

Estilo de vida Web

Entendendo Style Sheets - Parte 1

Este tutorial introduz inovações na maneira de programar páginas para a Internet. Como pré-requisito é necessário saber programar na linguagem html. Consulte nosso [Manual de HTML](#), se necessário.

OBS.: Para visualizar inovações de layout programadas em *Cascading Style Sheet* - também conhecido por *CSS* - é necessário um *Internet Explorer* versão 3.0 ou superior, ou *Netscape Navigator* a partir da versão 4.0. Se você ainda não possui um browser 4.0, pode atualizá-lo na nossa seção de [download](#). Se você já possui um navegador compatível, visite a [Microsoft CSS Gallery](#) <<http://www.microsoft.com/truetype/css/content.htm>> para ver alguns bons exemplos de *Style Sheets*.

PARTE I:

- [Introdução](#)
- [O que Style Sheets podem fazer](#)
- [Adicionando estilo a sua página](#)

PARTE II:

- [Aplicando CSS em Fontes](#)

PARTE III:

- [Parágrafos](#)

PARTE IV:

- [Imagens](#)

PARTE V:

- [Layers](#)

Introdução

Se você se interessa pela criação de home pages já deve ter ouvido falar em *Cascading Style Sheets*, ou *CSS* apenas. Trata-se de um padrão de formatação para páginas web que vai além das limitações impostas pelo html. Proposto pelo [W3 Consortium](http://www.w3.org) <<http://www.w3.org>> - uma espécie de comitê que define os padrões de programação para a WWW - o *CSS* foi introduzido pela primeira vez pela Microsoft, no lançamento do Internet Explorer 3.0.

O *Cascading Style Sheet* permite uma versatilidade maior na programação do layout de páginas web sem aumentar o seu tamanho em Kb, pois oferece várias possibilidades que antes só eram conseguidas com a utilização de gifs e jpgs. Basicamente, o *CSS* permite ao designer um controle maior sobre os atributos tipográficos de uma home page, como tamanho e cor das fontes, espaçamento entre linhas e caracteres, margem do texto, entre outros. Introduziu também às páginas a utilização de *layers*, permitindo a sobreposição de texto sobre texto ou texto sobre figuras.

Para visualizar inovações de layout programadas em *Cascading Style Sheet* é necessário um Internet Explorer versão 3.0 ou superior. O Netscape incorporou o *CSS* em seu código a partir da versão 4.0. Se você ainda não possui um browser 4.0, pode atualizá-lo na nossa seção de [download](#).

O que Style Sheets podem fazer

Utilizando uma boa combinação de *tags* do código de programação html, como , <I>, <H1> e , podemos criar páginas interessantes sob o ponto de vista do design. Entretanto, as possibilidades que elas oferecem são bastante limitadas se comparadas às oferecidas pelo *CSS*.

Comandos de Style Sheets podem ser aplicados a **toda e qualquer tag**, modificando seus atributos. Até mesmo a *tag* <P> possui um </P> opcional e permite que você defina os atributos de tudo o que for compreendido entre as duas.

Você pode, por exemplo, trocar os atributos de uma *tag* , que tradicionalmente adiciona negrito a uma palavra:

Este é o **negrito normal**.

Agora vamos trocar os atributos da *tag* para mostrar o negrito em uma cor diferente:

Este é o resultado final em *CSS*. Não esqueça de que você só vai visualizar este exemplo se estiver utilizando um browser que suporte *CSS*.

O *Style Sheets* permite a você, por exemplo, configurar todos os comandos em uma página ou em um site inteiro de uma só vez.

Adicionando Estilo a Sua Página

Style Sheets podem ser utilizados de três maneiras diferentes: [local](#) (modificando uma *tag* específica de uma página), [geral](#) (modificando determinados atributos para a toda a página) ou [global](#) (quando criamos um modelo que será aplicado a várias páginas simultaneamente).

• Adicionando estilo localmente:

Style Sheet pode modificar os atributos de uma única *tag* específica, em um determinado ponto de uma página.

Os comandos em *CSS* aplicados localmente seguem a seguinte sintaxe:

```
<tag STYLE="propriedade:valor; propriedade, valor;"...>
```

Estilo de vida Web

Entendendo Style Sheets - Parte 2

PARTE I:

- [Introdução](#)
- [O que Style Sheets podem fazer](#)
- [Adicionando estilo a sua página](#)

PARTE II:

- [Aplicando CSS em Fontes](#)

PARTE III:

- [Parágrafos](#)

PARTE IV:

- [Imagens](#)

PARTE V:

- [Layers](#)

Aplicando CSS a fontes

Comandos:

- [font-family](#)
- [font-size](#)
- [font-weight](#)
- [font-style](#)
- [text-transform](#)
- [text-decoration](#)

font-family é o comando CSS utilizada para indicar qual a fonte que uma *tag* ou documento irá mostrar. Em uma aplicação geral de CSS a uma página, a linha de comando parecerá com esta:

```
P { font-family: Trebuchet MS, Arial, Helvetica }
```

No exemplo acima, tudo o que estiver entre as *tag* <P> e </P> no seu documento html será visualizado em *Trebuchet MS*. A razão de definir mais de uma fonte deve-se ao fato de que nem todos os computadores possuem as mesmas fontes instaladas. No caso, se o computador não possuir a *Trebuchet MS* instalada em seu hard disk, automaticamente passará para a segunda opção e mostrará o texto em *Arial*.

Deve-se observar que o nome da fonte especificadas na linha de comando do CSS deve ser exatamente igual ao nome da mesma no diretório "fontes" do computador. A fonte "*Courier*", por exemplo, leva este nome na plataforma *Mac*, mas se chama "*Courier New*" no *Windows*.

font-size é utilizado para definir o tamanho pelo qual as fontes serão apresentadas na tela. Há três formas básicas para definição de tamanho:

1. pontos, pixels, cm e outras unidades de medida;
2. palavras-chave;
3. porcentagem

1. Unidades de medida:

Ponto é uma medida familiar para designers, e pode ser usada para definir o tamanho de um elemento em uma página html. Tem sido a medida mais utilizada na programação em CSS pois funcionam bem em todos os *browsers* e plataformas operacionais. A única observação sobre esta medida é que as fontes costumam aparecer um pouco maiores na tela dos PCs do que nos monitores da linha *Mac*. Uma linha de comando geral em CSS parecerá assim:

```
P { font-size: 20pt } para pontos
```

ou:

```
B { font-size: 20px } para pixels, por exemplo.
```

As unidades de medida que podem ser utilizadas são:

- pt: pontos
- px: pixels
- in: inches
- cm: centímetros
- mm: milímetros
- pc: picas
- em: ems
- ex: x-weight

Pixel é uma medida familiar aos *web designers*. Pode dar uma boa noção do tamanho que a fonte aparecerá na tela, mas infelizmente a utilização dessa medida tem causado problemas para na impressão das páginas em papel.

Como já foi dito anteriormente, não há limites para a especificação de tamanho de uma

Estilo de vida Web

Entendendo Style Sheets - Parte 3

PARTE I:

- [Introdução](#)
- [O que Style Sheets podem fazer](#)
- [Adicionando estilo a sua página](#)

PARTE II:

- [Aplicando CSS em Fontes](#)

PARTE III:

- [Parágrafos](#)

PARTE IV:

- [Imagens](#)

PARTE V:

- [Layers](#)



Aplicando CSS a parágrafos

Comandos:

- [line-height](#)
- [text-align](#)
- [text-indent](#)
- [margin-left, top...](#)
- [padding-right, bottom...](#)
- [border-color, width...](#)

line-height é o comando que controla o espaçamento entre as linhas de um texto (também conhecido pelo termo *leading*). O espaço é definido pela distância entre a base das linhas de um texto. Veja o exemplo a seguir:

```
P { line-height: 8pt }
```

"Acima da verdade estão os deuses.
A nossa ciência é uma falhada cópia
Da certeza com que eles
Sabem que há o Universo"

(...)

Os valores de `line-height` podem ser estipulados de três maneiras diferentes: por número, por unidade ou por porcentagem.

Estipulando o valor em número, o browser utilizará o `font-size` para definir o tamanho. Ele multiplicará o valor do `font-size` pelo número para definir o espaço entre as linhas. No exemplo a seguir teremos um espaçamento de 24 pontos:

```
P { font-size: 12pt; line-height: 2 }
```

(...)

"Tudo é tudo, e mais alto estão os deuses,
Não pertence à ciência conhecê-los,
Mas adorar devemos
Seus vultos como às flores

(...)

Estipulando o valor em unidade, podemos definir diretamente o valor do espaçamento. As unidades que podem ser utilizadas são as mesmas explicadas para o comando `font-size`: pt, px, in, cm, mm, pc, ex, em.

Estipulando o valor por porcentagem, o browser utilizará o valor do `font-size` como referência. No exemplo a seguir, ao estipularmos um valor de 60%, com uma fonte de 10 pontos, teremos um espaçamento de 6 pontos como resultado:

```
P { font-size: 10pt; line-height: 60% }
```

(...)

"Porque visíveis à nossa alta vista,
São tão reais como reais as flores
E no seu calmo Olimpo
São outra Natureza."

Ricardo Reis
(pseudônimo de Fernando Pessoa)

text-align é a comando utilizado para o alinhamento de parágrafos em *Style Sheets*. Ela só funciona em elementos que definam parágrafos, como `<P>`, `<H1>`, `<H2>`, `<BLOCKQUOTE>` e

Estilo de vida Web

Entendendo Style Sheets - Parte 4

PARTE I:

- [Introdução](#)
- [O que Style Sheets podem fazer](#)
- [Adicionando estilo a sua página](#)

PARTE II:

- [Aplicando CSS em Fontes](#)

PARTE III:

- [Parágrafos](#)

PARTE IV:

- [Imagens](#)

PARTE V:

- [Layers](#)

Aplicando CSS a imagens

Comandos:

- [color](#)
- [background-color](#)
- [background-image](#)
- [background-repeat](#)
- [background-attachment](#)
- [background-position](#)
- [background](#)

color

Podemos aplicar cores a qualquer elemento de uma página html. Para tanto, utilizamos nomes ou números hexadecimais da mesma forma que na programação em html básico. A novidade em CSS é que podemos utilizar diretamente valores RGB para definir a cor.

Exemplo: itálicos em vermelho podem ser definidos por:

Nome:

```
I { color: red }
```

Valor numérico:

```
I { color: #FF0000 }
```

Código RGB:

```
I { color: rgb (255, 0, 0) }
```

O Internet Explorer 3 não é compatível com a definição em código rgb, mas ambos Netscape 4 e IE 4 aceitam.

background-color da mesma forma que no comando color, podemos aplicar valores numéricos, nomes ou códigos RGB para configurar uma cor de fundo de qualquer elemento da página html, incluindo imagens.

```
P.cinza { background-color: gray }
```

Neste exemplo você pode observar a cor cinza aplicada a todo o fundo deste parágrafo.

Observações:

No *IE3* e no *Netscape 4* a cor aparece apenas atrás de cada palavra, enquanto no *IE4* teremos um retângulo de cor envolvendo todo o parágrafo.

O *IE3* não aceita o comando **background-color**, mas aceita o comando **background**, que será discutido mais a frente.

background-image

Com este comando podemos facilmente aplicar uma gif ou jpg atrás de qualquer elemento html.

```
<P STYLE="background-imagem: url (nome_da_imagem.gif)>
```

Como exemplo, aplicamos uma imagem atrás de todo este parágrafo. A figura se repete continuamente, da mesma forma que o background de uma página html.

Assim como podemos aplicar a imagem na tag <BODY> fazendo com que a figura se repita por toda a página, podemos aplicá-la a uma única palavra, somente. O valor : url (none) serve para desativar qualquer valor aplicado anteriormente ao background image.

Estilo de vida Web

Entendendo Style Sheets - Parte 5

PARTE I:

- [Introdução](#)
- [O que Style Sheets podem fazer](#)
- [Adicionando estilo a sua página](#)

PARTE II:

- [Aplicando CSS em Fontes](#)

PARTE III:

- [Parágrafos](#)

PARTE IV:

- [Imagens](#)

PARTE V:

- [Layers](#)

Utilização de Layers

Comandos:

- [position, left, top](#)
- [width](#)
- [z-index](#)

Vamos verificar agora a maior dentre as inovações introduzidas em Style Sheets: a utilização de layers. Sabemos que posicionamento utilizando o tradicional código *html* é algo delicado. Mesmo com a utilização de tabelas, certas vezes alguns problemas de posicionamento e alinhamento podem quebrar a cabeça do mais experiente programador. Style Sheets chegou para acabar com essas limitações, como veremos a seguir.

Nota: Por terem sido criados posteriormente, os comandos *Style Sheets* para posicionamento e *layers* não podem ser visualizados com o Internet Explorer 3.

[position, left, top](#)

Este são os comando *CSS* utilizados para posicionar os elementos na página *html*. O comando **left** determina a distância entre o elemento e a margem esquerda da página. O Comando **top** determina a distância até a parte superior da mesma.

Podemos utilizar valores de percentagem ou unidades de medida para determinar a posição do elemento através desses comandos. As unidades que podem ser utilizadas são as mesmas apresentadas anteriormente neste tutorial.

O comando **position** determina se o posicionamento dos elementos será **absoluto** ou **relativo**. Posicionamento absoluto significa que cada elemento terá sua posição determinada individualmente, através de coordenadas. Já no posicionamento relativo temos a posição de cada elemento determinada em relação aos outros.

Qualquer elemento pode ter sua posição determinada em *Style Sheets*: imagens, vídeos, párrafos ou uma única letra.

```
.exemplo1 {position: absolute; left: 200px; top: 100px}
```

Clique para ver um exemplo de [posicionamento absoluto](#).

```
<IMG STYLE="position: relative; left: 10px; top: 10px">
```

Clique para ver um exemplo de [posicionamento relativo](#).

[width](#)

Determina largura. Esse comando só pode ser aplicado a elementos com posicionamento absoluto em uma página. O tamanho pode ser especificado em valores de percentagem ou unidades de medida.

```
.exemplo3 { position: absolute; left: 30%; top: 100px; width:200px }
```

Clique para ver um [exemplo](#) da aplicação do comando `width`.

[z-index](#)

Através do posicionamento absoluto, podemos sobrepor vários elementos. O comando **z-index**, por sua vez, permite definir em que ordem eles serão visualizados.

Neste exemplo posicionamos os *layers* aparecem na ordem em que foram dispostos:

```
.layer1 { color: yellow; position: absolute; left: 210px; top: 110px }
```



Copyright © 1998 [ZAZ](#). Todos os direitos reservados. All rights reserved.