

Introdução à criação de sites com CSS3

O CSS é utilizado para estilizar a página HTML. Com ele você cria regras de estilo para elementos ou grupos de elementos.



Uma regra CSS é representada por um seletor ou um grupo de seletores, no nosso caso é o `<a>`, então dentro de um par de chaves adicionamos as declarações, no exemplo acima estamos alterando cor e tamanho da fonte dessa âncora, as declarações são formadas por uma propriedade e um valor.

Percebam que podemos colocar vários seletores em uma regra separando-os por vírgula.

No exemplo da imagem, foram utilizados seletores de tipo, pois representa elementos HTML. Eles são utilizados para estilizar os elementos, o que significa que ao estilizar um `<h1>`, todos os `<h1>` receberão o mesmo estilo.

Além dos seletores de tipo, existem também os seletores de classe (**class**) e **id**. Eles podem ser utilizados dentro de qualquer elemento HTML. O **id** é representado pelo símbolo # (hash) seguido de um nome para esse **id**. Classe: a classe é representada de forma parecida do **id**, mas é precedida por um ponto em vez do hash.

E a diferença mais importante entre eles é a forma como devem ser usados: o **id** só pode ser usado uma vez em uma página HTML enquanto a **class** não tem restrições.

Adicionando o CSS ao HTML:

Podemos adicionar CSS de duas formas, com o elemento `style`, e assim suas regras ficarão no arquivo HTML. Mas, a boa prática é criar um arquivo CSS e adicioná-lo na página através do elemento `link`.

Adicione os seletores nos devidos elementos do código HTML. Ex.:

```
<h1 id="title">  
<h2 class="sec_title">  
<h3 class="trd_title">
```

Agora, crie um elemento *link* dentro do head do seu arquivo e adicione os atributos *rel="stylesheet"* e *href="style.css"*, o *rel* denota o tipo de arquivo que estamos incluindo na página e o *href* é o caminho para o arquivo.

```
<link rel="stylesheet" href="style.css">
```

Na mesma pasta do arquivo *HTML* crie um arquivo chamado *style.css*. É nele que você irá chamar os seletores criados no código HTML e estilizar seus respectivos elementos. Ex.:

```
#title, .sec_title, .trd_title {  
    color: blue;  
}
```

```
.sec_title {  
    size: 14px;  
}
```

```
.trd_title {  
    font-style: italic;  
}
```

BOX MODEL

Quando estamos criando o layout de um site o navegador representa cada elemento HTML como uma caixa retangular, isso é o box-model. E com CSS nós alteramos a aparência dessa caixa (largura, altura, cor de fundo, etc.). Essa caixa é composta por 4 áreas: o conteúdo, o padding, a borda e a margem.

- As margens (margin) são espaçamentos entre elementos;
- As bordas (border) ;
- O padding é um espaçamento entre as bordas e o conteúdo, a diferença para as margens é que declarações de imagem de fundo funcionam nele;
- O conteúdo (content) é o que o seu bloco representa, um texto, uma imagem, um vídeo;

ESTILIZANDO ELEMENTOS

Margin e padding

Se quisermos atribuir tamanhos diferentes para cada lado do *box* nós podemos, e vamos ver três formas de fazer isso.

A primeira é colocando um valor para as partes superior e inferior e depois para os lados esquerdo e direito.

```
.post {  
  padding: 10px 5px;  
}
```

O valor de 10 *pixels* se refere ao eixo Y, ou partes superior e inferior, e os 5 *pixels* se referem aos lados esquerdo e direito.

A segunda forma é dando valores para cada lado do *box*.

```
.post {  
  padding: 15px 10px 5px 0;  
}
```

Então começamos pelo topo com 15 pixels, passamos o lado direito com 10 pixels, depois para a parte inferior com 5 pixels e por último o lado esquerdo com 0, e sempre nessa ordem.

Uma boa dica também é que quando o valor for 0 não precisamos colocar a unidade.

A terceira forma é com as propriedades específicas para cada lado, até agora tínhamos visto atalhos para essas propriedades.

```
.post {  
  padding-top: 15px;  
  padding-right: 10px;  
  padding-bottom: 5px;  
  padding-left: 0;  
}
```

Essa opção é mais usada quando temos o mesmo valor para 3 lados, e o quarto precisa ter um valor diferente, então usamos o padding com apenas um valor e uma dessas opções para representar o lado diferente.

Background

É possível mudar a cor de fundo, adicionar uma imagem, alinhar a imagem, etc.

```
.post {  
  background-color: green;  
  background-image: url("bg.png");  
  background-position: top;  
}
```

E aqui temos 3 formas de colocar uma cor de fundo, e ainda existem outras.

```
.post {  
  background-color: green;  
  background-color: #008800;  
  background: #008000;  
}
```

A primeira é pelo nome da cor em inglês, a segunda é pelo código hexadecimal e a terceira é usando apenas o atalho *background*.

Border

A propriedade *border* pode ter 3 valores: a largura, a cor e o estilo, mas existem algumas particularidades nisso.

A largura pode ser usada com várias unidades, como px, em e mm. A cor pode ser atribuída pelo nome ou por um código hexadecimal, assim como fizemos com o *background*, e o estilo é representada por palavras-chave, vamos ver algumas delas:

```
.post {  
  border: 3px solid blue;  
  border-top: 2px dotted green;  
  border-right: 4px dashed pink;  
}
```

solid: mostra uma borda simples e reta;

dotted: são bolinhas com um pequeno espaçamento entre elas;

dashed: forma uma linha tracejada.

É possível separar a estilização de cada lado de uma borda. Se você não quiser usar a propriedade *border* existem as propriedades específicas para cada aspecto de uma borda, são elas *border-width* para a largura, *border-color* para a cor e *border-style* para o estilo.

Aqui temos o mesmo código anterior de duas formas diferentes, a primeira com o atalho *border* e a segunda com cada propriedade específica.

```
.post {  
  border: 3px solid #505050;  
}  
  
.post {  
  border-width: 3px;  
  border-color: #505050;  
  border-style: solid;  
}
```

E depois disso podemos juntar os lados com os aspectos de uma borda e criar uma regra mais específica ainda.

```
.post {  
  border-top-width: 3px;  
  border-top-color: blue;  
  border-top-style: solid;  
}
```

E a última propriedade é o *border-radius*, ele permite arredondar os cantos de um elemento. Podemos usar várias unidades, mas as mais comuns são os pixels e a porcentagem.

Colocando apenas um valor mudamos todos os cantos do elemento, mas seguindo aquela mesma ordem que vimos no *padding* e *margin* - topo, direita, inferior e esquerda - conseguimos alterar cada canto separadamente.

`border-radius: 10px;`

`border-radius: 50%;`

`border-radius: 10% 20%;`

`border-radius: 10% 20% 15% 22%;`

Obs.: Se colocar o border radius com 50%, a borda fica redonda.

ESTILIZANDO TEXTOS

font-family

Com o font-family podemos alterar a fonte dos nossos textos, como uma fonte da internet ou uma que esteja instalada no nosso computador, mas vamos nos ater às fontes seguras, chamadas de web safe fonts.

Essas fontes são chamadas assim pois são encontradas em quase todos os sistemas e podem ser usadas sem preocupação.

font-size

O font-size nos ajuda a mudar o tamanho do texto, existem algumas unidades de medida para ele mas por enquanto os pixels são suficientes para nós.

font-style

Usamos o font-style para tornar um texto itálico, na maioria das vezes você usará apenas o valor *italic* para ele, mas se precisar tirar o itálico de um texto você pode usar o valor *normal*.

font-weight

Altera o peso do texto. o valor *normal* é o peso comum e o *bold* é o negrito.

text-transform

Alterna o texto entre maiúsculas e minúsculas.

Valores:

uppercase - transforma todas as letras em maiúsculas

lowercase - transforma todas as letras em minúsculas

capitalize - capitaliza todas as letras iniciais de cada palavra

text-decoration

Usamos o text-decoration para colocar uma linha acima, abaixo ou no meio do texto, deixando-o traçado. Os valores são:

underline - linha abaixo

overline - linha acima

line-through - coloca linha cortando o texto

ESTILIZANDO LISTAS

```
ul {  
  list-style-type: square;  
}  
  
ol {  
  list-style-type: upper-roman;  
}  
  
ul {  
  list-style-type: "\1F44D";  
}
```

```
ul {  
  list-style-image: url("rocket.png");  
}
```

DIMENSÕES E ALINHAMENTOS

Width

Height

Max-width

Max-height

Margin

Text align

Observações:

Se após definir novas margens para uma section, uma imagem ficar vazando dela, por ser mais larga que a nova largura da section, isso pode ser resolvido por definir a largura da imagem para, no máximo, 100%.

Para que o body ou uma section apareça alinhada ao centro em qualquer formato de tela, definir a margin como auto.

Para que a página se ajuste a qualquer tela, a largura do body deve ser definida como max-width em vez de width.