

The background features a collection of 3D cubes in various colors (purple, blue, yellow, green, orange, pink, grey) arranged in a scattered, overlapping pattern. A thin, multi-colored curved line (red, blue, green) starts from the bottom left and curves upwards towards the center. The title 'Introdução ao CSS3 8' is prominently displayed in the upper right area.

Introdução ao CSS3 8

- ✓ Conceito de camadas
- ✓ Folhas de estilo (CSS);
- ✓ Estilos CSS;
- ✓ Cascadeamento;
- ✓ Inserção de comentários;
- ✓ Atributos para formatação;
- ✓ Seletores;

8.1.Introdução

Sites da Web, sejam eles redes sociais, comércio eletrônico, portais de notícias ou outros, produzem milhões de novas informações todos os dias. Além de gerar conteúdo novo, tais sistemas são obrigados a atualizar periodicamente as páginas já existentes. Tal responsabilidade implica no envolvimento de vários profissionais, incluindo diagramadores, redatores, programadores e editores.

Uma mudança gráfica no layout do site, por exemplo, exigiria muito trabalho e tempo. A alteração manual dos arquivos de um site não é apenas uma tarefa trabalhosa, mas também suscetível a erros, principalmente quando lidamos com sites ou Web apps, cujos usuários produzem informação.

Para amenizar a complexidade dessas modificações, organizar o layout e deixá-lo limpo e atual, muitos profissionais utilizam as folhas de estilo. Estas podem ser bastante úteis quando precisamos alterar elementos do mesmo tipo, mas que pertencem a documentos diferentes.

Nesta leitura, abordaremos como trabalhar com folhas de estilo em cascata (CSS – Cascading Style Sheets) e com seus recursos principais.

8.2.Conceito de camadas

Segundo a definição desse conceito, os elementos de programação com a mesma finalidade devem ser agrupados no mesmo local. Além de facilitar a inclusão de modificações, o conceito de camadas determina uma estrutura mais organizada para o sistema. Dessa forma, é possível acrescentar ou atualizar informações sem comprometer o conteúdo atual.

A partir do conceito de camadas, é possível trabalhar com as folhas de estilo em cascata. As folhas de estilo CSS definem como o sistema será formatado de modo a facilitar a visualização do código e a organização do sistema como um todo.

8.3.Folhas de estilo (CSS)

A folha de estilos em cascata CSS é um documento responsável pela definição de critérios de formatação de páginas. Com isso, é possível estabelecer um padrão de formatação que será associado às páginas desejadas. A extensão utilizada para os arquivos de folhas de estilo é a **.css**.

Por meio de um arquivo de folha de estilo, podemos definir a formatação dos elementos referente à cor, fonte, tamanho, posicionamento e outros estilos de um documento HTML. No entanto, para que tais padrões sejam aplicados, será necessário criar um vínculo entre a folha de estilo e as páginas que serão formatadas.

8.3.1. Declarando estilos

Para que uma tag seja formatada por meio de uma folha de estilo, será preciso declarar a sintaxe a seguir:

```
elemento {  
    atributo:valor;  
    atributo:valor2;  
}
```

Em que:

- **elemento:** Trata-se da descrição do elemento que será formatado. Neste caso, este elemento será uma tag HTML sem os sinais de menor e maior;
- **atributo:** Define qual será o atributo utilizado para a aplicação do estilo, desde que o nome seja de um atributo CSS válido, como o tamanho da fonte (font-size);
- **valor:** Refere-se ao valor específico do atributo definido, como 15 pt (ou seja, 15 pontos) para o atributo de tamanho de fonte (font-size).

8.4.Estilos CSS

Os estilos de uma CSS podem ser aplicados em uma linha ou cabeçalho da própria página HTML ou por meio de um arquivo externo, o que é mais indicado.

8.4.1.Declarando estilos internamente

As folhas de estilo podem ser definidas dentro do próprio documento que será formatado. Isso significa que os estilos associados a uma tag serão sempre aplicados de acordo com os mesmos parâmetros ao longo do arquivo. É importante lembrar que essas definições internas são prioritárias em comparação às externas, ou seja, elas serão utilizadas primeiramente, caso também existam vínculos para uma CSS externa.

8.4.2.Declarando estilos no cabeçalho

As definições das tags podem ser descritas no cabeçalho do documento que sofrerá as alterações. Todas as tags seguintes com o mesmo nome serão exibidas da mesma maneira, cada qual com os seus respectivos parâmetros. A descrição dos parâmetros também é feita sempre da mesma forma para todos os modos de vínculo, como podemos observar no exemplo a seguir:

```
<head>
  <style type="text/css" rel="StyleSheet">
    p {
      font-size : 20pt;
      font-family : "verdana";
      color : #FF0000;
      text-align : center;
    }
  </style>
</head>
```

De acordo com o exemplo anterior, a tag **<p>** terá fonte Verdana com tamanho de 20 pt, cor vermelha e alinhamento central.

8.4.3. Declarando estilos diretamente na tag

Uma folha de estilo é classificada como inline quando as regras são declaradas dentro das próprias tags de um elemento HTML. Esse método deve ser utilizado apenas para aplicar um estilo a um só elemento, pois o usuário poderá confundir facilmente o conteúdo da tag com as regras de formatação, já que ambos estão no mesmo local.

É importante mencionar que essa não é uma boa prática de desenvolvimento Web, uma vez que se o estilo for aplicado diretamente no elemento, o código fica engessado, logo este recurso deve ser utilizado com cautela e em casos de exceções.

A sintaxe utilizada para a aplicação da folha de estilo inline pode ser vista a seguir:

```
<elemento style="atributo: valor"> </elemento>
```

No seguinte exemplo, será formatado um determinado parágrafo de modo que sua cor fique vermelha e a margem tenha 3 px (pontos):

```
<p style="color:#FF0000; margin: 3px;">texto </p>
```

8.4.4. Declarando estilos externamente

A definição de uma folha de estilo por meio de um arquivo externo é a maneira mais recomendada para trabalhar com esse tipo de recurso. Para que seja possível o uso de arquivos externos, devemos criar vínculos com os documentos que serão formatados. Com isso, todos os documentos vinculados à folha de estilo serão modificados de acordo com os valores configurados nas tags.

Um arquivo de folha de estilo com a extensão `.css` pode ser criado em qualquer programa editor de textos. Sua estrutura pode ser vista a seguir:

```
elemento {  
    atributo1: valor;  
}
```

O próximo exemplo facilita a compreensão dessa estrutura:

```
body {  
    background-color: #FF0000;  
}  
  
p {  
    color: #FFFFFF;  
    font-size: 20pt;  
    text-align: center;  
}
```

De acordo com o exemplo apresentado, o parágrafo terá a fonte com tamanho de 20 pt, cor branca e alinhamento centralizado. O fundo da página, por sua vez, será vermelho.

Para vincular a folha de estilo ao documento HTML, devemos utilizar a tag `<link />`. Os parâmetros dessa tag incluem o caminho de localização do arquivo `.css` e a descrição do tipo de arquivo externo, como podemos observar no próximo exemplo:

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

Como é possível verificar no exemplo anterior, a tag `<link />` deve ser inserida no cabeçalho (**head**) do documento HTML. Graças à correta localização e descrição da folha de estilo, os padrões CSS poderão ser importados para aplicação nesse documento.

8.5.Cascadeamento

Um documento HTML5 pode ser vinculado a uma ou várias folhas de estilo. Caso mais de uma CSS esteja associada a esse documento, o browser será obrigado a decidir quais são os padrões mais importantes, uma vez que a mesma regra pode aparecer em várias folhas de estilo.

A prioridade entre definições de estilo atende às seguintes regras, descritas em ordem de importância:

- Os estilos aplicados caso não existam outros padrões que possam ser aplicados no seu lugar: são os estilos definidos por omissão;
- Os estilos definidos por meio de uma folha de estilo interna (ou seja, dentro do elemento `<style>`) ou por meio de um arquivo externo;
- Os estilos definidos nos elementos de um documento HTML por meio do atributo `<style>` em uma folha de estilo inline.

Como podemos notar, os estilos definidos através do atributo `<style>` no próprio elemento são prioritários em relação aos demais estilos. Caso o browser reconheça diferentes versões para o mesmo estilo, será utilizada somente a mais recente.

A ordem de maior utilização dos estilos pode ser categorizada de maneira simples:

- **Externo:** Arquivo `.css`;
- **Incorporado:** Através da tag `<style>` declarada no cabeçalho da página;
- **Inline:** Tag `<style>` declarada em uma determinada linha do código da página.

8.6.Inserindo comentários

Para facilitar a compreensão de um determinado código, é possível incluir comentários nas definições de estilo CSS. Tais comentários explicam o código que será escrito, de forma que outros usuários possam entender rapidamente sua utilização.

Um comentário deve ser incluído da seguinte forma: em primeiro lugar, devemos escrever a sequência de caracteres `/*`. Em seguida, é preciso incluir a descrição do comentário. Depois de incluir a descrição do comentário, devemos fechá-lo escrevendo a sequência de caracteres `*/`, como mostra o exemplo a seguir:

```
<style type="text/css">
p {
  /* isso é um comentário*/
  text-align: center
}
</style>
```

É importante utilizar comentários para documentação do código, entretanto em versões minificadas (utilizadas em ambiente de produção), os comentários e espaços são retirados.

8.7.Atributos utilizados para formatação

Os arquivos de folhas de estilo podem apresentar atributos associados a diferentes estilos de formatação, como aprenderemos a partir de agora.

8.7.1.Declaração de cores

Uma folha de estilo permite que sejam utilizados diferentes métodos para a especificação de cores:

- **Nome da cor:** Uma cor pode ser especificada por meio do seu nome, como **green**;
- **#valor:** Determina um valor padronizado da cor que será utilizada;
- **Rgb(vermelho, verde, azul):** Define os valores de vermelho, verde e azul para a composição da cor final, como **rgb (0,255,0)**;
- **rgb(vermelho%,verde%,azul%):** Define os valores de vermelho, verde e azul como porcentagens em relação ao valor máximo de cada cor. Por exemplo, **rgb(15%,40%,30%)**;
- **Nome resumido:** Caso seja uma cor segura, pode ser declarado em vez de **#CC6699** como **#C69**;
- **rgba(vermelho, verde, azul, alpha):** Define os valores de vermelho, verde e azul semelhante ao **rgb**, porém o último parâmetro **alpha**, determina a transparência da cor, em que **1.0** é 100% visível, **0.50** é 50% transparente e **0.20** é 80% transparente.

8.7.2.Formatando texto

Vários atributos podem ser utilizados para a formatação do texto de um elemento. Veja quais são esses atributos:

Atributos	Valor	Definição
color	Valor em formato hexadecimal o nome da cor.	Cor do texto.
font-family	Nome da fonte.	Tipo da fonte.
font-size	Valor referente ao tamanho da fonte ou porcentagem.	Tamanho da fonte.
font-style	normal, oblique ou italic.	Estilo do texto.
font-variant	normal ou small-caps.	Os caracteres são convertidos para maiúsculo.
font-weight	normal, bold, bolder, lighter, [100 -900].	Estilo negrito.
letter-spacing	normal ou pt.	Valor para o espaço disponível entre as letras.
text-align	left, right, center ou justify.	Ajuste do alinhamento do texto.
text-decoration	overline, underline, line-through ou blink.	Decoração do texto ou hiperlink.
text-indent	Porcentagem ou valor desejado para o comprimento.	Identação do texto.
text-transform	capitaliza, lowercase ou uppercase.	Conversão do texto em formato minúsculo ou maiúsculo ou apenas a primeira letra.
vertical-align	bottom, middle, sub, super, text-bottom, text-top e top	Alinhamento vertical do texto.

No exemplo a seguir, o documento **style.css** possui alguns dos atributos que foram descritos anteriormente:

```
<style type="text/css">
p{
    font-family: verdana;
    font-size: 20pt;
    font-variant: small-caps;
    font-style: italic;
    color: #f00;
    text-align: center;
    word-spacing: 20px
}
</style>
```

Já o arquivo **aula8_1.html** possui a seguinte estrutura:

```
1  <!doctype HTML>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title>HTML5 Fundamentos - Imagens</title>
6          <link type="text/css" href="css/style.css"rel="StyleSheet" />
7      </head>
8      <body>
9          <header>
10             <h1>Trabalhando com CSS</h1>
11          </header>
12          <main role="main">
13              <p>
14                  Texto formatado por folhas de estilo.
15              </p>
16          </main>
17      </body>
18  </html>
```

O resultado do código renderizado pode ser visto a seguir:

Trabalhando com CSS

TEXTO FORMATADO POR FOLHAS DE ESTILO.

8.7.3.Formatando o plano de fundo

Para que o plano de fundo de uma página seja formatado com as regras CSS, podemos utilizar vários atributos em conjunto com o comando background:

Atributos	Valor	Definição
background-attachment	fixed, scroll ou local.	Define se a imagem de fundo permanecerá fixa enquanto a barra de rolagem é deslizada (fixed) ou se ela acompanhará esse movimento (scroll).
background-color	transparent, valor em formato hexadecimal, rgb , rgba ou nome da cor.	Determina a cor de fundo de um elemento.
background-image	url, linear-gradient	Uma imagem que será utilizada como plano de fundo. Podemos utilizar a funcionalidade linear-gradient que permite criar degradê utilizando apenas CSS3.
background-repeat	repeat-x, repeat-y, repeat, space, round, no-repeat.	Especifica como a imagem de fundo deve se comportar caso seja maior que a área do elemento em que foi aplicada.
background-size	Valor em pixels, porcentagem ou auto.	Especifica o tamanho das imagens de fundo.
background-position	center, [left right top bottom] [%]	Caso a propriedade background-image seja especificada, podemos determinar a localização da imagem e também o seu comportamento quando a janela for redimensionada.

A seguir veja um exemplo do arquivo **style.css**:

```
1
2  .btn {
3
4  color:#fff;
5  border-radius:4px;
6  border:1px solid #980021;
7  box-shadow: inset 0 2px 3px 0 rgba(255,255,255,.3),
8  inset 0 -3px 6px 0 rgba(0,0,0,.2),
9  0 3px 2px 0 rgba(0,0,0,.2);
10 cursor: pointer;
11 }
12
13 .btn-large{
14     width:180px;
15     padding: 10px 10px;
16     height:50px;
17     font-size:20px;
18     text-align:center;
19 }
20
21 .btn-primary{
22     background-image:
23     linear-gradient(to bottom, #E44D3A, #A7372B 130%);
24 }
```

E com o código HTML carregando os estilos e criando um botão para aplicar os estilos criados:

```
1  <!doctype HTML>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title>HTML5 Fundamentos - Imagens</title>
6          <link type="text/css" href="css/style.css"rel="StyleSheet" />
7      </head>
8      <body>
9          <header>
10             <h1>Trabalhando com CSS</h1>
11          </header>
12          <main role="main">
13              <p>
14                  <button class="btn btn-large btn-primary" id="btn-envio">
15                      Enviar
16                  </button>
17              </p>
18          </main>
19      </body>
20  </html>
```

Após a renderização, o código anterior produzirá o seguinte resultado:

Trabalhando com CSS



8.7.4.Formatando bordas

As bordas podem ser ajustadas de acordo com uma série de atributos, como podemos observar na tabela a seguir:

Atributos	Valor	Definição
border-bottom	border-bottom-width, border-color e border-style.	Todas as propriedades da linha de limite inferior da borda podem ser escritas em uma só declaração.
border-color	Cor	As cores das quatro linhas de limite são escolhidas.
border-style	none, hidden, dotted, dashed, solid, double, groove, ridge, inset ou outset.	O estilo das quatro linhas de limite da borda pode ser ajustado com um a quatro valores.
border-top	border-color, border-style e border-top-width.	Todas as propriedades da linha de limite superior são definidas por meio de uma só declaração.
border-radius	[px %] [px %]	Determina o grau da curva das margens de um objeto.

Veja um exemplo com o uso da formatação de bordas:

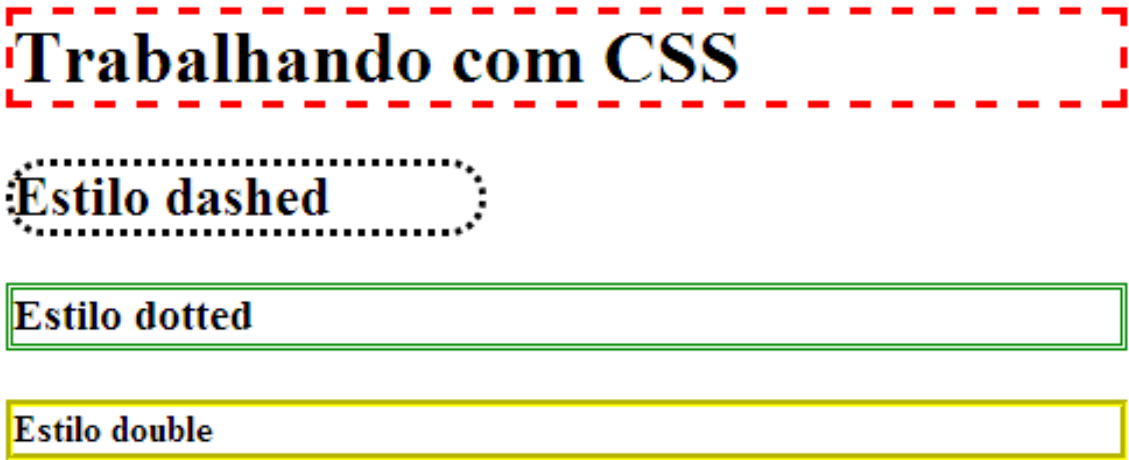
```
.borda-arredondada{
    border-radius: 25px;
    width:200px;
}

h1 {
    border-style: dashed;
    border-color: red
}
h2 {
    border-style: dotted;
    border-color: black
}
h3 {
    border-style: double;
    border-color: green
}
h4 {
    border-style: groove;
    border-color: yellow
}
```

O arquivo **aula8_3.html** com o conteúdo do documento HTML:

```
1  <!doctype HTML>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title>HTML5 Fundamentos - Imagens</title>
6          <link type="text/css" href="css/style.css"rel="StyleSheet" />
7      </head>
8      <body>
9          <header>
10             <h1>Trabalhando com CSS</h1>
11          </header>
12          <main role="main">
13              <p>
14                  <h2 class="borda-arredondada">Estilo dashed</h1>
15                  <h3> Estilo dotted</h2>
16                  <h4> Estilo double</h3>
17              </p>
18          </main>
19      </body>
20  </html>
```

Confira o resultado criado no navegador:



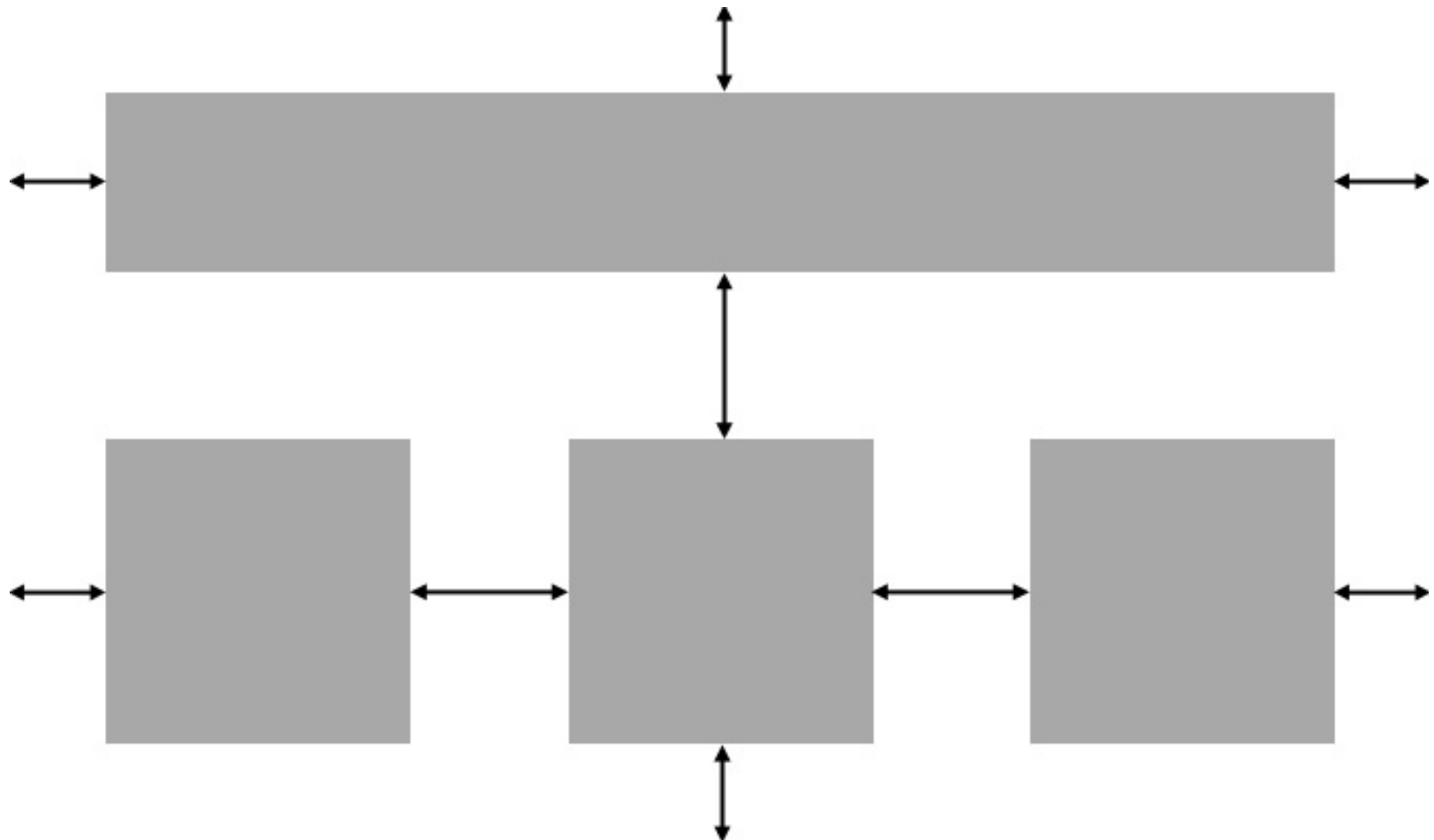
Observe no exemplo acima que o elemento h1 está chamando a classe **borda-arredondada**, criando assim um efeito com cantos arredondados sem o uso de imagens, o que não era possível em versões anteriores ao HTML5.

8.7.5.Formatando espaçamento entre conteúdo, margem e bordas.

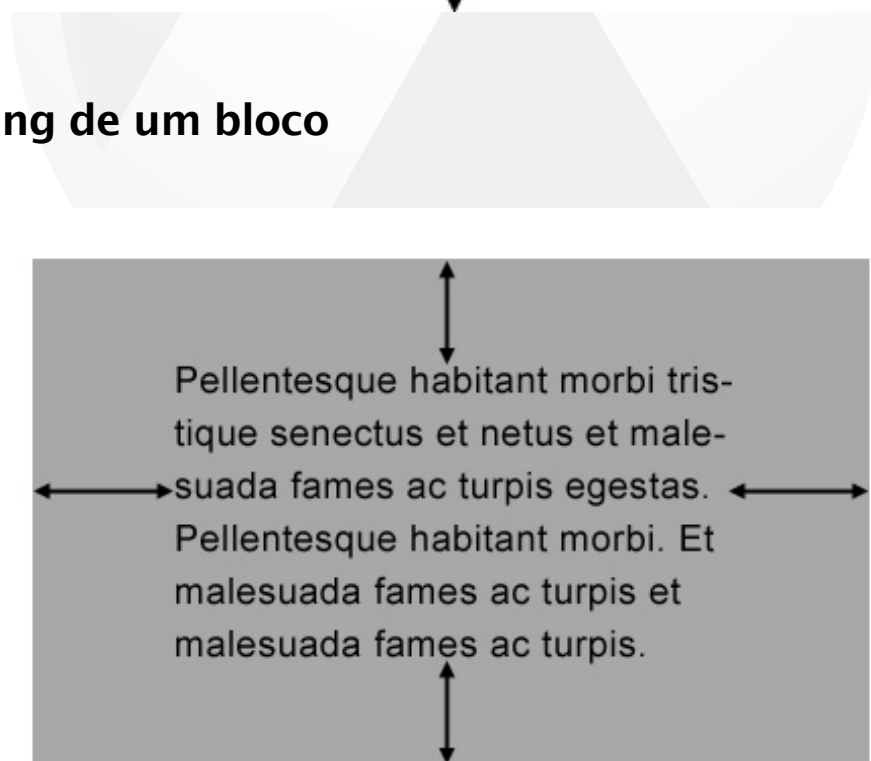
O espaço existente entre as bordas e o conteúdo dos elementos de um documento HTML pode ser ajustado conforme os seguintes atributos:

Atributos	Valor	Definição
padding-bottom	Porcentagem ou valor desejado para o comprimento.	O espaço inferior entre o conteúdo e as bordas é configurado.
padding-left	Porcentagem ou valor desejado para o comprimento.	O espaço do lado esquerdo entre o conteúdo e as bordas é configurado.
padding-right	Porcentagem ou valor desejado para o comprimento.	O espaço do lado direito entre o conteúdo e as bordas é configurado.
padding-top	Porcentagem ou valor desejado para o comprimento.	O espaço superior entre o conteúdo e as bordas é configurado.

- **Definição das margens de um objeto: margens a partir do topo, da base, da esquerda e da direita**



- **Definição do padding de um bloco**



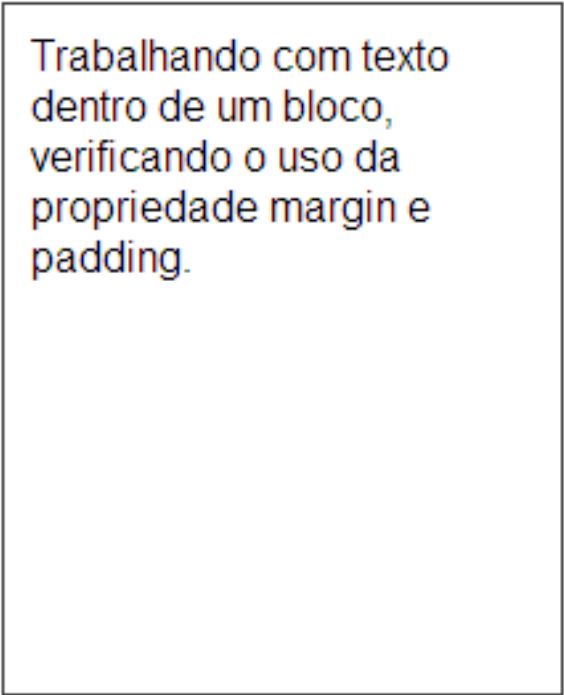
Criamos o estilo em um arquivo **style.css**:

```
.bloco{  
    width:200px;  
    height: 250px;  
    border:solid 1px #333;  
    font-family: arial;  
    margin:10px 10px;  
    padding-left:10px;  
    padding-top: 10px;  
}
```

Criamos o conteúdo HTML no arquivo **aula8_4.html**, definindo uma **section** com uma chamada para a classe **bloco**:

```
1  <!doctype HTML>  
2  <html>  
3      <head>  
4          <meta charset="utf-8" />  
5          <title>HTML5 Fundamentos - CSS</title>  
6          <link type="text/css" href="css/style.css"rel="StyleSheet" />  
7      </head>  
8      <body>  
9          <main role="main">  
10             <p>  
11                 <section class="bloco">  
12                     Trabalhando com texto dentro de  
13                     um bloco, verificando o uso da propriedade  
14                     margin e padding.  
15                 </section>  
16             </p>  
17         </main>  
18     </body>  
19 </html>
```

Então temos como resultado um bloco definindo as margens externas (**margin**) e internas (**padding**):

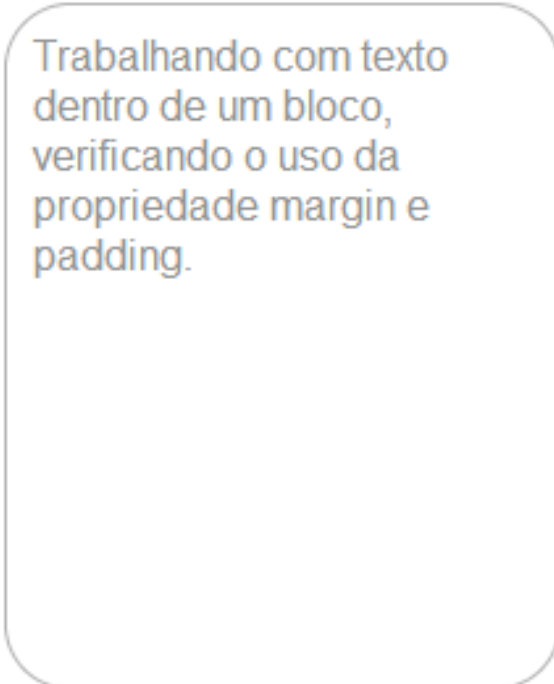


Trabalhando com texto
dentro de um bloco,
verificando o uso da
propriedade margin e
padding.

Com o clareamento da fonte e a adição de cantos arredondados (**border-radius**), temos um efeito interessante:

```
.bloco{  
  width:200px;  
  height: 250px;  
  border:solid 1px #aaa;  
  color:#888;  
  font-family: arial;  
  margin:10px 10px;  
  padding-left:10px;  
  padding-top: 10px;  
  border-radius: 25px;  
}
```

E o resultado renderizado será o seguinte:



Trabalhando com texto
dentro de um bloco,
verificando o uso da
propriedade margin e
padding.

As margens de um documento HTML também podem ser formatadas com a ajuda das folhas de estilo. Para isso, será necessário utilizar o atributo **margin**, como podemos observar no exemplo anterior.

8.7.6.Formatando links

Os links inseridos em um documento HTML também podem ser formatados. Para isso, algumas regras de estilo devem ser declaradas para as pseudoclasses do elemento `<a>` do HTML. As pseudoclasses permitem acrescentar efeitos a uma instância dos seletores ou aos próprios seletores.

As pseudoclasses dos links podem ser divididas em:

- **a: hover**: Define a formatação do link no momento em que passamos o mouse sobre ele;
- **a: link**: Define a formatação do link antes dele ser visitado;
- **a: visited**: Define a formatação do link depois que ele é visitado;
- **a: active**: Link ativo.

A sintaxe utilizada para a formatação de links é a seguinte:

```
seletor:pseudo-classe {propriedade: valor}
```

De acordo com o padrão, os links aparecem na cor azul e sublinhados. Para que ambas as características sejam exibidas apenas no momento em que passamos o mouse sobre o link, devemos utilizar o atributo **text-decoration**.

A seguir, podemos visualizar um exemplo desse tipo de formatação, no arquivo **style.css**:

```
a:link {text-decoration: none; color: red}
a:visited {text-decoration: none; color: green}
a:hover {text-decoration: underline; color: blue}
```

A estrutura HTML será utilizada da seguinte maneira:

```
1  <!doctype HTML>
2  <html>
3  <head>
4      <meta charset="utf-8" />
5      <title>HTML5 Fundamentos - CSS</title>
6      <link type="text/css" href="css/style.css"rel="StyleSheet" />
7  </head>
8  <body>
9      <main role="main">
10         <p>
11             <a href="http://www.impacta.com.br">
12                 Impacta Certificação e Treinamento
13             </a>
14         </p>
15     </main>
16 </body>
17 </html>
```

Confira o resultado renderizado:

- **Link ainda não visitado**

Impacta Certificação e Treinamento

- **Link já visitado**

[Impacta Certificação e Treinamento](#)

- **Link ao passar o mouse**

[Impacta Certificação e Treinamento](#)

8.8.Seletores

Os seletores permitem não só a definição de diferentes formatações em um mesmo elemento HTML, mas também a criação de estilos personalizados. Ambos os procedimentos serão vistos com detalhes a partir de agora.

8.8.1.Seletores de classe

Por meio dos seletores de classe, é possível configurar diferentes estilos no mesmo elemento de um documento HTML. Por exemplo, suponhamos que vários tipos de alinhamento de parágrafo serão definidos: um ao centro, outro à esquerda e um último à direita. Vejamos como os seletores de classe podem ser aplicados para que essa formatação seja realizada.

No primeiro exemplo, temos a seguinte estrutura no documento CSS:

```
.p1 {  
  font-family: verdana;  
  font-size: 14px;  
  color: red;  
  text-align: center  
}  
.p2 {  
  font-family: arial;  
  font-size: 14px;  
  color: blue;  
  text-align: left  
}  
.p3 {  
  font-family: courier;  
  font-size: 14px;  
  color: black;  
  text-align: right  
}
```

O **arquivo.html** apresentará a seguinte estrutura:

```
1  <!doctype HTML>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title>HTML5 Fundamentos - CSS</title>
6          <link type="text/css" href="css/style.css"rel="StyleSheet" />
7      </head>
8      <body>
9          <main role="main">
10             <p class="p1"> PARAGRAFO class=p1 </p>
11             <p class="p2"> PARAGRAFO class=p2 </p>
12             <p class="p3"> PARAGRAFO class=p3 </p>
13
14          </main>
15      </body>
16  </html>
```

E o resultado renderizado será apresentado da seguinte forma:

PARAGRAFO class=p1

PARAGRAFO class=p2

PARAGRAFO class=p3

8.8.2. Seletores de id

Ao contrário do seletor de classe, o seletor **id** é aplicado a um único elemento, pois os valores do atributo **id** não podem aparecer mais de uma vez em uma página. Isso significa que a quantidade de elementos que possuem **id** em uma página sempre será zero ou um. Conforme a regra do próximo exemplo, somente um elemento `<p>` com o valor **p1** no atributo **id** poderá ser aplicado pelo estilo.

No documento CSS, temos:

```
#p1 {  
  font-family: verdana;  
  font-size: 14px;  
  color: red;  
  text-align: center  
}  
#p2 {  
  font-family: arial;  
  font-size: 14px;  
  color: blue;  
  text-align: left  
}
```

Em seguida, a estrutura do arquivo **aula8_7.html** será:

```
1  <!doctype HTML>  
2  <html>  
3    <head>  
4      <meta charset="utf-8" />  
5      <title>HTML5 Fundamentos - CSS</title>  
6      <link type="text/css" href="css/style.css" rel="StyleSheet" />  
7    </head>  
8    <body>  
9      <main role="main">  
10        <p id="p1"> PARAGRAFO ID=p1 </p>  
11        <p id="p2"> PARAGRAFO ID=p2 </p>  
12      </main>  
13    </body>
```

PARAGRAFO ID=p1

PARAGRAFO ID=p2

