

CSS



Artist Image :Descourtilz, Jean-Théodore | Dates:179?-1855

Guia iniciante:

CSS

# ABOUT ME



*Brunna Croches*

***Developer Full Stack***

Brunna Croches é Dev FullStack, advogada e empreendedora.

Apaixonada por tech, vem adquirindo vasto conhecimento na área.

Desenvolveu projetos ricos em diversidade, buscando captar as próximas tendências e necessidades do mercado.

Neste e-book você aprenderá ou recapitulará de forma simplificada e otimizados conceitos de programação feito por ela.

*let's share*

# SUMMARY

---

## VISIBILIDADE DE OPACIDADE

**09**

*Exibindo ou ocultando um elemento em uma página e adicionar ou remover um elemento.*

## BACKGROUND

**10**

*As variadas formas de posicionar o background.*

## SOMBRA

**11**

*Teoria e prática de sombreamento usando a text-shadow e box-shadow.*

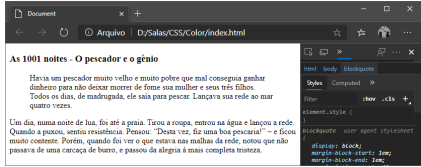
## POSICIONAMENTO

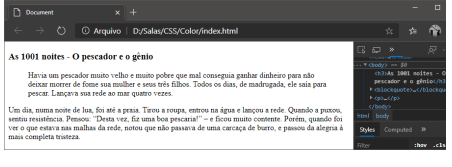
**12**

*Como controlar onde cada elemento em uma página vai ficar*

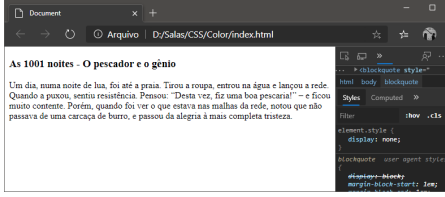
# 9- Visibilidade e Opacidade

<b>VISIBILITY</b>	Visibility permite <b>exibir/ocultar</b> um elemento em uma página.
<b>DISPLAY</b>	Display permite <b>adicionar ou remover</b> um elemento do layout, tornando ele visível ou não.

<b>VISIBILITY</b>	<p>Visibility permite <b>exibir/ocultar</b> um elemento em uma página. Elemento invisível na página Ela pode receber dois valores:</p> <ul style="list-style-type: none"> <li>• visible</li> <li>• hidden</li> </ul>		
<b>VISIBLE</b>	<p>O elemento fica <b>visível na página</b>. Esse é o valor padrão desta propriedade para a maioria dos elementos.</p>	<pre>blockquote {   visibility: visible; }</pre>	
<b>HIDDEN</b>	<p>O elemento <b>se torna invisível</b>, mas continua ocupando o espaço na página.</p>	<pre>blockquote {   visibility: hidden; }</pre>	

<b>DISPLAY</b>	<p>Visibility controla a visibilidade do elemento, mas não o remove do layout. Para <b>remover completamente um elemento do layout, como se ele não estivesse na página, usamos a propriedade display com o valor none.</b></p>			
<b>ELEMENTOS INLINE</b>	<p>O elemento <b>não inicia em uma nova linha e ocupa apenas o espaço necessário para apresentar o seu conteúdo.</b> <b>Pode fazer com que um elemento de bloco (block) se comporte como um elemento de linha (inline).</b></p>	<pre>&lt;a&gt; &lt;abbr&gt; &lt;acronym&gt; &lt;b&gt; &lt;bdo&gt; &lt;big&gt; &lt;button&gt; &lt;cite&gt; &lt;code&gt; &lt;dfn&gt; &lt;em&gt; &lt;i&gt; &lt;img&gt; &lt;input&gt; &lt;kbd&gt; &lt;label&gt; &lt;map&gt; &lt;object&gt; &lt;output&gt; &lt;q&gt; &lt;samp&gt; &lt;script&gt; &lt;select&gt; &lt;small&gt; &lt;span&gt; &lt;strong&gt; &lt;sub&gt; &lt;sup&gt; &lt;textarea&gt; &lt;time&gt; &lt;tt&gt; &lt;var&gt;</pre>	<pre>h3, blockquote, p {   display: inline; }</pre>	
<b>ELEMENTOS INLINE</b>	<p>O elemento inicia em uma nova linha e vai até o final dela. Pode fazer com que um elemento de linha (inline) se</p>	<pre>&lt;address&gt; &lt;article&gt; &lt;aside&gt; &lt;blockquote&gt; &lt;canvas&gt; &lt;dd&gt; &lt;div&gt; &lt;dl&gt; &lt;dt&gt; &lt;fieldset&gt;</pre>		



<b>ELEMENTOS BLOCK</b>	comporte como um elemento de bloco (block).	<pre>&lt;figcaption&gt; &lt;figure&gt; &lt;footer&gt; &lt;form&gt; &lt;h1&gt; -&lt;h6&gt; &lt;header&gt; &lt;hr&gt; &lt;li&gt; &lt;main&gt; &lt;nav&gt; &lt;noscript&gt; &lt;ol&gt; &lt;p&gt; &lt;pre&gt; &lt;section&gt; &lt;table&gt; &lt;tfoot&gt; &lt;ul&gt; &lt;video&gt;</pre>		
<b>DISPLAY BLOCK</b>	<p>None é um valor especial de display que oculta o elemento e causa a sensação de que ele foi removido página.</p> <p>Display none causa a sensação de que um elemento foi removido da página</p>	<pre>blockquote { display: none; }</pre>		

### display none vs visibility hidden

Usar display none causa um efeito diferente de visibility hidden.

```
blockquote {
visibility: hidden;
}
```

### Código 2. Elementos visibility hidden

Visibility hidden faz com que o elemento fique oculto, mas algo ainda parece estar no lugar dele na página.

# 10 -Background

Com background podemos definir a cor de fundo, imagem, o posicionamento dela, se deve se repetir, entre outras opções, agregando um diferencial ainda maior para o visual das nossas aplicações.

<b>Background</b>	A principal propriedade para manipulação do plano de fundo de uma página ou dos elementos nela declarados é a <b>background</b> . Com ela podemos manipular o plano de fundo utilizando uma sintaxe abreviada e que nos permite definir todas opções a ela relacionadas de forma mais rápida e em apenas uma linha de código.	<code>background: #FFFFFF url("pasta/Imagem.jpg") no-repeat right top fixed;</code>	No código ao lado, configuramos a <b>cor de fundo</b> , a <b>imagem a ser utilizada</b> , <b>se ela deve ser repetida para ocupar todo o fundo do elemento</b> , as posições horizontal e vertical e a forma como a imagem será anexada ao documento.
<b>Background-Color</b>	Utilizamos essa propriedade para alterar a cor do plano de fundo, seja de um elemento ou do documento como um todo. Para isso, basta atribuir a cor desejada à propriedade <b>background-color</b> , declarada dentro do seletor no qual ela deve ser aplicada.	<code>/* Plano de fundo em vermelho e opacidade de 50% */ background-color: rgba(255,0,0,0.5);</code>	
<b>Background-Image</b>	Esta propriedade é utilizada para inserir imagens como plano de fundo de um elemento. Para isso, fazemos uso da função <b>url()</b> , para a qual passamos como parâmetro o caminho da imagem a ser inserida.	<code>background-image: url("pasta/Imagem.jpg");</code>	
<b>Background-Repeat</b>	Utilizamos esta propriedade para definir a maneira como a imagem será repetida no elemento. Podemos especificar que a imagem se repetirá em todos os eixos, apenas no horizontal, apenas no vertical ou não se repetirá.	<code>background-repeat: no-repeat;</code>	<b>Opções para background-repeat</b> A tabela a seguir apresenta todos os valores que podemos atribuir a background-repeat. <b>repeat-y</b> Repete-se na vertical. <b>repeat-x</b> Repete-se na horizontal. <b>no-repeat</b> Não se repete. <b>repeat</b> Valor padrão, repete-se nas duas direções.
<b>Background-Multiplo</b>	Nossos documentos podem conter diversas imagens de fundo. Para isso, basta utilizar a propriedade <b>background-image</b> e incluir, separadas por vírgula, as URLs de cada imagem. A ordem de exibição das imagens é a mesma em que elas foram declaradas. Dessa forma, a primeira imagem será inserida antes da segunda e assim sucessivamente.	<code>background-image: url("pasta/Imagem1.jpg"), url("pasta/Imagem2.jpg");</code>	Caso desejemos customizar a exibição de cada imagem individualmente, devemos declarar a propriedade seguida dos valores separados por vírgula. Exemplo de uso com customização de múltiplas imagens: <code>background-image:  url("pasta/Imagem1.jpg"), url("pasta/Imagem2.jpg"); background-repeat: no-repeat, repeat-x; background-position: top right, center;</code>
<b>Background-Clip</b>			

## Position

<b>Background-Position</b>	Com esta propriedade podemos definir a <b>posição na qual a imagem será inserida. Essa posição pode ser especificada utilizando uma combinação de valores</b> , sendo o primeiro a <b>posição horizontal</b> e o segundo a <b>posição vertical</b> .	background-position: right top;
<b>Opções para background-position</b> Posição horizontal	<b>right</b> Sinaliza que a imagem será renderizadas à <b>direita</b> . <b>left</b> Sinaliza que a imagem será renderizadas à <b>esquerda</b> .	<b>center</b> Sinaliza que a imagem será renderizadas no <b>centro</b> .
<b>Opções para background-position</b> Posição vertical	<b>top</b> Sinaliza que a imagem será renderizadas no <b>topo</b> . <b>bottom</b> Sinaliza que a imagem será renderizadas na parte de <b>baixo</b> .	<b>center</b> Sinaliza que a imagem será renderizadas no <b>centro</b> .
<b>Opções para background-position</b> Posicionamento de uma imagem com mais precisão	Também podemos <b>especificar o posicionamento de uma imagem com mais precisão</b> , atribuindo à background-position valores em porcentagem, pixels ou outra unidade de medida das CSS. Além disso, podemos <b>configurar os eixos com diferentes unidades de medida</b> , como informar que a <b>imagem ficará a 20 pixels de distância da margem esquerda e 40% em relação ao topo da página</b> .	background-position: 20% 40%; /* 20% à direita e 40% abaixo */ background-position: 30px 60px; /* 30px à direita e 60px abaixo */ background-position: 20px 40%; /* 20px à direita e 40% abaixo */

## Attachment

Esta propriedade é utilizada para definir o que acontecerá com a imagem de fundo quando da interação com a barra de rolagem, **se ela ficará fixa, se ela se moverá junto com o movimento de scroll**, entre outras opções.

<b>Background-Attachment</b>	Esta propriedade é utilizada para definir o que acontecerá com a imagem de fundo quando da interação com a barra de rolagem, <b>se ela ficará fixa, se ela se moverá junto com o movimento de scroll</b> , entre outras opções.	background-attachment: fixed;
<b>Opções para background-attachment</b> <b>SCROLL</b>	Valor padrão assumido pela imagem de fundo. Define que ao <b>rolar</b> a página a imagem também irá rolar.	
<b>Opções para background-attachment</b> <b>FIXED</b>	Faz com que a imagem não se movimente junto com o documento, dando a impressão de que o conteúdo desliza sobre o fundo.	
<b>Opções para background-</b>	Especifica que a imagem de fundo de um documento interno vai se movimentar	

attachment <b>LOCAL</b>	junto com o fundo do documento principal.	
Opções para background-attachment <b>INHERIT</b>	Especifica que a propriedade recebe, por herança, a mesma configuração indicada para o elemento pai.	
Opções para background-attachment <b>INITIAL</b>	Indica à propriedade que o valor padrão deve ser utilizado.	

## Background Size

Background-Size <b>PIXELS</b>	Utilizamos esta propriedade para <b>especificar o tamanho da imagem de fundo</b> . Assim como acontece com background-position, existem diversas formas de declarar esse recurso. Uma delas é <b>definir os valores em pixels, sendo o primeiro valor para a largura e o segundo para a altura</b> .	<b>background-size:</b> 80px 60px; /* 80 pixels de largura e 60 pixels de altura */ <b>background-size:</b> 300px; /* 300 pixels de largura e altura automática */  <b>Nota:</b> Se informarmos apenas um valor, ele será atribuído à largura da imagem, enquanto a altura mudará de forma automática e proporcional.
Background-Size <b>PORCENTAGEM</b>	Podemos utilizar ainda a <b>porcentagem da página ou do elemento para definir a largura que a imagem deve ocupar</b> . Nesse caso a altura também assume valor automático proporcional à largura.	background-size: 50%;
Background-Size (valores pre definidos) <b>COVER</b>	<b>Amplia a imagem até o ponto que preencha toda a área da página ou elemento</b> .	
Background-Size (valores pre definidos) <b>CONTAIN</b>	<b>Amplia a imagem ao maior tamanho possível respeitando a área visível da página ou elemento</b> .	
Background-Size (valores pre definidos) <b>INHERIT</b>	<b>Herda a mesma configuração definida no elemento pai</b> .	
Background-Size (valores pre definidos) <b>INITIAL</b>	Utiliza o <b>valor padrão da propriedade</b> .	



## Background Origin

<b>Background- Origin</b>	Este recurso permite <b>escolher a posição em que a imagem será inserida</b> . Diferentemente da propriedade <b>background-position</b> , ela <b>configura o ponto de inserção em relação aos elementos do documento</b> .	background-origin: padding-box;
<b>Background- Origin PADDING BOX</b>	Define que a <b>renderização da imagem inicia-se junto ao padding</b> .	
<b>Background- Origin BORDER BOX</b>	Define que a <b>renderização da imagem inicia-se junto à borda</b> .	
<b>Background- Origin CONTENT BOX</b>	Define que a <b>renderização da imagem inicia-se junto ao conteúdo do elemento</b> .	

## Background-Clip

<b>Background-clip</b>	Esta propriedade é usada para selecionar a área do background que será pintada. Assim, podemos especificar que desejamos colorir apenas a borda, o padding ou o conteúdo.	background-clip: border-box;
<b>Background-clip BORDER BOX</b>	Preenche toda a área delimitada pelas bordas.	
<b>Background-clip PADDING BOX</b>	Preenche toda a área delimitada pela propriedade padding.	
<b>Background-clip CONTENT- BOX</b>	Preenche toda a área de conteúdo.	

## Background-blend-mode

	Esta propriedade nos permite <b>atribuir efeitos pré-definidos às imagens e cores de fundo</b> ,
--	--

Background- blend-mode	atribuindo a elas mais ou menos <b>luminosidade, saturação, brilho, entre outras</b> opções.
Background- blend-mode	background-blend-mode: darken;
Background- blend-mode	Mantém as <b>características normais do plano de fundo.</b>
<b>NORMAL</b>	
Background- blend-mode	Multiplica o valor da cor de um pixel, <b>deixando-o mais escuro.</b> Pixels de cor preta ou branca não serão alterados.
<b>MULTIPLY</b>	
Background- blend-mode	Funciona de forma inversa ao <b>multiply, deixando as cores mais claras.</b>
<b>SCREEN</b>	
Background- blend-mode	<b>Escurece os pixels que já são escuros e clareia os pixels que já são claros;</b>
<b>OVERLAY</b>	
Background- blend-mode	<b>Compara o objeto de cima com o de baixo e mantém o pixel que for mais escuro,</b> logo, <b>substitui os mais claros, escurecendo o conteúdo presente no elemento.</b>
<b>DARKEN</b>	
Background- blend-mode	Possui comportamento inverso ao do Darken. <b>Compara os dois objeto e mantém os pixels mais claros,</b> tornando mais claro o conteúdo presente no elemento.
<b>LIGHTEN</b>	
Background- blend-mode	Clareia os pixels e aumenta o contraste. <b>Costuma ser usado para criar efeitos brilhosos;</b>
<b>COLOR-DODGE</b>	
Background- blend-mode	<b>Reproduz a saturação do objeto de cima no objeto de baixo, mas o objeto de baixo continua com sua cor e brilho originais.</b>
<b>SATURATION</b>	
<b>N</b>	
Background- blend-mode	Reproduz a saturação e a cor do <b>objeto de cima no objeto de baixo,</b> mas o objeto de baixo continua com o seu <b>brilho original</b>
<b>COLOR</b>	
Background- blend-mode	<b>Reproduz o brilho do objeto de cima no objeto de baixo,</b> mas <b>o objeto de baixo</b>

## Exemplo prático

Para demonstrar as propriedades relacionadas à background na prática, criamos a seguinte folha de estilos.

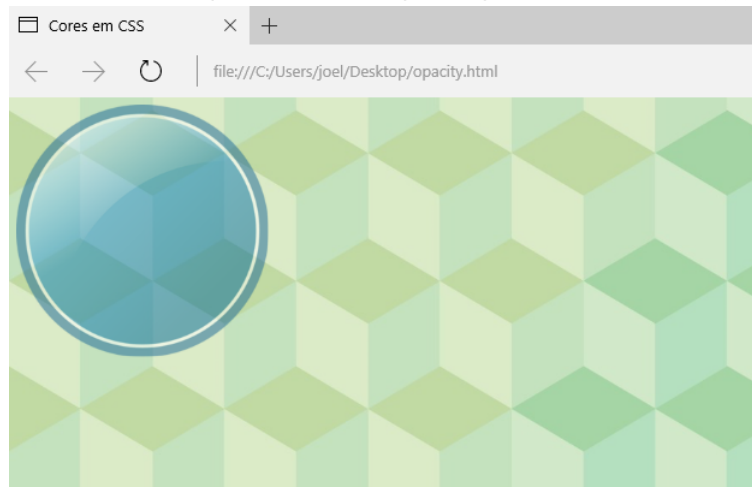
Exemplo de uso:

```
01 body {
02     background-image: url("img/imagem1.jpg"), url("img/imagem2.jpg");
03     background-repeat: no-repeat, repeat;
04     background-attachment: initial, fixed;
05     background-blend-mode: overlay;
06 }
```

## Visibilidade e Opacidade

<h3>Alterar a Opacidade</h3>	<p>A propriedade opacity do <b>CSS</b> nos permite <b>alterar a opacidade de um elemento, deixando-o mais ou menos transparente.</b></p> <p>Para <b>alterar a opacidade de um elemento, basta atribuir um valor entre 0 e 1 à propriedade, opacity.</b></p> <p>Neste caso, <b>o valor 0 indica um elemento completamente transparente, enquanto o valor 1 torna o elemento totalmente opaco.</b></p>	<p>opacity: [valor entre 0 e 1]</p>
<h3>Alterar a Opacidade EXEMPLO</h3>	<p>No exemplo podemos ver uma imagem cuja opacidade foi alterada para 50% (0.5), ou seja, ela está com 50% de transparência</p> <p>&gt;&gt;&gt;&gt;</p> <p>Linha 7: aqui estamos <b>apenas atribuindo um plano de fundo ao documento</b>, para que seja possível perceber que a imagem está transparente;</p> <p>Linha 8: nesta linha estamos, de fato, <b>modificando a opacidade da imagem, atribuindo a ela o valor 0.5, que equivale a 50%.</b></p>	<pre>&lt;!DOCTYPE html&gt; &lt;html lang="pt-br"&gt; &lt;head&gt;   &lt;meta charset="UTF-8"&gt;   &lt;title&gt;Cores em   CSS&lt;/title&gt;   &lt;style&gt;     body{ background-   image: url('bg.jpg'); }     img { opacity : 0.5 }   &lt;/style&gt; &lt;/head&gt; &lt;body&gt;   &lt;img src="circulo.png"   alt="Imagem"&gt; &lt;/body&gt; &lt;/html&gt;</pre>


O resultado desse código pode ser visto na figura a seguir. Note que é possível ver, através da imagem, o conteúdo que está atrás dela.



## 11 - Sombra

<b>Sombra</b> Sintaxe	Com a versão mais recente das CSS temos à disposição as propriedades <b>text-shadow</b> e <b>box-shadow</b> , que permitem aplicar sombras a textos e blocos.	Ambas as propriedades, text-shadow e box-shadow, podem ser configurados de três formas diferentes, dependendo da necessidade
<b>Sombra</b> Sintaxe	01 text-shadow: X Y; 02 box-shadow: X Y; 03 04 text-shadow: X Y Blur; 05 box-shadow: X Y Blur; 06 07 text-shadow: X Y Blur Cor; 08 box-shadow: X Y Blur Cor;	Linhas 1 e 2: Ao usar as propriedades dessa forma, os valores X e Y representam, respectivamente, o deslocamento da sombra para a direita em relação à lateral esquerda, e para baixo em relação ao topo do elemento. Valores negativos, neste caso, fazem com que a sombra seja deslocada para a esquerda, ou para cima.  Linhas 4 e 5: Nessa segunda forma, podemos definir um valor para o efeito de blur (borrão ou desfoque). Quanto maior esse valor, mais borrada/espalhada será a sombra.  Linhas 7 e 8: Além das coordenadas e do blur, podemos ainda definir a cor da sombra. Caso não se deseje aplicar o efeito blur, pode-se simplesmente omitir esse parâmetro e informar apenas a cor.

## 3 Exemplos do uso da Box-Shadow

<b>Box-Shadow</b>	<pre>01 &lt;!DOCTYPE html&gt; 02 &lt;html&gt; 03 &lt;head&gt; 04 &lt;style&gt; 05   div { width: 100px; 06         height: 100px; 07         border: 1px solid black; 08         float: left; 09         margin-right: 100px; } 10   #box1 { box-shadow: 20px 20px 11           } 12   #box2 { box-shadow: 20px 20px 13           10px 14           } 15   #box3 { box-shadow: 20px 20px 16           10px blue 17           } 18 &lt;/style&gt; 19 &lt;/head&gt; 20 &lt;body&gt; 21   &lt;div id="box1"&gt;&lt;/div&gt; 22   &lt;div id="box2"&gt;&lt;/div&gt; 23   &lt;div id="box3"&gt;&lt;/div&gt; 24 &lt;/body&gt; 25 &lt;/html&gt;</pre>	Linhas 5 a 9: aplicamos uma configuração básica às três divs, apenas para deixá-las visíveis lado a lado e facilitar sua observação;  Linhas 10 a 12: configuramos a sombra das divs de três formas distintas.	
-------------------	--	--	---

## Exemplos do uso da Tex-Shadow



## Text-Shadow

```
01 <!DOCTYPE html>
02 <html>
03 <head>
04 <style>
05 #titulo1 { text-shadow: 10px
10px;
06 #titulo2 { text-shadow:
10px 10px 5px;
07 #titulo3 { text-shadow:
10px 10px 5px blue;
08 </style>
09 </head>
10 <body>
11 <h1 id="titulo1">Título 1</h1>
12 <h1 id="titulo2">Título 2</h1>
13 <h1 id="titulo3">Título 3</h1>
14 </body>
15 </html>
```

Título 1  
Título 2  
Título 3

Observe que no primeiro caso o resultado final não parece muito eficiente, pois a sombra possui a mesma cor do texto e não está borrada, atrapalhando assim a visualização de ambos.

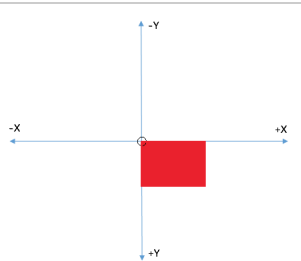
## Funcionamento

### Funcionament

o

das propriedades text-shadow e box-shadow

é importante observar que o **eixo X (horizontal) é crescente para a direita, enquanto o eixo Y (vertical) é crescente para baixo**. Esse comportamento é um pouco diferente do Plano Cartesiano convencional, em que o eixo Y cresce para cima, mas a lógica de funcionamento é a mesma.

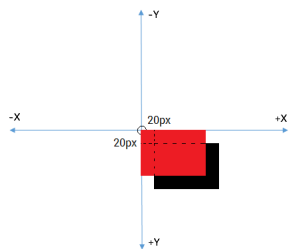


### Funcionament

o

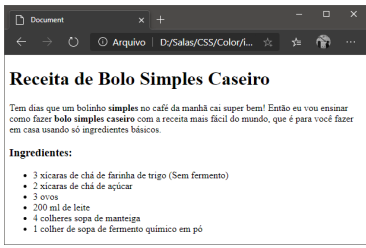
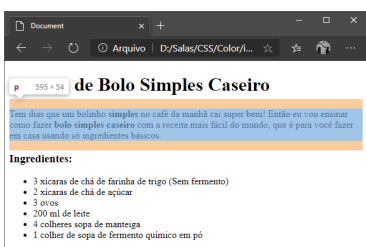
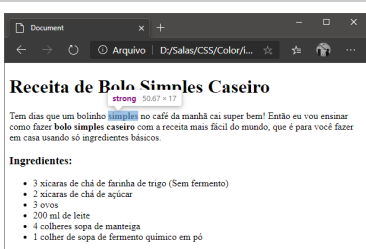
das propriedades text-shadow e box-shadow

Ao atribuir valores para as posições horizontal e vertical da sombra, teremos um novo elemento, igual ao original em dimensões e forma, porém **completamente preenchido com a cor que definirmos (preto por padrão)**. Esse elemento, que estava originalmente posicionado no mesmo local do original (mas atrás dele), **será então deslocado para a direita e para baixo (ou para a esquerda e para cima, caso se utilize valores negativos)**.



# 12- Posicionamento

## Posicionamento : o que é?

<p><b>Posicionament</b> <b>o</b></p>	<p><b>Como controlar onde cada elemento em uma página deve ficar.</b></p> <p>No <b>Slide 1</b> vemos uma página que contém os elementos <b>p (block)</b> e <b>strong (inline)</b></p>	
<p><b>Os níveis das caixas</b></p>	<p><b>Block</b></p> <p>O elemento <b>sempre vai se iniciar em uma nova linha</b> e ocupa todo o espaço horizontal disponível.</p>	
<p><b>Os níveis das caixas</b></p>	<p><b>Inline</b></p> <p>O elemento <b>nunca se inicia em uma nova linha</b> e ocupa apenas o espaço necessário para apresentar o conteúdo dele.</p>	
<p><b>Os níveis das caixas que envolve um elemento</b></p>	<ul style="list-style-type: none"> <li>• Bloco</li> <li>• Linha</li> </ul>	<p>As <b>caixas de nível de bloco, assunto deste curso, possuem características que são a base para posicionar elementos em uma página</b></p>

## Posicionamento Position

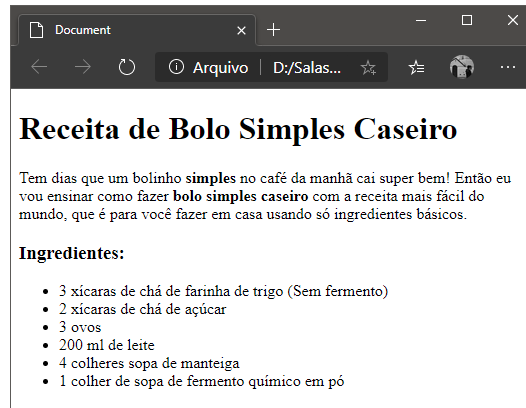
<p><b>O que é?</b></p>	<p>Neste capítulo estamos estudando como posicionar elementos na página. <b>Há duas propriedades que nos permitem fazer isso que são position e float.</b></p>	
<p><b>Por que isso é útil?</b></p>	<p>Como sabemos, <b>o CSS não é uma linguagem de programação. Por isso não posicionamos elementos utilizando geometria.</b> Em lugar dessa forma, o CSS nos dá modelos de posicionamento que utilizamos conforme a necessidade da página. Grande parte desses modelos <b>são representados pelos valores da propriedade position</b>, assunto deste curso.</p>	
<p><b>Qual técnica é necessária para se fazer isso?</b></p>	<p><b>A posição de um elemento em uma página pode ser controlada utilizando a propriedade position. Ela pode receber quatro valores.</b> Cada um deles representa um modelo de posicionamento do CSS:</p> <ul style="list-style-type: none"> <li>• Fluxo normal</li> <li>• Posicionamento relativo</li> <li>•</li> </ul>	

Posicionamento absoluto

- Posicionamento fixo

## Fluxo normal

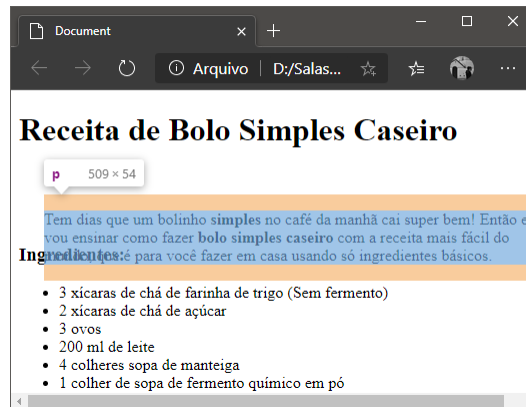
É o **modelo de posicionamento utilizado por padrão**. Nele **os elementos de bloco passarão a iniciar em uma nova linha, empurrando os demais para baixo**.



Veja na **Figura 1** que **dois elementos de bloco nunca dividem a mesma linha, mesmo que haja espaço disponível para isso**.

## Posicionamento relativo

Nesse modelo **o elemento pode ser movido para cima, baixo, esquerda ou direita tendo como ponto de partida a sua posição no fluxo normal**.



Na **Figura 2** vemos um **parágrafo sendo movido 25 pixels para a direita e para baixo**. Perceba que os elementos ao redor daquele que **está posicionado relativamente não são afetados**.

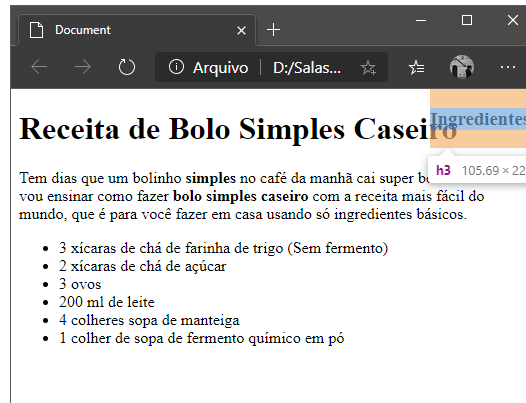
**Figura 2.** Parágrafo posicionado relativamente

## Posicionamento absoluto

**O elemento é removido do fluxo normal**.

Na **Figura 3** usamos esse posicionamento no h3 para **movê-lo para o topo e para a direita da página**.

Note que cada outro elemento na página se comporta como se o título não estivesse mais nela.



## Posicionamento fixo

Essa é uma forma de **posicionamento absoluto, mas o elemento não se move quando o usuário utiliza as barras de rolagem da janela do navegador**.

Na **Animação 1** o parágrafo foi posicionado de forma fixa, 25 pixels para a direita e para baixo.



**Animação 1.** Parágrafo posicionado fixamente

## HMTL

```
<!DOCTYPE html>
<html lang="pt-br">
```

```
<head>
<meta charset="UTF-8">
<title>Document</title>
</head>
```

```
<body>
<h1>Receita de Bolo Simples
Caseiro</h1>
```

```
<p>Tem dias que um bolinho
<strong>simples</strong> no café da
manhã cai super bem! Então eu vou
ensinar
como fazer
<strong>bolo simples
caseiro</strong> com a receita mais
fácil do mundo, que é para você fazer
em casa
usando só
ingredientes básicos.</p>
```

```
<h3>Ingredientes:</h3>
```

```
<ul>
<li>3 xícaras de chá de farinha de
trigo (Sem fermento)</li>
<li>2 xícaras de chá de açúcar</li>
<li>3 ovos</li>
<li>200 ml de leite</li>
<li>4 colheres sopa de manteiga</li>
<li>1 colher de sopa de fermento
químico em pó</li>
</ul>
</body>
```

```
</html>
```

### Ingredientes:

- 3 xícaras de chá de farinha de trigo (Sem fermento)
- 2 xícaras de chá de açúcar
- 3 ovos
- 200 ml de leite
- 4 colheres sopa de manteiga
- 1 colher de sopa de fermento químico em pó

```
<div class="right">
<h3>Ingredientes:</h3>
<ul>
<li>3 xícaras de chá de farinha de...</li>
<li>2 xícaras de chá de açúcar</li>
<li>3 ovos</li>
<li>200 ml de leite</li>
<li>4 colheres sopa de manteiga</li>
<li>1 colher de sopa de fermento q...</li>
</ul>
</div>
```

## Fluxo Normal na Prática

### Fluxo Normal

O valor padrão da propriedade **position** é **static**. Ele **representa o fluxo normal (normal flow)** de posicionamento dos elementos em uma página.

No fluxo normal os elementos:

- Seguem o fluxo da página, sendo posicionados na ordem em que foram declarados no HTML
- Elementos de nível de bloco ocupam todo o espaço disponível na linha em que estão

### Sintaxe

O valor padrão de position é static, **por isso geralmente não é necessário utilizá-lo**. Caso isso seja necessário fazemos dessa forma:

```
position: static;
```

### Exemplo fluxo normal como width

Nesse modelo de posicionamento **dois elementos de nível de bloco não podem ocupar uma mesma linha**, mesmo quando houver espaço para isso.

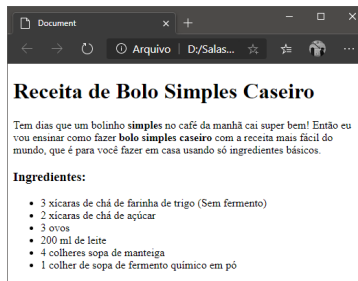
Quando a largura (**width**) **não for definida para elementos de nível bloco**, eles **vão se estender por toda a linha em que foram iniciados**. Com



(largura) não definido

isso outros elementos de nível de bloco **serão empurrados para baixo.**

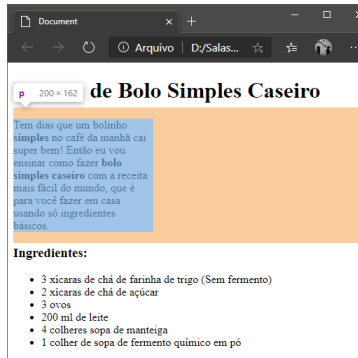
Na **Figura 1** vemos os elementos de nível de bloco h1, p, h3, ul e li sendo posicionados dessa forma.



Exemplo do fluxo normal com width (largura) definido

Para provar isso, no **Código 1** modificamos a largura do elemento p.

```
p {  
width: 200px;  
}
```



Veja na **Figura 2** que mesmo que haja espaço disponível na linha em que ele está, **os demais elementos se iniciam na linha seguinte.**

## Posicionamento Relativo na Prática

Posicionamento Relativo

O valor relativo de position **põe o elemento em posicionamento relativo.** Nesse modelo de posicionamento **o elemento se move em relação a sua posição no fluxo normal.**

### Sintaxe

Para colocar um elemento em posicionamento relativo utilizamos a seguinte sintaxe:

```
position: relative;
```

Quatro Propriedades para mover o elemento

Quando o posicionamento é relativo utilizamos outras quatro propriedades para mover o elemento, são elas:

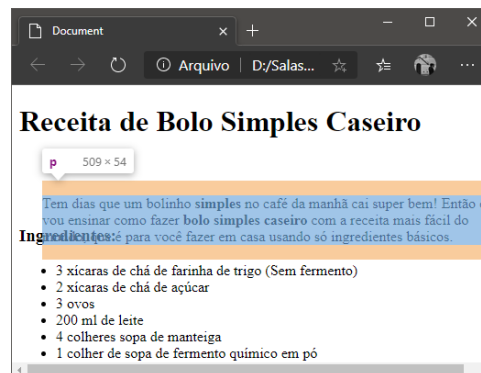
- top
- bottom
- left
- right

Essas são propriedades de deslocamento. Sendo assim, o elemento estará mais distante do fundo da página quanto maior for o valor de bottom e assim por diante.

Exemplo

No **Código 1** deslocamos o parágrafo 25 pixels para baixo e para a direita.

```
p {  
left: 25px;  
position: relative;  
top: 25px;  
}
```



### Código 1. Movendo o parágrafo para baixo e para a direita

Note que para isso **usamos as propriedades de deslocamento top (topo) e left (esquerda).** Isso porque mover o objeto para baixo e para a direita é distanciar-lo do topo (top) e da esquerda (left).

## Exemplo 2

```
.relative-parent {
  background: #156236;
}

.relative {
  background: #af8542;
  height: 100px;
  width: 300px;
  position: relative;
  top: 20px;
  left: 45px;
}
```

- **Linha 01:** declaração da classe **relative-parent**. **Define o estilo para o elemento pai em nosso posicionamento relativo.** Observe que não há nenhuma quanto à propriedade position. Assim, temos que o valor dela é static;
- **Linha 02:** Especifica **uma cor de fundo** para os elementos da classe **relative-parent**. É a única característica presente nessa classe;
- **Linha 05:** declaração da classe **relative**. Define o **estilo para o elemento filho, que terá seu posicionamento determinado de acordo com o posicionamento do elemento pai;**
- **Linha 06:** indica **acor de fundo para os elementos da classe relative;**
- **Linhas 07 e 08:** propriedades de **altura** (height) e **largura** (width) para o elemento, em pixels;
- **Linha 09:** indica a **propriedade position para nossa classe. Nesse caso, com o valor relative;**
- **Linhas 10 e 11:** **especifica as distâncias com relação ao topo (top) e esquerda (left) com relação ao elemento pai.** As distâncias, nesse caso, estão em pixels, mas **poderia ser utilizado porcentagem** ou em, uma medida relativa empregada no CSS. É importante ressaltar que esses valores podem ser negativos.

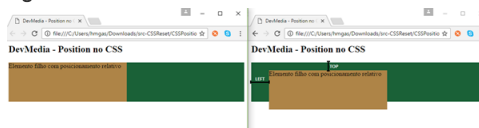
## HTML

```
HTML

<div class="relative-parent">
  <div class="relative">
    Elemento filho com
    posicionamento relativo
  </div>
</div>
```

Note que temos uma classe em cada elemento: uma representará o elemento pai e a outra, o elemento filho. O resultado do que fizemos pode ser observado na **Figura 1**. À esquerda, temos o resultado com um CSS diferente, sem as **linhas 9 a 11 da Listagem 1**. Nesse caso, não há um posicionamento relativo de um elemento em relação ao outro, pois isso ainda não foi definido. Já à direita, temos o resultado com o CSS da **Listagem 1** completo. Observe que o elemento filho está posicionado de forma relativa ao pai, por causa dos parâmetros que informamos na folha de estilos.

**Figura 1.** Position: relative



## Posicionamento Fixo na Pratica

### Posicionamento Fixo na Pratica

O posicionamento fixo é um tipo de **posicionamento absoluto**. Quando um elemento está posicionado fixamente **ele passa a ser ignorado pelos demais, como se não estivesse na página.**

Além disso, **aqui o elemento é posicionado em relação a janela do navegador e se mantém parado no mesmo lugar quando o usuário utiliza o scroll.**



### Sintaxe

Para que um elemento seja posicionado dessa maneira usamos essa sintaxe:

```
position: fixed;
```

A partir disso, **usamos as propriedades de deslocamento**

**top,**  
**bottom,**  
**left e**  
**right**

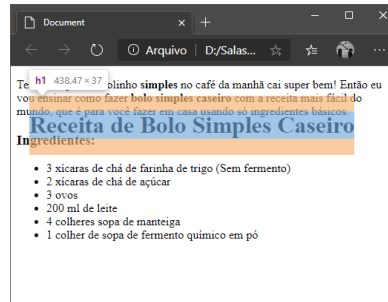
para controlar a posição do elemento na página.

No **Código 1** deslocamos o parágrafo 25

## Exemplo

pixels para baixo e para a direita.

```
h1 {  
  left: 25px;  
  position: fixed;  
  top: 25px;  
}
```



**Código 1.** Posicionando o elemento de forma fixa

O resultado deste código podemos ver na **Figura 1**.

Novamente **utilizamos as propriedades de deslocamento para mover o elemento em relação a sua posição no fluxo normal.**

Portanto, para mover o elemento para a direita definimos para ele uma distância da **esquerda** utilizando **left**. Da mesma forma, para mover o elemento para **baixo** distanciamos ele do topo usando **top**.

## HTML E CSS

HTML  
<div class="fixed">  
Elemento com posicionamento fixo  
</div>

CSS  
.fixed {  
background: #a4c467;  
height: 50px;  
width: 300px;  
position: fixed;  
bottom: 60px;  
right: 32px;  
}

- **Linha 01:** declaração da classe **fixed**. **Define as características dos elementos com posicionamento fixo;**
- **Linhas 02 a 04:** informa a cor de fundo background, altura height e largura width dos elementos da classe;
- **Linha 05:** com essa propriedade, passando o valor fixed, temos a **declaração do tipo de posicionamento que queremos para os nossos elementos;**
- **Linhas 06 e 07:** especificamos as distâncias dos elementos da classe em relação ao navegador. Repare que, dessa vez, especificamos bottom e right, distância da base e direita, respectivamente. Como estamos com um posicionamento fixo, é utilizado o navegador para calcular essa distância, ou seja, **o elemento permanecerá na mesma posição independentemente do lugar da página em que estivermos.**

## Posicionamento Absolute

### Posicionament o Absolute

Um elemento em posicionamento absoluto é removido do fluxo normal. Por isso ele não afeta ou é afetado pela posição dos demais elementos na página.

### Sintaxe

Para posicionar um elemento dessa forma use essa sintaxe:

```
position: absolute;
```

Novamente, **usamos as quatro propriedades de deslocamento para mover o elemento.** Relembrando, elas são:

- top
- bottom
- left

# Posicionament o Absoluto

## Exemplo

## HTML E CSS

- **right**

No **caso do posicionamento absoluto, essas propriedades funcionam um pouco diferente** do que vimos anteriormente. **Elas definem onde o elemento deve aparecer em relação àquele que o contém.**

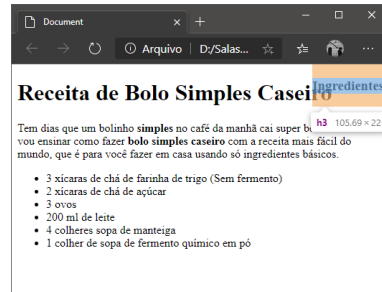
No **Código 1** usamos o posicionamento absoluto para mover o h3 para o topo e para a direita na página.

```
h3 {  
  right: 0;  
  position: absolute;  
  top: 0;  
}
```

**Código 1.** Movendo o h3 para o topo e para a direita da página

HTML  
<div class="absolute">  
Elemento com posicionamento absoluto  
</div>

CSS  
.absolute {  
 background: #c590f5;  
 height: 50px;  
 width: 120px;  
 position: absolute;  
 top: 35px;  
 right: 32px;  
}



O resultado deste código podemos ver na **Figura 1**.

Perceba que ao definir que o elemento deve se deslocar zero pixel a partir do topo e da direita do seu contêiner (a própria página) ele passa a ocupar exatamente essa posição.

- **Linha 01:** declaração da classe **absolute**. **Define o estilo para o elemento com posicionamento absoluto;**
- **Linha 02:** informar a **cor de fundo para os elementos da classe;**
- **Linhas 03 e 04:** atribuição dos valores para **altura height e largura width dos elementos da classe;**
- **Linha 05:** determina o **posicionamento dos elementos da classe, com a opção absolute;**
- **Linhas 06 e 07:** especifica duas propriedades de posicionamento: **right** e **top**, referentes à distância em relação à direita e ao topo da página. Note que essa é a mesma proposta vista anteriormente, no posicionamento relativo, mas com a diferença que estamos atribuindo a distância para a **página**, e não mais para um elemento pai.



# CONTACT



*Brunna Croches*

*Developer Full Stack*



*brunnacroches.dev*



*linkedin.com/brunnacroches*



*github.com/brunnacroches*



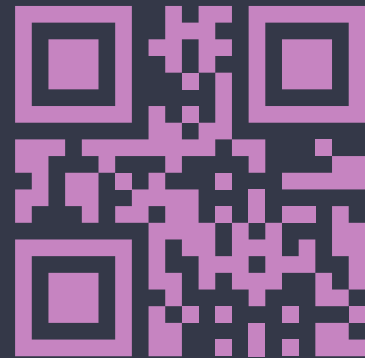
*@brunnacroches.dev*



*discord.com/brunnacroches*



*brunnacroches@gmail.com*



*let's share*