



**Fundação
Bradesco**

HTML COM ACESSIBILIDADE

INFORMÁTICA

Educação Profissional - Formação Inicial e Continuada

HTML COM ACESSIBILIDADE

Reservados todos os direitos patrimoniais e de reprodução à Fundação Bradesco
Homepage: www.fundacaobradesco.org.br

AUTORIA
Fundação Bradesco

COLABORADORES
Setor de Educação Profissional
DTE – Escola Virtual

PROJETO GRÁFICO/EDITORAÇÃO
Setor Geração de Recursos Didáticos

PUBLICAÇÃO: 2008

ÍNDICE

HTML com acessibilidade	1
Objetivo.....	1
Apresentação	1
Conteúdo do curso	1
Importância	2
HTML.....	3
O desenvolvimento em HTML	4
Os arquivos e extensões do HTML.....	5
Incompatibilidade para o reconhecimento de código entre os navegadores	5
Sobre o W3C	6
Etiquetas ou TAGS	6
Estrutura básica de uma página em HTML	8
O uso de acentos e caracteres especiais	10
Definindo cores nos documentos HTML.....	13
Definindo cores nos documentos HTML.....	16
Inserindo comentários	17
Formatando documentos para WEB	17
Quebra de linha	18
Parágrafos.....	19
Uso de parágrafos e da quebra de linha	20
Linhas horizontais.....	21
Formatando fontes	22
Tags complementares para formação das fontes	24
Formatação das fontes	26
Estilo de formatação para as fontes	27
Definição dos estilos de formatação para as fontes	28
Efeitos sobrescrito e subscrito	29
Uso de espaçamentos e tabulações	29
Cabeçalhos.....	30
Exemplo do uso de cabeçalhos	32
Texto pré-formatado.....	33
Hiperlinks	34
Sintaxe para inserção de links	34
Links internos	36
Links externos	38

Links para download de arquivos	38
Âncoras de marcação para documentos HTML.....	40
Links nulos.....	42
Link para e-mail.....	42
Formatando as cores dos links	44
Abrindo uma imagem no Microsoft Photo Editor	45
Imagens	46
O formato Gif	47
O formato JPEG	48
Sintaxe para inserção de imagens.....	49
O que é e como abrir o Microsoft Photo Editor.....	51
Convertendo formatos de imagem no Microsoft Photo Editor	52
Obtendo a largura e a altura ou redimensionando uma imagem no Microsoft Photo Editor	52
A aplicação de imagens nas páginas WEB	53
Inserindo uma imagem de fundo para a página	54
Exibindo um logotipo ou uma fotografia na página.....	54
Ilustrando as informações escritas.....	55
Inserindo botões gráficos de acesso.....	55
Usando thumbnails	56
Listas e tabelas.....	57
Lista desordenada ou não numerada.....	58
Lista ordenada ou numerada	59
Lista de definição ou glossário	62
Lista intercalada ou encadeada	63
Tabelas.....	66
Efeitos de animação e recursos multimídia	83
Inserindo movimento aos elementos HTML.....	84
Inserindo texto com efeito piscante.....	87
Inserindo recursos de áudio em páginas HTML	88
Áudio para o navegador Internet Explorer	89
Áudio para o navegador Netscape Navigator.....	90
Vídeo para o navegador Internet Explorer.....	91
Vídeo e áudio por streaming	92
Formulários eletrônicos.....	93
Criando um formulário.....	94
A classe input	95
Caixa de texto	136

Controle oculto.....	97
Caixa de verificação.....	98
Caixa de opção	99
Botões	100
Exemplo de um formulário eletrônico que utiliza os controles da classe input.....	101
A classe textarea	102
Exemplo de um formulário eletrônico que utiliza o controle da classe textarea	103
A classe select	104
Atributos da tag select	105
Cascading style sheets	107
Propriedades e valores aplicados a criação de estilos.....	109
Inserindo estilo de formatação em um elemento HTML.....	110
Entendendo a hierarquia dos elementos HTML	111
O uso de elementos containers	112
Regras para a definição dos estilos de formatação.....	113
Definindo regras a partir das classes.....	115
Definindo regras a partir de pseudoclasses.....	116
Definindo regras a partir de um selecionador ou de um grupo de selecionadores	118
Disponibilizando estilos de formatação para todas as páginas de um site.....	119
Exemplo de um arquivo de folha de estilos.....	120
Layout com frames	122
Arquivo de definição dos frames.....	124
Atributos da tag frameset	125
Atributos da tag frame	127
Arquivo do menu opções	128
Frames in-line.....	129
Exemplos de página contendo a estrutura de um layout com frames.....	130
Exemplo de uma página contendo um menu de navegação para um layout com frames	132
Exemplo de uma página de apresentação para um layout com frames	133
Arquivo de folha de estilos usado nos exemplos do layout com frames	134
E agora, o que devo fazer após ter desenvolvido a estrutura básica do meu site?.....	135
Dicas e considerações finais	136

HTML COM ACESSIBILIDADE

Objetivos

Um dos objetivos deste curso é possibilitar que ao concluí-lo, o aluno tenha adquirido o conhecimento necessário em HTML para desenvolver um site, utilizando-se dos principais recursos para a confecção de páginas para a Web.

Outro objetivo é fornecer ao aluno subsídios técnicos da linguagem HTML, tornando-o capaz de realizar análises mais criteriosas dos sites disponíveis na Internet.

Desta forma, identificará problemas referentes à acessibilidade e indicar possíveis soluções para a resolução dos mesmos, contribuindo de forma efetiva com a disseminação desta cultura entre a comunidade de desenvolvedores Web.

Por último, pretende-se acrescentar mais um diferencial na qualificação do aluno, ampliando as suas oportunidades de crescimento e de aperfeiçoamento profissional.

Apresentação

O curso de HTML com Acessibilidade, será dividido em quinze aulas, onde em cada uma das aulas, o aluno terá além da navegação pelo conteúdo exposto para assimilação, a realização de exercícios onde será feita uma análise do processo de aprendizado do aluno com base no conteúdo exposto até o momento.

Conteúdo do Curso

- Inicialmente faremos uma abordagem teórica demonstrando a importância, os princípios e os conceitos fundamentais da linguagem HTML.
- Apresentaremos os principais recursos disponíveis para a formatação de textos para a criação de documentos Web.

- Demonstraremos as diversas aplicações obtidas com a criação de link, considerado como um dos mais importantes recursos para a construção de sites para a Internet.
- Apresentaremos como usufruir da inserção de imagens, com base em um enfoque teórico e prático dentro das suas possibilidades de aplicação nas páginas Web.
- Estudaremos como organizar e alinhar informações em diferentes estilos de listas e em tabelas.
- Ensinaresmos a inserir alguns efeitos de animação e recursos multimídia, como áudio e vídeo.
- Veremos como construir formulários eletrônicos para a colheita de informações.
- Apresentaremos o mecanismo para a criação de estilos de formatação, trazendo mais flexibilidade e automação para o processo de formatação de textos e definição de layout de apresentação visual.
- Apresentaremos as etapas necessárias para se construir um site baseado em um layout com frames.

Importância

A Web tornou-se na última década o meio de comunicação de maior destaque, devido ao seu crescimento exponencial tanto de usuários como de informações e serviços disponibilizados.

Com a redução dos custos e com as facilidades de financiamentos para aquisição de computadores, além dos programas de Inclusão Digital e de democratização para o uso da Internet, cada vez, adeptos se conectam a essa rede, tornando-a um pólo atrativo para que grandes, médias, pequenas empresas. Além de pessoas comuns e profissionais liberais, enxerguem nela uma oportunidade de expor e oferecer seus produtos, serviços e/ou informações.

Uma das características que torna um grande diferencial na Internet sobre os demais meios de comunicação é a interatividade, onde seus usuários podem definir o que quer ver e por quais caminhos deseja seguir. Por outro lado, o Site precisa ser o mais atrativo possível, a fim de conquistar o usuário e mantê-lo o maior tempo possível navegando por suas seções.

Mas criar um vínculo de fidelidade com os visitantes de um Site não é uma tarefa tão simples. É necessário apresentar características que satisfaçam à preferência do maior número possível de pessoas.

Para isso, é preciso considerar algumas variáveis como o conteúdo exposto, o aspecto visual, o dinamismo, a complexidade quanto à navegação por parte dos visitantes e muitas outras, que devem ser criteriosamente analisadas a fim de se obter bons resultados e com certeza, um bom desenvolvimento em HTML contribui positivamente para uma boa receptividade por parte dos internautas.

HTML

○ HTML (HyperText Markup Language) que trouxe a possibilidade de se interligar documentos indiferentemente do seu endereço físico na Web, foi originado da linguagem SGML (Standard Generalized Markup Language), que apresenta uma rica estrutura de comandos padronizados para a formatação de documentos.

○ HTML é muito mais do que simplesmente uma linguagem de formatação para o desenvolvimento de páginas para a Internet.

Além de possibilitar a utilização de recursos multimídia como vídeo, som e imagem permitem definir tamanho, estilos, cores, efeitos para as fontes, além da aplicação de parágrafos, alinhamentos, tabulações, inclusive fazer o uso de listas, tabelas, formulários e diversos outros recursos que dão um formato muito mais agradável às informações que serão apresentadas pelos Web Browsers.

○ Web Browser ou navegador é como chamamos o aplicativo responsável em entender e exibir as páginas desenvolvidas com os recursos para a Internet.

O desenvolvimento em HTML

Por apresentar uma estrutura simples de comandos, ser escrita na forma de texto e em qualquer editor que aceite salvar seus documentos no formato ASCII (texto puro), o HTML é considerado uma linguagem prática e de fácil desenvolvimento.

Seus comandos são expressos na forma de tags (etiquetas), que formam um conjunto de palavras chaves que permitem ao navegador identificar o que deverá fazer com a informação que será apresentada.

Apesar de existirem programas para realizar a conversão de formatos e a edição de documentos em HTML como Front Page, Dreamweaver e muitos outros que tornam muito mais simples esta tarefa. Nós utilizaremos o Bloco de Notas do Windows como ferramenta para programar em HTML, pois deste modo, teremos uma compreensão melhor da linguagem como um todo e das especificidades das diversas tags que a compõe.

Algumas das vantagens de se programar deste modo é a familiarização obtida através do contato direto na utilização dos comandos e a flexibilidade de escolhermos soluções a fim de gerarmos páginas mais organizadas e livres de trechos de códigos desnecessários.

Para abrir o Bloco de Notas caso o mesmo esteja devidamente instalado no seu computador, siga os passos descritos abaixo:

- Ative o menu Iniciar pressionando (ctrl + esc).
- Desça com a seta até o item de menu Programas e pressione a seta para a direita.
- Caso o primeiro item apresentado seja os Acessórios, pressione novamente a seta para a direita ou em caso contrário, desça com a seta até localizá-lo e em seguida pressione a seta para a direita.
- Dentro do menu Acessórios desça com a seta até localizar o item Bloco de Notas e pressione a tecla (enter) para abri-lo. Pronto, o Bloco de Notas estará preparado para a digitação das tags que comporão a sua página em HTML.

Os arquivos e extensões do HTML

Os arquivos gerados em HTML devem ser salvos com as extensões (.htm ou .html).

A página principal de um site, também conhecida por Home Page, na maioria dos servidores que hospedam sites recomenda que este arquivo deva ter o nome de index, portanto terá o seguinte formato: index.htm ou index.html. Caso seja adotada uma nomenclatura diferente, esta informação é fornecida pelos administradores do servidor de hospedagem.

Para testar os documentos que desenvolveremos no decorrer deste curso, não é necessário estar conectado à Internet, basta localizar no Windows Explorer a pasta onde foi salvo o arquivo com uma das extensões referidas acima e pressionar a tecla (enter) sobre o mesmo, para que este seja aberto e executado automaticamente pelo Web Browser.

Caso o seu navegador esteja configurado para exibir automaticamente a janela para estabelecer uma conexão, basta cancelar esta operação e em seguida a página requisitada será carregada.

Com o navegador aberto, uma outra forma de visualizar os resultados após o desenvolvimento de um documento HTML salvo localmente, é acessando o menu Arquivo e na opção abrir, digitar o caminho completo deste documento.

Incompatibilidade para o reconhecimento de código entre os navegadores

A linguagem HTML, por ser interpretada, pode apresentar variações de acordo com o navegador padrão usado para visualizar as páginas da Internet, portanto, alguns comandos poderão não surtir o efeito desejado caso o navegador no qual está apresentando uma determinada página, não esteja preparado ou configurado para entender e executar tal recurso.

Mas fique tranquilo, pois os principais browsers disponíveis no mercado e que são utilizados por mais de 95% dos usuários da Internet, o Internet Explorer da Microsoft e o Netscape Navigator da Netscape suportam todos os principais recursos disponíveis da linguagem HTML, diferenciando-se às vezes apenas na sintaxe de algumas etiquetas ou na oferta de algum recurso diferenciado.

Pelo fato dos navegadores Internet Explorer e Netscape Navigator serem os mais utilizados pela grande maioria dos usuários da Internet, estaremos direcionando os nossos estudos com base principalmente nestes dois aplicativos.

Sobre o W3C

O World Wide Web Consortium (W3C), fundado em outubro de 1994 por Tim Berners-Lee, é o órgão responsável em propor e regulamentar novos recursos mais sofisticados, analisar sugestões e implementar novas versões para o padrão da linguagem HTML.

Este consórcio é representado e liderado pelo MIT/LCS (Massachusetts Institute of Technology / Laboratory for Computer Science), pelo CERN (European Laboratory for Particle Physics), pelo INRIA (Institut National de Recherche en Informatique et Automatique). Na Europa a partir de abril de 1995 e em seguida, pelo Keio (University of Japan) na Ásia a partir de 1996.

Para saber mais sobre o W3 Consortium, acesse o seguinte endereço na Internet (em inglês):

<http://www.w3.org/>

Etiquetas ou TAGS

Os comandos usados no desenvolvimento de páginas em HTML são chamados de marcações, etiquetas ou pelo seu termo correspondente em inglês, tags, que são utilizadas para indicar ao navegador qual ação deverá ser executada durante a montagem da página.

A maioria das tags é formada e a diferença existente nas demais, é que estas não precisam ser fechadas. No geral, todas são identificadas pelos sinais delimitadores:

< >

As que são formadas em pares, a primeira parte significa a abertura da tag, em seguida, o objeto no qual sofrerá alguma ação (este objeto pode ser uma imagem, um controle, um texto etc.) e por último, o fechamento da tag indicando o término desta ação.

Exemplo representativo de uma tag formada em par:

```
<tag_latir> OBJETO DA AÇÃO </tag_latir>
```

Verifique que a tag está delimitado pelos sinais de < (menor que) e > (maior que) e que no fechamento da tag, se usa o sinal / (barra) juntamente com o seu respectivo nome.

A maioria das tags possui atributos que definem suas particularidades. Normalmente, este atributo é definido por seu nome, seguido pelo sinal de = (igual) e por último, seu valor declarado entre os sinais de "" (aspas). Quando declarado mais de um atributo para uma tag, estes deverão estar separados por um espaço em branco.

Exemplo representativo de uma tag com atributos:

```
<tag_cachorro cor_do_pelo="amarelo" altura="80" raça="labrador"> OBJETO DA AÇÃO  
</tag_cachorro>
```

Neste caso, a nossa tag (tag_cachorro) define através dos seus atributos, quais serão as características do objeto cachorro. Esta mesma tag pode conter outros atributos que não foram modificados e, portanto, estes atributos terão seus valores padrões já pré-estabelecidos.

As tags podem ser utilizadas em conjunto para se atingir um resultado esperado. Neste caso, a única restrição é que essas tags obedecem à mesma ordem de fechamento conforme a sua abertura de modo que a primeira tag aberta deverá ser a última a ser fechada, a segunda, deverá ser a penúltima a ser fechada e assim sucessivamente.

Exemplo representativo do uso conjunto de tags:

```
<tag_cachorro raça="vira-lata"><tag_latir>
```

OBJETO DA AÇÃO

```
</tag_latir></tag_cachorro>
```

A linguagem HTML não faz distinção entre a utilização de letras maiúsculas e minúsculas na escrita das tags, mas para deixar o documento com um aspecto mais organizado, é interessante manter sempre um padrão. Pelo contrário, como em qualquer outra linguagem de programação, a sintaxe dos comandos deve ser rigorosamente respeitada, para que se consiga atingir o resultado desejado.

Por exemplo:

`<tag_latir>` (em minúscula) é igual a `<TAG_LATIR>` (em maiúscula)
`<tag_latir>` (com o sinal de sublinhado) é diferente de
`<taglatir>` (sem o sinal de sublinhado)

No primeiro exemplo, a diferença existente pela utilização de letras maiúsculas e minúsculas, não afetará a funcionalidade desta tag, porém no segundo exemplo, a falta do sinal de `_` (sublinhado), fará com que o browser ignore esta tag, não executando a sua ação.

Durante o desenvolvimento de uma página em HTML, a utilização de espaços em branco e de quebras de linha, não são respeitados pelo browser na execução dessa página. Existem tags específicas para se atingir tais efeitos.

Portanto, uma página em HTML pode ser escrita totalmente em uma única linha do editor de textos e ao ser executada, apresentar um belo aspecto visual, porém para a realização de manutenções para alteração ou atualização, o ideal é que o código fonte esteja bem escrito e de uma forma bem organizada.

Código fonte em HTML, é o conjunto escrito de tags que ao ser executado pelo navegador, compõem a página final em HTML.

No decorrer deste curso, nas demonstrações práticas, desenvolveremos esta idéia da estrutura e da organização dos códigos fontes.

Estrutura básica de uma página em HTML

Agora que você está mais familiarizado com os termos e conceitos do HTML, vamos aprofundar um pouco mais os nossos estudos nesta linguagem.

A estrutura básica de um documento HTML, é composta por três blocos principais.

O primeiro bloco engloba os dois outros, sendo aberto logo no início do documento e fechado por último.

É esse bloco que indica ao browser que o documento é uma página HTML.

A tag que define este bloco é:

`<html>` (abertura do bloco)

`</html>` (fechamento do bloco)

O segundo bloco trata-se do cabeçalho e vem logo após a abertura da página HTML que foi demonstrada acima.

São através do cabeçalho que o browser identifica várias informações importantes para que possa exibir uma página requisitada, mas para nós desenvolvedores em HTML, basta sabermos que é obrigatório declararmos apenas o título dentro do cabeçalho, pois as demais informações são negociadas pelo próprio navegador.

É importante que fique bem claro que este título, é referente ao topo da janela do navegador e não ao título do documento que será exposto na página.

É interessante que este tenha relação com o conteúdo do documento, pois alguns sites de busca utilizam este título da janela como identificador do conteúdo apresentado na página.

Tags usadas na declaração do cabeçalho:

`<head>` (abertura do bloco de cabeçalho)

`<title>` (abertura da tag usada para colocar o título da janela)

Meu título desta janela (texto que aparecerá como título)

`</title>` (fechamento da tag usada para colocar o título na janela)

`</head>` (fechamento do bloco de cabeçalho)

O terceiro bloco é referente ao corpo da página, onde realmente serão aplicados todos os recursos multimídia e de formatação disponíveis na linguagem HTML para a exibição de seu conteúdo.

Tag utilizada para delimitar o corpo da página:

`<body>` (abertura do corpo da página)
`</body>` (fechamento do corpo da página)

Esta tag, delimitadora do corpo da página, possui alguns atributos no qual estudaremos mais tarde como e para que utilizá-los.

Veja agora, um exemplo completo de uma estrutura básica de uma página em HTML. Observe bem a distribuição das tags na montagem do documento.

```
<html>  
<head>  
<title> MEU TÍTULO DA JANELA </title>  
</head>  
<body>  
</body>  
</html>
```

O uso de acentos e caracteres especiais

A maioria das versões atuais dos navegadores em português aceita o uso de acentos e caracteres específicos do nosso idioma, porém essa simbologia não é usada na grafia de muitos outros países e, portanto, seus navegadores não estão preparados para interpretar e exibir estes símbolos automaticamente, além dos navegadores de versões mais antigas.

Para suprir esta diferença existente na grafia de outros idiomas permitindo que os sites estrangeiros ao serem visitados conservassem suas particularidades lingüísticas, foi criada uma tabela padronizada desses caracteres especiais e deste modo, os navegadores são capazes de identificá-los através de tags e exibi-los normalmente, indiferentemente da versão ou do idioma padrão do navegador.

Algumas características das tags utilizadas para o uso de caracteres especiais:

- São delimitadas pelos sinais & (e comercial) e ; (ponto e vírgula)
- São escritas somente em letras minúsculas
- Não possuem parte de fechamento além dos sinais delimitadores
- O caracter que receberá a ação (acento ou símbolo gráfico) na maioria das vezes está embutido imediatamente após o primeiro sinal delimitador da tag e este, pode estar tanto em maiúscula como em minúscula conforme a necessidade.

Para inserir um acento ou um caracter especial, utilize a tag neste formato:
&*tag_modifica;

Onde o sinal asterisco, representa o caracter (maiúscula ou minúscula no caso de letras) que sofrerá um efeito determinado pela descrição da tag_modifica.

Vejamos agora, alguns exemplos do uso desses caracteres especiais, mais comumente usados durante a edição de textos em HTML.

Uso do acento agudo:

- Árvore = árvore
- Capítulo = capítulo
- Paletó = paletó

Uso do acento grave ou crase:

- À = à
- Àquela = àquela

Uso do acento circunflexo:

- Êxito = êxito
- Importância = importância
- Avô = avô

Uso do trema:

- Tranqüilidade = tranqüilidade
- Líqüido = líqüido

Uso do acento til:

- Não = não
- Aviões = aviões

Uso do cedilha:

- Caça = ca¸a
- Informação = informa¸ão
- Maça = ma¸ã

Uso de números ordinais:

- 1 (primeiro) = 1º
- 1 (primeira) = 1ª
- 3 (terceiro) = 3º

Uso de frações (existem somente as possibilidades descritas abaixo):

- 1/4 (um quarto) = ¼
- 1/2 (meio) = ½
- 3/4 (três quartos) = ¾

Uso de expoentes (existem somente as duas possibilidades descritas abaixo):

- 5² (5 ao quadrado) = 5²
- 2³ (2 ao cubo) = 2³
- 9² (9 ao quadrado) = 9²
- 6³ (6 ao cubo) = 6³

Alguns outros caracteres especiais:

- < (menor que) = <
- > (maior que) = >
- & (& comercial) = &
- " (aspas) = "

Uma lista completa dos caracteres especiais poderá ser acessada no seguinte endereço (em inglês):

<http://www.w3.org/TR/1999/REC-html401-19991224/sgml/entities>

Definindo cores nos documentos HTML

Ao desenvolver em HTML, existem muitas oportunidades para se fazer o uso de um recurso visual muito explorado, a aplicação de cores.

No decorrer deste curso, estaremos aprendendo a aplicar cor para o fundo da página, para as fontes, para as tabelas e em outras situações que for possível o seu uso.

O bom senso e a criatividade são os ingredientes necessários para uma excelente utilização deste recurso.

Uma das regras básicas que nos auxilia na hora de escolhermos as cores que aplicaremos em uma página, é a utilização de cores contrastantes entre o fundo da página em relação à cor da fonte utilizada, pois deste modo, o leitor não precisará de grandes esforços para a sua visualização. Logo se o fundo da página estiver preenchido por uma cor clara, a fonte utilizada deverá ter uma tonalidade escura e vice-versa.

Exemplos:

- Uma página com o fundo branco ou amarelo, fará contraste principalmente com a utilização de fontes com as cores vermelha (mais utilizada nos títulos de documentos), azul, preta etc.
- Caso o fundo do documento esteja em vermelho, azul ou preto, para contrastar deveremos utilizar principalmente as fontes com as cores branca ou amarela.

Não é muito recomendado o uso de cores escuras para o fundo da página, mas como foi colocada anteriormente, a definição das cores dependerá do conteúdo que será exposto, das características comuns do seu público-alvo e sempre deverá ser ponderada pelo bom senso e pela sua criatividade.

Em todo o caso, é sempre bom solicitar a opinião de uma ou mais pessoas, para que através de seus relatos, você possa identificar as cores que serão mais apropriadas para compor os elementos da página. Lembre-se sempre que o desenvolvimento de um site deve visar agradar seus visitantes e não somente ao seu criador.

A definição de cores em páginas HTML é feita com base no padrão RGB. Este padrão é baseado nas três cores do qual originou seu nome (red "vermelho", green "verde" e blue "azul") e a partir dessas cores básicas, são derivadas as demais cores.

As cores são geralmente definidas através de um atributo específico no qual recebe como valor, um código do padrão RGB em hexadecimal equivalente à cor que se deseja aplicar, precedido do sinal #.

Algumas cores podem ser definidas diretamente pelo seu nome em inglês, mas deste modo, não é possível explorar características como a intensidade do brilho e a tonalidade das cores, por esta vantagem, recomendamos a utilização do padrão RGB com seus valores em hexadecimal, apesar de ser um pouquinho mais complicado o seu uso.

A diferença entre a base decimal para a hexadecimal, é a quantidade de unidades representativas existentes em cada uma dessas.

- Base decimal (10 unidades):

0<1<2<3<4<5<6<7<8<9

- Base hexadecimal (16 unidades):

0<1<2<3<4<5<6<7<8<9<a<b<c<d<e<f

- Sintaxe genérica para aplicação de cores:

<tag_que_permite_cor atributo_cor="#RRGGBB">

- Objeto da ação:

</tag_que_permite_cor>

Onde: A (tag_que_permite_cor) é uma tag no qual definirá cor seja para o fundo da tela usando a tag (<body>), seja para definir cores para as fontes através da tag (), para definir cores para o fundo ou célula de uma tabela etc.

O (atributo_cor) é um atributo genérico que pode ser usado para definir cor do fundo "bgcolor", cor das fontes "color", bordas de objetos quando presentes "bordercolor" etc.

O valor recebido pelos atributos acima, é composto por seis dígitos em hexadecimal, sendo (2 dígitos) para a cor vermelha "red", (2 dígitos) para a cor verde "green" e mais (2 dígitos) para a cor azul "blue". O sinal # deve sempre preceder estes (6 dígitos) como o demonstrado na sintaxe acima.

Os dois dígitos em hexadecimal usado para cada uma das cores, representam um único valor e pode variar desde seu valor mínimo (00 na base hexadecimal = 00 na base decimal) que significa a ausência total da cor, até seu valor máximo (ff na base hexadecimal = 255 na base decimal) que significa o total da tonalidade de uma cor aplicada.

Veja os exemplos:

- atributo_cor="#ff0000" – Representa a cor vermelha, pois nos dois primeiros dígitos que são referentes a esta cor, está representado com o seu valor máximo (ff). Perceba que os dígitos referentes às cores verde e azul estão com ausência total de cor (0000).
- atributo_cor="#00ff00" – Neste exemplo, a cor aplicada será a verde, por estar com seus dois dígitos preenchidos pelo valor máximo, enquanto a cor vermelha que a antecede e a cor azul que é posterior estão respectivamente com seus valores representados por (00).
- atributo_cor="#0000ff" – Neste momento, você já deve ter entendido o funcionamento do padrão RGB e deve estar pensando que desta forma, está representada a cor azul. Se pensar assim, parabéns, pois os dois dígitos referentes a esta cor, estão preenchidos com o seu valor máximo (ff).

Quanto maior for o valor empregado em cada uma das cores, esta tende a ser mais clara e quanto menor o valor empregado, a cor tende a ser mais escura.

Exemplos:

- A cor verde representada por "#00ff00" é mais clara do que o verde representado por "#00bb00", que por sua vez ainda é mais claro do que o verde representado por "#008000".
- Atributo_cor="#ffffff" – Representa a cor branca (foi empregado o valor máximo em todas as cores).
- Atributo_cor="#000000" – Representa a cor preta (ausência de todas as cores).

Pelo que notamos até o momento, através da combinação entre as três cores básicas do padrão RGB, é possível originar as demais cores além de poder definir as suas tonalidades.

Veja na próxima página a tabela com as 16 cores reconhecidas pelos navegadores (mais utilizadas), tanto pelo nome da cor em inglês (neste caso é indiferente o uso de letras maiúsculas ou minúsculas), como pelo seu código hexadecimal correspondente do padrão RGB.

Definindo cores nos documentos HTML

Navegue na tabela, utilizando-se da tecla (tab) observe que serão apresentados na coluna da esquerda os nomes das cores e na coluna da direita, seu respectivo código em hexadecimal.

COLOR	VALUE
Aqua	#00FFFF
Fuchsia	#FF00FF
Lime	#00FF00
Silver	#C0C0C0
Yellow	#FFFF00
Black	#000000
Blue	#0000FF
Gray	#808080
Green	#008000
Maroon	#800000
Navy	#000080
Olive	#808000
Purple	#800080
Red	#FF0000
Teal	#008080

<http://www.w3.org/Graphics/Color/sRGB>

Para obter uma lista com a representação das demais cores no padrão RGB, acesse o endereço (em inglês):

<http://www.w3.org/TR/css3-color>

Inserindo comentários

Para contribuir com a organização e identificação dos trechos de código HTML, você pode contar com a possibilidade de inserir linhas de comentário durante o processo de implementação de uma página.

Um comentário pode ser curto, abrangendo uma única linha ou ocupar várias linhas ao mesmo tempo, inclusive linhas em branco.

Um comentário é composto por três etapas:

- Abertura do bloco de comentário – Utiliza-se o sinal de < (menor que), um ponto de ! (exclamação), seguido por – (dois hífen).
- Inserção do comentário – Você pode inserir informações diversas ou tags da linguagem HTML (um comentário não é exibido pelo navegador, o mesmo somente pode ser visualizado durante o modo de edição da página).

Fechamento do bloco de comentário – Para encerrar um comentário utiliza-se – (dois hífen), seguido pelo sinal de > (maior que).

Veja abaixo a inserção de um comentário abrangendo uma única linha:

```
<!-- Este é um pequeno comentário. -->
```

Observe a inserção de um comentário em múltiplas linhas:

```
<!-- Observação: Os itens: Ótimo Bom Regular Ruim Péssimo
```

```
Deverão ser inseridos na análise do conteúdo do site. -->
```

Formatando documentos para WEB

Neste módulo, vamos estudar os diversos recursos de formatação que a linguagem HTML nos disponibiliza, para a composição de documentos para a Web.

A tarefa de formatar um documento comum quando feita, por exemplo, em um editor de textos automatizado como no Microsoft Word, se torna extremamente simples devido ao fato dos inúmeros comandos de formatação, estarem dispostos dentro de uma interface gráfica, permitindo realizar as ações de formatação de um modo intuitivo e prático.

Para a aplicação das formatações em HTML quando não utilizado uma ferramenta de rápido desenvolvimento, faz-se necessário à utilização de um conjunto de tags específicas que são usadas desde a realização de ações simples como uma quebra de linha, a inserção de um novo parágrafo ou um alinhamento, até a definição de detalhes das características da fonte, como o tipo, o tamanho, a cor, o estilo etc.

Estudaremos os principais recursos de formatação disponíveis na linguagem HTML, visando entender as suas particularidades conceituais, bem como a sua aplicação prática.

Quebra de linha

Aprenderemos agora a aplicar o recurso de quebra de linha e parágrafo na construção de páginas em HTML.

Como vimos anteriormente, no HTML não é possível quebrar uma linha ou inserir um parágrafo simplesmente com o pressionamento da tecla (enter) como num editor de textos normal. Existem tags próprias para realizar tais formatações.

Para efetuarmos uma quebra de linha, deve-se usar a tag `
`, do inglês break, que fará a interrupção da linha atual e forçará o deslocamento do elemento seqüente para a próxima linha.

Vamos entender por elemento dados como (texto, imagem, objeto de vídeo, objeto de som e controles) de acordo com o contexto empregado.

Resultado em HTML:

**Esta linha,
Foi quebrada após a vírgula.**

Perceba que imediatamente após o uso da tag `
`, o restante do texto, foi passado para a próxima linha. Caso houvesse utilizado duas vezes a tag `
`, seria inserido uma linha vazia entre as palavras antecedentes e posteriores às tags, como abaixo:

Exemplo de uma quebra de linha:

Esta linha, `

` foi quebrada após a vírgula.

Resultado em HTML:

Esta linha,

foi quebrada após a vírgula.

Parágrafos

O parágrafo é um recurso bastante utilizado na edição de textos, onde permite separar as informações de forma a torná-las mais fáceis de serem compreendidas e portanto, torna-se indispensável a sua utilização também nos documentos para a Web.

Tag utilizada para aplicação de parágrafos:

`<p>` Elemento HTML

Quando utilizada dessa forma, essa tag mantém o elemento alinhado à esquerda e como observado acima, não é necessário fechá-la com `</p>`.

Para realizar parágrafos com outros alinhamentos, esta tag, possui o atributo (align) que pode assumir os seguintes valores:

- **Align="left" » alinhamento à esquerda (padrão)**
- **Align="right" » alinhamento à direita**
- **Align="center" » alinhamento centralizado**
- **Align="justify" » alinhamento justificado**

Neste caso, usando-se o atributo (align), torna-se obrigatório o fechamento da tag com o `</p>`, para delimitar qual trecho do texto, imagem ou controle, sofrerá o referido alinhamento.

Também podemos centralizar um elemento na página, utilizando a tag descrita abaixo:

`<center> ELEMENTO </center>`

A tag `<center>` é muito utilizada na centralização de títulos de documentos e imagens.

Uso de parágrafos e da quebra de linha

Observe com atenção a distribuição dos comandos HTML (TAGS) na construção do documento, navegando no código exposto na caixa de texto ao lado.

Para melhor compreensão das TAGS, navegue soletrando os seus caracteres.

```
<html>
<head>
<title> Parágrafos e quebra de linha </title>
</head>
<body>
<center> Texto alinhado ao centro </center>
<br><br>
```

`<p align="justify">` Acima, utilizamos duas vezes a tag que faz a quebra de linha, para que dessa forma pudéssemos deixar linhas vazias entre o título do documento e este parágrafo que está usando um alinhamento justificado. Perceba que é interessante usar o alinhamento justificado quando o texto se estender por mais de uma linha, pois assim, este alinhamento fará com que o texto fique alinhado tanto do lado direito, como do lado esquerdo. `</p>`

`<p>` Este parágrafo, está alinhado à esquerda por padrão, portanto, não precisamos fechá-lo com `</p>`.

`<p>` Agora, vamos quebrar esta linha após a vírgula, `
` usando a tag `
`.

`<p>` No parágrafo abaixo, usaremos um alinhamento à direita para colocarmos a assinatura, logo após utilizarmos o recurso para a quebra de linha.

```
<br><br>
<p align="right"> Assinado: Programador </p>
</body>
</html>
```

Linhas horizontais

É interessante utilizarmos uma linha horizontal sempre que desejarmos separar o título do restante do documento, ou ainda, quando julgarmos apropriado dividir em blocos um documento que contenha informações diferenciadas.

Para inserirmos uma linha horizontal, devemos utilizar a tag ímpar `<hr>` (horizontal rules), que não possui uma parte referente ao seu fechamento.

Elemento ou elementos HTML após a linha horizontal

Os principais atributos da tag `<hr>` são:

size » Este atributo determina a altura do tracejado da linha. Podemos utilizar um valor absoluto de "1" a "100".

Exemplo: `<hr size="50">`

width » Este atributo é utilizado para definir o tamanho da linha horizontalmente, em relação a página. Podemos utilizar um valor absoluto (em pixels) ou um valor relativo (em porcentagem).

Exemplo:

`<hr size="10" width="50%">` - A linha terá a metade do tamanho da página horizontalmente.

noshade » Inserindo este atributo é eliminado a sombra da linha, dando um efeito tridimensional.

Exemplo: `<hr noshade>`

align » Com este atributo podemos definir um alinhamento para a linha, podendo ser:

- **Align="left"** – Alinha à esquerda da página (padrão).
- **Align="center"** – Alinha ao centro da página.
- **Align="right"** – Alinha à direita da página.

Exemplo:

```
<hr size="10" width="50%" align="center" noshade>
```

No exemplo, definimos que a linha horizontal terá "10" unidades de altura, ocupará "50%" do tamanho horizontal da página, estará alinhada ao centro e com o efeito tridimensional.

DICA: Ao encontrar uma linha horizontal, o leitor de telas Virtual Vision a pronuncia como "separador"

Formatando fontes

A possibilidade de se escolher e de se modificar os principais atributos na formatação das fontes, é um importante recurso que quando utilizado com criatividade e bom senso, torna-se um grande aliado na composição visual de uma página em HTML.

Existe uma tag na linguagem HTML exclusiva para a formatação das fontes. Utilizando esta tag, podemos definir através dos atributos disponíveis, detalhes como o tipo, o tamanho e a cor da fonte.

Sintaxe da tag ****:

```
<font face="nome_do_tipo_da_fonte" size="tamanho" color="cor">
```

Texto que receberá a formatação ****

Onde: Todos os valores definidos nos atributos, serão aplicados no texto delimitado pela abertura e pelo fechamento da tag font.

O atributo **face**:

Este atributo define o nome do tipo da fonte no qual a informação será apresentada.

É importante saber que a fonte definida no atributo (face), deve estar devidamente instalada no computador da pessoa que está requisitando esta página, por isso, recomenda-se a utilização apenas das fontes mais tradicionais e que já vêm instaladas com o sistema operacional ou por aplicativos que são comumente utilizados como processadores de textos.

Como exemplos de fontes mais usadas podemos citar: Arial, Verdana, Helvetica, Sans-Serif, Times New Roman etc.

Podemos declarar separados por vírgula, mais de uma fonte no atributo (*face*), pois deste modo, caso a primeira fonte não esteja instalada no computador do visitante, será utilizada a segunda e caso a situação se repita com a segunda fonte, a terceira será utilizada e assim sucessivamente.

O ideal é disponibilizarmos por precaução no máximo quatro fontes e caso não haja nenhuma das fontes instaladas, será utilizada a fonte padrão para as páginas em HTML, definida nas configurações do navegador que originalmente é a fonte Times New Roman.

A sintaxe para declarar mais de uma fonte é a seguinte:

```
<font face="Arial, Verdana, Helvetica, Sans-Serif"> TEXTO </font>
```

O atributo **size**: Este atributo é utilizado para definir o tamanho da fonte, podendo ser feita de duas formas:

- **Valor absoluto:** Informamos ao atributo um valor referente ao tamanho da fonte, podendo ser utilizados números de "1" à "7", onde quanto maior for o número inserido, maior será o tamanho aplicado de acordo com a escala de tamanho das fontes.
- **Valor relativo:** Com base no tamanho padrão da fonte que é igual a "3", somamos ou subtraímos um valor pelo qual definirá o tamanho atual da fonte.

Exemplos da utilização do atributo (*size*):

```
<font face="arial" size="1"> valor (1) equivalente ao tamanho (9) </font>  
<font face="arial" size="2"> valor (2) equivalente ao tamanho (10) </font>  
<font face="arial" size="3"> valor (3) equivalente ao tamanho (12) </font>  
<font face="arial" size="4"> valor (4) equivalente ao tamanho (14) </font>  
<font face="arial" size="5"> valor (5) equivalente ao tamanho (18) </font>  
<font face="arial" size="6"> valor (6) equivalente ao tamanho (24) </font>  
<font face="arial" size="7"> valor (7) equivalente ao tamanho (36) </font>
```

Os tamanhos das fontes indicados ao lado, são equivalentes às definições das fontes utilizadas pelo editor de textos Microsoft Word. Serão válidas somente, caso as configurações originais do navegador referente às fontes, não tenham sido modificadas.

○ atributo **color**:

○ atributo (color) é utilizado para definir a cor da fonte que será aplicada no texto delimitado pela abertura e fechamento da tag font.

Para a definição das cores das fontes, é utilizado o padrão RGB que segue as regras contidas no módulo I, no texto "A aplicação de cores em páginas Web".

Exemplo da utilização do atributo (color):

```
<font face="verdana" size="4" color="red"> texto em vermelho </font>  
<font face="comic sans ms" size="+1" color="#ff0000"> texto em vermelho </font>  
<font face="arial" size="3" color="black"> texto com a cor preta </font>  
<font face="times new roman" size="3" color="#000000"> texto com a cor preta </font>
```

Tags complementares para a formatação das fontes

Vejamos agora mais três tags adicionais, utilizadas para a formatação das fontes. São elas:

- <basefont>
- <small>
- <big>

As três tags citadas acima, são utilizadas em par, portanto possuem uma parte referente a sua abertura e outra referente ao seu fechamento.

A tag <basefont> deve ser aberta logo no início do corpo do documento e fechada somente no final deste.

Isso se deve pelo fato de que todas as características de formatação das fontes, definidas nos atributos (face), (size) e (color) declarados na abertura da tag <basefont>, serão aplicadas como padrão a todas as partes do documento que não sofreram nenhuma modificação através da tag .

A sintaxe utilizada na declaração da tag `<basefont>` é idêntica à utilizada pela tag `` como o demonstrado pelo exemplo abaixo:

`<body>` - Abertura do bloco referente ao corpo do documento.

`<basefont face="arial" size="3" color="#000088">`

(IMPLEMENTAÇÃO DO CORPO DO DOCUMENTO)

`</basefont>`

`</body>` - Fechamento do bloco referente ao corpo do documento.

Tanto a tag `<small>`, como a `<big>`, podem ser utilizadas como um complemento do atributo (size), tendo a função de diminuir ou aumentar o tamanho da fonte respectivamente conforme a necessidade.

O aumento ou a diminuição do tamanho da fonte estará sempre relacionado ao tamanho atual definido anteriormente pela tag `<basefont>` ou pela tag ``.

Exemplo da diminuição do tamanho da fonte:

``

Observe a `<small>` diminuição `</small>` do tamanho da fonte.

``

No exemplo acima, a palavra "diminuição" será exibida com a fonte menor do que o restante da frase.

Exemplo do aumento do tamanho da fonte:

`<basefont face="arial" size="3" color="#000000">`

Observe o `<big>` aumento `</big>` do tamanho da fonte utilizada.

`</basefont>`

No exemplo acima, a palavra "aumento" será exibida com a fonte maior do que o restante da frase.

Formatação das fontes

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> FORMATANDO FONTES </title>
</head>
<body>
<br><br>
<center><font face="comic sans ms, verdana, sans-serif" size="4" color="#ff0000">
FORMATAÇÃO DAS FONTES </font></center>
<hr size="5" width="40%" align="center" noshade>
<br><br>
<p align="justify"><font face="arial" size="3" color="#000000"> O título deste documento, está
em vermelho. Este parágrafo está utilizando o tamanho da fonte menor e com a cor preta.
</font></p>
<p align="justify"><font face="arial" size="3" color="#000099"> Este 2º parágrafo, está
utilizando a fonte com a cor azul escuro. Perceba que para alterar a formatação da fonte utilizada, é
necessário delimitar o texto no qual deseja formatar
com a tag "font", com seus respectivos valores dos atributos definidos. </font></p>
<font face="verdana" size="3" color="#000000">
<p align="justify"> Utilizamos cores e tipos de fontes diferenciados no corpo deste documento, para
exemplificar bem a aplicação desse recurso, mas essas formatações devem ser utilizadas com muito
bom senso e com bastante criatividade de acordo com as necessidades presentes no documento.
</p>
<p align="justify"> Nestes últimos dois parágrafos, estamos usando as mesmas características para
a fonte. Observe que com apenas uma tag "font", abrangemos a formatação para dois parágrafos
subseqüentes, abrindo a tag antes e a fechando somente após o término dos mesmos. </p>
</font>
<br><br>
<hr width="20%" align="right">
<p align="right"><font face="arial" size="2" color="#000000"> PROGRAMADOR INTERNET
</font></p>
</body>
</html>
```

Estilos de formatação para as fontes

Exemplos:

- Usamos o negrito na definição de títulos do documento ou para evidenciar uma determinada palavra ou parte do texto.
- Normalmente utilizamos o itálico para se indicar expressões lingüísticas informais, termos técnicos ou estrangeirismo.
- Sublinhado não deve ser utilizado freqüentemente em páginas Web, pois este estilo é aplicado automaticamente nas ligações (links) entre as páginas, podendo confundir o usuário, porém quando necessário, seu uso se faz presente para se destacar (grifar) uma determinada palavra ou trecho do texto. Este estilo é somente aplicado ao navegador Internet Explorer da Microsoft.
- O uso de caracteres monoespaçados não é muito utilizado nas páginas Web, mas algumas vezes este estilo torna-se apropriado, como por exemplo, quando se deseja alinhar caracteres em linhas subsequentes ou para a definição do título do documento. A desvantagem de se utilizar este estilo no corpo do documento é a utilização desnecessária de espaços, pois note que o espaço na horizontal utilizado pelo carácter "i", é muito menor do que o espaço utilizado pelo carácter "m".

Tags para a definição dos estilos de formatação das fontes:

Para inserir o efeito negrito (**bold**), utilize:

` TEXTO EM NEGRITO `

Ou:

` TEXTO EM NEGRITO `

Para inserir o efeito itálico:

`<i> TEXTO EM ITÁLICO </i>`

Para inserir o efeito sublinhado:

`<u> TEXTO SUBLINHADO </u>`

Para inserir caracteres monoespaçados:

`<tt> CARACTERES MONOESPAÇADOS </tt>`

Note que os estilos de formatação, podem ser combinados e utilizados em conjunto para se obter um resultado desejado, conforme demonstra os exemplos abaixo.

Exemplos:

```
<b><i> TEXTO EM NEGRITO E ITÁLICO </i></b>  
<b><i><u> TEXTO EM NEGRITO, ITÁLICO E SUBLINHADO </u></i></b>  
<tt><u> TEXTO MONOESPAÇADO E SUBLINHADO </u></tt>
```

Definição dos estilos de formatação para as fontes

Observe com atenção a distribuição dos comandos HTML (tags) na construção do documento na caixa de texto ao lado.

```
<html>  
<head>  
<title> FORMATAÇÃO DE ESTILOS </title>  
</head>  
<body>  
<br><br>  
<center><font face="Verdana" size="4" color="#ff0000"><b> TÍTULO DO DOCUMENTO  
</b></font></center>  
<br><br>  
<font face="Arial" size="3" color="#0000cc">  
<p align="justify"> Acima, foi colocado o título do documento em negrito, centralizado e com a  
fonte Verdana, tamanho 4 e cor vermelha. Este parágrafo, está escrito com a fonte Arial, tamanho 3  
e cor azul, além de usar um alinhamento justificado. </p>  
<p align="Justify"> Usando estes recursos para a formatação de estilos das fontes, a  
<i><u>"World Wide Web"</u></i>, também conhecida simplesmente por  
<tt><i>"Web"</i></tt>, torna-se muito mais agradável a visualização de seus documentos.  
</p>  
<p align="Justify"> Tanto neste parágrafo quanto no acima, <i><u>"World Wide  
Web"</u></i> está entre aspas, em itálico e sublinhado. A palavra <tt><i>"Internet"</i></tt>,  
se apresenta em ambos parágrafos entre aspas, em itálico e com caracteres monoespaçados. </p>  
<p align="Justify"> Abaixo, a assinatura está alinhada à direita e delimitada pela <i>tag</i>  
<tt> </tt>.  
<br><br>  
<p align="right"> ASSINADO: <tt> Programador Internet </tt></p>  
</font>  
</body>  
</html>
```

Efeitos sobrescrito e subscrito

Como na edição de documentos comuns, a construção de documentos para a Web conta com a possibilidade de aplicar os efeitos sobrescrito (elevando) ou subscrito (rebaixando) um ou mais caracteres de um texto em relação à posição da linha.

Este recurso é indispensável para a formatação de equações químicas, matemáticas e demais situações similares.

Para inserir os efeitos de sobrescrito e subscrito, existem as respectivas tags **<sup>** e **<sub>**, que são utilizadas em par, tendo uma parte referente à sua abertura e outra referente ao seu fechamento.

Observe que a sintaxe usada na definição destas tags é bastante simples, não sendo necessário a declaração de nenhum atributo complementar.

Sintaxe para a inserção do efeito sobrescrito:

^{VALOR-SOBRESCRITO}

Sintaxe para a inserção do efeito subscrito:

_{VALOR-SUBSCRITO}

Veja os exemplos abaixo:

- A equação (x ao quadrado mais quatro que é igual a vinte) descrita literalmente, pode ser matematicamente esquematizada como:

x² + 4 = 20

- A fórmula química do Ácido Sufúrico que contém duas moléculas de Hidrogênio, uma de Enxofre e quatro de Oxigênio, pode ser esquematizada como:

H₂SO₄

O uso de espaçamentos e tabulações

Aprendemos que ao editar um documento em HTML, os espaços adicionais são ignorados pelo browser ao interpretar este documento.

Para que possamos inserir mais de um espaço em seqüência, devemos utilizar o caracter especial ** **; tal for correspondente ao tamanho do espaçamento desejado, por exemplo:

Para inserirmos três espaços entre duas palavras, devemos codificar da seguinte forma:

PALAVRA1 PALAVRA2

Observe que utilizamos o caracter especial ** **; por três vezes para se alcançar o objetivo proposto no exemplo acima.

Em algumas situações é necessário saltar um espaço em relação à margem de um texto ou ainda, inserir tabulações entre partes de um documento. Para a inserção de tabulações em HTML, contamos com a tag **<blockquote>** usada em par.

A sintaxe da tag **<blockquote>** é bastante simples, não havendo atributos complementares como o demonstrado abaixo:

<blockquote> ELEMENTO **</blockquote>**

O espaço saltado a cada vez que utilizamos a tag **<blockquote>** é de aproximadamente 0,63 CM, podendo ser ampliado com o alinhamento repetitivo desta tag como o feito na demonstração abaixo:

<blockquote><blockquote><blockquote> ELEMENTO
</blockquote></blockquote></blockquote>

Na demonstração acima, o espaço saltado será de aproximadamente 1,9 CM.

Cabeçalhos

Na linguagem HTML podemos utilizar até 6 níveis de tópicos de cabeçalho para a definição dos títulos e subtítulos dos documentos.

Ao definir um cabeçalho, estaremos aplicando ao texto delimitado pela tag correspondente, o estilo negrito e um tamanho específico à fonte de acordo com o nível do tópico selecionado.

O nível do tópico (1) apresentará o tamanho da fonte maior do que o nível (2), que por sua vez será maior do que o nível (3) e assim sucessivamente até o nível (6), no qual apresentará o menor tamanho da fonte.

Sintaxe geral para a definição de cabeçalhos:

```
<h* align="alinhamento"> CABEÇALHO DO NÍVEL DO TÓPICO * </h*>
```

Onde:

O (*) representa o valor correspondente ao nível do tópico do cabeçalho no qual deseja aplicar.

O "alinhamento" é o valor que deverá ser passado ao atributo (align), podendo assumir:

- **Align="left"** – Alinhamento à esquerda (alinhamento padrão não havendo necessidade de ser declarado).
- **Align="center"** – Alinhamento ao centro.
- **Align="right"** – Alinhamento à direita.
- **Align="justify"** – Alinhamento justificado.

Exemplos:

```
<h1 align="center"> CABEÇALHO DO NÍVEL 1, centralizado e COM O TAMANHO DA FONTE 24 </h1>
```

```
<h2 align="left"> CABEÇALHO DO NÍVEL 2, à esquerda e COM O TAMANHO DA FONTE 18 </h2>
```

```
<h3> CABEÇALHO DO NÍVEL 3, à esquerda e COM O TAMANHO DA FONTE 14 </h3>
```

```
<h4 align="center"> CABEÇALHO DO NÍVEL 4, centralizado e COM O TAMANHO DA FONTE 12 </h4>
```

```
<h5 align="right"> CABEÇALHO DO NÍVEL 5, à direita e COM O TAMANHO DA FONTE 10 </h5>
```

```
<h6 align="justify"> CABEÇALHO DO NÍVEL 6, justificado e COM O TAMANHO DA FONTE 8 </h6>
```

Os tamanhos das fontes indicados anteriormente, são equivalentes às definições das fontes utilizadas pelo editor de textos Microsoft Word e somente serão válidas caso as configurações padrões do navegador sejam mantidas.

Um cabeçalho pode ser utilizado em conjunto com a tag de formatação das fontes, porém, o atributo size, deverá ser omitido, pois o tamanho da fonte é definido pelo nível do cabeçalho como foi demonstrado acima. Caso não se use a tag para a formatação da fonte, ao cabeçalho será atribuído à fonte padrão, definida nos navegadores dos usuários.

Exemplo da declaração do cabeçalho com o uso da formatação da fonte:

```
<font face="comic sans ms" color="#ff0000"><h2 align="center"> TÍTULO DO DOCUMENTO </h2></font>
```

No exemplo acima, inserimos o título do documento Centralizado, com a fonte Comic Sans MS, cor vermelha e com o tamanho 18, atribuído pelo uso do nível (2) do cabeçalho.

Exemplo do uso de cabeçalhos

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> USANDO CABEÇALHOS </title>
</head>
<body>
<br><br>
<font face="comic sans ms, verdana" color="#ff0000"> <h2 align="center"> CAPÍTULO I -
TÍTULO PRINCIPAL DO DOCUMENTO </h2></font>
<br><br>
<blockquote><font face="arial" color="#0000ff"> <h3> 1.1 PRIMEIRO SUBTÍTULO
</h3></font></blockquote>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> Este, é o 1º parágrafo deste subtítulo. </p>
<p align="justify"> Este, é o 2º parágrafo deste subtítulo. </p>
<p align="justify"> Continuação deste texto até a sua conclusão. </p>
</font>
<br><br>
<blockquote><font face="arial" color="#0000ff"> <h3> 1.2 SEGUNDO SUBTÍTULO
</h3></font></blockquote>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> Este, é o 1º parágrafo deste subtítulo. </p>
<p align="justify"> Este, é o 2º parágrafo deste subtítulo. </p>
<p align="justify"> Continuação deste texto até a sua conclusão. </p>
</font>
<br><br>
<blockquote><font face="arial" color="#0000ff"> <h3> 1.3 TERCEIRO SUBTÍTULO
</h3></font></blockquote>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> Este, é o 1º parágrafo deste subtítulo. </p>
<p align="justify"> Este, é o 2º parágrafo deste subtítulo. </p>
<p align="justify"> Continuação deste texto até a sua conclusão. </p>
</font>
</body>
</html>
```

Texto pré-formatado

Como uma exceção à regra geral para a formatação de textos em HTML, existe a tag `<pre>` que possibilita a inserção de textos pré-formatados, onde automaticamente é adotada para a exibição uma fonte monoespaçada como o tipo Courier.

Diferentemente da fonte proporcional, onde, por exemplo, o caracter "i" ocupa um menor espaço horizontalmente do que o caracter "m", o tipo de fonte mono espaçada exhibe todos os caracteres com uma mesma largura, semelhante à fonte de um texto gerado através do uso de uma máquina de datilografia..

A principal diferença entre a tag `<pre>` usada para a inserção do recurso de texto pré-formatado da tag `<tt>` apresentada anteriormente, está no fato em que as seqüências de espaços livremente inseridos durante a digitação são mantidas na visualização da página, bem como a quebra de linha feita com o pressionamento da tecla (enter).

Observe um exemplo do uso da tag `<pre>`:

```
<pre>
```

Este texto está delimitado pelo uso da tag `<pre>` e por este motivo não foi preciso utilizar o caracter especial (` `) para inserir os três espaços antes desta frase.

Não foi preciso inserir o `
` para poder fazer a quebra de linha acima.

```
</pre>
```

Qualquer texto inserido após o fechamento da tag `</pre>`, será automaticamente transferido para a próxima linha livre.

A utilização do recurso de texto pré-formatado é ideal para a exposição de esquemas, tabelas simples de dados literais e/ou numéricos, códigos fontes de programação e em outras situações similares.

A sintaxe da tag `<pre>` é bastante simples, sendo utilizada em par e não possuindo atributos complementares.

Na área delimitada pela tag `<pre></pre>` não é aconselhado inserir outras tags de formatação de texto, pois caso contrário, o seu efeito pode ser cancelado. Por um outro lado é fortemente recomendado o uso dos caracteres especiais no processo de edição, principalmente quando for necessário gerar sinais como: `<` (menor que), `>` (maior que), `&` (& comercial), etc.

Hiperlinks

Vamos aprender a utilizar o recurso mais fantástico da linguagem HTML, os hiperlinks, no qual através da simplicidade de acesso a outros documentos interligados por estas marcações, originou-se o conceito de hipertextos, responsável por este fenômeno tão atrativo que se tornou a Internet.

Os hiperlinks, além da sua principal utilização que é a interligação entre as páginas Web, indiferentemente da sua localização física ou entre partes dentro de um mesmo documento, podem ser aplicados para algumas outras finalidades como o download de arquivos e disponibilização de um canal de contato através do correio eletrônico.

A diferença básica na aplicação dos tipos de hiperlinks, está mais relacionada com uma questão conceitual do que com a sintaxe de comando propriamente dita, apresentando apenas na maioria dos casos, uma sutil diferença na passagem do valor para o seu principal atributo.

Vamos apresentar a sintaxe da tag para a inserção de links com seus atributos, além das diferentes aplicações deste interessantíssimo recurso nas páginas em HTML.

Sintaxe para inserção de links

Para inserir um link em HTML, devemos utilizar a tag em par `<a>` derivada da palavra (anchor = âncora), que possui uma parte referente à sua abertura onde são inseridos os atributos e outra referente ao seu fechamento. Abaixo, segue a sintaxe geral para a inserção de links em HTML:

```
<a href="url"> DESCRIÇÃO DO LINK </a>
```

Onde:

(href):

É o atributo principal da tag anchor, usado para indicar a localização do arquivo de destino ao qual o link se refere. O valor "url" representado acima, deverá ser substituído pelo endereço (protocolo://www.servidor.sigla.país/caminho/nome_do_arquivo) no qual deseja referenciar.

Dependendo da aplicação do link, o valor passado para o atributo (href), sofrerá algumas variações, como por exemplo: utilização apenas de parte da URL, uso de um endereço de correio eletrônico etc.

Descrição do link:

Esta descrição será apresentada na página com destaque (a maioria dos Web browsers a coloca sublinhada ou com uma cor diferente do restante do texto). O cursor ao ser posicionado sobre esta descrição, automaticamente é transformado na imagem de uma (mãozinha) indicando ser um hiperlink e ao ser acessado, trará o arquivo referenciado pelo atributo (href).

Verifique no item posterior, os atributos alternativos (accesskey) e (title) que podem ser utilizados como um recurso muito interessante para a promoção da acessibilidade.

(accesskey):

Este atributo é utilizado para atribuir uma tecla de atalho ao link. A tecla de atalho deverá ser composta por um caracter alfanumérico e este, ao ser combinado com a tecla (alt) durante a navegação, focalizará diretamente o link correspondente, indiferentemente da posição atual do cursor dentro da página ativa.

(title):

Este atributo apresentará uma descrição alternativa para o link, no instante em que este receber o foco do cursor. Esta descrição aparecerá dentro de uma moldura e caso o usuário esteja utilizando um programa leitor de telas para navegar por esta página, esta descrição lhe será falada.

Exemplo genérico do uso dos atributos (accesskey) e (title):

```
<a href="url" accesskey="l" title="Descrição usando o atributo title"> Descrição escrita no corpo do documento </a>
```

Note que no exemplo acima, usamos o atributo (accesskey="l"), onde fará com que este link receba o foco quando o usuário ao navegar por esta página, pressionar simultaneamente as teclas (alt+l).

Este link, ao receber o foco, trará a descrição que foi passada como valor ao atributo (title), no caso a mensagem: "Descrição usando o atributo title"

Links internos

Os links internos são utilizados para interligar páginas que estão dentro de um mesmo sistema. Neste sentido, podemos entender sistema como o conjunto de pastas, subpastas e os seus respectivos arquivos que compõem a estrutura organizacional de um site dentro de um servidor.

A pasta principal de um site onde estão contidas as demais subpastas é chamada de pasta raiz.

Nela está o arquivo que contém a Home Page do site (página principal), representada ou chamada automaticamente de um arquivo nomeado como (index.htm ou index.html).

A partir dos links contidos na Home Page, se acessa as demais páginas que compõem as diversas seções de um site, que transparente aos olhos dos visitantes, estão internamente estruturadas e divididas em pastas e subpastas de modo que cada página fique organizada de acordo com o seu respectivo conteúdo, facilitando assim, os processos de alteração e atualização das informações do site.

Neste caso, como os links vão referenciar outras páginas que estão gravadas internamente em um mesmo servidor, não se faz necessário a indicação completa da URL, bastando informar o caminho das pastas e o nome do arquivo a ser chamado.

Caso a referência seja feita entre dois arquivos salvos em um mesmo nível hierárquico de pasta, basta passar como valor apenas o nome do arquivo ao atributo (href).

Imagine a descrição abaixo da estrutura organizacional de um site sobre turismo, para ilustrarmos as possibilidades do uso de links internos em HTML.

Pasta raiz (Brasil):

Contém duas subpastas: nordeste e sul

Arquivo: index.html (Home Page do site com links para os menus dos estados da região Nordeste e do Sul)

Subpasta (nordeste):

Contém dez arquivos: index.html (menu com links para as páginas dos estados da região Nordeste), al.html, ba.html, ce.html, ma.html, pb.html, pe.html, pi.html, se.html e rn.html

Subpasta (sul):

Contém quatro arquivos: index.html (menu com links para as páginas dos estados da região Sul), pr.html, sc.html e rs.html

Vejamos as seguintes situações:

Definição dos links da Home Page para as páginas de menu da região Nordeste e Sul:

```
<a href="/nordeste"> Região Nordeste </a>  
<a href="/sul"> Região Sul </a>
```

Perceba que não foi necessário informar o nome do arquivo referente aos menus, pois os mesmos estão nomeados como (index.html), sendo automaticamente requisitados e carregados pelo navegador.

Links externos

A inserção de links externos permite fazer ligações com qualquer página devidamente conectada à rede mundial de computadores, indiferentemente da sua localização física, desde que esta não possua nenhuma restrição de acesso.

Caso a referência seja feita para a Home Page do site, basta passar como valor para o atributo (href) o endereço deste site na WWW. Agora caso seja uma página interna ao site, deverá ser passado como valor ao atributo (href), a URL completa, incluindo o caminho das subpastas e o nome do arquivo HTML, quando necessário.

Um interessante atributo recomendado para utilizar ao definir links externos, é o (target), onde deverá ser passado como valor o parâmetro (window). Desta forma, quando o visitante clicar em um link externo existente no seu site, a página indicada por você será aberta em uma nova janela sobreposta, mantendo aberta no navegador, a janela anterior com o seu site, facilitando deste modo, o retorno deste visitante.

Exemplos do uso de links externos:

Link para a Home Page do site do W3 Consortium

```
<a href="http://www.w3.org/">http://www.w3.org</a> target="window"> W3 Consortium </a>
```

Link para a lista de cores no padrão RGB, existente numa seção interna do site W3 Consortium

```
<a href="http://www.w3.org/TR/css3-color"> http://www.w3.org/TR/css3-color</a>  
target="window"> Lista de cores </a>
```

Links para download de arquivos

Podemos utilizar o recurso dos links para realizar a transferência de arquivos entre o servidor e a máquina cliente dos visitantes de um site. A este processo dá-se o nome de download.

A prática do download é bastante difundida nos sites da Internet. Através deste processo, é possível disponibilizar para os visitantes inúmeros tipos e formatos de arquivos diferentes do HTML, como por exemplo:

- Documentos do Microsoft Word
- Planilhas do Microsoft Excel
- Apresentações do Microsoft PowerPoint
- Arquivos executáveis
- Arquivos de vídeo e de áudio em diversos formatos etc.

Caso o arquivo referenciado por um link não possa ser aberto automaticamente pelo navegador, será apresentado ao usuário uma caixa de diálogo onde o mesmo deverá optar por salvá-lo (indicando o caminho de destino) ou abri-lo se possuir o aplicativo correspondente para a execução de tal ação.

Observe algumas dicas interessantes para a inserção de links para o download de arquivos em uma página Web:

- Colocar na descrição do link, informações que identificam o tipo ou formato do arquivo, bem como o seu tamanho.
- Recomenda-se a compactação dos arquivos que serão disponibilizados para download, reduzindo deste modo o tempo de conexão utilizado para a realização deste processo.

Para a compactação dos arquivos que serão disponibilizados para o download, é aconselhável utilizar os formatos de maior uso popular como, por exemplo, o .zip.

- Ao disponibilizar para o download um aplicativo composto por vários arquivos resultando em um tamanho muito extenso, fragmente-o compactando-o em diversas unidades pequenas, para que os visitantes interessados em baixar este aplicativo, possam o fazer por etapas, reduzindo deste modo, os problemas causados com as eventuais quedas de conexão.

A sintaxe usada na criação de um link para o download, é quase idêntica a utilizada para referenciar uma página em HTML, onde a única diferença está no fato de que após a indicação do caminho, devemos informar o nome do arquivo no qual será disponibilizado para o download.

Esta referência será feita usando o atributo (href) e deverá seguir os mesmos conceitos usados na construção dos links internos e dos links externos de acordo com o endereço do arquivo ao qual se deseja referenciar.

Veja abaixo, o exemplo genérico da criação de um link para o download de arquivo:

```
<a href="protocolo://www.servidor.sigla.país/pasta/arquivo.extensão"> Arquivo.extensão (1.2 mb)
</a>
```

Âncoras de marcação para documentos HTML

A utilização de âncoras permite a inserção de marcações em determinados pontos de um documento, possibilitando o deslocamento direto para estes pontos, através da criação de referências.

É possível criar referências para marcações existentes dentro de uma mesma página, para um ponto dentro de uma outra página interna do site, ou ainda, para uma marcação presente em uma página de um outro site qualquer, disponível na Internet.

O recurso da criação de âncoras é bastante utilizado como índice de documentos, facilitando o deslocamento do leitor diretamente para um ponto desejado.

Um link pode ao mesmo tempo, referenciar e ser uma marcação. Essa prática é utilizada em páginas demasiadamente extensas possibilitando que o usuário se desloque do início para o fim do documento ou vice-versa. Como valor deste atributo devemos informar o nome da marcação, precedida do sinal #, caso esta marcação pertença à mesma página da referência. Caso a referência seja para uma outra página, o atributo (href) receberá como valor o endereço da página em HTML, seguido do sinal # com o nome da marcação contida nesta página referenciada ao qual se deseja atingir.

Exemplos genéricos de referências para marcações:

```
<a href="#fimdodocumento"> Vai para o fim do documento </a>
```

Esta referência atingirá uma marcação de nome (fimdodocumento), criada dentro desta mesma página onde se encontra esta referência.

```
<a href="/nordeste/ce.html#praias"> Praias cearenses </a>
```

Destino: A marcação será o objeto alvo de uma referência. Com ela, é possível definir um ponto dentro da página, que poderá ser acessado diretamente a partir de referências criadas em qualquer página disponível na Internet, desde que na PÁGINA DESTINO ONDE ESTÁ INSERIDA A MARCAÇÃO, não haja nenhuma restrição de acesso.

Uma marcação de âncora é definida utilizando o atributo (name) que receberá como valor o nome pela qual esta marcação será referenciada.

Pode se utilizar uma marcação de âncora juntamente com outras tags da linguagem HTML, portanto ao criar uma referência para um subtítulo em um texto, o mesmo pode estar, por exemplo, em negrito, centralizado, com uma fonte diferenciada etc.

Exemplos da declaração de marcações:

```
<a name="fimdodocumento"> Fim deste documento. </a>
```

Clicando no link ao qual possui uma referência para esta marcação, é dado um salto no texto diretamente para o local onde a mesma foi inserida, possibilitando que o visitante visualize a descrição: Fim deste documento.

```
<center><font face="comic sans ms, verdana" size="4" color="#000099"><b><a  
name="praias"> PRAIAS CEARENSES </a></b></font></center>
```

Ao se clicar no link que faz referência a esta marcação, será dado um salto para este ponto do texto onde foi inserida esta marcação.

Exemplo da criação de âncoras que fazem referência e são marcações:

```
<a name="iniciododocumento" href="#fimdodocumento"> Vai para o final deste documento </a>  
<!-- corpo de um documento extenso -->  
<a name="fimdodocumento" href="#iniciododocumento"> Vai para o início deste documento  
</a>
```

Analisando o exemplo acima, ao clicar no link com a descrição "Vai para o final deste documento", o foco será jogado para o link que contem a descrição "Vai para o início deste documento", localizado no final da página, no contrário, caso o visitante clique sobre o link do final da página no qual está referenciando o link do início, o foco será lançado para este último.

Links nulos

Um link nulo é aquele que não oferece ligação para nenhuma outra página ou parte de um documento em HTML.

O objetivo principal de se criar um link nulo é permitir que o texto inserido como descrição para este link, possa ser focalizado.

Esta é mais uma forma de se destacar uma determinada informação de interesse.

Podemos criar um link nulo de duas maneiras:

- Passando como valor ao atributo (href) o nome do arquivo da própria página, pois deste modo, quando este link for ativado, apenas carregará novamente a página ao qual pertence.

Passando como valor ao atributo (href) o sinal #, usado para fazer referência a âncoras de marcação em documentos HTML.

Exemplo da criação de links nulos:

Supondo que na página praias.htm, necessitamos inserir um link nulo para destacar uma determinada informação. Portanto podemos criar este link das duas formas abaixo:

```
<a href="praias.htm"> NAVEGUE PELAS PRAIAS NORDESTINAS! </a>
```

```
<a href="#"> NAVEGUE PELAS PRAIAS NORDESTINAS! </a>
```

Link para e-mail

Com o objetivo de reforçar a idéia de que conquistar e fidelizar o maior número possível de adeptos, é algo realmente importante para o sucesso de um site na Internet e para isso, se torna necessário entre outras coisas, estreitar o relacionamento entre a pessoa responsável pelo site (WebMaster) e seus visitantes.

Apresentaremos a criação de um canal de comunicação bastante simples porém funcional, permitindo que os visitantes colaborem, critiquem, interajam, participem, enviando sugestões e contribuindo de forma direta no desenvolvimento e aperfeiçoamento do site.

Este canal de comunicação consiste na criação de um link que ao ser clicado, abrirá a janela para o envio de mensagens do aplicativo padrão de correio eletrônico, devidamente instalado e configurado no computador cliente, de modo que automaticamente o campo "Para:" e o campo "Assunto:", sejam preenchidos respectivamente com o endereço de e-mail do WebMaster e com o título da mensagem, facilitando o contato por parte do visitante que só precisará informar o texto no qual deseja enviar.

Observe abaixo, a sintaxe utilizada para a criação de um link para o envio de mensagens de correio eletrônico, onde através do atributo (href), são passados os parâmetros necessários:

Onde:

(mailto:) É o protocolo utilizado para informar o endereço de E-Mail que será inserido no campo "Para:".

(?): O sinal de interrogação (?) é utilizado para indicar a passagem de parâmetros.

(subject): Parâmetro no qual se atribui como valor, a informação que será exibida no campo "Assunto:". Perceba que para se atribuir o valor (DÚVIDAS E/OU SUGESTÕES) ao parâmetro (subject), é usado o sinal de igual (=).

A descrição do link utilizado para contato deve estar bem clara de forma que os visitantes entendam claramente, o seu objetivo. Normalmente são usadas expressões como:

Fale conosco

Fale com o WebMaster

E-Mail para o WebMaster

Dúvidas ou sugestões? Clique aqui!

Etc.

É evidente que a expressão empregada, dependerá do contexto atual e do objetivo pela qual o link para contato foi disponibilizado.

Formatando as cores dos links

Com o objetivo de facilitar a navegação dos visitantes pelas diversas seções de um site, além de podermos definir uma cor para o fundo da página usando o atributo (bgcolor) e com o atributo (text) uma cor padrão para os textos que não possuem a sua própria formatação de fonte, também é possível atribuir cores diferentes aos links nas seguintes situações:

Links carregados: É a cor inicialmente atribuída aos links no carregamento da página e antes de sofrerem qualquer ação pelo usuário.

Link ativo: É a cor atribuída a um determinado link, quando este recebe o foco através do posicionamento do cursor.

Links visitados: É a atribuição de cor aos links já acessados.

Por padrão, aos links carregados e aos ativos, é atribuída a cor azul e aos visitados, é atribuída a cor magenta.

Caso queiramos alterar essas definições, podemos especificar novos valores com as cores do padrão RGB aos atributos (link, alink e vlink) pertencentes à tag <body>, usada na abertura do bloco referente ao conteúdo do corpo da página.

Sintaxe para a definição de cores para os links:

```
<body link="#rrggbb" alink="#rrggbb" vlink="#rrggbb">
```

Onde:

Link="#rrggbb" - Define a cor para os links inicialmente carregados.

Alink="#rrggbb" - Define a cor para os links ao se tornarem ativos.

Vlink="#rrggbb" - Define a cor para os links ao serem visitados.

Exemplo:

```
<body bgcolor="#ffffff" text="#000000" link="#808080" alink="#ff0000" vlink="#00ff00">  
(corpo da página)  
</body>
```

No exemplo acima, definimos que o fundo da página terá a cor branca, os textos sem formatação de fonte terão a cor preta, os links ao serem carregados estarão da cor cinza, ao serem ativados, assumirão a cor vermelha e ao serem acessados, ficarão da cor verde.

Abrindo uma imagem no Microsoft Photo Editor

No Microsoft Photo Editor, você terá opções para abrir os seguintes formatos de imagem:

Graphics Interchange Format (*.gif)
Bitmap Padrão do Windows (*.bmp)
Joint Photographic Engineering Group (*.jpg)
Tagged Image File (*.tif)
Portable Network Graphics (*.png)
Kodak Photo CD (*.pcd)
PC Paintbrush (*.pcx)
Targa (*.tga)

Para abrir uma imagem desejada, com o programa Microsoft Photo Editor ativo, siga os passos descritos abaixo:

Acessar o menu "Arquivo" atalho (alt+a)

Descer com a seta até o item de menu "Abrir" e pressionar (enter). Pressionando simultaneamente as teclas (ctrl+a), consegue-se atingir o mesmo resultado obtido nos dois passos citados acima, a ativação da caixa de diálogo "Abrir".

Com a caixa de diálogo "Abrir" ativa, pressionando a tecla (tab) você deverá se deslocar até a lista de "Pastas e Arquivos", focalizada logo após o controle "Examinar".

Pressionando a tecla (backspace), você descerá um nível na lista hierárquica das pastas e pressionando a tecla (enter) sobre uma pasta selecionada, você abrirá seu conteúdo (o nome da pasta atual, sempre será exibido no controle "Examinar").

Após abrir a pasta que contém o arquivo de imagem desejado, selecione o mesmo e pressione a tecla (enter) para abri-lo.

Uma outra forma direta de abrir um arquivo de imagem no programa Microsoft Photo Editor após ativar a caixa de diálogo "Abrir" com (Ctrl+A), é digitar corretamente o caminho completo com o nome e a extensão da imagem no campo "Nome do arquivo" e em seguida, teclar (enter) para abri-lo.

Imagens

Neste módulo, será apresentado um recurso bastante explorado pelos desenvolvedores Web e que contribui de maneira significativa com a composição visual de um site, trata-se da inserção de imagens. O uso deste recurso quando bem elaborado, torna a navegação por um site muito mais agradável, tornando verdadeira a afirmação que "uma imagem vale mais do que mil palavras".

É notável que o uso de imagens facilite visualmente a navegação por um site, tornando-a muito mais intuitiva, mas como bons desenvolvedores Web, devemos sempre disponibilizar uma descrição alternativa das imagens na forma de texto, pois além de alguns Browsers não terem a capacidade de exibi-las, devemos manter um padrão de acessibilidade universal, possibilitando o acesso ao nosso site por pessoas com limitação visual, onde estes fazem o uso de programas leitores de tela para inclusive navegar na Internet (programa que captura as informações escritas na tela do computador na forma de texto e as devolve como uma resposta falada a seus usuários, através de sistemas de síntese de voz).

Os formatos de imagem mais apropriados para as páginas Web, são o GIF e o JPEG, devido a algumas particularidades que propiciam a utilização destes formatos, além de possuírem sistemas eficazes de compressão, reduzindo o tamanho em bytes das imagens, se comparados por exemplo, com o formato bitMap.

É aconselhável criar na raiz do seu sistema uma pasta exclusiva para armazenar todas as imagens que serão utilizadas. Algumas ferramentas usadas na construção de sites para a Internet, costumam atribuir a esta pasta o nome de "images", facilitando a referência à imagem indiferentemente da localização da página que a exibirá, dentro das pastas que compõem a estrutura organizacional do site.

O tamanho em bytes e a quantidade de imagens que serão utilizadas é algo que precisa ser considerado, pois imagens muito grandes poderão retardar o carregamento de uma página, prejudicando deste modo, o público do seu site. Neste sentido, uma dica interessante é sempre que possível, utilizar uma mesma imagem em várias páginas, como por exemplo, logotipos e botões de acesso, pois uma vez carregada, a imagem fica armazenada no Cache da máquina do visitante até que o mesmo seja limpo, ficando disponível caso haja outras requisições.

As imagens para a utilização na Internet, podem ser obtidas através de profissionais que trabalham com programas de editoração gráfica, através de máquinas fotográficas digitais, além de poder serem salvas da própria Internet, bastando para isso, ao localizar a imagem no qual deseja salvar, clicar sobre ela com o botão direito do mouse e na opção "Salvar figura como...", indicar o local no seu computador onde deseja salvá-la.

Como na aplicação de cores, vamos unir o útil ao agradável, solicitando a opinião de outras pessoas para a seleção e utilização mais adequada das imagens, lembrando sempre que um site deve agradar principalmente aos seus visitantes.

Observe as particularidades dos formatos mais utilizados na Internet, a sintaxe para a utilização da tag na inserção de imagens com seus respectivos atributos, bem como a sua aplicação nas páginas Web.

O formato Gif

As imagens (.gif), extensão originada da abreviação de Graphics Interchange Format, é o formato mais utilizado nas imagens inseridas nas páginas Web e reconhecido pela grande maioria dos navegadores disponíveis no mercado, como pelos populares Internet Explorer da Microsoft e o Netscape Navigator da Netscape.

O formato GIF é bastante eficaz na compressão de imagens que apresentam áreas compostas por seqüências monocromáticas, reduzindo significativamente o tamanho em bytes dos arquivos que armazenam imagens nesta condição.

As imagens no formato GIF aceitam no máximo 256 cores (padrão de 8 bits), por este motivo, este formato é recomendado no desenho de imagens com um aspecto gráfico mais simples, como em animações, botões de opções, barras de menus, planos de fundo e em outras situações semelhantes onde não se faça necessário, uma cópia idêntica rica em detalhes gráficos e em cores.

Algumas importantes características alcançadas com a aplicação de efeitos muito interessantes nas imagens (.gif), colaboram para que este formato seja preferivelmente o mais utilizado nas páginas Web. Em alguns editores gráficos, através da opção de salvar a imagem no formato GIF89, torna-se possível inserir alguns efeitos propícios à Internet.

Abaixo, com o objetivo descritivo, serão relacionadas três aplicações de efeitos com a utilização do formato GIF:

1. **Interlaceamento:** Efeito que possibilita a composição progressiva da imagem durante o estágio de carregamento da página.
2. **Transparência:** Efeito que consiste na eliminação da cor do plano de fundo da imagem, preservando sob o seu fundo, a aparência do papel de parede ou a cor do plano de fundo da página. Este efeito, é bastante interessante para a montagem de animações gráficas.
3. **Antialias:** Efeito utilizado para suavizar a visualização da tonalidade da cor da borda da imagem, em relação ao plano de fundo da página.

O formato JPEG

O formato JPEG (Joint Photographic Engineering Group), é o segundo formato mais utilizado pelos Web Designers nas páginas de Internet.

Este formato faz a utilização de complexos e de sofisticados algoritmos matemáticos para a compactação das imagens, onde em alguns casos, consegue reduzir em até dez vezes o tamanho em bytes usado para armazenar uma imagem, se comparado com o formato BITMAP.

O formato JPEG, trabalha com o padrão de cores de 24 bits, portanto, aceita o uso de até 16777216 tonalidades de cores, superando inclusive, a capacidade de distinção do olho humano. Por esta capacidade de resolução, o formato JPEG é fortemente recomendado na geração de imagens fotográficas como, por exemplo, nas de objetos da vida real, paisagens, animais, pessoas ou em logotipos e esquemas gráficos que exijam uma alta definição etc.

Alguns programas de editoração gráfica permitem a conversão de formatos, mas deve-se analisar criteriosamente esta ação, pois às vezes devido a algumas características individuais de cada formato, podem acarretar na perda da resolução gráfica da imagem. Um exemplo prático desta situação seria a conversão de uma imagem no formato JPEG com uma alta resolução de cores, para o formato GIF que permite apenas, a utilização de 256 cores.

Modernos equipamentos eletrônicos como máquinas fotográficas e câmeras digitais, utilizam na sua grande maioria o formato JPEG para a geração das imagens fotográficas, facilitando e agilizando os processos de obtenção de material gráfico para o uso na Internet.

Devido às características citadas acima, como a elevada taxa de compressão do tamanho em bytes das imagens e a alta resolução gráfica, este formato se tornou bastante adequado para o uso nas páginas da Internet, sendo superado pelo formato GIF, principalmente por não permitir a aplicação do efeito de transparência para a cor do plano de fundo das imagens, apesar de possibilitar a utilização de outros efeitos gráficos.

Sintaxe para inserção de imagens

A sintaxe da tag utilizada para a inserção de imagens é muito simples e semelhante com a usada para a inserção de links, diferenciando-se principalmente por não possuir uma parte aplicada ao seu fechamento.

Basicamente, possui o seguinte formato:

```

```

O atributo (src) é usado para referenciar uma imagem e a sua utilização é idêntica a do atributo (href), podendo criar uma referência para uma imagem interna ou externa ao seu sistema. O procedimento usado é o mesmo, independentemente do formato da imagem.

Além da sintaxe básica demonstrada acima, a inserção de imagens poderá ser implementada, utilizando-se dos atributos descritos abaixo:

(height):

Define (em pixels) a altura da imagem que será exibida na página.

(width):

Define (em pixels) a largura da imagem que será exibida na página.

A utilização dos atributos (height) e (width) é sempre recomendada, pois através dos valores passados, o navegador reserva as áreas que serão utilizadas na página pelas imagens e prossegue com o carregamento do restante do documento, agilizando deste modo, este processo.

Os valores passados aos atributos (height) e (width) deverão ser idênticos ou proporcionais às dimensões reais da imagem para não ocasionar problemas de distorção. Estes valores poderão ser obtidos através de programas de editoração gráfica.

(align):

Este atributo é usado para alinhar a imagem em relação ao texto ou à página. Os valores mais comuns são:

Align="top" - Alinha a imagem com a parte superior do texto.

Align="middle" - Alinha a imagem com o centro do texto.

Align="bottom" - Alinha a imagem com a parte inferior do texto.

Align="left" - Alinha a imagem à esquerda da página.

Align="right" - Alinha a imagem à direita da página.

(border):

Define a espessura (em pixels) da borda da imagem. Caso receba o valor "0", a borda não será exibida.

(Alt):

Este atributo recebe como valor, um texto alternativo que será apresentado quando a imagem receber o foco do cursor.

A utilização deste atributo é essencial para promover a acessibilidade, pois na falta desta descrição, pessoas que se utilizam de programas leitores de telas ou de navegadores que não suportam o uso de imagens, serão prejudicadas no entendimento da informação relacionada a esta imagem.

(hspace):

O uso deste atributo reserva uma margem horizontal externa à imagem, onde o valor definido em pixels é assumido tanto pelo lado direito como pelo lado esquerdo da imagem.

(vspace):

O uso deste atributo reserva uma margem vertical externa à imagem, onde o valor definido em pixels é assumido tanto pelo lado superior como pelo lado inferior da imagem.

O que é e como abrir o Microsoft Photo Editor

O Microsoft Photo Editor é uma ferramenta de editoração gráfica, disponível no pacote Office da Microsoft. Tomaremos como base a versão 3.0 disponível no Office 97.

Apesar de ser um programa bastante simples, permite a utilização de diversos efeitos gráficos, como por exemplo, a suavização da qualidade da imagem, a ênfase das bordas, a aplicação do efeito de textura, colocar a cor do fundo transparente, além de muitos outros recursos disponíveis para se trabalhar com a edição de imagens.

Para abrir o programa Microsoft Photo Editor, caso o mesmo esteja devidamente instalado no seu computador, siga os passos descritos abaixo:

Acessar o Botão "Iniciar" (ctrl+esc)

No menu "Iniciar", descer com a seta até o item de menu "Programas"

Teclar (enter) ou (seta para a direita) para ter acesso aos itens deste submenu

Em seguida, descer com a seta até localizar o item de menu "Microsoft Photo Editor"

Após localizá-lo, pressionar a tecla (enter) e o programa será aberto.

Convertendo formatos de imagem no Microsoft Photo Editor

Para a comunidade de desenvolvedores Web, a possibilidade de se realizar a conversão entre os vários formatos de imagem. É algo muito interessante, pois dessa forma, torna-se possível utilizar uma gama muito maior de opções de imagens, convertendo os diversos formatos disponíveis para os dois mais aceitos pelos sites da Internet, o GIF (*.gif) e o JPEG (*.jpg).

É possível converter todos os formatos de imagem suportados pelo Microsoft Photo Editor, listados no seu procedimento de abertura de arquivos, para os formatos relacionados abaixo:

Graphics Interchange Format (*.gif)
Bitmap Padrão do Windows (*.bmp)
Joint Photographic Engineering Group (*.jpg)
Tagged Image File (*.tif)
Portable Network Graphics (*.png)
PC Paintbrush (*.pcx)
Targa (*.tga).

Obtendo a largura e a altura ou redimensionando uma imagem no Microsoft Photo Editor

O objetivo principal desta aula, é obter através do Microsoft Photo Editor, as dimensões referentes a largura e a altura de uma imagem, a fim de utilizar estes valores para o preenchimento dos atributos (width) e (height), ao se inserir esta imagem em uma página em HTML.

A opção que será utilizada para alcançar o objetivo proposto acima tem uma outra finalidade dentro do programa Microsoft Photo Editor, que é a realização do redimensionamento de uma imagem, permitindo aumentar ou diminuir seu tamanho.

Para obter ou redimensionar a largura e a altura de uma imagem, após abri-la no Microsoft Photo Editor, acesse o menu "Imagem" atalho (alt+i), com a seta desça até o item de menu "Redimensionar" e pressione a tecla (enter).

Com a caixa de diálogo "Redimensionar" aberta, você terá as seguintes opções que poderão ser acessadas com o pressionamento da tecla (tab) ou através das suas respectivas teclas de atalho:

- Largura (controle caixa de texto editável – atalho alt+l): Apresentará o valor numérico referente à largura da imagem.
- Altura (controle caixa de texto editável – atalho alt+a): Apresentará o valor numérico referente à altura da imagem.
- Percentual da largura (controle caixa de texto editável): Apresenta o valor em porcentagem referente à largura da imagem e o padrão é (100%).
- Percentual da altura (controle caixa de texto editável): Apresenta o valor em porcentagem referente à altura da imagem e o padrão é (100%).
- Permitir distorção (controle caixa de verificação): O ideal é manter desmarcada esta opção, pois caso deseje redimensionar a imagem atribuindo um novo valor para a largura, por exemplo, o valor da altura é recalculado automaticamente mantendo a proporcionalidade.

A aplicação de imagens nas páginas WEB

A inserção de imagens, combinada com outros recursos do HTML, podem ter diversas aplicações dentro das páginas Web.

Entre muitas outras possibilidades, podemos destacar:

Decorar o fundo de uma página

Exibir um logotipo ou uma fotografia

Ilustrar uma informação escrita

Servir como um botão sensível de acesso a uma outra página em HTML

Carregar uma outra imagem equivalente, mas de maior tamanho e com uma melhor resolução gráfica etc.

Inserindo uma imagem de fundo para a página

Podemos inserir nas páginas HTML uma imagem de fundo, bastando apenas que na abertura do bloco do corpo do documento, seja passado à localização da imagem desejada como valor ao atributo (background).

A Microsoft acrescentou ao navegador Internet Explorer o atributo (bgproperties) que ao receber o valor "fixed", permite que o texto ao ser rolando na tela durante a sua visualização, flutue sobre a imagem de fundo que permanece fixa. A este efeito é dado o nome de Marca D'água.

Exemplo da declaração do efeito da Marca D'água:

```
<body background="/images/fundo.jpg" bgproperties="fixed">  
(corpo do documento em HTML)  
</body>
```

Exibindo um logotipo ou uma fotografia na página

É comum que os sites de organizações exibam uma imagem com seu logotipo, dos seus parceiros ou dos seus patrocinadores, quando existentes, a fim de divulgar suas marcas.

Não muito diferente, alguns desenvolvedores de sites pessoais, costumam inserir na Home Page do site, sua fotografia, para que os visitantes possam conhecer as suas características físicas, dando uma impressão de maior proximidade, estreitando a relação entre o visitante e o Webmaster.

Na maioria das vezes, as imagens com logotipos ou fotografias, são posicionadas no topo da página e com o alinhamento centralizado, para que em destaque, possam ser visualizadas imediatamente após o seu carregamento.

Observe o exemplo abaixo:

```
<center></center>
```

Ilustrando as informações escritas

As imagens utilizadas para ilustrar as informações escritas nas páginas em HTML, deverão ser inseridas conforme a necessidade, utilizando o atributo de alinhamento para posicionar a imagem em relação ao texto e à página.

Devemos sempre ter o cuidado de não fazer um uso excessivo de imagens com um tamanho (em bytes) muito grande, para não tornar o acesso a esta página demasiadamente lento, prejudicando deste modo a navegação para os visitantes do site.

É extremamente importante lembrarmos que todas as imagens devem conter a sua respectiva descrição em texto através da utilização do atributo (alt).

Observe o exemplo abaixo:

```
</p>
```

Inserindo botões gráficos de acesso

Para tornar a navegação muito mais agradável e intuitiva, juntamente com o recurso da inserção de links, podemos utilizar pequenas imagens (ícones) como botões sensíveis para fazer referência a uma outra página em HTML. Neste caso, devemos nos lembrar sempre de retirar a borda da imagem, passando o valor "0" ao atributo (border) e ao atributo (alt), uma descrição textual referente a ação que será executada..

Exemplos:

Uma imagem que ao ser clicada dá acesso ao Portal do Deficiente Visual:

Perceba nos exemplos, no lugar da descrição do link, inserimos uma imagem e esta ao ser clicada, executará uma ação de acordo com o valor passado ao atributo (href) do link, que poderá ser o carregamento de uma nova página, a abertura do aplicativo de correio eletrônico, o deslocamento para um determinado ponto dentro da página atual, a realização do procedimento de download de arquivos etc.

Uma imagem (.gif) animada com o sinal de "@" girando, que ao ser clicada abre a janela do aplicativo padrão de correio eletrônico:

```
<a href="http://www.deficientevisual.org.br"></a>  
<a href="mailto:webmaster@servidor.sigla.país?subject=Fale com o WebMaster"></a>
```

Caso uma imagem represente um fator relevante para o entendimento dentro do contexto geral da página, podemos associá-la a um link nulo, permitindo que esta imagem seja focalizada através da navegação feita pelo uso dos atalhos de teclado e deste modo, garantimos que pessoas impossibilitadas de utilizarem o mouse ou que utilizem programas leitores de tela, também tenham acesso à sua descrição fornecida através do atributo (alt).

Observe o exemplo abaixo:

De acordo com o exemplo, o visitante ao navegar com o uso da tecla (tab) pela página, terá acesso ao logotipo de acessibilidade e com isso, saberá que o desenvolvedor deste site tem a intenção de possibilitar o acesso a todas as pessoas indiferentemente dos seus recursos computacionais ou das suas habilidades físicas e ou sensoriais.

```
<a href="#"></a>
```

Usando thumbnails

Às vezes, é necessário exibir uma imagem com maior tamanho e com uma melhor resolução gráfica, para que o visitante possa obter uma maior riqueza de detalhes na sua visualização.

Para não carregarmos diretamente em uma página várias imagens grandes sobrecarregando a mesma, podemos disponibilizar miniaturas destas imagens e estas ao serem clicadas, carregarão sua imagem correspondente com uma maior dimensão. A esta técnica, é dado o nome de thumbnail.

Observe no exemplo abaixo:

No exemplo anterior, quando o visitante clicar na imagem pequena, será carregada em uma nova janela (devido a utilização do atributo `target="window"`) a imagem grande.

Podemos exibir uma imagem referenciada por uma descrição textual.

Exemplo:

```
<a href="/images/pantanal_grande.jpg" target="window"> Veja uma foto do Pantanal Mato-grossense </a>
```

Neste exemplo, quando o visitante clicar sobre o texto "Veja uma foto do Pantanal Mato-grossense", a imagem será carregada em uma nova janela.

```
<a href="/images/paisagem_grande.jpg" target="window"> </a>
```

Listas e tabelas

Neste módulo, vamos apresentar a criação de listas e tabelas, recursos utilizados para promover uma melhor apresentação dos conteúdos expostos nos sites da Internet.

A utilização das listas apenas será viável, quando houver a necessidade de se organizar as informações em itens, de modo a facilitar a sua compreensão, quantificação e classificação.

Na versão atual da linguagem HTML, são utilizados três diferentes estilos de listas, conforme a relação abaixo:

- Lista desordenada ou não numerada
- Lista ordenada ou numerada
- Lista de definição ou glossário

Nas primeiras versões da linguagem HTML, eram utilizadas as listas de menus e de diretórios, mas ambas se tornaram obsoletas.

Os diferentes estilos de listas podem ser utilizados em conjunto, o que originou mais um estilo, ao qual damos o nome de lista intercalada ou encadeada.

A possibilidade de disponibilizar as informações em tabelas, veio para suprir a falta de comandos mais elaborados na realização eficaz do posicionamento e alinhamento dos textos e objetos, principalmente com relação aos links, imagens e controles, dando um novo realce quanto ao aspecto da apresentação visual destes elementos, quando inseridos nas páginas em HTML.

Tanto as listas como as tabelas podem usufruir dos diversos comandos de formatação de texto, disponíveis na linguagem HTML e estudados nos módulos anteriores.

Observe a sintaxe e os atributos utilizados na criação das listas e tabelas, para que possamos implementar mais estes recursos na construção de páginas em HTML.

Lista desordenada ou não numerada

O estilo de lista desordenada, também conhecido por não numerada, insere um símbolo gráfico (bullet) ao lado esquerdo de cada uma das informações, a fim de identificá-las individualmente como itens de uma lista. O resultado apresentado por este estilo de lista é idêntico ao efeito produzido na inserção do recurso de marcadores, disponível em alguns editores de textos, como por exemplo, no Microsoft Word.

Sintaxe utilizada para a inserção de listas desordenadas:

`` - Tag utilizada para abertura da lista desordenada (UL = Unordered List).

`` - Tag utilizada para a inserção de cada novo item da lista.

`` - Tag utilizada para o fechamento da lista desordenada.

A tag `` deverá ser inserida antes de cada novo item da lista, não sendo necessário fechá-la com `` ao término da descrição.

Atributo (**type**):

O atributo (**type**) é utilizado para definir qual símbolo gráfico (bullet) entre os três disponíveis deverá ser inserido juntamente com os itens da lista. Os valores possíveis são:

Type="disc" – Insere um círculo preenchido.

Type="circle" – Insere um círculo vazio.

Type="square" – Insere um quadrado preenchido.

O atributo (**type**) poderá ser usado tanto com a tag (definindo o símbolo para todos os itens da lista), quanto com a tag personalizando apenas os itens desejados.

Observe com atenção a distribuição dos comandos HTML (tags) na construção do documento, no exemplo ao lado.

```
<html>
<head>
<title> LISTA DESORDENADA OU NÃO NUMERADA </title>
</head>
<body>
<br><br>
<center><font face="comic sans ms, verdana, sans-serif" size="4" color="#000099"> <b> LISTA
DESORDENADA OU NÃO NUMERADA </b></font></center>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> O território brasileiro é dividido em cinco regiões.</p>
<br>
<p align="justify"> Regiões do Brasil:</p>
<ul type="square">
<li> Norte
<li> Nordeste
<li> Centro-oeste
<li> Sudeste
<li> Sul
</ul>
</font>
</body>
</html>
```

Lista ordenada ou numerada

O estilo de lista ordenada, também conhecido por lista numerada, insere ao lado esquerdo de cada nova informação, um algarismo do sistema numérico cardinal,

ou do sistema numérico romano, ou ainda, um caracter do sistema alfabético, classificando estas informações, como itens ordenados em uma lista.

O formato produzido na utilização de uma lista ordenada é idêntico a inserção do recurso de numeração, disponível em alguns editores de textos, como por exemplo, no Microsoft Word.

Sintaxe utilizada para a inserção de listas ordenadas:

- **** - Tag utilizada para abertura da lista ordenada (OL = Ordered List).
- **** - Tag utilizada para a inserção de cada novo item da lista.
- **** - Tag utilizada para o fechamento da lista ordenada.

A tag **** deverá ser inserida antes de cada novo item da lista, a fim de lhe atribuir um índice de classificação seqüencial, não sendo necessário o seu fechamento com **** ao término da descrição.

Atributo (**type**):

O atributo (**type**) na lista ordenada, é utilizado para definir qual sistema de ordenação entre os disponíveis deverá ser aplicado aos itens da lista. As formas possíveis são:

Type="1" – Algarismos do sistema numérico cardinal (1, 2, 3 etc.)

Type="i" – Algarismos em minúscula do sistema numérico romano (i, ii, iii etc.)

Type="I" - Algarismos em maiúscula do sistema numérico romano (I, II, III etc.)

Type="a" – Caracteres em minúscula do sistema alfabético (a, b, c etc.)

Type="A" - Caracteres em maiúscula do sistema alfabético (A, B, C etc.)

Atributo (**start**):

O atributo (**start**) define o ponto de partida para a classificação dos itens de uma lista ordenada. A sua utilização é feita da seguinte forma:

Start="*" – O sinal (*) deverá ser substituído por um número que será tomado como referência para o início da classificação em uma lista ordenada.

Exemplos da utilização do atributo (**start**):

- Quando (type="1") e (start="0"), a classificação será iniciada no número 0 (valor aplicado apenas quando for usado o sistema cardinal para a classificação).
- Quando (type="A") e (start="11"), a classificação será iniciada no caracter K.
- Quando (type="I") e (start="10"), a classificação será iniciada no algarismo romano X.
- Tanto o atributo (type) como o (start), deverão ser inseridos dentro da tag , definindo um mesmo estilo para todos os itens da lista.

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> LISTA ORDENADA OU NUMERADA </title>
</head>
<body>
<br><br>
<center><font face="comic sans ms, verdana, sans-serif" size="4" color="#000099"> <b> LISTA
ORDENADA OU NUMERADA </b></font></center>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> O Brasil possui 26 estados e mais um distrito federal, onde está localizado
Brasília, a capital do país.</p>
<br>
<p align="justify"> Estados brasileiros:</p>
<ol type="a">
<li> Acre
<li> Alagoas
<li> Amapá
<li> Amazonas
<li> Bahia
<li> Ceará
<li> Espírito Santo
<li> Goiás
<li> Mato Grosso
<li> Mato Grosso do Sul
<li> Maranhão
<li> Minas Gerais
<li> Pará
<li> Paraíba
<li> Paraná
```

```
<li> Pernambuco
<li> Piauí
<li> Rio de Janeiro
<li> Rio Grande do Norte
<li> Rio Grande do Sul
<li> Roraima
<li> Rondônia
<li> Santa Catarina
<li> São Paulo
<li> Sergipe
<li> Tocantins
</ol>
</font>
</body>
</html>
```

Lista de definição ou glossário

Este estilo de lista é diferenciado pelo fato de que cada um dos seus itens é composto por um termo e por sua respectiva definição, justificando o segundo nome dado a este estilo de lista, por caracterizar o formato de um glossário.

Sintaxe utilizada para a criação da lista de definição ou glossário:

```
<dl> - Tag utilizada para a abertura da lista de definição
<dt> - Tag utilizada para a inserção de um novo termo à lista
<dd> - Tag utilizada para a inserção de uma definição associada ao seu
respectivo termo.
</dl> - Tag utilizada para o fechamento da lista de definição
```

As tags **<dt>** e **<dd>**, deverão ser inseridas tanto quanto for à quantidade de itens (termos e definições) disponíveis para a montagem da lista.

A tag **<dd>** também pode ser usada individualmente fora de uma lista de definição, inserindo um espaço em branco à esquerda da informação escrita na sua seqüência.

Exemplo:

```
<dd> Texto com um espaço à esquerda.
```

No próximo exemplo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> LISTA DE DEFINIÇÃO OU GLOSSÁRIO </title>
</head>
<body>
<br><br>
<center><font face="comic sans ms, verdana, sans-serif" size="4" color="#000099"><b> LISTA
DE DEFINIÇÃO OU GLOSSÁRIO </b></font></center>
<br><br>
<font face="arial" size="3" color="#000000">
<p align="justify"> O Brasil está dividido em 5 regiões, conforme o esquema abaixo:</p>
<br>
<dl>
<dt> NORTE
<dd> A região norte do Brasil é subdividida em 7 estados.
<dt> NORDESTE
<dd> O nordeste brasileiro, está subdividido em 9 estados.
<dt> CENTRO-OESTE
<dd> A região centro-oeste, além do Distrito Federal onde está localizado Brasília, a capital do
país, possui 3 estados.
<dt> SUDESTE
<dd> O sudeste brasileiro, possui 4 estados.
<dt> SUL
<dd> A região sul, é composta por 3 estados.
</dl>
</font>
</body>
</html>
```

Lista intercalada ou encadeada

Este estilo é resultante da intercalação ou encadeamento dos demais estilos de listas disponíveis, não havendo uma sintaxe própria para a sua implementação, mas sim, a utilização da sintaxe de um ou mais estilos de listas em conjunto.

As listas poderão estar dispostas seqüencialmente ou aninhadas internamente a outras listas.

Apenas diante de determinadas situações específicas é que você irá identificar a quantidade e quais estilos de listas, serão mais apropriados para melhor esquematizar as informações por itens.

No exemplo a seguir, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> LISTA INTERCALADA OU ENCADEADA </title>
</head>
<body>
<br> <br>
<center><font face="comic sans ms, verdana, sans-serif" size="4" color="#000099"> <b> LISTA
INTERCALADA OU ENCADEADA </b> </font> </center>
<br> <br>
<font face="arial" size="3" color="#000000">
<p align="justify"> O Brasil está dividido em 5 regiões e estas, estão subdivididas em estados que
apresentam características geográficas semelhantes, como: </p>
<!-- Lista desordenada ou não numerada. -->
<ul type="disc">
<!-- Itens da lista desordenada ou não numerada. -->
<li>Clima
<li>Vegetação
<li>Solo etc
</ul>
<br>
<p align="justify"> Regiões do Brasil: </p>
<br>
<!-- lista de definição ou glossário. -->
<dl>
<!-- Item da lista de definição ou glossário. -->
<dt> NORTE
<dd> A região norte do Brasil é subdividida em 7 estados.
<br>
<!-- Lista ordenada ou numerada. -->
<ol type="1">
<!-- Itens da lista ordenada ou numerada. -->
<li>Acre
<li>Amapá
<li>Amazonas
<li>Pará
<li>Roraima
<li>Rondônia
<li>Tocantins
</ol>
<br> <br>
<!-- Item da lista de definição ou glossário. -->
<dt> NORDESTE
<dd> O nordeste brasileiro, está subdividido em 9 estados.
<br>
<!-- Lista ordenada ou numerada. -->
```

```
<ol type="1">
<!-- Itens da lista ordenada ou numerada. -->
<li>Alagoas
<li>Bahia
<li>Ceará
<li>Maranhão
<li>Paraíba
<li>Pernambuco
<li>Piauí
<li>Rio Grande do Norte
<li>Sergipe
</ol>
<br><br>
<!-- Item da lista de definição ou glossário. -->
<dt> CENTRO-OESTE
<dd> A região centro-oeste, além do Distrito Federal onde está localizado Brasília, a capital do
país, possui 3 estados.
<br>
<!-- Lista ordenada ou numerada. -->
<ol type="1">
<!-- Itens da lista ordenada ou numerada. -->
<li>Goiás
<li>Mato Grosso
<li>Mato Grosso do Sul
</ol>
<br><br>
<!-- Item da lista de definição ou glossário. -->
<dt> SUDESTE
<dd> O sudeste brasileiro, possui 4 estados.
<br>
<!-- Lista ordenada ou numerada. -->
<ol type="1">
<!-- Itens da lista ordenada ou numerada. -->
<li>Espírito Santo
<li>Minas Gerais
<li>Rio de Janeiro
<li>São Paulo
</ol>
<br><br>
<!-- Item da lista de definição ou glossário. -->
<dt> SUL
<dd> A região sul, é composta por 3 estados.
<br>
<!-- Lista ordenada ou numerada. -->
<ol type="1">
<!-- Itens da lista ordenada ou numerada. -->
```

```
<li>Paraná  
<li>Santa Catarina  
<li>Rio Grande do Sul  
</ol>  
</dl>  
</font>  
</body>  
</html>
```

Tabelas

O uso de tabelas permite posicionar textos e objetos com muita eficácia, tomando por base as células, que são originadas através do cruzamento entre linhas e colunas. É possível dentro das células de uma tabela, criar outras tabelas interpostas e assim sucessivamente conforme a necessidade.

Uma tabela pode conter várias linhas (posição horizontal) e uma linha por sua vez, pode conter várias colunas (posição vertical).

Uma linha pode conter várias colunas.

A sintaxe necessária para a implementação das tabelas, é composta por diversas tags pares, usadas para delimitar a abertura e o fechamento da tabela, das linhas, das colunas etc. Além das tags, estão disponíveis vários atributos, para que possamos formatar a aparência da tabela conforme a nossa vontade ou necessidade, definindo imagens ou cores de fundo, borda, alinhamentos, espaçamentos entre células etc.

Tags utilizadas na implementação de tabelas:

<table> e **</table>** - Delimitam o início e o fim da tabela

<caption> e **</caption>** - Insere um título para a tabela

<tr> e **</tr>** - Delimitam o início e o fim de uma linha

<th> e **</th>** - Insere cabeçalhos para as colunas. Devem ser inseridos conforme o número de colunas da tabela e delimitados pela tag **<tr>** e **</tr>**. A informação inserida entre **<th>** e **</th>**, por padrão, ficará centralizada e no estilo negrito.

<td> e **</td>** - Insere colunas à tabela. Devem estar alinhados dentro da tag **<tr>** e **</tr>**.

Atributos que podem ser declarados tanto na abertura da tabela **<table>**, quanto na abertura das colunas **<th>** ou **<td>**:

(width) e **(height)**:

Estes atributos definem respectivamente a largura e a altura da tabela em relação à página ou ainda, de uma coluna em relação à tabela. Os valores passados a estes atributos deverão ser inseridos em pixels ou em porcentagem.

(bgcolor): Define a cor de fundo para a tabela ou para uma célula específica, usando o padrão RGB.

(background): Define uma imagem de fundo para a tabela ou para uma célula específica.

(border): Define a espessura (em pixels) da borda da tabela. Caso receba o valor "0", a borda será omitida.

(bordercolor): Define a cor principal da borda usando o padrão RGB.

(bordercolordark): Define a cor da sombra mais escura da borda usando o padrão RGB.

(bordercolorlight): Define a cor da sombra mais clara da borda usando o padrão RGB.

(cellspacing): Este atributo define em pixels a distância entre uma célula e outra.

(cellpadding): Este atributo define a margem entre o texto ou objeto inserido na célula e a sua borda.

Atributos utilizados na abertura das colunas `<th>` ou `<td>`:

(align): Este atributo permite alinhar um texto ou objeto na horizontal dentro da célula. Os valores disponíveis são:

Align="center" – Alinhamento centralizado

Align="left" – Alinhamento à esquerda (padrão)

Align="right" – Alinhamento à direita

(valign): Este atributo permite alinhar um texto ou objeto na vertical dentro da célula. Os valores disponíveis são:

- `vAlign="middle"` – Alinha ao centro da célula.
- `vAlign="bottom"` – Alinha à parte inferior da célula.
- `vAlign="top"` – Alinha à parte superior da célula.

(colspan): Este atributo recebe como valor um número, definindo a quantidade de colunas ocupadas por uma célula.

(rowspan): Este atributo recebe como valor um número, definindo a quantidade de linhas ocupadas por uma célula.

Observem no exemplo abaixo os códigos utilizados para elaboração de uma tabela.

```
<html>
<head>
<title> TABELA BÁSICA DAS CORES DO PADRÃO RGB </title>
</head>
<body>
<br><br>
<CENTER>
<TABLE BORDER CELSPACING=1 BORDERCOLOR="#000000" CELLPADDING=4
WIDTH=181 >
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffff">
<b>
COLOR
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffff">
<b>
VALUE
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP"> </TD>
```

```
<TD WIDTH="54%" VALIGN="TOP"> </TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FFFF">
<b>
Aqua
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FFFF">
<b>
#00FFFF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FF00FF">
<b>
Fuchsia
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FF00FF">
<b>
#FF00FF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FF00">
<b>
```

```
Lime
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FF00">
<b>
#00FF00
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#c0c0c0">
<b>
Silver
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#c0c0c0">
<b>
#C0C0C0
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffffff">
<b>
White
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
```

```
<font face="arial" size="3" color="#ffffff">
<b>
#FFFFFF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FFFF00">
<b>
Yellow
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FFFF00">
<b>
#FFFF00
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000000">
<b>
Black
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000000">
<b>
#000000
</b>
</font>
</a>
</TD>
```

```
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#0000ff">
<b>
Blue
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#0000ff">
<b>
#0000FF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808080">
<b>
Gray
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808080">
<b>
#808080
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008000">
<b>
Green
```

```
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008000">
<b>
#008000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800000">
<b>
Maroon
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800000">
<b>
#800000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000080">
<b>
Navy
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000080">
```

```
<b>
#000080
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808000">
<b>
Olive
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808000">
<b>
#808000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800080">
<b>
Purple
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800080">
<b>
#800080
</b>
</font>
</a>
</TD>
</TR>
```

```
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ff0000">
<b>
Red
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ff0000">
<b>
#FF0000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008080">
<b>
Teal
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008080">
<b>
#008080
</b>
</font>
</a>
</TD>
</TR>
</TABLE>
</CENTER>
</body>
</html>
```

Observe no exemplo abaixo os códigos utilizados para elaboração de uma tabela.

```
<html>
<head>
<title> TABELA BÁSICA DAS CORES DO PADRÃO RGB </title>
</head>
<body>
<br><br>
<CENTER>
<TABLE BORDER CELSPACING=1 BORDERCOLOR="#000000" CELLPADDING=4
WIDTH=181 >
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffff">
<b>
COLOR
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffff">
<b>
VALUE
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP"> </TD>
<TD WIDTH="54%" VALIGN="TOP"> </TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FFFF">
<b>
Aqua
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
```

```
<font face="arial" size="3" color="#00FFFF">
<b>
#00FFFF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FF00FF">
<b>
Fuchsia
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FF00FF">
<b>
#FF00FF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FF00">
<b>
Lime
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#00FF00">
<b>
#00FF00
</b>
</font>
</a>
</TD>
```

```
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#c0c0c0">
<b>
Silver
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#c0c0c0">
<b>
#C0C0C0
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffffff">
<b>
White
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ffffff">
<b>
#FFFFFF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#000000">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FFFF00">
<b>
Yellow
```

```
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#FFFF00">
<b>
#FFFF00
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000000">
<b>
Black
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000000">
<b>
#000000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#0000ff">
<b>
Blue
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#0000ff">
```

```
<b>
#0000FF
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808080">
<b>
Gray
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808080">
<b>
#808080
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008000">
<b>
Green
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008000">
<b>
#008000
</b>
</font>
</a>
</TD>
</TR>
```

```
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800000">
<b>
Maroon
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800000">
<b>
#800000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000080">
<b>
Navy
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#000080">
<b>
#000080
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808000">
<b>
Olive
</b>
```

```
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#808000">
<b>
#808000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800080">
<b>
Purple
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#800080">
<b>
#800080
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ff0000">
<b>
Red
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#ff0000">
<b>
```

```
#FF0000
</b>
</font>
</a>
</TD>
</TR>
<TR bgcolor="#ffffff">
<TD WIDTH="46%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008080">
<b>
Teal
</b>
</font>
</a>
</TD>
<TD WIDTH="54%" VALIGN="TOP">
<a href="#">
<font face="arial" size="3" color="#008080">
<b>
#008080
</b>
</font>
</a>
</TD>
</TR>
</TABLE>
</CENTER>
</body>
</html>
```

Efeitos de animação e recursos multimídia

Neste módulo, vamos estudar a implementação de animações e a utilização de recursos multimídia, com o objetivo de enriquecer a apresentação visual e de entreter os visitantes, tornando a interação destes com o site muito mais agradável e atrativo.

Usando os recursos disponíveis para a realização de animações nos documentos HTML, podemos dar movimento aos elementos, ou ainda, inserir o efeito de piscar em textos. Estes recursos geralmente são utilizados com o propósito de provocar um apelo visual, direcionando a atenção dos visitantes para algo que desejamos colocar em destaque, contudo, devemos dosar bem quais são as situações realmente ideais para a aplicação das animações, pois a exploração excessiva deste recurso pode surtir um efeito contrário, tornando a visualização do site bastante cansativa.

Com a utilização de recursos multimídia, podemos despertar a atenção dos visitantes através de estímulos audiovisuais, além de poder usufruir das inúmeras vantagens com a possibilidade de se disponibilizar áudio e/ou vídeo, seja como uma opção de entretenimento ou como um complemento à informação escrita.

Devemos nos lembrar que nem todos os navegadores estão preparados para exibir tais recursos, sendo prudente utilizar juntamente com as animações, outros recursos da linguagem HTML que permitem destacar uma informação, como por exemplo, o alinhamento centralizado e o uso do estilo de formatação negrito.

No caso do áudio e do vídeo, é recomendado inserir uma descrição alternativa em texto, pois deste modo, possibilitamos a compreensão da informação por parte das pessoas com deficiência auditiva, visual ou por demais pessoas que não possuam os seus computadores equipados com um kit multimídia.

Inserindo movimento aos elementos HTML

Podemos dar movimento a textos, links, imagens e a outros elementos HTML, colocando-os em evidência quando a situação for conveniente, pois conforme orientamos, não é recomendado o uso excessivo deste recurso. Para a realização desta tarefa, contamos com a tag `<marquee>`, que é reconhecida e executada somente pelo navegador Internet Explorer da Microsoft, sendo ignorada pelos demais navegadores.

A tag `<marquee>` é utilizada em par, possuindo abertura e fechamento. Na parte referente a sua abertura, inserimos diversos atributos que controlam características do seu funcionamento.

A declaração básica da tag `<marquee>` é:

```
<marquee> elemento </marquee>
```

Vejamos agora, a descrição dos principais atributos da tag `<marquee>`:

(behavior)

Com este atributo, definimos o estilo de comportamento executado pela animação, podendo ser utilizado um entre os três valores disponíveis:

Behavior="scroll" - Este é o comportamento padrão utilizado pela tag <marquee>. Faz com que o elemento entre por um lado da tela e saia pelo outro, em um ciclo.

Behavior="slide" - Faz com que o elemento entre por um lado da tela e se desloque até o outro lado.

Behavior="alternate" - Faz com que o elemento se desloque de um lado para o outro da tela, em um ciclo.

(direction): Com este atributo, definimos para qual direção será movimentado o elemento:

Direction="left" - Da direita para a esquerda (direção padrão).

Direction="right" - Da esquerda para a direita.

(height): -Com este atributo, definimos em pixels ou em porcentagem da tela, a dimensão vertical que será reservada para a animação.

(width): Com este atributo, definimos em pixels ou em porcentagem da tela, a dimensão horizontal que será reservada para a animação.

(hspace): Com este atributo, podemos estabelecer uma margem horizontal externa à área da animação, onde o valor definido em pixels será assumido tanto pelo lado esquerdo como pelo lado direito.

(vspace): Com este atributo, podemos estabelecer uma margem vertical externa à área da animação, onde o valor definido em pixels será assumido tanto pelo lado superior como pelo lado inferior.

(align): Com este atributo, definimos o alinhamento vertical do elemento, dentro do espaço reservado para a animação. Podemos utilizar um entre os três valores:

Align="top" – Alinha à parte superior.

Align="middle" – Alinha ao centro (alinhamento padrão).

Align="bottom" – Alinha à parte inferior.

(bgcolor): Este atributo nos permite definir uma cor de fundo para a área reservada para a animação, utilizando o padrão RGB.

(loop): Com este atributo, podemos definir o ciclo de repetições que será aplicada a um comportamento estabelecido para a animação:

(Loop="3") – A animação executará o movimento estabelecido por três vezes.

Loop="infinite" – O ciclo de movimentos será executado ininterruptamente.

(scrollamount): Este atributo nos permite definir em pixels, o espaço para cada movimento realizado pelo elemento na animação.

(scrolldelay): Este atributo nos permite definir o tempo em milissegundos, para cada movimento realizado pelo elemento na animação.

Note que com os atributos (scrollamount) e (scrolldelay), podemos modificar conforme a nossa preferência, a velocidade do deslocamento feito pelo elemento na animação, pois quanto maior for o espaço utilizado para cada movimento, maior será a velocidade e quanto maior for o tempo utilizado para cada movimento, menor será a velocidade.

Caso necessário, podemos combinar estas duas variáveis de modo a atingir a velocidade que desejamos.

0

Vejamos alguns exemplos da sua utilização:

```
1.<center>
<marquee behavior="alternate"><font face="Comic Sans MS, verdana, helvetica, Sans-Serif"
size="5" color="#ff0000">
<b>BEM-VINDO</b>
</font></marquee></center>
```

No exemplo acima, definimos que a mensagem de boas-vindas, ficará movimentando na tela horizontalmente de um lado para o outro. Perceba que complementamos a animação, definindo uma formatação para a fonte da mensagem exibida.

```
2. <table border="0" cellspacing="0" cellpadding="0" height="70" width="95%"><tr><td>
<marquee behavior="slide" direction="right" height="60" width="95%" hspace="5" vspace="5">

</marquee></td></tr></table>
```

No exemplo anterior, definimos que a imagem será deslocada do lado esquerdo para o lado direito, percorrendo 95% da dimensão horizontal da janela do navegador. Caso o navegador não suporte o uso de animações com o elemento `marquee`, por precaução, inserimos a imagem em uma tabela com 95% de largura em relação a janela e com o alinhamento à direita, garantindo o posicionamento final idealizado para a imagem.

3. `<center>`

```
<marquee loop="5">  
<font face="Comic Sans MS, verdana, helvetica, Sans-Serif" size="5" color="#ff0000">  
<b> BEM-VINDO </b> </font> <br>  
  
</marquee>  
</center>
```

No exemplo acima, utilizamos o comportamento "scroll" que é o padrão do elemento `marquee`, para que a mensagem de boas-vindas e a fotografia rolem horizontalmente da direita para a esquerda (direção padrão) por cinco vezes consecutivas. Caso o navegador não seja o Internet Explorer da Microsoft, a mensagem aparecerá sobre a fotografia e ambos alinhados ao centro.

Inserindo texto com efeito piscante

Para inserirmos um texto com o efeito piscante, basta utilizar o elemento `blink`, que é válido apenas para os navegadores Netscape Navigator da Netscape.

Como da mesma forma do que no uso do elemento `marquee`, o efeito de piscar deve ser utilizado com ponderação, para que a sua visualização não se torne cansativa para os visitantes do site.

A declaração do elemento `blink` é bastante simples, pois utilizamos apenas uma tag em par com abertura e fechamento, não sendo necessário a inserção de atributos para o controle das suas características de funcionamento.

Sintaxe do elemento `blink`:

```
<blink> TEXTO PISCANDO </blink>
```

Quando houver a necessidade de se destacar um texto, utilize o elemento `blink` juntamente com o elemento `marquee`, pois desta forma, você contemplará com animações distintas, tanto os navegadores Netscape Navigator, como os navegadores Internet Explorer.

Exemplo:

```
<marquee behavior="alternate">  
<center>  
<font face="Comic Sans MS" size="4" color="#ff0000">  
<b>  
<blink>
```

Texto piscando (Netscape Navigator) e em movimento (Internet Explorer)

```
</blink>  
</b>  
</font>  
</center>  
</marquee>
```

Inserindo recurso de áudio em páginas HTML

Através dos recursos de áudio disponíveis, podemos por exemplo inserir uma trilha sonora de fundo, um áudio complementar à informação escrita ou um som de alerta para quando o visitante acessar uma determinada página do site.

Apesar das versões mais atuais dos principais navegadores reconhecerem o (.mp3), os formatos de áudio mais recomendados para serem executados a partir de páginas Web, são: (.mid) para as trilhas sonoras musicais e (.wav) para os demais trechos de áudio.

É importante ressaltar que tanto o navegador Internet Explorer da Microsoft, como o Netscape Navigator da Netscape, possui suas próprias tags para a execução de áudio. Por este motivo, é aconselhável utilizar em conjunto numa mesma página as tags reconhecidas por ambos navegadores, para que possamos contemplar seus distintos usuários.

Recomendamos que as tags utilizadas para a reprodução do áudio, sejam inseridas no final do corpo do documento, antecedendo o fechamento da tag </body>. Com isso, o carregamento do arquivo de áudio não prejudicará o carregamento do restante da página.

Devemos nos lembrar sempre de que para fazer uma visita ao nosso site, os visitantes dependem de uma conexão de rede e por este motivo, não devemos desnecessariamente inserir arquivos de áudio com um tamanho em bytes muito grande, para não prejudicarmos o desempenho do acesso, tornando-o demasiadamente lento e desgastante.

Note que para a reprodução do áudio, será necessário que o visitante do site possua no mínimo o seu computador equipado com um kit multimídia básico (placa de som e um dispositivo de saída e amplificação do áudio).

Áudio para o navegador Internet Explorer

Para inserirmos áudio às páginas acessadas pelo navegador Internet Explorer, devemos utilizar a tag ímpar `<bgsound>` complementada pelos seguintes atributos:

(src): Através deste atributo indicamos o arquivo de áudio no qual deve ser executado. A sua sintaxe é semelhante a do atributo (href) usado na referência dos links.

(loop): Através deste atributo, indicamos o número de vezes pela qual o arquivo de som deve ser tocado. Caso este atributo receba como parâmetro o valor "infinite", a seqüência de repetição do áudio será ininterrupta até que o visitante mude de página ou pressione a tecla (esc).

Exemplos da inserção de áudio para uma página acessada pelo Internet Explorer:

1. No exemplo abaixo, o arquivo trilha.mid será tocado continuamente até que o visitante mude de página:

```
<bgsound src="trilha.mid" loop="infinite">
```

2. No exemplo abaixo, o arquivo som.wav será tocado 5 vezes seguidas:

```
<bgsound src="som.wav" loop="5">
```

Caso omitamos o atributo (loop), o áudio será executado apenas uma única vez.

Áudio para o navegador Netscape Navigator

A sintaxe usada para a inserção de áudio às páginas acessadas a partir do navegador Netscape Navigator, é bastante semelhante com a usada para o navegador Internet Explorer.

Caso queiramos, o navegador Netscape Navigator nos permite exibir um console com algumas opções disponíveis para que o visitante tenha um controle da reprodução do áudio, como: Tocar, Parar, Pausar entre outras.

Através do uso da tag ímpar `<embed>` e de alguns dos seus atributos, podemos controlar as características do console apresentado, bem como a reprodução do áudio.

Vejamos os principais atributos da tag `<embed>`:

(src): Através deste atributo indicamos o arquivo de áudio no qual deve ser executado. A sua sintaxe é semelhante a do atributo `(href)` usado na referência dos links.

(loop): Caso este atributo receba o valor "true", a reprodução do áudio será feita de forma ininterrupta até que o visitante abandone esta página ou pressione a tecla `(esc)`.

(autostart): Caso este atributo receba o valor "true", a reprodução do áudio será inicializada juntamente com o carregamento da página.

(hidden): Caso este atributo receba o valor "true", o console para o controle da reprodução do áudio será omitido.

(height): Com este atributo definimos a altura do console em pixels, caso nós não o omitamos com o uso do atributo `(hidden="true")`.

(width): Com este atributo definimos a largura do console em pixels, caso nós não o omitamos com o uso do atributo `(hidden="true")`.

Exemplo da inserção de um som de fundo que contemple os navegadores Netscape Navigator e Internet Explorer:

```
<body>
(corpo do documento)
<embed src="trilha.mid" loop="true" hidden="true" autostart="true">
<bgsound src="trilha.mid" loop="infinite">
</body>
```

Observe que no exemplo anterior, o navegador Netscape Navigator não exibirá o console para o controle do áudio e em ambos navegadores, o som de fundo será tocado continuamente.

O navegador Internet Explorer da Microsoft ao encontrar a tag **(embed)**, reproduzirá o arquivo de som indicado através da tag **(src)**, caso na máquina do visitante esteja devidamente instalado o aplicativo Windows Media Player.

Vídeo para o navegador Internet Explorer

A Microsoft adicionou à tag utilizada para a exibição de imagens, o atributo (dynsrc) que suporta a execução de clipe de filmes disponíveis no formato AVI (Audio Video Interleave).

Para os demais navegadores que não suportam o uso do atributo (dynsrc), podemos inserir normalmente em conjunto os demais atributos da tag , disponibilizando alternativamente uma imagem estática que será exibida em substituição ao vídeo.

Vejamos os principais atributos usados na execução de vídeos para o navegador Internet Explorer da Microsoft:

(dynsrc): Através deste atributo indicamos o caminho do arquivo de vídeo que será executado. A sua declaração é semelhante a do atributo (href) usado na referência dos links.

(start): Este atributo nos permite indicar o método para a execução do vídeo. Podemos utilizar um entre os dois valores abaixo:

(Start="fileopen"): O vídeo é executado automaticamente com o carregamento da página.

(Start="mouseover"): A execução ocorrerá quando o visitante posicionar o cursor do mouse sobre um ponto dentro da região reservada para a exibição do vídeo, definida através do uso dos atributos (height) e (width).

(loop): Com este atributo podemos definir o ciclo de repetição do vídeo que pode ser um valor absoluto ou "infinite", onde com este último, a execução do vídeo será feita de forma ininterrupta até que o visitante mude de página ou pressione a tecla (esc).

Exemplo da inserção de um vídeo para ser visualizado através de um acesso feito à página com o navegador Internet Explorer:

```

```

Vídeo e áudio por streaming

Caso o usuário possua devidamente instalado e configurado em seu computador players para a execução de vídeo e áudio, estes arquivos quando requisitados através de um link, são automaticamente abertos e executados por seu aplicativo correspondente, como por exemplo, os populares: Windows Media Player da Microsoft e o Real Player da RealNetworks.

Para proteger e melhor organizar concentrando os arquivos multimídia em pastas específicas, é recomendado criar um documento ASCII (texto puro) com a extensão asx, contendo o caminho completo do áudio ou vídeo a ser executado, substituindo o protocolo HTTP pelo protocolo MultiMedia Streaming MMS.

A transmissão via Streaming, consiste no armazenamento inicial de um buffer contendo um trecho do arquivo a ser executado antes de começar a reproduzi-lo, para que desta forma possa se reduzir ou eliminar as falhas causadas pelas interrupções durante a transmissão dos dados.

O link para a chamada do arquivo video.asx segue as mesmas regras usadas na criação de um link convencional:

Suponha que o arquivo institucional.wmv esteja salvo na pasta "videos", portanto o arquivo video.asx deve conter a chamada da seguinte forma:

```
<a href="video.asx"> Víde&iacute;o institucional </a>
```

De acordo com o exemplo abaixo, caso o visitante possua devidamente instalado em seu computador o utilitário Windows Media Player da Microsoft, o vídeo será carregado e exibido.

```
<A
```

```
  HREF="mms://www.servidor.sigla.país/videos/institucional.wmv">mms://www.servidor.sigla.país/vid  
eos/institucional.wmv</A>
```

Formulários eletrônicos

Neste módulo, vamos estudar a criação dos formulários eletrônicos, um importantíssimo recurso que possibilita uma valorosa interação com os visitantes do site.

O uso de formulários tem por objetivo possibilitar a colheita de informações através de controles, podendo posteriormente gravar estas informações em um banco de dados, salvá-las em arquivos de texto, apenas processá-las a fim de retornar alguma informação para o usuário, enviá-las para um endereço de e-mail etc.

A interação com os visitantes do site é promovida através do uso de controles, dando-lhes a oportunidade de digitar uma informação em uma caixa de texto, selecionar um item em uma lista Drop Down, marcar opções através de caixas de verificação, confirmar enviando as informações para o servidor ou desistir limpando o formulário através de botões específicos etc.

Os controles estão basicamente divididos em três classes: input, textarea e select.

Com certeza você já deve ter interagido com um formulário eletrônico na Internet, diante de uma ou mais das seguintes situações:

- Ao informar dados na realização de um cadastro.
- Ao pesquisar sobre um determinado assunto nos sites de busca.
- Ao responder a uma enquete.
- Ao enviar uma mensagem a um fórum de discussão.
- Ao autenticar-se, informando "nome de usuário" e "senha" para ter acesso a uma área restrita de um site.
- Ao participar de um chat.
- Ao consultar seus e-mails através de um webmail etc.

Observe bem a sintaxe aplicada à criação dos formulários eletrônicos com seus respectivos controles, para que você possa explorar ao máximo as inúmeras possibilidades advindas deste recurso.

Criando um formulário

Para criar um formulário, usamos uma tag em par que delimita a sua abertura e o seu fechamento.

Dentro de um formulário, podemos inserir normalmente, todos os recursos disponíveis da linguagem HTML, podendo aplicá-los na formatação e no alinhamento dos textos e dos controles.

Uma página em HTML pode conter mais de um formulário, porém, nunca um formulário poderá ser criado dentro de um outro.

Sintaxe utilizada para a criação de formulários:

<form> - abertura do formulário.

(Inserção dos controles e tags da linguagem HTML)

</form> - Fechamento do formulário.

Principais atributos que devem ser declarados na abertura do formulário:

(name) - Usado para definir um nome para o formulário.

Name="nome_do_formulario" – Onde "nome_do_formulario" deverá ser substituído pelo nome que deseja atribuir ao formulário.

(method) - Através deste atributo definimos qual será o método utilizado para a transmissão dos dados inseridos nos controles do formulário. Existem dois métodos: "get" e "post".

Method="get" – Envia a informação juntamente com a URL. É comum observarmos a utilização deste método em alguns sites de busca.

Method="post" – Por enviar as informações em blocos internamente e invisíveis aos olhos do visitante, este método tornou-se o mais utilizado e recomendado para a transmissão de dados via formulário na Internet, principalmente em situações onde o volume de informações é maior.

(action) - Este atributo é declarado da mesma forma que o atributo (href) usado na inserção de links, onde o arquivo referenciado será responsável em processar as informações enviadas do formulário.

Action="arquivo_processa" – Onde "arquivo_processa" deverá ser substituído pelo endereço do arquivo que irá processar as informações no servidor.

Este arquivo poderá ser um CGI (Common Gateway Interface – protocolo utilizado em programas executáveis para processar informações enviadas ao servidor), um arquivo script (conjunto de comandos que são inseridos juntamente com tags da linguagem HTML, mas que são interpretados e executados pelo servidor e não pelo browser) e outros.

Exemplo genérico da criação de um formulário:

```
<form name="form1" method="post" action="cgibin/mail.cgi">  
(controles do formulário)  
</form>
```

Normalmente, a maioria dos servidores para hospedagem de sites, inclusive os gratuitos, oferecem esta possibilidade para seus usuários juntamente com as descrições e códigos necessários para a sua implementação.

Você deverá solicitar maiores informações com os administradores do servidor onde deseja hospedar o seu site.

Existem atualmente na Internet, diversos programas CGIs disponíveis para o uso público, com a finalidade de processar e encaminhar para um e-mail informado, os dados inseridos em um formulário. Isso significa que você poderá incorporar este recurso em seu site.

A Classe input

Usando a classe input, podemos criar os seguintes controles:

- Caixa de texto (text).
- Controle oculto (hidden).
- Caixa de verificação (checkbox).
- Caixa de opção (radio).
- Botões (button, submit e reset).

A criação dos controles da classe input apresenta uma sintaxe bastante semelhante, onde inclusive alguns atributos são comuns a todos os seus controles.

Vejam agora, quais são estes atributos:

(type): Através deste atributo, informamos qual o tipo de controle que desejamos criar, por exemplo: Uma caixa de texto, uma caixa de verificação, um botão etc.

(name): Com este atributo, definimos um nome interno para o controle.

É através deste nome que a aplicação no servidor identificará os diversos dados informados no formulário, portanto, é sempre recomendado dar ao controle um nome que tenha uma relação com a informação que será inserida nele.

(value): Este atributo recebe e armazena as informações inseridas nos controles para que estas sejam lidas e processadas pela aplicação no servidor.

O atributo (value) de uma caixa de texto é preenchido quando o usuário digita algum dado na mesma, por isso, caso deseje que a caixa de texto apareça sem nenhuma informação prévia, declare-a como vazia, colocando após o sinal de igual (=), duas aspas da seguinte forma:

Value="" - A tag utilizada para a criação dos controles da classe input, não necessita de uma parte aplicada ao seu fechamento.

Sintaxe genérica usada na criação de controles da classe input:

```
<input type="tipo_controle" name="nome_controle" value="vazio_ou_valor_controle">
```

Além dos atributos apresentados acima, alguns controles possuem também seus atributos específicos, a fim de definirem as suas particularidades.

Caixa de texto

Este controle permite a entrada de dados, digitados em uma única linha e deve ser usado quando forem solicitadas informações como: nome, endereço, bairro, cidade, CEP, telefone, e-mail etc.

Dica: O texto usado para descrever a informação que será inserida em um controle, deverá estar bem próximo ao mesmo, posicionado acima ou à sua esquerda. Este posicionamento é facilmente conseguido com a utilização de tabelas.

Sintaxe utilizada para a criação de uma caixa de texto:

```
<input type="text" name="curso" value="" size="40" maxlength="50">
```

Onde: Usamos o atributo (type) para indicar que o controle criado será uma caixa de texto.

No atributo (name), declaramos que o nome interno da caixa de texto será "curso".

No atributo (value), não inserimos nenhum valor para que a caixa de texto seja carregada vazia. Caso queiramos que seja apresentado algum valor inicial, devemos inseri-lo entre as aspas, exemplo:

```
<input type="text" name="curso" value="Curso de HTML" size="40" maxlength="50">
```

O atributo (size) define o tamanho no qual a caixa de texto será apresentada na página. Neste caso, definimos seu tamanho em "40" caracteres.

Usando o atributo (maxlength), definimos a quantidade de caracteres que serão aceitos pela caixa de texto. Neste caso, serão aceitos até "50" caracteres mesmo que na tela sejam visíveis somente "40".

Controle oculto

Inserimos um controle oculto no formulário, para que determinadas informações sejam enviadas ao servidor, sem que o usuário possa visualizá-las na apresentação da página.

Sintaxe utilizada para a criação de um controle oculto:

```
<input type="hidden" name="descrição" value="cadastro para receber novidades sobre o site">
```

Onde: Com o atributo (type) definimos a criação de um controle oculto "hidden".

Através do atributo (name), informamos que o nome interno do controle oculto será "descrição".

Usando o atributo (value), inserimos a informação que será enviada ao servidor para ser processada.

O ideal é que o controle oculto seja inserido no topo do formulário, logo após a sua abertura.

Neste caso, você receberá um e-mail contendo na primeira linha a seguinte informação:

Descrição: cadastro para receber novidades sobre o site.

Quando em seu site contiver mais de um formulário, é recomendado inserir em cada um deles, um controle oculto com a sua descrição para que ao receber as informações no seu e-mail, por exemplo, você saiba a que elas se referem.

Caixa de verificação

Uma caixa de verificação é um controle lógico, que apresenta sempre dois estados (marcado ou desmarcado), onde o usuário deve decidir se aceita ou não, uma condição que lhe foi proposta. Na maioria das vezes, este tipo de controle é usado diante de situações que expressem um sentido de questionamento.

Sintaxe utilizada para a criação de uma caixa de verificação:

```
<input type="checkbox" name="informações" value="sim">
```

Onde: Através do atributo (type), definimos que o controle criado será uma caixa de verificação "checkbox".

Com o atributo (name), definimos o nome interno da caixa de verificação.

No atributo (value), definimos qual deverá ser a informação enviada ao servidor caso o usuário aceite a condição que lhe foi proposta, marcando a caixa de verificação.

Da forma que foi declarada acima, a caixa de verificação aparecerá desmarcada. Para que a mesma apareça marcada inicialmente, devemos usar seu atributo específico (checked) da seguinte forma:

```
<input type="checkbox" name="informações" value="sim" checked>
```

Caso omitamos o atributo (value) e o usuário marque a caixa de verificação, o valor enviado ao servidor será "on" e caso a caixa de verificação não seja marcada, nenhuma informação será enviada ao servidor.

Caixa de opção

Uma caixa de opção é um controle que disponibiliza para o usuário, várias opções e destas, o mesmo poderá selecionar apenas uma.

Regras aplicadas na criação de uma caixa de opção:

- Cada nova opção adicionada deve ser antecedida pela tag usada na criação deste controle.
- Por uma questão de acessibilidade, as opções devem estar dispostas verticalmente (uma por linha), para que os programas leitores de tela possam identificá-las individualmente. Para tanto, ao final da descrição de cada opção, utilize a tag (
 - quebra de linha).
- Todas as opções pertencentes a uma mesma caixa de opção, devem receber a mesma descrição no atributo (name), para que sejam agrupadas em um único controle.

Para melhor demonstrar a sintaxe da tag utilizada na criação do controle caixa de opção, imagine a seguinte situação:

Suponha que peçamos para os visitantes classificarem o nosso site como: Ótimo, Bom, Regular, Ruim ou Péssimo.

A sintaxe utilizada para a criação de uma caixa de opção que contemple a situação exposta acima será:

```
<input type="radio" name="classificação" value="Ótimo"> <br>  
<input type="radio" name="classificação" value="Bom"> Bom <br>  
<input type="radio" name="classificação" value="Regular" checked> Regular <br>  
<input type="radio" name="classificação" value="Ruim"> Ruim <br>  
<input type="radio" name="classificação" value="Péssimo"> Onde: Com o atributo (type)  
declaramos que o controle criado será uma caixa de opção "radio".
```

Através do atributo (name) definimos um nome interno para o controle e que deverá ser inserido em todos os itens que compõem a caixa de opção.

No atributo (value) definimos a informação em que cada item da caixa de opção enviará ao servidor quando selecionado.

O atributo (checked) define qual das opções estará selecionada inicialmente. Perceba que na demonstração acima, nós determinamos que a opção que deverá aparecer selecionada inicialmente é a que contém a descrição "regular".

Botões

O botão é um tipo de controle utilizado para executar uma determinada ação, com os dados inseridos nos demais controles do formulário. Na classe input existe três tipos de botões disponíveis. São eles:

Button: Este tipo de botão não envia automaticamente as informações do formulário para o servidor. É comumente utilizado quando o processamento das informações é realizado no lado cliente, através das linguagens de script locais, embutidas nos próprios navegadores como: JavaScript e VBScript.

Submit: Este tipo de botão ao ser pressionado, envia automaticamente todas as informações inseridas no formulário para que estas possam ser processadas por uma aplicação no servidor.

Reset: Este tipo de botão ao ser pressionado, limpa todos os campos existentes no formulário, resgatando as suas definições iniciais.

A sintaxe utilizada para a criação de botões, é idêntica independentemente do seu tipo.

Por atender bem aos nossos objetivos, vamos utilizar apenas os botões do tipo "submit" e "reset" que possibilitam respectivamente, o envio das informações ao servidor e a limpeza dos campos disponíveis no formulário.

Sintaxe utilizada na criação de botões:

```
<input type="submit" name="enviar" value="Enviar">  
<input type="reset" name="limpar" value="Limpar">
```

Onde: Utilizando o atributo (type), definimos o tipo de botão "submit" ou "reset".

O atributo (name) é utilizado para definir os nomes internos para os botões criados. Com o atributo (value), definimos qual será o rótulo (caption) apresentado pelos botões. No caso acima, o botão do tipo "submit" apresentará como descrição a palavra "Enviar" e o botão do tipo "reset" apresentará como descrição a palavra "Limpar". A descrição inserida em um botão deverá condizer com a ação que o mesmo executará ao ser pressionado.

Exemplo de um formulário eletrônico que utiliza os controles da classe input

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>  
<head>  
<title> FORMULÁRIO ELETRÔNICO - CONTROLES DA CLASSE INPUT </title>  
</head>  
<body>  
<br>  
<center><font face="comic sans ms, verdana, sans-serif" size="4"  
color="#000099"> <b>FORMULÁRIO ELETRÔNICO - CONTROLES DA CLASSE  
INPUT</b></font></center>  
<br><br>  
<center>  
<table border="0" cellspacing="0" cellpadding="0">  
<form name="form1" method="get" action="http://www.meuservidor.com.br/cgibin/mail.cgi">  
<input type="hidden" name="Identificação" Value="Exemplo utilizando os controles da classe  
input">  
<tr>  
<td align="right"><b> Nome: </b></td>  
<td align="left"> <input type="text" name="Nome" value="" size="50" maxlength="50"> </td>
```

```
</tr>
<tr>
<td align="right"><b> Sexo: </b></td>
<td align="left"> <input type="radio" name="sexo" value="m"> Masculino <br>
<input type="radio" name="sexo" value="f"> Feminino </td>
</tr>
<tr>
<td align="right"><b> Deseja receber mensalmente o boletim com novidades sobre o site?
</b></td>
<td align="left"> <input type="checkbox" name="informacoes" value="sim" checked> </td>
</tr>
<tr>
<td align="right"> <input type="submit" value="Enviar"> </td>
<td align="left"> <input type="reset" value="Limpar"> </td>
</tr>
</form>
</table>
</center>
</body>
</html>
```

A classe textarea

A classe textarea permite a criação de um único tipo de controle, conhecido na linguagem HTML, pelo próprio nome da classe.

O controle textarea, difere-se do controle caixa de texto "text" da classe input, por permitir ao usuário a possibilidade de digitar uma informação que abranja mais de uma linha de texto. Este tipo de controle é comumente utilizado em situações que exijam um espaço maior para a entrada de dados, como campos para observações, comentários, opiniões etc.

Para a criação de um controle textarea é utilizado uma tag em par, possuindo partes referentes à sua abertura e ao seu fechamento.

Sintaxe genérica para a criação do controle textarea:

```
<textarea name="nome" rows="n_linhas" cols="n_colunas"> valor_inicial
</textarea>
```

Onde: No atributo (name) devemos substituir o valor "nome" pela palavra que definirá o nome interno para o controle.

Devemos nos lembrar que o nome interno de um controle deve ser sempre condizente com a informação que será inserida no mesmo.

Com o atributo (rows), definimos a quantidade de linhas para o controle textarea, portanto devemos substituir na declaração acima, o valor "n_linhas" pelo número que corresponda à quantidade de linhas desejada.

Com o atributo (cols), definimos a quantidade de colunas para o controle textarea, portanto devemos substituir na declaração acima, o valor "n_colunas" pelo número que corresponda à quantidade de colunas desejada.

Para que uma informação seja previamente inserida no controle textarea, devemos substituir na declaração acima, valor_inicial pela informação desejada. Caso queiramos que o controle textarea não apresente nenhuma informação inicial, basta que a tag seja fechada imediatamente após a definição da parte referente à sua abertura.

Veja um exemplo da criação de um controle textarea, criado com o objetivo de permitir que os visitantes informem as suas opiniões sobre o nosso site. Para tanto, o controle textarea deverá conter 10 linhas, 50 colunas e não deverá apresentar nenhuma informação inicial:

```
<textarea name="opiniões" rows="10" cols="50"></textarea>
```

Exemplo de um formulário eletrônico que utiliza o controle da classe textarea

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> FORMULÁRIO ELETRÔNICO - CONTROLE DA CLASSE TEXTAREA </title>
</head>
<body>
<br>
<center><font face="comic sans ms, verdana, sans-serif" size="4"
color="#000099"> <b>FORMULÁRIO ELETRÔNICO - CONTROLE DA CLASSE
TEXTAREA</b></font></center>
<br><br>
<center>
<table border="0" cellspacing="0" cellpadding="0">
```

```
<form name="form2" method="get" action="http://www.meuservidor.com.br/cgi-bin/mail.cgi">
<input type="hidden" name="Identificação" Value="Exemplo utilizando o controle da classe
textarea">
<tr>
<td align="right"><b> Nome: </b></td>
<td align="left"> <input type="text" name="Nome" value="" size="50" maxlength="50"> </td>
</tr>
<tr>
<td align="right"><b> Sexo: </b></td>
<td align="left"> <input type="radio" name="sexo" value="m"> Masculino <br>
<input type="radio" name="sexo" value="f"> Feminino </td>
</tr>
<tr>
<td align="right"><b> Deseja receber mensalmente o boletim com novidades sobre o site?
</b></td>
<td align="left"> <input type="checkbox" name="informacoes" value="sim" checked> </td>
</tr>
<tr>
<td align="right"><b> Digite seus comentários: </b></td>
<td align="left"> <textarea name="Comentários" rows="10" cols="50">< /textarea> </td>
</tr>
<tr>
<td align="right"> <input type="submit" value="Enviar"> </td>
<td align="left"> <input type="reset" value="Limpar"> </td>
</tr>
</form>
</table>
</center>
</body>
</html>
```

A classe select

Usando a classe select, podemos inserir em nosso formulário eletrônico, controles como listas e caixas combinadas.

O controle lista, possibilita a apresentação dos seus itens em grupos e com quantidades preestabelecidas. Por apresentar esta característica de visualização, é ideal para permitirmos ao usuário caso seja necessário, a seleção de um ou mais itens, entre os disponíveis neste controle.

O controle caixa combinada possibilita inicialmente a visualização apenas do item que foi previamente selecionado. Para ter acesso aos demais itens é necessário abrir a caixa combinada para a seleção do item desejado.

Dica (1): Durante a navegação, o atalho de teclado utilizado para abrir uma caixa combinada disponível em um formulário eletrônico, é (alt + "seta para baixo"). Após a seleção com as setas do item desejado, o fechamento da caixa combinada se dá automaticamente com a perda do foco do cursor, ou com o pressionamento da tecla (enter) sobre a mesma.

Diferentemente da caixa de opção "radio" da classe input, utilizaremos os controles da classe select, para situações que exijam um número maior de opções, ou quando houver a necessidade do usuário realizar uma múltipla seleção entre os itens disponibilizados em uma lista.

É importante destacar que as listas de definição, desordenadas, ordenadas e as intercaladas, apresentadas anteriormente para o posicionamento de informações em páginas HTML, não possuem nenhuma relação com as listas da classe select, geradas como controles de formulários eletrônicos.

A diferença básica entre a sintaxe utilizada para a criação de uma lista ou de uma caixa combinada, está relacionada com a definição da quantidade de itens que serão exibidos simultaneamente e com a capacidade de permitir ou não, a múltipla seleção dos itens inseridos em cada um destes controles.

Sintaxe utilizada na criação de listas e caixas combinadas:

```
<select name="nome_controle" size="n_itens"> - Parte referente à abertura da tag select.  
<option value="valor_item1" selected> Descrição da primeira opção  
<option value="valor_item2"> Descrição da segunda opção  
<option value="valor_item3"> Descrição da terceira opção  
<option value="valor_item4"> Descrição da quarta opção  
<option value="etc."> etc.  
</select> - Parte referente ao fechamento da tag select.
```

A tag select delimita a abertura e o fechamento de uma lista ou caixa combinada. Dentro da tag select, inserimos a tag option para cada novo item adicionado ao controle.

Atributos da tag select

(name): A variável "nome_controle" deve ser substituída pelo nome interno referente ao controle.

(size): A variável "n_itens" deve ser substituída por um número que definirá a quantidade de itens que serão exibidos simultaneamente pelo controle criado. Caso o atributo (size) seja omitido, ele assumirá o seu valor padrão "1" e o controle criado será uma caixa combinada. Caso o atributo (size) receba como valor um número maior do que "1", o controle criado será uma lista.

(multiple): A inserção deste atributo permite a seleção de múltiplos itens em uma lista.

Dica (2): Quando for necessário utilizar o atributo (multiple), permita que o usuário visualize um conjunto de itens simultaneamente, para que o mesmo possa verificar as opções anteriormente selecionadas nesta lista.

Atributos da tag option:

(value): Com este atributo, definimos a informação que cada item enviará ao servidor, caso o mesmo seja selecionado.

(selected): Com este atributo, definimos entre os itens inseridos, qual aparecerá previamente selecionado.

No exemplo abaixo observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> FORMULÁRIO ELETRÔNICO - CONTROLE DA CLASSE SELECT </title>
</head>
<body>
<br>
<center><font face="comic sans ms, verdana, sans-serif" size="4"
color="#000099"><b>FORMULÁRIO ELETRÔNICO - CONTROLE DA CLASSE
SELECT</b></font></center>
<br><br>
<center>
<table border="0" cellspacing="0" cellpadding="0">
<form name="form3" method="get" action="http://www.meuservidor.com.br/cgi-bin/mail.cgi">
<input type="hidden" name="Identificação" Value="Exemplo utilizando o controle da classe select">
<tr>
<td align="right"><b> Nome: </b></td>
<td align="left"> <input type="text" name="Nome" value="" size="50" maxlength="50"> </td>
```

```
</tr>
<tr>
<td align="right"><b> Sexo: </b></td>
<td align="left"> <input type="radio" name="sexo" value="m"> Masculino <br>
<input type="radio" name="sexo" value="f"> Feminino </td>
</tr>
<tr>
<td align="right"><b> Deseja receber mensalmente o boletim com novidades sobre o site?
</b></td>
<td align="left"> <input type="checkbox" name="informacoes" value="sim" checked> </td>
</tr>
<tr>
<td align="right"><b> Digite seus comentários: </b></td>
<td align="left"> <textarea name="Comentários" rows="10" cols="50"> </textarea> </td>
</tr>
<tr>
<td align="right"><b> Classifique o nosso site em: </b></td>
<td align="left"> <select name="Classificação">
<option value="Ótimo"> Ótimo
<option value="Bom"> Bom
<option value="Regular" selected> Regular
<option value="Ruim"> Ruim
<option value="Péssimo"> Péssimo
</select> </td>
</tr>
<tr>
<td align="right"> <input type="submit" value="Enviar"> </td>
<td align="left"> <input type="reset" value="Limpar"> </td>
</tr>
</form>
</table>
</center>
</body>
</html>
```

Cascading style sheets

Neste módulo, vamos estudar a implementação dos CSS (Cascading Style Sheets), ou simplesmente, folhas de estilo, que é um mecanismo complementar incorporado inclusive aos navegadores Microsoft Internet Explorer e Netscape Navigator, recomendado pelo W3C (World Wide Web Consortium), órgão que regulamenta as tecnologias aplicadas a Web.

A utilização deste recurso, nos permite um controle abrangente sobre os elementos da linguagem HTML, definindo características de formatação como fontes, cores, alinhamentos, espaçamentos etc.

Podemos implementar os estilos de formatação de três formas diferentes:

Na tag de um elemento: Os efeitos da formatação serão aplicados exclusivamente neste elemento.

No cabeçalho da página: Os efeitos da formatação poderão ser aplicados a qualquer elemento desta página.

Em um arquivo de estilos: Os efeitos da formatação poderão ser aplicados aos elementos de qualquer página, desde que esta faça referência ao arquivo que contém as definições dos estilos de formatação.

A primeira forma descrita acima, deve ser utilizada somente em situações específicas ou provisórias, pois conforme foi descrito, os efeitos definidos no estilo de formatação, atinge apenas a um único elemento.

Para que a sua aplicação seja ampliada, devemos utilizar elementos containers, pois estes possuem a capacidade de agrupar outros elementos e lhes transmitir as suas características de formatação.

Com a terceira forma descrita, podemos criar diversos estilos de formatação dentro de um arquivo, para que sejam aplicados a conteúdos de diversas páginas do site e que possuam características semelhantes, padronizando a apresentação destes, além de facilitar futuras alterações na visualização dos mesmos, pois basta que sejam alteradas apenas as informações definidas nos estilos de formatação, para que estas mudanças sejam assumidas automaticamente por todos os elementos que fazem referência às formatações contidas no arquivo de estilos.

O mecanismo CSS está na sua segunda edição, o CSS2 (desde maio de 1998), mas por não ter sido totalmente implementado aos diversos navegadores, vamos basear os nossos estudos no CSS1 (desde dezembro de 1996), que além de ter sido incorporado ao CSS2, é reconhecido pelos navegadores de maior uso no mercado.

Atualmente, está em desenvolvimento o CSS3.

Vamos estudar o mecanismo CSS, a fim de beneficiarmos das suas vantagens na implementação e alteração das características de formatação de um site.

Propriedades e valores aplicados a criação de estilos

Para a criação dos estilos, utilizamos uma lista de propriedades específicas do CSS, onde a estas propriedades, são atribuídos valores que definem as características de formatação aplicadas a um ou mais elementos da linguagem HTML.

É com o uso destas propriedades que determinamos, por exemplo: o nome da fonte, o tamanho, a cor, o estilo de formatação (negrito - itálico - sublinhado), o alinhamento do texto etc.

A sintaxe utilizada na definição das propriedades é bastante simples, basta inserirmos entre a propriedade e o seu valor, o sinal de atribuição ":" (dois pontos) e separarmos todas as propriedades com seus respectivos valores com o sinal ";" (ponto e vírgula).

Exemplo genérico:

- Propriedade(1): valor(1);
- Propriedade(2): valor(2);
- Propriedade(x): valor(x);

Lista de algumas propriedades e dos seus respectivos valores, que são mais comumente utilizados na definição dos estilos de formatação:

Font-family: Arial; - Define a fonte utilizada e neste caso, será aplicada a fonte Arial. Para definirmos outras fontes por precaução, devemos separar os demais nomes com "," (vírgulas) e apenas após o último nome, encerrarmos a declaração desta propriedade com ";" (ponto e vírgula). Exemplo:

Font-family: Comic Sans MS, Verdana, Helvetica;

Font-size: 14pt; - Define o tamanho da fonte em pontos equivalentes às definições utilizadas pelos editores de textos, como por exemplo, no Microsoft Word. Neste caso, usamos o tamanho 14. Podemos ainda, definir o tamanho da fonte com base na altura em pixels da área que temos a intenção de reservar para a linha de texto, para tanto, devemos substituir o complemento pt por px. Exemplo:

Font-size: 12px; - Neste caso a altura da área reservada para a exibição do texto será de 12 pixels e comportará a fonte tamanho 9 em pontos.

Font-style: italic; - Caso declarado define o estilo itálico para a fonte.

Font-weight: bold; - Caso declarado define o estilo negrito para a fonte.

Text-decoration: underline; - Caso declarado define o estilo sublinhado para a fonte.

Text-Align: center; - Define o alinhamento horizontal para o texto inserido em um elemento. Neste caso, atribuímos o alinhamento centralizado. Pode ser ainda: (left) à esquerda, (right) à direita e (justify) justificado.

- **Vertical-align:** middle; - Define o alinhamento vertical. Neste caso, alinhamos ao centro. Entre outros valores, podemos utilizar: (top) superior e (bottom) inferior.
- **Height: 30px;** - Define em pixels a altura da área utilizada para a aplicação de um estilo de formatação. Neste caso, reservamos 30 pixels.
- **Width: 120px;** - Define em pixels a largura da área utilizada para a aplicação de um estilo de formatação. Neste caso, reservamos 120 pixels.
- **Color: #000000;** - Define a cor da fonte usando o padrão RGB. Neste caso, aplicamos a cor preta.
- **Background-color: #ffffff;** - Define a cor de fundo utilizando o padrão RGB. Neste caso, atribuímos a cor branca.
- **Background-image:** url("url_da_imagem"); - Define uma imagem de fundo. O endereço da imagem deve ser passado como valor ao atributo (url).

Esta relação contém apenas algumas das propriedades mais utilizadas entre as inúmeras disponíveis no CSS1. Para ter acesso a uma lista completa das propriedades e dos seus respectivos valores, bem como da aplicação destas nos elementos HTML, acesse a seguinte publicação pertencente ao W3C (em inglês):

<http://www.w3.org/TR/REC-CSS1>

Inserindo estilo de formatação em um elemento HTML

Um estilo de formatação inserido dentro da tag de um elemento da linguagem HTML, terá os efeitos definidos, aplicados somente a este elemento.

A sintaxe utilizada para definir um estilo de formatação para um elemento, consta na declaração do atributo (`style`), que recebe como valor um conjunto de propriedades com seus respectivos valores.

Exemplos:

1. Para definir o plano de fundo de uma página:

```
<body style="background-color: #ffffff; background-image: url("fundo.gif");">
```

2. Para definir as características da fonte em um parágrafo:

```
<p style="font-family: Verdana; font-size: 14pt; color: #ff0000; font-weight: bold; text-align: center;"> Este texto utiliza a fonte Verdana, tamanho 14, cor vermelha, em negrito e com o alinhamento centralizado. </p>
```

Deste modo, podemos definir características individuais de formatação para os diversos elementos da linguagem HTML.

Entendendo a hierarquia dos elementos HTML

A linguagem HTML está estruturada sobre um modelo de objetos (DOM – Document Object Model), onde os seus elementos estão classificados dentro de uma cadeia hierárquica e deste modo, os elementos inseridos em níveis mais altos desta hierarquia, podem ter as suas propriedades modificadas pelos elementos que estão em um nível mais baixo.

Esta técnica está diretamente relacionada com o modelo de programação orientado a objetos, que acrescentou inúmeras vantagens sobre os outros modelos de programação.

Para entendermos bem este relacionamento na linguagem HTML, vamos analisar as duas situações descritas logo abaixo:

Perceba que ao definirmos na tag `<body>` a formatação de cores para os links carregados (`link`), ativos (`alink`) e visitados (`vlink`), estas características são herdadas automaticamente pelos elementos `<a>` (links) que estão definidos dentro do bloco `<body>` (corpo do documento), desde que nestes, não seja aplicada uma formatação individual, pois primeiramente são respeitadas as características particulares de cada elemento.

O elemento `<body>`, só consegue alterar as propriedades dos elementos `<a>` (links), por estar em um nível hierárquico menor.

Perceba que ao definirmos na tag `<body>` a formatação de cores para os links carregados (`link`), ativos (`alink`) e visitados (`vlink`), estas características são herdadas automaticamente pelos elementos `<a>` (links) que estão definidos dentro do bloco `<body>` (corpo do documento), desde que nestes, não seja aplicada uma formatação individual, pois primeiramente são respeitadas as características particulares de cada elemento.

O elemento `<body>`, só consegue alterar as propriedades dos elementos `<a>` (links), por estar em um nível hierárquico menor.

Para realizar esta segunda análise, observe o trecho de código no exemplo ao lado:

```
<body>
<font style="font-family: Verdana; font-size: 14pt; color: #0000ff;">
<p align="justify"> Texto (1), usando a fonte Verdana, tamanho 14 e cor azul. </p>
<p align="justify"><font style="font-family: Arial; font-size: 12pt; color: #000000; font-weight:
bold;"> Texto (2), usando a fonte Arial, tamanho 12, cor preta e em negrito. </font> </p>
<p align="justify"> Texto (3), usando a fonte Verdana, tamanho 14 e cor azul. </p>
</font>
</body>
```

Os três parágrafos acima, estão delimitados pelo elemento ``, porém, as suas características de formatação são herdadas somente pelos parágrafos que contém respectivamente o "Texto (1)" e o "Texto (3)", pois o parágrafo que contém o "Texto (2)" possui internamente a sua própria definição de fonte.

O uso de elementos containers

Agora que já conhecemos bem o funcionamento da relação hierárquica entre os elementos HTML, podemos estudar sem mistérios a utilização dos elementos containers para a aplicação dos estilos de formatação.

Os elementos containers são aqueles que possuem a capacidade de agrupar outros elementos e desta forma, conseguem lhes transmitir as suas características de formatação.

Na linguagem HTML existem dois elementos containers especiais, o `` e o `<div>`, muito utilizados na geração de efeitos dinâmicos em páginas HTML (a tecnologia que possibilita esta prática, é chamada de DHTML) onde estes, podem ser utilizados com êxito na aplicação dos estilos de formatação a um grupo de elementos que compartilhem características comuns de formatação.

A diferença básica entre os elementos `` e `<div>`, está no fato em que o elemento ``, posiciona os elementos lado a lado, desde que os elementos agrupados não forcem uma quebra de linha e o elemento `<div>`, força uma quebra de linha automaticamente logo após o fechamento da sua tag. Para que você entenda bem esta diferença, observe o trecho de código exemplo ao lado:

```
<body>
<span style="background-color: #00ff00; font-family: Arial; font-size: 20px; color: #ffff00; font-weight: bold; height: 30px; vertical-align: middle; text-align: center;"> AMARELO </span>
<span style="background-color: #ffff00; font-family: Arial; font-size: 20px; color: #00ff00; font-weight: bold; height: 30px; vertical-align: middle; text-align: center;"> VERDE </span>
<br><br>
<div style="background-color: #00ff00; font-family: Arial; font-size: 20px; color: #ffff00; font-weight: bold; height: 30px; vertical-align: middle; text-align: center;"> AMARELO </div>
<div style="background-color: #ffff00; font-family: Arial; font-size: 20px; color: #00ff00; font-weight: bold; height: 30px; vertical-align: middle; text-align: center;"> VERDE </div>
</body>
```

Como resultado da aplicação do exemplo anterior, obteremos:

1. Utilizando o elemento ``: A palavra "AMARELO" será exibida com a fonte da cor amarela sobre o fundo verde e logo ao lado, a palavra "VERDE" será exibida com a fonte da cor verde sobre o fundo amarelo.
2. Utilizando o elemento `<div>`: A palavra "AMARELO" será exibida com a fonte da cor amarela sobre o fundo verde e logo abaixo devido a quebra de linha, a palavra "VERDE" será exibida com a fonte da cor verde sobre o fundo amarelo.

Perceba que no exemplo acima, aplicamos o estilo de formatação definido no elemento container apenas em uma palavra, mas o seu uso se torna mais interessante quando estendido a vários elementos que compartilhem características comuns de formatação.

Regras para a definição dos estilos de formatação

No CSS, para que os estilos de formatação sejam utilizados de uma forma mais flexível, contamos com a possibilidade de definirmos regras através de classes, ids, pseudo-classes, selecionadores, grupos de selecionadores etc.

Na definição, para se distinguir a declaração de uma classe de um id, devemos utilizar antes do nome atribuído como referência à classe, o sinal "." (ponto) e antes do nome atribuído como referência a um id, o sinal "#" (sustenido).

Na chamada, um elemento deve fazer referência a uma classe através do atributo (`class="nome_da_classe"`) ou a um id através do atributo (`id="nome_do_id"`), ambos inseridos na abertura da tag de cada elemento que receberá as formatações previamente definidas.

Apesar da sintaxe de criação e utilização das classes e dos ids serem bastante semelhantes, na prática existe uma diferença, pois um id tem seu foco voltado ao elemento, funcionando como uma referência lógica do mesmo, sendo o seu uso mais apropriado para a geração de efeitos através do HTML dinâmico (DHTML), em quanto uma classe tem o seu foco voltado às definições de formatação, podendo ser aplicada e compartilhada por diversos elementos simultaneamente.

Uma pseudo-classe consiste na declaração de uma regra específica, sendo aplicada na formatação das características de apresentação das âncoras (links), onde podemos definir aparências individuais para os links não visitados, visitados, para o link ativo ou para o atualmente posicionado sob o cursor do mouse.

Através de um selecionador ou de um grupo de selecionadores, podemos redefinir parâmetros de formatação considerados como padrão de apresentação dos elementos de uma página em HTML.

Na prática, um selecionador corresponde a um elemento HTML, que pode ser um parágrafo `<p>`, uma lista desordenada ``, uma lista ordenada ``, uma tabela `<table>`, ou ainda, partes destes elementos como os itens de uma lista ``, as linhas de uma tabela `<tr>`, as colunas de uma tabela `<td>` etc. Logo, as formatações definidas para um grupo de selecionadores, modificarão as características de apresentação consideradas como padrão para um grupo predeterminado de elementos.

Ao se definir um estilo de formatação para um grupo de selecionadores, os elementos envolvidos neste grupo deverão estar separados por vírgulas e sem os sinais delimitadores de tags `<` (menor que) e `>` (maior que), além de não ser preciso utilizar uma chamada para este tipo de regra, pois já constam explícitos na referência da regra, os elementos ao qual serão aplicadas as formatações definidas.

A sintaxe utilizada para a definição de uma regra, independentemente do uso de qualquer uma das possibilidades descritas, estará enquadrada dentro de um mesmo formato que consiste em duas partes:

Referência: Nome, elemento ou grupo de elementos, ao qual será atribuído à regra.

Bloco de propriedades: Delimitado pelos sinais "{" (abre chave) e "}" (fecha chave), contém a declaração das propriedades e dos seus respectivos valores, compondo o estilo de formatação.

Exemplo genérico da definição de uma regra de formatação:

```
REFERÊNCIA_DA_REGRA
{
PROPRIEDADE(1): VALOR(1);
PROPRIEDADE(2): VALOR(2);
PROPRIEDADE(X): VALOR(X);
}
```

Pelo fato dos ids estarem mais relacionados com a geração de efeitos dinâmicos em páginas HTML, vamos concentrar os nossos estudos na definição das classes, das pseudo-classes, dos selecionadores e dos grupos de selecionadores.

Definindo regras a partir de classes

Observe abaixo, dois exemplos da definição de classes:

```
.texto_centro
```

```
{
font-family: Comic Sans MS, Verdana, Helvetica;
font-size: 14pt;
color: Red;
font-weight: bold;
text-align: center;
}
```

```
texto_normal
```

```
{
font-family: Arial, Sans-Serif;
font-size: 12px;
color: #000099;
text-align: justify;
}
```

Apesar da atribuição do tamanho e da cor da fonte ter sido feita de forma diferente em cada uma das classes, ambos os exemplos procedem de acordo com as regras de definição dos estilos de formatação.

Observe abaixo, a aplicação das classes definidas, no trecho de página que segue:

```
<body>
<br> <br>
<p class="texto_centro"> T&acut;TULO DA P&Aacute;GINA </p>
<br> <br>
<p class="texto_normal"> Este &eacute; o 1&ordm; par&aacute;grafo utilizando uma
formata&ccedil;&atilde;o de texto normal. </p>
<p class="texto_normal"> Este &eacute; o 2&ordm; par&aacute;grafo utilizando uma
formata&ccedil;&atilde;o de texto normal. </p>
</body>
```

No exemplo anterior, inserimos inicialmente um parágrafo com o texto centralizado, usando a fonte Comic Sans MS, tamanho 14, cor vermelha e em negrito.

Perceba que todas as formatações aplicadas no exemplo acima, foram definidas anteriormente nas classes, com isso temos a flexibilidade de se trabalhar o layout de formatação independente do conteúdo que será exposto.

Definindo regras a partir de pseudoclasses

A definição da referência de uma pseudoclasse, contém primeiro a tag <a> (anchor) sem os sinais delimitadores "<>", seguida do sinal ":" e por último, a opção desejada para a formatação dos links da página. Observe abaixo, as opções mais utilizadas de pseudoclasses:

- A:link – Define a formatação para os links carregados e ainda não visitados
- A:visited – Define a formatação para os links já visitados
- A:active – Define a formatação para o link ativo
- A:hover – Define a formatação para o link imediatamente posicionado sob o cursor do mouse.

Suponhamos que você idealizou as seguintes regras de formatação para a apresentação dos links do seu site:

- Os links carregados e não visitados deverão ser exibidos com a cor azul, em negrito e sem o sublinhado.
- Os links já visitados deverão ser exibidos com a cor vermelha, em itálico e sem o sublinhado.
- O link sob o cursor do mouse deverá ser exibido com a cor amarela sobre o fundo verde, em negrito e sublinhado.
- O link ativo deverá ser exibido com a cor branca sobre o fundo vermelho, em negrito e sublinhado.

Observe a codificação das regras sugeridas no exemplo anterior:

```
a:link {
color: blue;
font-weight: bold;
text-decoration: none; }
a:visited {
color: red;
font-style: italic;
text-decoration: none; }
a:hover{
color: #ffff00;
background-color: #00ff00;
font-weight: bold;
text-decoration: underline; }
a:active {
color: #ffffff;
background-color: #ff0000;
font-weight: bold;
text-decoration: underline; }

a:active
{
color: #ffffff;
background-color: #ff0000;
font-weight: bold;
text-decoration: underline;
}
```

Diferentemente do uso das classes, uma regra definida através de uma pseudo-classe não necessita de um atributo para a sua chamada, pois é automaticamente incorporada aos links por ser de uso específico para a formatação destes elementos.

O único cuidado que se deve ter é com relação a hierarquia dos elementos, pois caso um link possua alinhado internamente a sua própria formatação de fonte, esta será priorizada, sendo esta regra válida para todos os demais elementos HTML.

Definindo regras a partir de um selecionador ou de um grupo de selecionadores

Ao utilizar um selecionador ou um grupo de selecionadores para a definição de uma regra de formatação, devemos informar previamente na sua referência, qual o elemento ou elementos que serão contemplados por esta regra e por este motivo, não se faz necessário uma chamada para a sua aplicação no corpo do documento.

Quando a regra for definida para um grupo de selecionadores, devemos separar os elementos envolvidos por uma vírgula e omitir os sinais delimitadores de tags "<>". Observe as duas regras abaixo, onde a primeira foi definida a partir de um selecionador e a segunda, a partir de um grupo de selecionadores.

Definição de uma regra para o elemento <h1>:

H1

{

font-family: Comic Sans MS, Verdana, Helvetica, Sans-Serif;

font-size: 16pt;

color: #ff0000;

font-weight: bold;

text-align: center;

text-decoration: none;

}

Quando no corpo de um documento HTML for inserida a tag <h1> para a inserção de um cabeçalho de nível 1, com a aplicação da regra acima, as formatações definidas serão assumidas automaticamente pelo texto delimitado por esta tag.

Definição de uma regra para os elementos <tr>, <td>, e <p>:

Tr, td, li, p

```
{  
font-family: Arial;  
font-size: 12pt;  
color: #000099;  
text-align: justify;  
text-decoration: none;  
vertical-align: middle;  
}
```

Quando no corpo do documento for inserido as tags `<tr>`, `<td>`, `` e `<p>`, com a aplicação da regra acima, as características de alinhamento e as definições de formatação da fonte serão automaticamente assumidas pelo texto delimitado por qualquer uma dessas tags.

Disponibilizando estilos de formatação para todas as páginas de um site

Para que os estilos de formatação criados possam abranger não somente a um elemento ou a uma única página, mas sim ao site como um todo, devemos concentrar as regras de formatação em um arquivo, fazendo referência ao mesmo por todas as páginas que compõem o site. Este arquivo é conhecido como arquivo de folha de estilos.

O arquivo de folha de estilos poderá ser criado em qualquer editor de texto puro (ASCII), como, por exemplo, no Bloco de Notas do Windows. Para melhor identificá-lo, é aconselhável que este arquivo seja salvo em uma pasta exclusiva, criada na raiz do sistema do site, nomeada como `css` (Cascading Style Sheets) e que o arquivo também contenha a sigla `css` como sua extensão.

A referência ao arquivo de estilos de formatação deverá ser feita dentro do bloco de cabeçalho da página, utilizando para tanto a tag ímpar `<link>`, complementada pelos seguintes atributos descritos abaixo.

(rel): Através deste atributo definimos que a referência será relacionada a um arquivo de folha de estilos de formatação, para tanto, devemos passar ao atributo `(rel)` o valor `"stylesheet"`. Exemplo:

```
Rel="stylesheet"
```

(href): Com este atributo fazemos a referência ao arquivo de folha de estilos seguindo os mesmos conceitos utilizados na referência dos hiperlinks. Exemplo:

```
Href="css/arquivo_estilos.css"
```

(type): A este atributo devemos passar o valor "text/css", indicando ao browser o tipo do conteúdo do arquivo referenciado. Exemplo:

```
Type="text/css"
```

Observe no trecho de código abaixo, um exemplo de como referenciar um arquivo de folha de estilos:

```
<html>
<head>
<title> TÍTULO DA PÁGINA </title>
<link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
(Implementação do corpo da página)
</body>
</html>
```

Exemplo de um arquivo de folha de Estilos

Observe atentamente a sintaxe das propriedades e dos seus respectivos valores, utilizados na definição das regras que compõem o arquivo de folha de estilos de formatação.

```
texto_centro
```

```
{
font-family: Comic Sans MS, Verdana, Helvetica;
font-size: 14pt;
color: Red;
font-weight: bold;
text-align: center;
}
```

```
texto_normal
```

```
{
font-family: Arial, Sans-Serif;
font-size: 12px;
color: #000099;
text-align: justify;
}
```

a:link

```
{  
  
color: blue;  
font-weight: bold;  
text-decoration: none;  
  
}
```

a:visited

```
{  
  
color: red;  
font-style: italic;  
text-decoration: none;  
  
}
```

a:hover

```
{  
  
color: #ffff00;  
background-color: #00ff00;  
font-weight: bold;  
text-decoration: underline;  
  
}
```

a:active

```
{  
  
color: #ffffff;  
background-color: #ff0000;  
font-weight: bold;  
text-decoration: underline;  
  
}
```

h1

{

```
font-family: Comic Sans MS, Verdana, Helvetica, Sans-Serif;  
font-size: 16pt;  
color: #ff0000;  
font-weight: bold;  
text-align: center;  
text-decoration: none;
```

}

Tr, td, li, p

{

```
font-family: Arial;  
font-size: 12pt;  
color: #000099;  
text-align: justify;  
text-decoration: none;  
vertical-align: middle;
```

}

Layout com Frames

Neste módulo, vamos estudar a implementação de um novo layout, utilizando uma estrutura de frames, que consiste na divisão da janela do navegador em quadros.

Cada quadro criado possui a autonomia de exibir uma página em HTML independente, possibilitando a visualização simultânea de páginas, diferenciando-se do layout utilizado anteriormente, por permitir apenas a visualização de uma única página no navegador.

Utilizando um layout com frames, podemos dividir a janela do navegador em quadros postados horizontalmente (em linhas) ou verticalmente (em colunas).

Um frame, pode ainda suportar a implementação de outros frames, caso haja a necessidade de subdividi-lo em áreas menores, mas na prática, esta situação não é tão comum.

A maioria dos sites que fazem o uso dos frames tem por objetivo facilitar o processo de navegação, bastando, por exemplo, que seja disponibilizado um frame de tamanho reduzido do lado esquerdo da janela do navegador, com uma página de menu contendo links para as seções deste site e do lado direito, um outro que ocupe o restante do espaço da janela, responsável em exibir uma determinada página quando o visitante clicar sobre o seu link , inserido no menu do frame ao lado.

Deste modo, o menu sempre estará disponível para que rapidamente possamos acessar uma página pertencente a uma outra seção do site.

Para entendermos melhor o funcionamento de um site com um menu construído sobre uma estrutura de frames, vamos fazer uma analogia, comparando-o com um aparelho de televisão:

Primeiramente, imagine toda a extensão da parte da frente do aparelho de televisão.

Agora, do lado direito, está a tela plana, com vinte polegadas e inicialmente exibindo a programação do canal (1).

Do lado esquerdo, com 5 polegadas (20% da dimensão horizontal do aparelho), está o menu com os botões para a sintonia de todos os canais suportados por este aparelho de televisão.

Observe que mesmo ao mudar de canal, sendo exibida outra programação na tela da TV, o menu para a troca de canais, estará sempre ali disponível do lado esquerdo.

Para construir um site com esta característica de navegação, são necessários basicamente três arquivos:

Arquivo com as definições da estrutura e divisão dos frames.

Arquivo contendo o menu de navegação.

Arquivo com a página inicial a ser exibida.

A divisão da janela do navegador em frames é apenas uma opção a mais de layout que você poderá usar caso julgue conveniente.

Vejamos agora, as particularidades de cada um dos arquivos listados acima, para que possamos aprender a implementar este novo layout.

Arquivo de definição dos Frames

O arquivo com a definição dos frames, deve ser carregado primeiramente, para tanto, como vimos no início do curso, o mesmo tem que ser salvo na pasta principal da sua aplicação, com o nome de (index.html) ou com o nome indicado pelo servidor para esta finalidade, pois neste arquivo, estarão definidas informações importantes para a visualização dos frames, como por exemplo:

- As páginas associadas a cada frame;
- A quantidade de frames e o espaço que ocupará cada um deles;
- A distância entre os frames;
- A utilização ou não de bordas etc.

O arquivo de definição da estrutura de um layout com o uso de frames, diferencia-se da criação de uma página comum, apenas por incorporar o bloco que contém o corpo do documento dentro das suas tags.

Após a abertura do bloco principal e da declaração do cabeçalho da página, inserimos as tags referentes à definição da estrutura dos frames.

A tag em par `<frameset>` e `</frameset>`, possui atributos que definem características gerais da estrutura de frames e encadeia o uso da tag `<frame>`, que define as particularidades de cada frame criado.

Uma pequena porcentagem entre os navegadores disponíveis, não suportam a exibição dos frames.

Para contornar esta situação, existe a tag em par `<noframes>` e `</noframes>`, que permite carregarmos o bloco com o corpo de uma página comum em HTML, com links para as demais páginas do site que não sofrerão nenhuma modificação em decorrência da utilização dos frames ou podemos ainda, inserir um texto informando ao visitante que o seu navegador não suporta o uso deste recurso e disponibilizarmos para ele, um link para fazer o acesso a uma versão do site sem a utilização dos frames.

Vejam agora, um exemplo genérico da criação da estrutura de um layout com dois frames, sem a declaração dos seus respectivos atributos:

<html> - Abertura do bloco principal da página.

<head> - Abertura do cabeçalho.

<title> Define o título da janela do navegador, ignorando os demais títulos inseridos nas páginas exibidas pelos frames **</title>**

</head> - Fechamento do cabeçalho.

<frameset> - Abertura da tag para a definição da estrutura dos frames.

<frame> - Definição das características para o primeiro frame.

<frame> - Definição das características para o segundo frame.

<noframes> - Abertura da tag para o acesso dos navegadores que não suportam o uso dos frames.

<body> - Abertura do bloco com o corpo da página.
(Implementação do conteúdo da página ou exibição de uma mensagem informativa)

</body> - Fechamento do bloco com o corpo da página.

</noframes> - Fechamento da tag para o acesso dos navegadores que não suportam o uso dos frames.

</frameset> - Fechamento da tag para a definição da estrutura dos frames.

</html> - Fechamento do bloco principal da página.

Atributos da tag frameset

Devemos declarar os atributos da tag frameset na parte referente à sua abertura. Vejamos agora, os seus principais atributos:

(cols): Com este atributo definimos a quantidade e o espaço ocupado pelos frames que estarão posicionados verticalmente (em colunas), ou seja, um ao lado do outro.

(rows): Com este atributo definimos a quantidade e o espaço ocupado pelos frames que estarão posicionados horizontalmente (em linhas), ou seja, um acima do outro.

Tanto no atributo (cols) como no atributo (rows), definimos o espaço ocupado para todos os frames, separando-os por uma vírgula de acordo com as seguintes possibilidades:

- Valor percentual: Esta é a forma mais utilizada. Através de um valor, definimos a porcentagem da janela do navegador que será ocupada pelo frame.
- Valor absoluto: Define em pixels o espaço ocupado pelo frame em relação à janela do navegador. Não é recomendado utilizar apenas valores absolutos na definição do tamanho dos frames, pois existe diferentes resoluções de vídeo.
- Valor relativo: Usando a máscara "*", informamos que o frame ocupará o espaço restante, disponível na janela do navegador.

Os valores citados acima, podem ser utilizados em conjunto conforme a situação.

Exemplos:

<frameset cols="20%,*"> - Define o frame da esquerda com vinte por cento do tamanho da janela horizontalmente e o frame da direita ocupará o restante do espaço disponível.

<frameset rows="20%*,100"> - Neste exemplo definimos 3 frames. O primeiro, posicionado na parte superior, ocupará 20 por cento do tamanho total da janela do navegador. O segundo, ocupará o espaço excedente pela ocupação do primeiro e do terceiro frame. O terceiro, posicionado na parte inferior da janela do navegador, ocupará o espaço de 100 pixels.

(framespacing): Este atributo define em pixels, a distância entre os frames.

(frameborder): Reconhecido pelo navegador Internet Explorer, este atributo define em pixels, a espessura da borda dos frames. Para que os frames não apresentem uma borda, basta passar como referência o valor "0".

(border): Reconhecido pelo navegador Netscape, este atributo define em pixels, a espessura da borda dos frames. Para que os frames não apresentem uma borda, basta passar como referência o valor "0".

Ao criarmos um layout com um menu de navegação, é interessante eliminarmos as bordas e o espaço de separação entre os frames. Para que cause o efeito desejado tanto no Internet Explorer como no Netscape Navigator, além do atributo (framespacing) devemos declarar os atributos (frameborder) e (border), como no exemplo abaixo:

```
<frameset cols="20%,*" framespacing="0" frameborder="0" border="0">
```

Atributos da tag frame

A tag frame não possui uma parte aplicada ao seu fechamento e deve ser inserida conforme o número de frames declarados no atributo (cols) ou (rows) da tag frameset. Seus principais atributos são:

(name): Com este atributo definimos um nome interno para o frame. Este nome servirá para indicarmos qual frame será o alvo do carregamento de uma determinada página.

(src): Com este atributo definimos a página que será carregada inicialmente pelo frame. A sua declaração é muito semelhante a do atributo (href) utilizado na criação dos links, podendo referenciar uma página interna ou externa ao sistema.

(scrolling): O uso deste atributo nos permite definir se um frame apresentará ou não as barras de rolagem. Normalmente quando o texto inserido ultrapassa o tamanho da janela, as barras de rolagem são exibidas automaticamente.

Caso o atributo (frameborder) seja declarado dentro da tag frame, a especificação da borda será aplicada exclusivamente a este frame.

(n): Com este atributo, definimos em pixels uma margem interna para o frame na horizontal. O espaço será reservado tanto no lado esquerdo como no lado direito do frame.

(marginheight): Com este atributo, definimos em pixels uma margem interna para o frame na vertical. O espaço será reservado tanto na parte superior como na parte inferior do frame.

(noresize): Este atributo não recebe nenhum valor como referência. Quando utilizado, o tamanho do frame será fixo, não podendo ser redimensionado pelo visitante.

Arquivo do menu opções

No frame de menor tamanho, este arquivo conterá um menu com links que estarão presentes na janela do browser durante toda a navegação, facilitando o acesso às diversas páginas do site.

É aconselhável reservar 20 por cento do espaço disponível da janela do navegador, ao frame que exibirá o arquivo com o menu de opções, definindo-o primeiramente na estrutura dos frames, para que o mesmo possa ser carregado no lado esquerdo da tela (caso tenhamos utilizado o atributo cols) ou na parte superior (caso tenhamos utilizado o atributo rows), ambos da tag frameset.

A única diferença existente na implementação do arquivo com o menu de navegação, é a inserção do atributo (target) na tag de criação dos links, indicando qual será o frame alvo que carregará a página referenciada pelos links do menu.

O atributo (target), deve receber como valor, o nome exato dado ao frame alvo, definido na implementação da estrutura dos frames, sendo inclusive respeitado a utilização de letras maiúsculas e ou minúsculas.

Existem alguns outros valores específicos que podem ser aplicados ao atributo (target). Para que causem o efeito desejado, estes devem ser digitados com letras minúsculas e precedidos pelo sinal _ (sublinhado).

Vejamos algumas opções:

Target="_blank" – Carrega a página em uma nova janela do navegador.

Target="_self" – Carrega a página no próprio frame.

Target="_top" – Carrega a página por toda a extensão da janela do navegador, ignorando a estrutura dos frames.

Exemplo:

Suponhamos que no frame que contém o arquivo com o menu de navegação, tenhamos um link para exibir no frame "principal" a página de links interessantes. Então a sua sintaxe será a seguinte:

```
<a href="links.html" target="principal"> Links interessantes </a>
```

Quando o visitante clicar sobre este link, a página links.html será aberta no frame de maior tamanho, chamado "principal".

Devemos discretamente informar no menu de navegação que "Este site contém frames", para que os visitantes que possuem limitação visual possam tomar conhecimento da utilização deste recurso e através dos atalhos de teclado (ctrl + tab) ou (f6) alternarem o foco do cursor entre os frames.

Frames in-line

O frame in-line é um interessante recurso que nos permite inserir em qualquer página, um quadro exibindo parte ou um documento HTML completo. Deste modo, havendo um trecho de código HTML comum a várias páginas de um site, podemos disponibilizá-lo em um arquivo, possibilitando a sua visualização em todas as páginas que fizerem referência ao mesmo através de um frame in-line.

Note que: Caso seja necessário alterar este trecho comum, basta modificar apenas o arquivo compartilhado e todas as páginas estarão atualizadas automaticamente.

Para a implementação de um frame in-line, utilizamos a tag em par <iframe>, onde na sua abertura, declaramos alguns atributos também disponíveis na tag <frame>, como name, frameborder, scrolling, noresize etc.

Sintaxe básica da tag <iframe>: <iframe
src="arquivo_compartilhado.htm"> </iframe>

Outros atributos possíveis para a tag <iframe> são:

- **(align)**: Define o alinhamento horizontal para o quadro, podendo assumir os seguintes valores: Align="left" - Alinha à esquerda.

- `Align="center"` - Alinha ao centro.
- `Align="right"` - Alinha à direita.
- `(height)`: Define a altura do quadro em pixels ou em porcentagem da tela.
- `(width)`: Define a largura do quadro em pixels ou em porcentagem da tela.

Observe um exemplo genérico da implementação de um frame in-line no trecho de código abaixo:

```
<body><iframe name="cabecalho" src="include/cabecalho.htm" frameborder="0" border="0"
height="10%" width="100%" noresize></iframe> (corpo do documento) <iframe name="rodape"
src="include/rodape.htm" frameborder="0" border="0" height="10%" width="100%"
noresize></iframe></body>
```

Aplicando o exemplo acima em todas as páginas de um site, estaremos facilitando futuras alterações tanto para o cabeçalho, como para o rodapé das páginas.

Exemplos de página contendo a estrutura de um layout com frames

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> LAYOUT COM FRAMES </title>
<link rel="stylesheet" href="css/estilo.css" type="text/css">
</head>
<frameset cols="20%,*" framespacing="0" frameborder="0" border="0">
<frame name="menu" src="menu.html" scrolling="no" marginwidth="3" marginheight="3"
noresize>
<frame name="principal" src="apresentacao.html">
<noframes>
<body>
<br>
<marquee behavior="alternate"><blink><font class="titulo">BEM-
VINDO</font></blink></marquee>
<br>
<center></center>
```

```
<br> <br>
<font class="texto">
<p>Espero que possa encontrar neste espaço, informações que sejam do seu agrado, justificando o
seu retorno constante a este site que foi inteiramente desenvolvido pensando em você. </p>
<p>Participe enviando seus comentários, críticas e/ou sugestões, para que este site se torne cada
vez melhor segundo as suas preferências. </p>
</font>
<br>
<center>
<table border="0" cellspacing="0">
<tr>
<td> <font class="titulo"> MENU DE NAVEGAÇÃO </font> </td>
</tr>
<tr>
<td> </td>
</tr>
<tr>
<td>
<a href="autor.html"> <font class="menu"> Sobre o autor </font> </a>
</td>
</tr>
<tr>
<td>
<a href="historico.html"> <font class="menu"> Histórico do site </font> </a>
</td>
</tr>
<tr>
<td>
</td>
</tr>
<tr>
<td>
<a href="download.html"> <font class="menu"> Download de arquivos </font> </a>
</td>
</tr>
<tr>
<td>
<a href="formulario.html"> <font class="menu"> Envie-nos seus comentários </font> </a>
</td>
</tr>
<tr>
<td>
<div align="center"> <a
href="mailto:seuemail@seudominio.sigla.pais?subject=CONTATO">  </a> </div>
```

```
</td>  
</tr>  
</table>  
</center>  
</body>  
</noframes>  
</frameset>  
</html>
```

Exemplo de página contendo um menu de navegação para um layout com frames

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>  
<head>  
<title> MENU DE NAVEGAÇÃO </title>  
<link rel="stylesheet" href="css/estilo.css" type="text/css">  
</head>  
<body>  
<br>  
<center>  
<table border="0" cellspacing="0">  
<tr>  
<td> <font class="titulo"> MENU DE NAVEGAÇÃO </font> </td>  
</tr>  
<tr>  
<td> </td>  
</tr>  
<tr>  
<td>  
<a href="autor.html" target="principal"> <font class="menu"> Sobre o autor </font> </a>  
</td>  
</tr>  
<tr>  
<td>  
<a href="historico.html" target="principal"> <font class="menu"> Histórico do site </font> </a>  
</td>  
</tr>  
<tr>  
<td>  
<a href="links.html" target="principal"> <font class="menu"> Links interessantes </font> </a>  
</td>  
</tr>
```

```
<tr>
<td>
<a href="download.html" target="principal"> <font class="menu">Download de
arquivos</font> </a>
</td>
</tr>
<tr>
<td>
<a href="formulario.html" target="principal"> <font class="menu">Envie-nos seus
comentários</font> </a>
</td>
</tr>
<tr>
<td>
<a href="apresentacao.html" target="principal"> <font class="menu">Página
principal</font> </a>
</td>
</tr>
<tr>
<td>
<div align="center"> <a
href="mailto:seuemail@seudominio.sigla.pais?subject=CONTATO">  </a> </div>
</td>
</tr>
</table>
</center>
</body>
</html>
```

Exemplo de uma página de apresentação para um layout com frames

No exemplo abaixo, observe com atenção a distribuição dos comandos HTML (tags) na construção do documento.

```
<html>
<head>
<title> APRESENTAÇÃO </title>
<link rel="stylesheet" href="css/estilo.css" type="text/css">
</head>
<body>
<br>
<marquee behavior="alternate"> <blink> <font class="titulo">BEM-
VINDO</font> </blink> </marquee>
<br>
<center>  </center>
<br> <br>
<font class="texto">
```

```
<p>Espero que possa encontrar neste espaço, informações que sejam do seu agrado, justificando o seu retorno constante a este site que foi inteiramente desenvolvido pensando em você. </p>  
<p>Participe enviando seus comentários, críticas e/ou sugestões, para que este site se torne cada vez melhor segundo as suas preferências. </p>  
</font>  
</body>  
</html>
```

Arquivo de folha de estilos usado nos exemplos do layout com frames

Observe as definições de formatação inseridas no arquivo de folha de estilos, aplicadas às páginas que compõem a estrutura do layout com frames.

.menu

```
{  
font-family:arial;  
font-size:14pt;  
font-style:normal;  
font-weight:bold;  
text-align:justify;  
text-decoration:none;  
}
```

titulo

```
{  
font-family:comic sans ms, verdana, helvetica;  
font-size:16pt;  
color:#8b0000;  
font-style:normal;  
font-weight:bold;  
text-align:center;  
text-decoration:none;  
}
```

texto

```
{  
font-family:arial;  
font-size:12pt;  
color:#000000;  
font-style:normal;  
text-align:justify;  
text-decoration:none;  
}
```

E agora, o que devo fazer após ter desenvolvido a estrutura básica do meu site?

Como pôde perceber durante a realização deste curso, toda a estrutura básica de um site pode ser perfeitamente desenvolvida e testada localmente no seu computador, sem a necessidade de estar conectado a Internet.

Mas agora que você já concluiu o desenvolvimento da estrutura do seu site, não faz mais nenhum sentido mantê-lo isolado como uma ilha. É chegada a hora de compartilhá-lo com o restante do mundo.

O primeiro passo a ser dado é definir um servidor para a hospedagem do seu site e para isso, existem inúmeras opções entre serviços pagos e gratuitos disponíveis na Internet (utilize um buscador para pesquisar e identificar um servidor de hospedagem que melhor atenda às suas necessidades).

Podemos entre outras possibilidades, tomar as seguintes variáveis como critério para definir a escolha de um bom servidor de hospedagem:

Espaço livre oferecido para a hospedagem do seu site

- Se haverá ou não a inclusão obrigatória de propagandas no seu site mantidas pelo servidor de hospedagem.
- A preocupação com a segurança e a integridade das informações do seu site.
- Se além da hospedagem do site, também serão oferecidos serviços como contador de acessos, livro de visitas, CGI público para o processamento de formulários eletrônicos, contas de e-mail etc. É válido ressaltar que estes serviços também podem ser conseguidos individualmente na Internet.
- Boas condições de infra-estrutura e suporte técnico.
- Se o servidor oferece a facilidade de atualização de arquivos e pastas via FTP.

FTP significa File Transfer Protocol (protocolo de transferência de arquivos). Um protocolo é um conjunto de normas que permite a comunicação entre computadores.

O FTP é utilizado para possibilitar a transferência de arquivos do seu computador para o servidor de hospedagem. Para realizar este acesso, você precisará de permissões especiais como um nome de usuário e uma senha.

O endereço Internet (URL) de um servidor FTP tem uma aparência ligeiramente diferente do URL que você usa em uma página da Web comum, por esta utilizar o protocolo HTTP.

Acessando os sites FTP diretamente, pode trabalhar com arquivos e pastas nos servidores FTP da mesma forma que trabalha com eles no seu próprio computador.

Observe as informações necessárias para acessar diretamente sites FTP através do Microsoft Internet Explorer:

Você pode visualizar, fazer download, carregar e excluir arquivos e pastas. Se precisar de permissão do servidor FTP para realizar qualquer uma dessas ações, você será instruído a fornecer o seu nome de usuário e senha.

- Na barra de endereços, digite o endereço Internet (URL) do site FTP ao qual você deseja se conectar. Por exemplo: ftp://ftp.domínio.sigla.país
- Para fazer o download de um arquivo ou pasta, clique com o botão direito do mouse em um item da página e depois clique em **Fazer o download para**.
- Para efetuar logon como um usuário diferente nesse site FTP, clique no menu **Arquivo** e depois clique em **Efetuar logon como**.
- Para renomear ou excluir itens de uma pasta FTP ou colar itens em um site FTP, você pode utilizar os mesmos comandos e ações usados no Windows Explorer.

Existem vários programas clientes de FTP disponíveis no mercado, pesquise e escolha o que melhor lhe atenda.

Dicas e considerações finais

Após publicar o seu site na Internet, basta apenas divulgá-lo para que o mesmo se torne conhecido pelo público de interesse, recebendo a cada dia, um número crescente de visitantes.

Para realizar a divulgação do seu site, na grande maioria das vezes você poderá contar com o apoio do seu próprio servidor de hospedagem, com os serviços oferecidos pelos grandes portais Web, com o cadastro em sites populares, com as ferramentas de busca na Internet etc.

Você poderá ainda, inserir na página principal (index.htm) metatags que recebem como parâmetros expressões curtas ou palavras chaves associadas ao conteúdo abordado pelo seu site, onde estes valores são aproveitados por alguns motores de busca na Internet.

As metatags são tags especiais normalmente relacionadas a definições de configuração, por isso são declaradas dentro do bloco de cabeçalho para que o browser possa priorizá-las durante o processo de interpretação da página. A sua sintaxe é bastante simples, onde no atributo (name) indicamos uma variável e no atributo (content) associamos o seu valor correspondente.

Observe um exemplo genérico da declaração de metatags relacionadas com a exposição de valores para a identificação do site na Internet:

```
<head> <meta name="author" content="Nome do autor do site">  
<meta name="description" content="Descrição do site">  
<meta name="keywords" content="conjunto,de palavras,  
chaves,que identificam, o site,recomendamos,  
menos de,16 expressões, todas separadas,com vírgulas">  
<title>Título principal do site</title> </head>
```

Lembre-se sempre de que a melhor campanha de divulgação é aquela feita "boca-a-boca" ou melhor, "e-mail-a-e-mail". Portanto mantenha sempre o seu site atualizado com muitas novidades e inovações.

Você já é capaz de entender, analisar e adaptar os códigos gerados pelas ferramentas de edição de documentos HTML, portanto a partir deste momento o uso de aplicativos como o Dreamweaver, Front Page, ou até mesmo a opção de salvar um texto como no formato HTML existente nos aplicativos do pacote Office, poderão contribuir dando mais agilidade na composição de páginas com muitas informações escritas.

A tecnologia aplicada à Web está em constante aprimoramento com o surgimento de recursos inovadores e por este motivo, você não pode ficar do lado de fora desta evolução.

Dê continuidade ao seu processo de aprendizado, utilizando a própria Internet como um meio de inteirar-se cada vez mais sobre este fantástico universo de possibilidades, limitado apenas ao alcance da sua criatividade.