

## CSS 3

A propriedade `float`: faz com que um elemento possa ser jogado para a direita e para a esquerda e permite que outros elementos fiquem em volta dele. É muito usado para imagens mas também é útil quando se trabalha com layouts.

Os elementos são fluídos na horizontal (para a direita e para a esquerda) mas não na vertical. Um elemento fluído é jogado tão longe quanto possa. Os elementos que vem depois do elemento fluído ficarão em volta dele. Os elementos que vem antes não sofrem modificação. Se uma imagem é fluída para a direita, o texto que vem depois fica em volta dela, à esquerda. Veja o exemplo:

```
<style> img { float: right; }
</style>
...
<body>
<p>No parágrafo abaixo, incluiu-se uma imagem
com float<b>float:right</b>. O resultado:
a imagem fluirá. <p>
Repita esta linha umas 10 vezes para ver o
resultado
```

## Exercício 1

Copie o exemplo acima e depois altere o valor de `float`: para `left`. Brinque com `margin`, `padding` e `border`.

Quando há diversos elementos fluindo, um depois do outro, eles aparecerão um ao lado do outro enquanto houver espaço.

## Exercício 2

Crie um arquivo chamado `galeria.html` e `galeria.css` colocando nele 12 figuras todas do mesmo tamanho. Defina no CSS

```
.ima { float: left; width: __px;
height: __px; margin: 5px; }
```

Use no estilo um tamanho máximo, maior que todas as imagens que vai mostrar. No arquivo `html`, faça

```
`, `<p>`, `<div>` etc) para efeitos de layout, há alguns truques. Para centralizar elementos de bloco pode-se estabelecer as margens direita e esquerda para `auto`. Veja um exemplo

```
<style>
.center { margin-left: auto;
margin-right: auto; width: 70%;
background-color: #b0e0e6; }
</style> </head>
...
<body>
<div class="center">
<p>Escreva aqui um texto de umas 3 linhas</p>
</div> </body>
```

Veja como fica o exemplo acima num browser.

O assunto agora são combinadores. Um combinador é algo que explica o relacionamento entre seletores. Um seletor CSS pode conter mais do que um seletor simples. Entre os seletores simples, pode-se incluir um combinador. Há 4 combinadores no CSS3. São eles: descendentes (um espaço); filho ("`>`"); irmãos adjacentes ("`+`") e irmãos gerais ("`~`"). Vamos a eles:

O descendente captura todos os elementos que são descendentes de um elemento específico. O exemplo abaixo pega todos os `<p>` que estão dentro de um elemento `<div>`.

```
div p {background-color: aqua;}
```

Já o combinador filho pega os elementos que são filhos imediatos do elemento específico. O combinador irmãos adjacentes e irmãos gerais fazem combinações de critérios. A coisa é sutil. Veja em [www.w3schools.com](http://www.w3schools.com) no tutorial CSS3 a lição CSS combinators e veja as sutilezas em ação.

A pseudo-classes é usada para definir um estado especial de um elemento. Existem diversas pseudo-classes que funcionam como se classes fossem, mas usam palavras reservadas, fixas, cada uma delas com uma especificação. Por exemplo, uma pseudo classe é a `first-child` que se aplica ao primeiro filho daquele elemento. Por exemplo, ao fazer

```
p:first-child { color: blue; }
```

fará com que o primeiro elemento `<p>` seja escrito em azul. Há uma lista de mais de 30 pseudo-classes. Fica como lição de casa estudá-las. Todas elas começam seu nome com `:`.

Os pseudo-elementos são usados para aplicar estilos a partes de um elemento. Seus nomes começam com `::`. São eles `::first-line`, `::first-letter`, `::before`, `::after` e `::selection`. Este último estiliza o que o usuário selecionar.

## Exercício 4

Digite o arquivo a seguir e veja-o num browser.

```
<style>
p::first-letter { font-size: 200%;
color: blue; }
</style> </head> <body>
<h1>Bem vindo à minha página</h1>
<p>Meu nome é José.</p>
<p>Eu moro em Curitiba.</p>
```

O tutorial de CSS em [www.w3schools.com](http://www.w3schools.com) tem uma sessão denominada CSS Image Gallery que descreve de maneira fácil e adequada como construir uma galeria de imagens, na qual, quando se clica numa imagem ela é ampliada e mostrada com detalhes. Vale a pena estudar o código.

Ainda sobre imagens, em CSS3 há uma propriedade chamada opacidade (`opacity`) que regula a transparência de uma imagem. O valor é um número entre 0.0 e 1.0. Quanto mais próximo de 1 mais opaca (menos transparente) é a imagem. Com um valor de 0.0 não há imagem nenhuma. no IE esta propriedade só funciona a partir da versão 9.

## Exercício 5

Construa um arquivo `html` e `css` denominados `flor` e coloque neles pelo menos 3 imagens de flores. Daí aplique este código e veja o que acontece quando o cursor passeia pelas imagens.

```
<style>
img { opacity: 0.4; }
img:hover { opacity: 1.0; }
</style> </head> <body>
<h1>Transparência de imagens</h1>


```

Algumas propriedades CSS estão dirigidas para alguma mídia específica. Por exemplo, a propriedade `voice-family`: só tem sentido para dispositivos que "falem". Ou então a propriedade `font-size`: talvez deverá ter valores diferentes para a tela e para a impressora.

Assim, surge a regra `@media` que permite diferentes construções (para diferentes mídias) na mesma folha de estilo. Veja o exemplo descrito acima (tela e impressora)

```
@media screen {
p { font-family: verdana,sans-serif;
font-size: 14px; } }
@media print {
p { font-size: 20px; color: red; } }
```

Os tipos de mídia permitidos são:

tipo	descrição
all	todos os tipos
aural	synthesized de som
braille	dispositivos tácteis
embossed	impressoras braille
handheld	dispositivos pequenos
print	impressoras
projection	projetores
screen	tela de computador
tty	dispositivos de caracter não proporcional
tv	televisões

Vai se ver agora como aplicar estilos para elementos que tenham atributos específicos. Usa-se o seletor `[atributo]`. Por exemplo, para estilizar todos os elementos `<a>` que tenham o atributo `target`, faz-se

```
a[target] { background-color: aqua;}
```

Já quando quer-se escolher apenas elementos com atributo valendo algum valor, usa-se o seletor `[atributo=valor]`. Por exemplo, para estilizar elementos `<a>` cujo `target` seja igual a `_blank`, escreve-se

```
a[target="_blank"] { color: green;}
```

Para escolher elementos que tenham uma palavra específica no seletor usa-se `[atributo~="palavra"]`. No exemplo a seguir serão selecionados todos os elementos que tenham um atributo de título contendo a palavra "flor".

```
[title~="flor"] { border: 5px solid; }
```

As outras opções são

<code> =</code>	seleciona elementos cujo atributo comece com o valor especificado. (o valor tem que ser uma palavra completa)
<code>^=</code>	seleciona elementos cujo atributo comece com o valor especificado. (o valor NÃO tem que ser uma palavra completa)
<code>=\$</code>	seleciona elementos cujo atributo termina com o valor especificado. (o valor NÃO tem que ser uma palavra completa)
<code>*=</code>	seleciona elementos cujo atributo contém o valor especificado. (o valor NÃO tem que ser uma palavra completa)

## Para você fazer

1. Escreva aqui e se não couber continue no verso desta folha um resumo das tags estudadas nesta aula:

---

---

---

---

---

---

---

---

---

---

2. Imprima os arquivos pedidos (galeria `html` e `css` e `flor` `html` e `css`) grampeie-os nesta folha e devolva tudo ao professor.

