;
- ⇒ **SRC="URL"**: define a página que será exibida no frame;
- ⇒ **ALIGN="posição"**: define o alinhamento do conteúdo do frame. Posição pode ser:
  - **top**: o conteúdo é alinhado no topo do frame;
  - **middle**: o conteúdo é alinhado no meio do frame;
  - **bottom**: o conteúdo é alinhado na base do frame;
  - **left**: o conteúdo é alinhado à esquerda do frame;
  - **right**: o conteúdo é alinhado à direita do frame;
- ⇒ **FRAMEBORDER="1/0"**: o valor 1 exibe uma borda 3D, o valor 0 não exibe borda;
- ⇒ **MARGINHEIGHT="altura"**: define a altura da margem em pixels;
- ⇒ **MARGINWIDTH="largura"**: define a largura da margem em pixels;
- ⇒ **SCROLLING="yes/no/Auto"**: define se o frame terá ou não barra de rolagem;
- ⇒ **NORESIZE**: evita o redimensionamento do frame;

#### Exemplo: Frames.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<FRAMESET COLS="40%,*" FRAMEBORDER="1" BORDER="5">
  <FRAME NAME="Frame_1" SCROLLING="No" NORESIZE SRC="Pagina1.htm">
  <FRAME NAME="Frame_2" SCROLLING="Auto" NORESIZE SRC="Pagina2.htm">
</FRAMESET>
</HTML>
```

#### Exemplo: Pagina1.htm

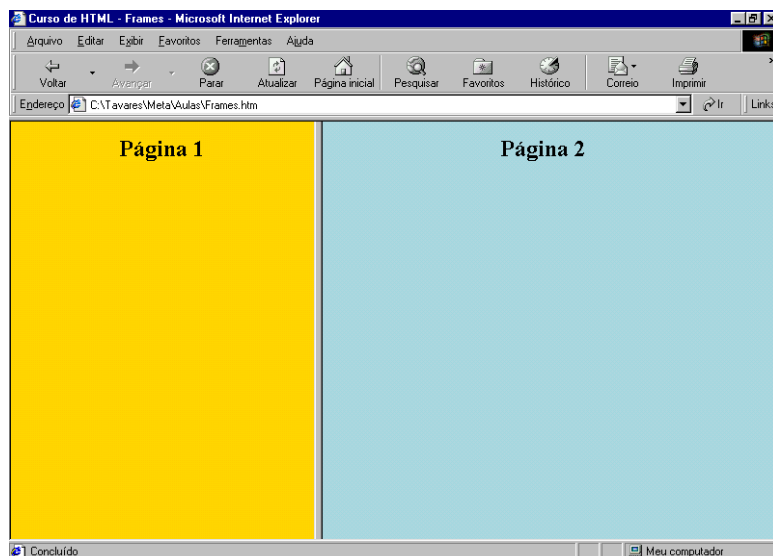
```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<BODY BGCOLOR="Gold">
<CENTER><H2>Página 1</H2></CENTER>
</BODY>
</HTML>
```

#### Exemplo: Pagina2.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<BODY BGCOLOR="LightBlue">
<CENTER><H2>Página 2</H2></CENTER>
</BODY>
</HTML>
```

---

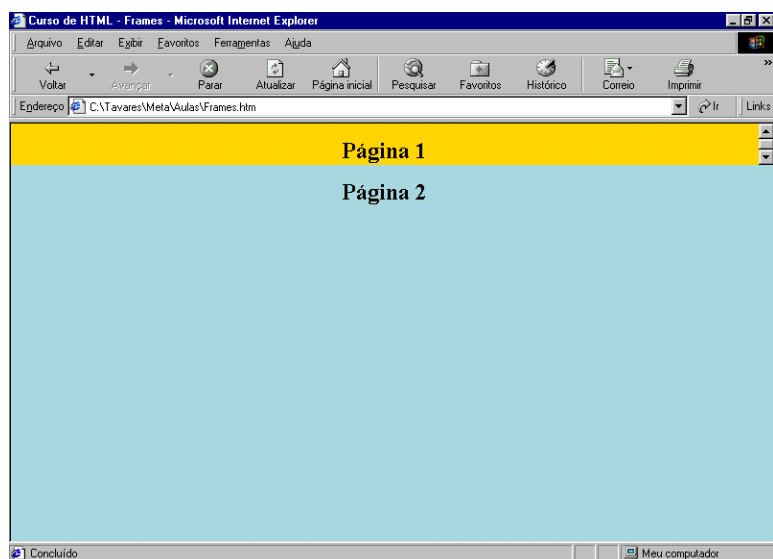
## Como fica no Browser:



## Exemplo: Frames\_2.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<FRAMESET COLS="40%,*" FRAMEBORDER="1" BORDER="5">
  <FRAME NAME="Frame_1" SCROLLING="No" NORESIZE SRC="Pagina1.htm">
  <FRAME NAME="Frame_2" SCROLLING="Auto" NORESIZE SRC="Pagina2.htm">
</FRAMESET>
</HTML>
```

## Como fica no Browser:



**Exemplo: Frames\_3.htm**

```

<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<FRAMESET ROWS="10%,*" FRAMEBORDER="0" BORDER="0">
  <FRAME NAME="Frame_1" SCROLLING="no" NORESIZE SRC="Pagina1.htm">
  <FRAMESET COLS="10%,*" FRAMEBORDER="0">
    <FRAME NAME="Frame_2" SCROLLING="Auto" NORESIZE SRC="Pagina2.htm">
    <FRAME NAME="Frame_3" SCROLLING="Auto" NORESIZE SRC="Pagina3.htm">
  </FRAMESET>
</FRAMESET>
</HTML>

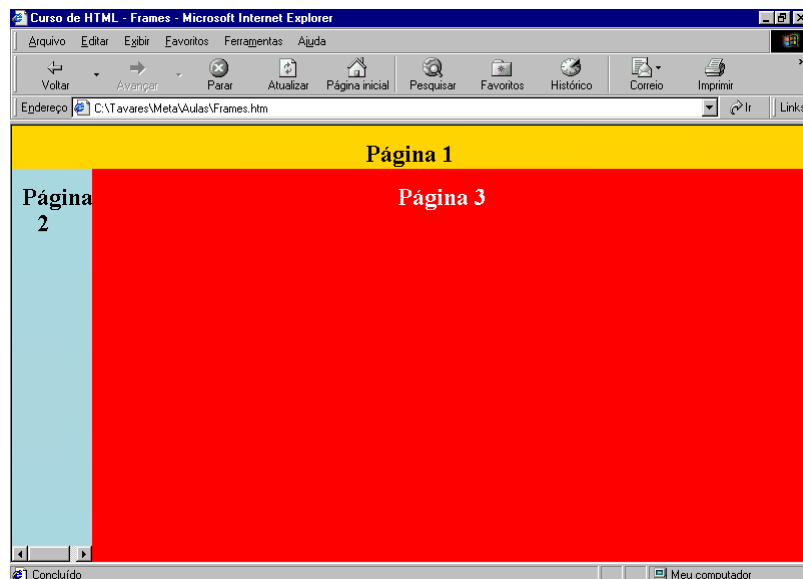
```

**Exemplo: Pagina3.htm**

```

<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<BODY BGCOLOR="Red">
<CENTER><H2><FONT COLOR="White">Página 3</FONT></H2></CENTER>
</BODY>
</HTML>

```

**Como fica no Browser:****7.2. LINKS COM FRAMES**

Sempre que se aciona um link dentro de uma página, o *default* é que a página referente a esse link seja carregada na mesma janela da página anterior, porém é possível fazer com que um link abra uma página em qualquer frame, para isso usamos o atributo TARGET na tag <A></A>

**Sintaxe:**

**<A HREF="" TARGET="frame">Link</a>**

**Onde:**

- ⇒ **TARGET="janela"**: indica em qual a janela ou frame que a página do link deve ser carregada. Janela pode ser;
- **\_top**: limpa a tela e abre a página ocupando toda a tela;
  - **\_blank**: abre uma nova janela do browser para exibir a página;
  - **\_self**: carrega a página no frame atual. É o default;
  - **nome do frame**: carrega a página no frame especificado;

**Exemplo: Frames.htm**

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<FRAMESET COLS="20%,*" FRAMEBORDER="0" BORDER="0">
  <FRAME NAME="Frame_1" SCROLLING="no" NORESIZE SRC="Pagina1.htm">
  <FRAME NAME="Frame_2" SCROLLING="Auto" NORESIZE SRC="Pagina2.htm">
</FRAMESET>
</HTML>
```

**Exemplo: Pagina1.htm**

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<BODY BGCOLOR="Gold">
<H2>Página 1</H2>
<A HREF="Pagina3.htm" TARGET="_top">Página 3 em _top</A><BR>
<A HREF="Pagina3.htm" TARGET="_blank">Página 3 em _blank</A><BR>
<A HREF="Pagina3.htm" TARGET="Frame_2">Página 3 em Frame_2</A><BR>
</BODY>
</HTML>
```

**Teste no Browser.**

## 8. OUTROS COMANDOS

Neste tópico vamos abordar alguns outros comandos interessantes, porém sem aprofundar em nenhum deles.

### 8.1. REFRESH PAGE

São páginas normalmente sem links, que chamam outras depois de um determinado tempo, sem nenhuma interferência do usuário. Este refresh pode ser para a própria página.

Para fazer uma página com refresh basta colocar no documento a seguinte linha de comando:

Sintaxe:

```
<META HTTP-EQUIV="REFRESH" CONTENT="segundos; URL=pagina">
```

Onde:

- ⇒ **segundos**: número de segundos para ocorrer o refresh.
- ⇒ **pagina**: página que deve ser carregada após transcorrer o número de segundos especificados.

Exemplo: Refresh.htm

```
<HTML>
<HEAD>
<META HTTP-EQUIV="REFRESH" CONTENT="5; URL=Pagina2.htm">
<TITLE>Curso de HTML - Refresh</TITLE>
</HEAD>
<BODY>
  <H1>Aguarde...</H1>
  <H2>Após 5 segundos será carregada a Pagina2.htm</H2>
</BODY>
</HTML>
```

Teste no Browser.

### 8.2. AUDIO E VÍDEO

Vamos ver duas maneiras de colocar músicas em sua home page. A primeira é com o TAG <BGSOUND> que só é reconhecida pelo Internet Explorer. Para utilizar este TAG siga o exemplo :

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

Onde *arquivo.mid* é o arquivo de música. Caso você queira que a música repita-se, adicione o atributo *loop="infinite"*.



A segunda maneira é com o TAG <EMBED> que, por ser reconhecido pelo Internet Explorer e Netscape, é recomendado. Para utilizá-lo, siga o exemplo :

**<EMBED SRC="arquivo.mid">**

É recomendado usar arquivos midi por serem bem menores que os wavs.  
Com o comando EMBED também é possível rodar vídeos, a sintaxe é a mesma que para músicas.

#### Exemplo: Multimidia.htm

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Frames</TITLE>
</HEAD>
<BODY BGCOLOR="LightBlue">
<CENTER><H2>Página 2</H2></CENTER>

<BGSOUND SRC="laser.wav">
<!-- <EMBED SRC="testsnd.mid"> -->
<!-- <EMBED SRC="laser.wav"> -->
</BODY>
</HTML>
```

Teste no Browser.

## 8.3. TEXTO ROLANTE

Com esta tag você pode colocar uma faixa decorativa com um texto animado em sua página.  
Para entende-lo melhor dê uma olhada no exemplo abaixo:

Sintaxe:

**<MARQUEE ALIGN="posição" BEHAVIOR="animação" BGCOLOR="#RRGGBB" DIRECTION="direção" HEIGHT="n%" WIDTH="n%" HSPACE="n" VSPACE="n" LOOP="quantide" SCROLLAMOUNT="n" SCROLLDELAY="n">Texto</MARQUEE>**

Onde:

- ⇒ **ALIGN:** Usado para alinhar a faixa, pode ser LEFT, RIGHT, MIDDLE.
- ⇒ **BEHAVIOR:** Define animação do texto, pode ser SCROLL, SLIDE ou ALTERNATE, eu poderia falar o que cada um desses tributos faz mas acho melhor você mesmo testa-los um a um.
- ⇒ **BGCOLOR:** Como você já deve saber define o fundo da faixa.
- ⇒ **DIRECTION:** Direção do texto, pode ser LEFT ou RIGHT
- ⇒ **HEIGHT/WIDTH:** Define o tamanho da faixa em pixels
- ⇒ **LOOP:** Define o número de vezes que o texto passará pela faixa, o valor atribuído no exemplo foi INFINITE que anima o texto um número infinito de vezes.

- ⇒ **HSPACE/VSPACE:** Define o tamanho das margens onde a faixa deve ser posicionada.
- ⇒ **SCROLLAMOUNT/SCROLLDELAY:** Definem a velocidade do texto, o SCROLLAMOUNT define a quantidade de informação que será movimentada, prefira os valores entre 2 e 5. O SCROLLDELAY define a velocidade entre os intervalos entre cada desenho, um valor aconselhável seria em torno de 20.

**Exemplo: Texto\_Rolante.htm**

```
<HTML>
<HEAD>
  <TITLE>Curso de HTML - Texto Rolante</TITLE>
</HEAD>
<BODY BGCOLOR="LightBlue">
<CENTER><H2>Usando a tag MARQUEE</H2></CENTER>
<MARQUEE
  ALIGN=middle
  BEHAVIOR=scroll
  BGCOLOR=Gold
  DIRECTION=right
  HEIGHT=5%
  WIDTH=80%
  HSPACE=10
  VSPACE=10
  LOOP=infinite
  SCROLLAMOUNT=2
  SCROLLDELAY=20>
  Apostila de HTML
</MARQUEE>
<BR>
<MARQUEE
  ALIGN=middle
  BEHAVIOR=alternate
  BGCOLOR=#FFFFFF
  DIRECTION=right
  HEIGHT=5%
  WIDTH=80%
  HSPACE=10
  VSPACE=10
  LOOP=infinite
  SCROLLAMOUNT=2
  SCROLLDELAY=15>
  Curso de HTML
</MARQUEE>
</BODY>
</HTML>
```

**Teste no Browser.**



## 9. CARACTERES ESPECIAIS

Á	&Aacute;	á	&aacute;	Â	&Acirc
â	&acirc;	À	&Agrave;	à	&agrave;
Å	&Aring;	å	&aring;	Ã	&Atilde;
ã	&atilde;	Ä	&Auml;	ä	&auml;
Æ	&AElig;	æ	&aelig;	É	&Eacute;
é	&eacute;	Ê	&Ecirc;	ê	&ecirc;
Ê	&Egrave;	è	&egrave;	Ë	&Euml;
ë	&euml;	Ð	&ETH;	ð	&eth;
Í	&Iacute;	í	&iacute;	Î	&Icirc;
î	&icirc;	Ì	&Igrave;	ì	&igrave;
Ï	&Iuml;	ÿ	&iuml;	Ó	&Oacute;
ó	&oacute;	Ô	&Ocirc;	ô	&ocirc;
Ò	&Ograve;	ò	&ograve;	Ø	&Oslash;
ø	&oslash;	Õ	&Otilde;	õ	&otilde;
Ö	&Ouml;	ö	&ouml;	Ú	&Uacute;
ú	&uacute;	Û	&Ucirc;	û	&ucirc;
Û	&Ugrave;	ù	&ugrave;	Ü	&Uuml;
ü	&uuml;	Ç	&Ccedil;	ç	&ccedil;
Ñ	&Ntilde;	ñ	&ntilde;	<	&lt;
>	&gt;	&	&amp;	"	&quot;
®	&reg;	©	&copy;	Ý	&Yacute;
ý	&yacute;	Þ	&THORN;	þ	&thorn;
ß	&szlig;	°	&#186;	<sup>a</sup>	&#170;
<sup>1</sup>	&#185;	<sup>2</sup>	&#178;	<sup>3</sup>	&#179;
f	&#131;	†	&#134;	‡	&#135;
‰	&#137;	¢	&#162;	£	&#163;
«	&#171;	±	&#177;	»	&#187;
•	&#183;	¼	&#188;	½	&#189;
¾	&#190;	¿	&#191;	×	&#215;
÷	&#247;	¡	&#161;	¤	&#164;

## 10. TABELA DE CORES

aliceblue	#f0f8ff	dodgerblue	1e90ff	mistyrose	ffe4e1
antiquewhite	faebd7	firebrick	b22222	moccasin	ffe4b5
aquamarine	7fffd4	floralwhite	fffaf0	navajowhite	ffdead
azure	f0ffff	forestgreen	228b22	navy	000080
beige	f5f5dc	gainsboro	dcdcdc	oldlace	fdf5e6
bisque	ffe4c4	ghostwhite	f8f8ff	olivedrab	6b8e23
black	000000	gold	ffd700	orange	ffa500
blanchedalmond	ffebed	goldenrod	daa520	orangered	ffa500
blue	0000ff	gray	7f7f7f	orchid	da70d6
blueviolet	8a2be2	green	008000	palegoldenrod	eee8aa
brown	a52a2a	greenyellow	adff2f	palegreen	98fb98
burlywood	deb887	honeydew	f0fff0	paleturquoise	afeeee
cadetblue	5f9ea0	hotpink	ff69b4	palevioletred	db7093
chartreuse	7fff00	indianred	cd5c5c	papayawhip	ffefd5
chocolate	d2691e	ivory	fffff0	peachpuff	ffdab9
coral	ff7f50	khaki	f0e68c	peru	cd853f
cornflowerblue	6495ed	lavender	e6e6fa	pink	ffc0cb
cornsilk	fff8dc	lavenderblush	fff0f5	plum	dda0dd
cyan	00ffff	lightseagreen	#20b2aa	powderblue	b0e0e6
darkblue	00008b	lightskyblue	87cefa	purple	800080
darkcyan	008b8b	lightslateblue	778899	red	ff0000
darkgoldenrod	b8860b	lightslategray	888888	rosybrown	bc8f8f
darkgray	a9a9a9	lightsteelblue	b0c4de	royalblue	4169e1
darkgreen	006400	lightyellow	ffffe0	saddlebrown	8b4513
darkkhaki	bdb76b	limegreen	00ff00	salmon	fa8072
darkmagenta	8b008b	linen	faf0e6	sandybrown	f4a460
darkolivegreen	556b2f	magenta	ff00ff	seagreen	2e8b57
darkorange	ff8c00	maroon	800000	seashell	fff5ee
darkorchid	9932cc	mediumaquamarine	66cdaa	sienna	a0522d
darkred	8b0000	mediumblue	0000cd	skyblue	87ceeb
darksalmon	e9967a	mediumorchid	ba55d3	slateblue	6a5acd
darkseagreen	8fbc8f	mediumpurple	9370db	slategray	708090
darkslateblue	483d8b	mediumseagreen	3cb371	snow	fffafa
darkslategray	2f4f4f	mediumslateblue	7b68ee	springgreen	00ff7f
darkturquoise	00ced1	mediumspringgreen	00fa9a	steelblue	4682b4
darkviolet	9400d3	mediumturquoise	48d1cc	tan	d2b48c
deeppink	ff1493	mediumvioletred	c71585		
deepskyblue	00bfff	midnightblue	191970		
dimgray	696969	mintcream	f5fffa		

## 11. REFERÊNCIAS

- HTML Dinâmico. *Ramalho, José Antônio Alves*. Berkeley Brasil, 1999.
- A Arte Web. *Site Sobre HTML, HTML Dinâmico e Design*. ([www.aarte-web.com.br](http://www.aarte-web.com.br)).
- Tutorial HTML. *Tutorial mantido pela USP desde 1995*. ([www.icmc.usp.br/manuals/HTML](http://www.icmc.usp.br/manuals/HTML)).