

CSS e efeito cascata das estilizações

- Qual a razão do termo “cascata” na terminologia CSS? A forma como as regras de estilo são aplicadas obedece a 3 princípios básicos: herança, especificidade do seletor e proximidade do elemento estilizado.
- Folhas de estilo em cascata podem se originar de três fontes:
 - Autor da aplicação;
 - Usuário;
 - Agente de usuário (navegador).
- Lembrar que a maioria das propriedades de um elemento são as mesmas propriedades de seu elemento-pai. Este mecanismo é chamado de herança. Outras propriedades não são herdadas automaticamente. Exemplo: tipo de fonte, cor de fonte são herdadas. Margens, bordas e fundos não são.

Origem das folhas de estilo em cascata

- *Do autor*: é o procedimento mais comum, como o que estamos fazendo agora: o autor cria as folhas de estilo e as serve incorporadas aos seus documentos;
- *Do usuário*: o usuário de uma página web pode criar folhas de estilo de acordo com suas preferências e necessidades. Os modernos navegadores oferecem recursos para isso. Exemplo: no IE, vá em Opções da Internet -> acessibilidade
- *Do agente de usuário*: todo fabricante de navegadores dota seu navegador com uma folha de estilo própria, interna, que aplica um estilo-padrão aos elementos (X)HTML de uma página.

Colisão de propriedades

- Imagine a seguinte situação:
 1. O autor estilizou suas fontes na cor vermelha;
 2. O usuário, na cor azul;
 3. Enquanto que o estilo-padrão para fontes do navegador é preta.
- Em que cor a fonte será exibida na tela? Há uma precedência na hora da estilização?
- Neste momento, surge o conceito do efeito cascata sobre as regras de estilo, de acordo com sua especificidade e proximidade do elemento a se estilizado;
- Vence sempre a regra que tiver o maior peso.

Ordem de aplicação de uma regra

- Quando houver conflito onde as mesmas propriedades são aplicadas a um seletor, mas com valores diferentes, temos a seguinte ordem de precedência – da mais alta para a mais baixa:

- 1) Folha de estilos do autor:

- A) estilo inline;

- B) estilo incorporado;

- C) estilo linkado (externo).

- 2) Folhas de estilo do usuário;

- 3) Folhas de estilo do navegador.

Observação importante: o usuário pode ter a maior prioridade em determinada regra, bastando acrescentar nesta a diretiva `!important`. Isto sobrescreve a mesma propriedade definida pelo autor. Exemplo:

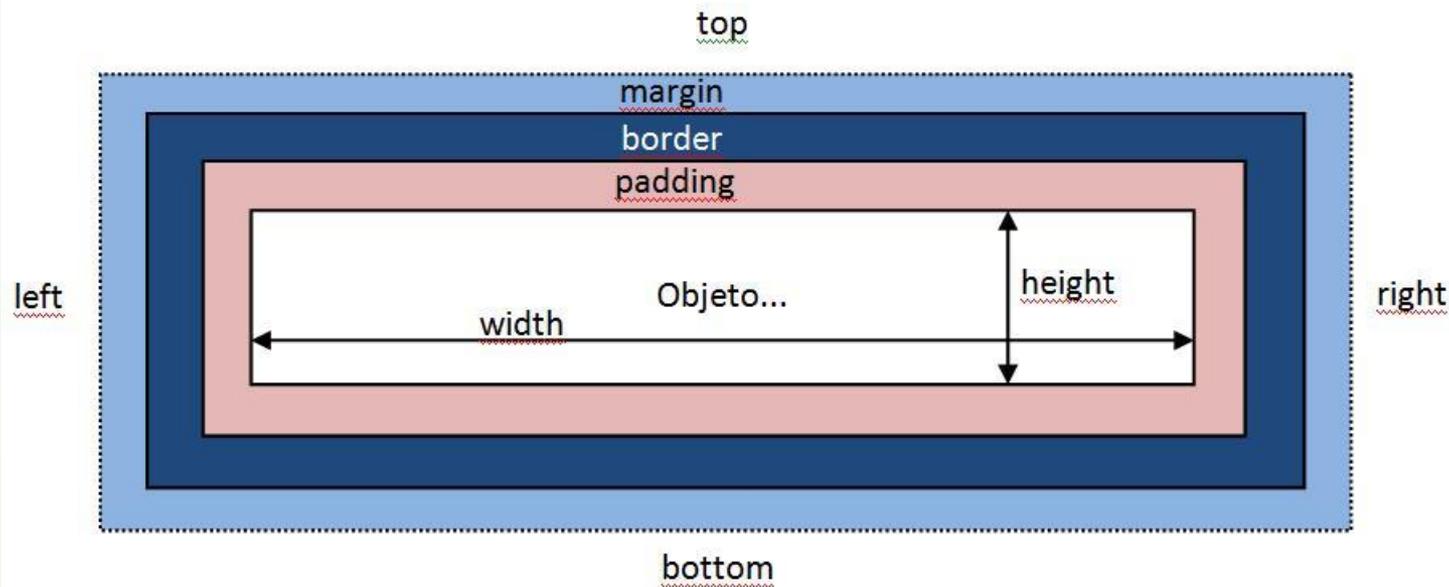
```
p {  
  color: red !important;  
}
```

O modelo de caixa das CSS

- Cada elemento em um documento HTML permite que o CSS crie, em torno deste elemento, uma caixa (box), com diversas características, como, por exemplo:
 - Dimensões (altura, largura);
 - Bordas (espessura, estilo da borda, cor);
 - Preenchimento para afastar o conteúdo das bordas internas do elemento (padding);
 - Afastamento desta caixa em relação aos outros objetos (margens).

Modelo de caixa de um objeto em CSS

- A grande maioria dos marcadores cria em torno de si um modelo de formatação visual chamado de Box Model;
- Elementos de um Box Model:
 - width e height – largura e altura do objeto;
 - padding – área de preenchimento ao redor das bordas internas do objeto;
 - border – borda situada externamente ao padding;
 - margin – espaço que separa um objeto do outro na página



Propriedades CSS para o modelo de caixa

- margin {
margin-top
margin-right
margin-bottom
margin-left

- padding {
padding-top
padding-right
padding-bottom
padding-left

- border {
border-width: espessura da borda
border-style: estilo da borda
border-color: cor da borda

As propriedades que se referem a espessura admitem qualquer uma das unidades de comprimento aceitas pelo CSS: px, em, ex, pt, mm, cm, etc..., exceto %.

A propriedade border-style se refere a um dos 8 estilos diferentes disponíveis para ela, além de none e hidden.

border-width

- Podemos definir **espessuras** diferentes para cada uma das quatro bordas:
 - border-top-width
 - border-right-width
 - border-bottom-width
 - border-left-width
- Os valores destas propriedades podem ser uma medida CSS (pixel, porcentagem, ex, em, in, cm, mm, pt, etc...) ou
- Uma das três palavras-chave:
 - *thin*
 - *medium*
 - *thick*

border-color

- Podemos definir cores diferentes para cada uma das quatro bordas:
 - border-top-color
 - border-right-color
 - border-bottom-color
 - border-left-color
- Os valores destas propriedades podem ser:
 - Uma cor em hexadecimal;
 - Uma cor usando seu nome;
 - Uma cor em RGB ou RGBA;
 - Uma cor em HSL ou HSLA;
 - Ou o valor *transparente (sem cor)*.

border-style

- Podemos definir estilos diferentes para cada uma das quatro bordas:

- border-top-style
- border-right-style
- border-bottom-style
- border-left-style

Estilo	Descrição
none	Define espessura 0 para a borda
hidden	Idem ao anterior
dotted	Borda pontilhada
dashed	Borda tracejada
solid	Borda contínua sólida
double	Borda formada por duas linhas contínuas
groove	Borda com aparência entalhada
ridge	Borda com aparência de resalto
inset	Borda em baixo relevo
outset	Borda em alto relevo

Forma abreviada da propriedade border

```
border: espessura estilo cor;
```

- Exemplo: border: 2px dotted red;
- Equivale a border-width: 2px; border-style: dotted; border-color: white;
- Nota 1: aplica as mesmas características aos 4 lados da caixa CSS;
- Nota 2: se o estilo não for declarado, o navegador atribui automaticamente **none** e não mostra a borda ao redor do elemento. Quando não declaramos nenhum valor para a propriedade border, são usados os valores-padrão de cada navegador:
 - border-width: *medium*;
 - border-style: *none*;
 - border-color: *a mesma cor da fonte do elemento rodeado pela borda.*