

Unidade 4: Introdução à Tecnologia CSS

Prof. Daniel Caetano

Objetivo: Apresentar conceitos de introduzir o uso de Folhas de Estilo em Cascata.

Bibliografia: W3, 2009; CASCADE, 2006; RAMALHO, 1999; NIELSEN, 2000.

INTRODUÇÃO

Conceitos Chave:

- XHTML => descrever estrutura
 - * Página feia!
- Páginas modernas: XHTML + CSS
- CSS => descrever apresentação visual

Até a presente aula, vimos várias tags do XHTML que servem para descrever a função estrutural de cada trecho do texto dentro da página XHTML: o que é um título de seção, o que é um subtítulo, o que é parágrafo... e assim por diante.

Muitos alunos devem ter se perguntado por que um site feito desta forma é tão feio, e como é que outros sites na Web são tão coloridos e variados. A resposta para isso é que uma página Web atual não é feita apenas de XHTML.

Além do XHTML, que descreve os elementos existentes em uma página, é necessário também um arquivo de folhas de estilo, mais conhecido como arquivo CSS (Cascade Style Sheets ou, em bom português, Folhas de Estilo em Cascata).

Nesta e nas próximas aulas estaremos estudando como construir um arquivo CSS e como usá-lo para dar à nossa página a aparência que desejamos.

1. FOLHAS DE ESTILO EM CASCATA (CSS)

Conceitos Chave:

- Documento de Estilos
 - * Define a apresentação de cada estrutura do XHTML
 - * Não dá para fazer no XHTML?
 - + Praticidade e Eficiência
 - + Facilidade de modificação de layout
 - + Melhor uso do cache

Uma folha de estilo em cascata é nada mais do que um documento onde são definidas as características de apresentação de cada tipo de estrutura do XHTML.

Alguns alunos podem se perguntar: "Mas por que não fazer isso dentro do próprio XHTML?". Bem, originalmente era assim que as coisas eram feitas. Aliás, muita gente programa ainda daquela forma, embora a definição de apresentação dentro do XHTML seja muito pouco recomendável hoje em dia. É preciso tomar cuidado, já que a grande maioria dos livros e sites ainda ensina HTML antigo, induzindo os novos programadores aos erros dos velhos programadores.

Mas, se antigamente se usava a codificação visual dentro do próprio XHTML, por que hoje isso não é mais recomendado? Bem, as razões para isso chamam-se praticidade e eficiência.

Quando separamos os detalhes de apresentação em um arquivo separado, podemos usar o mesmo arquivo de definição de apresentação para **todas** as páginas. Em outras palavras, apesar do trabalho inicial para criar o arquivo de estilos em separado, ele **economiza** digitação nas páginas seguintes, acelerando o desenvolvimento.

Além disso, com um (ou poucos) arquivo de estilos, é muito mais fácil alterar a aparência de todo um site, de forma consistente e **sem ter que editar todas as páginas do site!**

Adicionalmente, os arquivos de CSS ficam no cache dos navegadores. Como ele é o mesmo para todas as páginas, acaba sendo uma economia de banda de transferência usá-lo, além de fazer com que o site fique mais rápido para o usuário.

1.1. Que Recursos as CSSs Possuem?

Conceitos Chave:

- Modificar a propriedade visual de qualquer elemento do XHTML
- Exemplos:
 - * Fonte de texto
 - * Alinhamento de texto
 - * Posição de imagens
 - * Cores de elementos
 - * Cores de botões
 - * Posições de tabelas
 - * ...

Basicamente, as folhas de estilo CSS são capazes de modificar qualquer propriedade visual de qualquer coisa que apareça em uma página XHTML.

As folhas de estilo permitem modificar as fontes de texto, os alinhamentos de texto, a posição das imagens, cores dos elementos da página, cores de botões, posição de tabelas... enfim, uma infinidade de recursos.

Nesta aula serão vistos alguns destes recursos, e outros seguirão nas próximas aulas.

1.2. Porque o nome "Em Cascata"

Conceitos Chave:

- Cascata?
 - * Navegador => Arquivo => No cabeçalho => Na tag
 - * Classes e subclasses (opcional!)
 - + <H1>
 - + <H1.editorial> => herda propriedades de <H1>

As razões mais práticas para considerar o termo "Cascata" são:

- 1) Podemos ter quatro níveis de definição de estilo:
 - a) O definido pelo navegador
 - b) Em um arquivo separado
 - c) No próprio arquivo XHTML, no topo da página
 - d) No próprio arquivo XHTML, dentro da tag

2) Se definirmos o estilo de uma tag como <H1> e depois definirmos um estilo derivado de <H1>, como por exemplo <H1.editorial>, este estilo derivado "herdará" todas as mudanças de estilo que foram realizadas na tag <H1>.

O primeiro conceito pode ser explicado assim: normalmente o navegador tem estilos definidos (que podem ser configurados pelo usuário). O programador da página define estilos globais em arquivos separados, com a extensão .CSS. Isso permite o uso de todos os benefícios citados anteriormente para os arquivos CSS. Se quisermos que apenas em alguma página um dos estilos seja modificado, podemos redefinir este estilo no topo da página, sendo que a mudança terá efeito apenas nesta página. Se quisermos, ainda, modificar um estilo apenas em uma tag, podemos redefinir o estilo dentro dela, e terá efeito apenas na tag. Em geral, usaremos o arquivo de estilo separado, por ser a maneira mais "correta" e limpa de usar folhas de estilo.

Com relação ao segundo conceito, sua assimilação é mais fácil com o uso. Na prática é como dizer que se mudamos a fonte do H1 para Arial (ao invés de Times), automaticamente o estilo <H1.editorial> passará a ser também em Arial.

1.3. Passos para a Criação de um Arquivo de Estilo (CSS)

Conceitos Chave:

- 1) Criar arquivo com definições (.CSS)
- 2) Indicar este arquivo no cabeçalho do XHTML
- 3) Editar o estilo conforme desejado

São três os passos básicos para criar um arquivo de estilo:

- 1) Criar arquivo para inserir as definições de estilo (normalmente com extensão .css)
- 2) Indicar este arquivo na página XHTML, obviamente na seção <head>...</head>
- 3) Editar o arquivo de estilo e a página até que tudo fique como desejado.

2. USANDO ARQUIVOS CSS

Conceitos Chave:

- Primeiro passo: criar arquivo vazio (ex.: *estilo.css*)
- Indicar no XHTML

```
* <link href="estilo.css" rel="stylesheet" type="text/css" />
```

Quando vamos criar uma página com folhas de estilo, a primeira coisa é criar um arquivo texto vazio, com um nome qualquer (por exemplo: **estilos.css**), para armazenar os estilos de uma dada página.

Criado este arquivo, temos de indicá-lo no arquivo XHTML, de forma que o navegador possa encontrá-lo e usar os estilos definidos no mesmo. Como a informação do arquivo de folhas de estilo é para o navegador, esta indicação virá dentro da seção `<head>...</head>`.

Esta indicação é feita da seguinte forma:

```
<link href="estilo.css" rel="stylesheet" type="text/css">
```

Notando que "estilo.css" é o nome do arquivo de estilos criado pelo desenvolvedor. **rel** e **type** são definições para que o navegador saiba o que fazer com as informações que encontrar neste arquivo.

2.1. Estrutura de um Arquivo .CSS

Conceitos Chave:

- Estrutura do arquivo .CSS

```
tag {  
    propriedade1: valor1;  
    propriedade2: valor2;  
    ...  
    propriedadeN: valorN;  
}
```

- Cuidado com erros!

- Exemplo:

```
h1 {  
    text-align: center;  
    font-size: 3em;  
}
```

- Se houver unidades, não deixar espaço!

- Múltiplas Tags

```
tag1, tagG2, tagN {  
    propriedade1: valor1;  
    propriedade2: valor2;  
    ...  
    propriedadeN: valorN;  
}
```

Dentro do arquivo de estilo (que é um arquivo texto comum), devemos seguir estritamente uma estrutura para que ele funcione. Um erro neste arquivo e a página não funcionará corretamente.

A estrutura é a seguinte:

```
tag {  
  propriedade1: valor1;  
  propriedade2: valor2;  
  ...  
  propriedadeN: valorN;  
}
```

Por exemplo:

```
h1 {  
  text-align: center;  
  font-size: 3em;  
}
```

Isto indica, no arquivo de estilo, que os textos do tipo `<h1>` serão centralizados na tela (`text-align`) e seu tamanho terá uma ênfase (`em`) de 300%, ou seja, ficará 3x maior.

NOTA IMPORTANTE: sempre que houver uma "unidade" no valor de um parâmetro ("em", no caso, é uma "unidade"), lembre-se de NÃO deixar espaço entre o número e a unidade. Caso contrário, o efeito desejado NÃO será obtido.

É possível ainda modificar duas (ou mais) tags simultaneamente, da seguinte forma:

```
tag1, tag2, tagN {  
  propriedade1: valor1;  
  propriedade2: valor2;  
  ...  
  propriedadeN: valorN;  
}
```

Por exemplo:

```
h1, h2 {  
  font-family: verdana, arial, sans-serif;  
}
```

Que altera o tipo de fonte dos títulos e subtítulos simultaneamente.

É possível definir sub-tipos das tags também, mas isso será visto mais adiante.

3. TIPOS DE ESTILOS FUNDAMENTAIS

Conceitos Chave:

- Plano de Fundo
- Texto
- Fonte
- Bordas
- Linhas de Contorno
- Margens
- Espaço de Contorno
- Marcadores de Lista
- Propriedades de Tabelas

Os estilos não são definidos "por tag", mas sim por característica que se deseja alterar. Por exemplo: há propriedades de estilo para planos de fundo, para textos, para células de tabela... e assim por diante. Todos eles estão detalhadamente descritos no Guia de Referência CSS Volume 1.

É fortemente sugerido que tal guia de referência seja seu livro de cabeceira neste semestre.

4. ALGUNS EXEMPLOS DE ESTILOS PARA TAGS BÁSICOS

Conceitos Chave:

- body
- h1 a h6
- p
- a
- hr
- Classes e Centralização de Imagens

Obviamente, os estilos que podem ser definidos variam de acordo com a tag que está sendo modificada. Por exemplo: não faz sentido mudar propriedades de texto em uma imagem. Por essa razão, se isso for tentado, nada acontecerá.

Nesta parte serão indicados alguns tags e alguns modificadores que podem ser usados com os mesmos. Entretanto, esta lista não é, de forma alguma, exaustiva. Para maiores detalhes sobre os parâmetros e valores possíveis, consulte a seção anterior.

Tag body:

É possível mudar, por exemplo, as margens (margin-left, margin-right, margin-top, margin-bottom), a cor do fundo (background-color), imagem de fundo (background-image), se a imagem de fundo estará repetida (background-repeat), dentre outros.

Por exemplo, vamos definir um fundo com margem de 20 pixels em cada lateral da tela, com um fundo creme claro:

```
body {  
  margin-left: 20px;  
  margin-right: 20px;  
  background-color: rgb(255,255,200);  
}
```

Tags h1, h2, h3, h4, h5 e h6:

É possível mudar o alinhamento (text-align), o tamanho da fonte (font-size), o tipo de fonte (font-family), realce de fonte (font-weight), margem (margin-left, margin-right, margin-top, margin-bottom), cor (color), dentre outras.

Por exemplo, vamos redefinir h1 centralizado, com fonte 1.6x maior que o normal, usando uma fonte sem serifa (verdana, arial ou fonte sem serifa padrão do navegador), com texto em realce (mais grosso) e com cor azul escuro:

```
h1 {  
  text-align: center;  
  font-size: 1.6em;  
  font-family: verdana, arial, sans-serif;  
  font-weight: bold;  
  color: rgb(0,0,50);  
}
```

Outro exemplo, vamos redefinir h2 com fonte 1.3x maior que o normal, usando uma fonte sem serifa (verdana, arial ou fonte sem serifa padrão do navegador), com texto em realce (mais grosso) e com cor azul escuro:

```
h2 {  
  font-size: 1.3em;  
  font-family: verdana, arial, sans-serif;  
  font-weight: bold;  
  color: rgb(0,0,50);  
}
```

Tag p:

É possível mudar muitas coisas da tag de parágrafo, dentre elas: o alinhamento (text-align), o tamanho da fonte (font-size), o tipo de fonte (font-family), realce de fonte (font-weight), margem (margin-left, margin-right, margin-top, margin-bottom), cor (color)...

Por exemplo, vamos redefinir P como texto "justificado", com fonte 1.2x maior que o normal, usando uma fonte com serifa (garamond, times new roman ou fonte com serifa padrão do navegador), em cor azul desbotado:

```
p {
  text-align: justify;
  font-size: 1.2em;
  font-family: garamond, times new roman, serif;
  color: #004080;
}
```

Um desejo comum dos usuários é colocar o "espaço inicial no parágrafo" na tag <P>. Neste caso usamos o seguinte, para colocar 200% do tamanho de uma letra como "indentação":

```
p {
  text-indent: 2em;
}
```

Tag a:

É possível mudar várias coisas também na tag de âncora (links), e é muito comum que se mude a aparência de um link. Dentre as coisas possíveis de modificar estão: o tamanho da fonte (font-size), o tipo de fonte (font-family), realce de fonte (font-weight), margem (margin-left, margin-right, margin-top, margin-bottom), cor (color), a decoração da fonte (text-decoration)...

Por exemplo, vamos redefinir **a** como usando uma fonte com serifa (garamond, times new roman ou fonte com serifa padrão do navegador), em letra mais forte (negrito) e sem sublinhado, mas em cor azul mais claro:

```
a {
  font-family: garamond, times new roman, serif;
  font-weight: bold;
  text-decoration: none;
  color: #3030D0;
}
```

Tag hr:

Também é possível mudar muitas coisas na linha divisória. Dentre as coisas possíveis de modificar estão: a altura da barra (height), a largura da barra (width), a cor da barra (background-color), a borda (border), a imagem de fundo (background-image), a repetição da imagem (image-repeat) e assim por diante.

Por exemplo, vamos redefinir **hr** como 7 pixel de altura, 75% da largura da tela, cor de fundo azul escuro, sem borda 3D:

```
hr {  
  height: 7px;  
  width: 75%;  
  background-color: #000480;  
  border: none;  
}
```

4.1. Classes e Centralização de Imagens (OPCIONAL!)

Para centralizar uma imagem, podemos fazer o seguinte: definir um novo tipo de parágrafo `<p class="centrado">` que será centralizado, e colocar a imagem dentro deste parágrafo.

Primeiramente definiremos a nova classe de parágrafo:

```
p.centrado {  
  text-align: center;  
}
```

E em seguida alteramos o documento original, indicando esta versão alternativa de parágrafo no parágrafo da figura:

```
<p class="centrado">  
    
</p>
```

Na verdade, o conceito de classes pode ser explorado em qualquer uma das tags. Como um exemplo, vamos definir a tag **p** de duas formas:

```
p {
  text-indent: 2em;
  text-align: justify;
  font-size: 1.2em;
  font-family: garamond, times new roman, serif;
  color: #004080;
}

p.centrado {
  text-indent: 0;
  text-align: center;
}
```

Sempre que usarmos `<p>`, teremos como resultado um parágrafo justificado e com o espaçamento inicial na margem da esquerda, tamanho 20% maior que o normal na fonte garamond (ou uma de suas alternativas) em uma cor azulada.

Sempre que usarmos `<p class="centrado">`, teremos como resultado um parágrafo centralizado e sem o espaçamento adicional na margem da esquerda. Como a classe "p.centrado" é uma "especialização" de "p", ela herdará as outras definições: fonte garamond 20% maior e cor azulada.

5. BIBLIOGRAFIA

CASCADE Style Sheets, level 2 revision 1: CSS 2.1 Specification - W3C Working Draft 06 November 2006. Disponível em < <http://www.w3.org/TR/CSS21/> >. Visitado em 21 de Dezembro de 2006.

W3 schools - CSS Tutorial. Disponível em < <http://www.w3schools.com/> >. Visitado em 10 de Março de 2009.

RAMALHO, J.A. *HTML 4 Prático e Rápido*. Editora Berkeley, 1999.

BOENTE, A. *Programação Web Sem Mistérios*. Editora Brasport, 2006.

NIELSEN, J. *Projetando Websites*. Ed. Campus, 2000.

EXEMPLO 1 (SEM CSS)**pir_ap04ex.html**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-br">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
  <title>Primeira Pá&ccedil;&atilde;gina Web com CSS</title>
</head>
<body>

  <h1>Primeira p&ccedil;&atilde;gina XHTML com CSS</h1>
  <hr />
  <h2>Primeira Se&ccedil;&atilde;o</h2>
  <p>
  </p>
  <p>A primeira se&ccedil;&atilde;o ir&ccedil;&atilde;o conter dois par&ccedil;&atilde;grafos.
    Note, tamb&ccedil;&atilde;m, que o t&ccedil;&atilde;tulo desta se&ccedil;&atilde;o &ccedil;&atilde;
    em H2, sendo que o t&ccedil;&atilde;tulo da p&ccedil;&atilde;gina foi em H1.
  </p>
  <p>Este &ccedil;&atilde; o segundo par&ccedil;&atilde;grafo da primeira se&ccedil;&atilde;o.
  </p>

  <h2>Segunda Se&ccedil;&atilde;o</h2>
  <p>Esta &ccedil;&atilde; a segunda se&ccedil;&atilde;o e tamb&ccedil;&atilde;m cont&ccedil;&atilde;m
    dois par&ccedil;&atilde;grafos. O t&ccedil;&atilde;tulo desta se&ccedil;&atilde;o
    tamb&ccedil;&atilde;m &ccedil;&atilde; em H2.
  </p>
  <p>Este &ccedil;&atilde; o segundo par&ccedil;&atilde;grafo da segunda se&ccedil;&atilde;o.
  </p>

  <p>Mais informa&ccedil;&atilde;es na
  <a href="http://www.caetano.eng.br/aulas/psw/" title="Disciplina
    Programa&ccedil;&atilde;o para Internet Rica">p&ccedil;&atilde;gina da
    disciplina de Prorgama&ccedil;&atilde;o para Internet Rica</a>.
  </p>
</body>
</html>

```

EXEMPLO 2 (Com CSS)**pir_ap04ex1.html**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-br">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
  <title>Primeira Pá&ccedil;&atilde;gina Web com CSS</title>
  <link href="estilo.css" rel="stylesheet" type="text/css" />
</head>
<body>

  <h1>Primeira p&ccedil;&atilde;gina XHTML com CSS</h1>
  <hr />
  <h2>Primeira Se&ccedil;&atilde;o</h2>
  <p>
  </p>
  <p>A primeira se&ccedil;&atilde;o ir&ccedil;&atilde;o conter dois par&ccedil;&atilde;grafos.
    Note, tamb&ccedil;&atilde;m, que o t&ccedil;&atilde;tulo desta se&ccedil;&atilde;o &ccedil;&atilde;
    em H2, sendo que o t&ccedil;&atilde;tulo da p&ccedil;&atilde;gina foi em H1.
  </p>
  <p>Este &ccedil;&atilde; o segundo par&ccedil;&atilde;grafo da primeira se&ccedil;&atilde;o.
  </p>

  <h2>Segunda Se&ccedil;&atilde;o</h2>
  <p>Esta &ccedil;&atilde; a segunda se&ccedil;&atilde;o e tamb&ccedil;&atilde;m cont&ccedil;&atilde;m
    dois par&ccedil;&atilde;grafos. O t&ccedil;&atilde;tulo desta se&ccedil;&atilde;o
    tamb&ccedil;&atilde;m &ccedil;&atilde; em H2.
  </p>
  <p>Este &ccedil;&atilde; o segundo par&ccedil;&atilde;grafo da segunda se&ccedil;&atilde;o.
  </p>

```

```

    <p>Mais informa&cedil;&otilde;es na
    <a href="http://www.caetano.eng.br/aulas/psw/" title="Disciplina
      Programa&cedil;&atilde;o para Internet Rica">p&acute;gina da
      disciplina de Prorgama&cedil;&atilde;o para Internet Rica</a>.
    </p>
  </body>
</html>

```

estilo.css

```

body {
  margin-left: 20px;
  margin-right: 20px;
  background-color: rgb(255,255,200);
}

h1 {
  text-align: center;
  font-size: 1.6em;
  font-family: verdana, arial, sans-serif;
  font-weight: bold;
  color: rgb(0,0,50);
}

h2 {
  font-size: 1.3em;
  font-family: verdana, arial, sans-serif;
  font-weight: bold;
  color: rgb(0,0,50);
}

p {
  margin: 5px;
  text-indent: 2em;
  text-align: justify;
  font-size: 1.1em;
  font-family: verdana, arial, sans-serif;
  color: rgb(0,60,120);
}

a {
  font-family: verdana, arial, sans-serif;
  font-weight: bold;
  text-decoration: none;
  color: rgb(50,50,200);
}

hr {
  height: 7px;
  width: 50%;
  background-color: rgb(0,60,120);
  border: none;
}

```

EXEMPLO 3 (Com CSS)**pir_ap04ex2.html**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-br">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
  <title>Primeira P&acute;gina Web com CSS</title>
  <link href="estilo2.css" rel="stylesheet" type="text/css" />
</head>
<body>

  <h1>Primeira p&acute;gina XHTML com CSS</h1>
  <hr />
  <h2>Primeira Se&cedil;&atilde;o</h2>
  <p>
  </p>
  <p>A primeira se&cedil;&atilde;o ir&acute; conter dois par&acute;grafos.
    Note, tamb&eacute;m, que o t&iacute;tulo desta se&cedil;&atilde;o &eacute;

```

```

    em H2, sendo que o título da página foi em H1.
</p>
<p>Este é o segundo parágrafo da primeira seção.
</p>

<h2>Segunda Seção</h2>
<p>Esta é a segunda seção e também contém
dois parágrafos. O título desta seção
também é em H2.
</p>
<p>Este é o segundo parágrafo da segunda seção.
</p>

<p>Mais informações na
<a href="http://www.caetano.eng.br/aulas/psw/" title="Disciplina
Programa para Internet Rica">página da
disciplina de Programa para Internet Rica</a>.
</p>
</body>
</html>

```

estilo.css

```

body {
border-color: rgb(200,200,160);
border-width: 20px;
border-style: solid;
margin-left: 100px;
margin-right: 100px;
background-color: rgb(240,240,220);
padding: 10px;
}

h1 {
margin-left: 50px;
margin-right: 50px;
text-align: center;
font-size: 2.0em;
font-family: verdana, arial, sans-serif;
font-weight: normal;
background-color: rgb(50,50,100);
color: rgb(255,255,200);
}

h2 {
font-size: 1.3em;
font-family: verdana, arial, sans-serif;
font-weight: bold;
font-variant: small-caps;
color: rgb(0,0,50);
}

p {
margin: 5px;
text-indent: 2em;
text-align: justify;
font-size: 1.1em;
font-family: verdana, arial, sans-serif;
color: rgb(0,60,120);
}

a {
font-family: verdana, arial, sans-serif;
font-weight: bold;
text-decoration: none;
font-variant: small-caps;
color: rgb(50,50,200);
}

hr {
height: 0px;
width: 0%;
background-color: rgb(0,60,120);
border: none;
}

```