

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

COMO FUNCIONA O CSS?

01

Neste capítulo será abordado conceitos básicos do CSS tais como seleção, classe, elementos entre outros.

ADICIONANDO O CSS NO HTML

02

Exemplo e explicação de como adicionar o CSS no HTML.

ESTILIZANDO O TEXTO

03

Tutorial de como aplicar fontes, tipografia, sombra, altura, espaçamento e alinhamento no texto em CSS.

ESTILIZANDO O FUNDO

04

Ajustando o tamanho da imagem do fundo, posicionamento, repetição e alterar a cor do fundo.

1 - Como funciona o CSS?

HTML + CSS

VINCULAR LINK HTML + CSS

Para vincular o arquivo **stylesheet.css** a página **index.html**. Abra o arquivo index.html e insira o seguinte conteúdo dentro do elemento head, abaixo de title.

SELETOR E DECLARAÇÃO

SELETOR	DECLARAÇÃO
Seletor é a parte da regra que informa a qual ou quais elementos ela deve ser aplicada .	Declaração é a parte da regra que informa como os elementos referenciados pelo seletor devem ser modificados .
p é o seletor <pre>p { font-family: Arial; }</pre>	e todo o conteúdo entre chaves é a declaração. <pre>p { font-family: Arial; }</pre>

DECLARAÇÃO

DECLARAÇÃO

As declarações são compostas por pares de chave e valor separados por dois pontos ::

```
p {  
  font-family: Arial;  
}
```

Essa regra informa para o navegador que **todos os elementos p encontrados na página devem ter a fonte Arial**.

ESTILIZAR UM OU MAIS NO CSS

Regra com mais de uma propriedade	Para que todos os elementos p encontrados na página devem ter a fonte Arial .	<pre>p { font-family: Arial; }</pre>
Regra com mais de uma propriedade	Essa regra informa para o navegador que os elementos h1 e h2 devem ter a fonte Arial e a cor yellow.	<pre>h1, h2 { font-family: Arial; color: yellow; }</pre>

NA PRÁTICA

```
<!DOCTYPE html>  
<html>  
  
<head>
```


	ATENÇÃO: um id só pode pertencer a um único elemento HTML e o uso de classes são para selecionarmos um grupo de elementos .	propriedade color modificada)	
*SELECIONAR APENAS UM ELEMENTO E NAO SE REPETIR	Seletor de ID para atribuir um estilo a apenas um elemento , visto que o valor desse atributo não deve se repetir na página . Para isso utilizamos o nome do elemento precedido de um # (cerquilha) .	#cor-azul { color: blue; }	
*SELECIONAR APENAS HREF	Para que apenas os elementos que tiverem o atributo href com o valor cursos serão afetados pela declaração desta regra.	[href="cursos"] { color: blue; }	

Classe ou ID?

SELETOR DE CLASSE	O seletor de classe deve ser usado quando muitos elementos compartilham do estilo definido na regra . Por exemplo , se queremos alterar todos os parágrafos em uma página utilizamos o seletor de classe. lembre-se de usar id se o elemento não se repete na página. Em outro caso, use o seletor de classe.
SELETOR DE ID	O seletor de id deve ser utilizado quando apenas um elemento deve receber o estilo definido na regra . Dessa forma, o elemento terá um estilo único na página e geralmente ele não se repete nela. Por exemplo , o menu lateral de um site , o avatar do usuário na barra de navegação ou o rodapé são candidatos a serem selecionados por ids e não classes, dada sua singularidade na página .

AGRUPAMENTO DE SELEÇÃO

AGRUPAMENTO POR ELEMENTOS	consiste em inserir mais de um elemento numa regra de estilo para que as alterações nas propriedades sejam iguais para ambos.	<pre>h1, p { color: blue; }</pre> <p>aplicamos o mesmo estilo a dois elementos diferentes, apenas especificando cada elemento e os separando com vírgula. Desta forma, todas as propriedades declaradas na regra serão aplicadas em ambos os elementos.</p>	
AGRUPAMENTO POR DESCENDÊNCIA DE ELEMENTOS	Sua sintaxe é composta pelo elemento que queremos selecionar e o elemento que ele descende separados por um espaço em branco	<pre>div p { color: blue; }</pre> <p>queremos aplicar a regra criada ao elemento p que pertence a tag div. Desta forma, na declaração da nossa regra especificamos o elemento na qual p descende para que</p>	

<p>AGRUPAMENTO POR CLASSES</p>	<p>selecionaremos os elementos que tiverem como atributo duas classes. Sua sintaxe é semelhante a seleção de uma classe, sendo que ambas serão especificadas juntas</p>	<p>somente ele receba o estilo e não as demais tags do mesmo tipo</p> <pre>.corAzul .corClaro { color: lightblue; }</pre> <p>Neste exemplo temos duas tags <code>p</code> que possuem classes definidas como atributo, mas apenas uma delas tem duas classes especificadas. Dessa maneira para aplicarmos uma regra de estilo nesta tag foram especificadas ambas as classes <code>corAzul</code> e <code>corClaro</code> para que apenas esse elemento receba o estilo especificado.</p>	<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> <p style="text-align: center; background-color: orange; color: white; margin: 0;">HTML</p> <pre><!DOCTYPE html> <html> <head> <title>CSS: Seletores</title> <link rel="stylesheet" type="text/css" href="style.css"> </head> <body> <div class="corAzul corClaro"> Texto simples</div> <div class="corAzul"> Texto simples 2</div> <div class="corClaro"> Texto simples 3</div> </body> </html></pre> </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> <p style="text-align: center; background-color: blue; color: white; margin: 0;">CSS</p> <pre>.corAzul .corClaro { color: lightblue; }</pre>  </div> </div>
<p>AGRUPAMENTO POR DESCENDENCIA DE CLASSES</p>	<p>Essa seleção é feita como na descendência de elementos, contudo, ao invés de utilizarmos diretamente os elementos, serão especificadas as classes que têm dependência com outras classes</p>	<pre>corClaro .corAzul { color: lightblue; }</pre>	<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> <p style="text-align: center; background-color: orange; color: white; margin: 0;">HTML</p> <pre><!DOCTYPE html> <html> <head> <title>CSS: Seletores</title> <link rel="stylesheet" type="text/css" href="style.css"> </head> <body> <div class="corClaro"> <div class="corAzul"> Texto simples 1</div> <div class="corAzul"> Texto simples 2</div> <div class="corAzul"> Texto simples 3</div> </div> </body> </html></pre> </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> <p style="text-align: center; background-color: blue; color: white; margin: 0;">CSS</p> <pre>.corClaro .corAzul { color: lightblue; }</pre>  </div> </div>

ESTILIZANDO

	CSS	SIGNIFICADO
Colocar fonte da letra CSS Mudar a cor da Letra CSS	<pre>h1 { font-family: Arial; color: #333; }</pre>	informa que esse elemento deve possuir fonte Arial e um tom de cinza como cor
Tamanho da Letra CSS	<pre>p { font-size: 14px; }</pre>	seletor <code>p</code> , informa que esse elemento deve ter o tamanho da sua fonte modificado para 14px
Perca o recuo CSS	<pre>blockquote { margin: 0; }</pre>	seletor <code>blockquote</code> , modifica a margem desse elemento de forma que ele perca o recuo .

ESTILIZAR UM OU MAIS NO CSS

Regra com mais de uma propriedade	Para que todos os elementos p encontrados na página devem ter a fonte Arial.	<pre>p { font-family: Arial; }</pre>
Regra com mais de uma propriedade	Essa regra informa para o navegador que os elementos <code>h1</code> e <code>h2</code> devem ter a fonte Arial e a cor yellow.	<pre>h1, h2 { font-family: Arial; color: yellow; }</pre>

<p>*SELECIONAR APENAS UM ELEMENTO E NAO SE REPETIR</p>	<pre>#cor-azul { color: blue; }</pre>	<p>par atribuir um estilo a apenas um elemento, visto que o valor desse atributo não deve se repetir na página. Para isso utilizamos o nome do elemento precedido de um # (cerquilha).</p>
<p>*SELECIONAR APENAS HREF</p>	<pre>[href="cursos"] { color: blue; }</pre>	<p>Para que apenas os elementos que tiverem o atributo href com o valor cursos serão afetados pela declaração desta regra.</p>

SELETORES E COMBINADORES

Atribuir um mesmo estilo a diferentes elementos.

<p>p, a { color: blue; }</p>	<p>Tanto o elemento p quanto a terão a cor de seu texto alterada para azul. Utilizamos a vírgula para determinar que o estilo deve ser aplicado a ambos.</p>
<p>p a { color: blue; }</p>	<p>Descendant Selector: onde utilizamos um espaço entre os seletores para determinar que o estilo deve ser aplicado se a for um descendente de p. Nesse caso, a receberá a cor azul se estiver contido em um elemento p, pois na hierarquia de elementos, a é um descendente de p.</p>
<p>p > a</p>	<p>Child Selector Nos permite selecionar um elemento a que seja um filho direto de outro elemento p.</p>
<p>p + a</p>	<p>Adjacent Sibling Selector Seleciona o elemento a após p, desde que os dois estejam no mesmo container.</p>
<p>p ~ a</p>	<p>General Sibling Selector Seleciona qualquer elemento a após p, desde que ambos estejam no mesmo container.</p>

Com **exceção do descendant selector**, os combinadores vêm sendo cada vez menos utilizados, pois **dificultam a leitura do código**. Um dos motivos para isso é a dificuldade em alterar apenas o HTML da página mantendo o CSS já escrito, uma vez que os combinadores estão vinculados a ordem na qual os elementos foram declarados.

Cascata

<pre>p { color: blue; } p { background-color: red; }</pre>	<p>o elemento p terá a cor de seu texto alterada para azul devido a aplicação da primeira regra e, logo após, sua cor de fundo será alterada para vermelho na aplicação na regra seguinte.</p>
<pre>p { color: blue; } p { color: red; }</pre>	<p>a cor final do elemento será vermelha, pois o valor atribuído à propriedade color na declaração da segunda regra sobreescreve o</p>

```
<p class="classe"
id="identificador">Texto
do parágrafo</p>
```

```
.classe { color: blue; }
#identificador { color:
red; }
```

valor da primeira. Isso acontece porque o **CSS leva em consideração a ordem na qual as regras foram declaradas no código.**

Neste caso a cor do parágrafo será vermelho, pois o peso da regra que utiliza o seletor de classe é menor em comparação aquela que utiliza o seletor de id.

2 - Adicionando o CSS no HTML

HTML + CSS

VINCULAR LINK HTML + CSS

Para vincular o arquivo **stylesheet.css** a página **index.html**. Abra o arquivo **index.html** e insira o seguinte conteúdo dentro do elemento **head**, abaixo de **title**.

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

rel é **stylesheet** nomeia a relação entre o recurso que está sendo adicionado no documento atual.

o atributo **href** contém a URL para o recurso que estamos adicionando.

3 - Estilizando o TEXTO

ESTILIZANDO

	CSS	SIGNIFICADO
Colocar fonte da letra CSS Mudar a cor da Letra CSS	<pre>h1 { font-family: Arial; color: #333; }</pre>	informa que esse elemento deve possuir fonte Arial e um tom de cinza como cor
Tamanho da Letra CSS	<pre>p { font-size: 14px; }</pre>	seletor p, informa que esse elemento deve ter o tamanho da sua fonte modificado para 14px
Perca o recuo CSS	<pre>blockquote { margin: 0; }</pre>	seletor blockquote, modifica a margem desse elemento de forma que ele perca o recuo.

ALTERAR ELEMENTOS DO TEXTO

NEGRITO REMOVER	A propriedade font-weight é utilizada para aplicar ou remover negrito de um elemento de texto. Os valores numéricos são 100, 200, 300, 400, 500, 600, 700, 800 e 900. Dentre eles, 400 corresponde a aparência padrão da maioria das fontes e 700 ao negrito.	<pre>p { font-weight: 400; }</pre>
NEGRITO ADICIONAR	font-weight : temos a aplicação de negrito com bold ou 700.	<pre>p { font-weight: 700; } /* ou */ p { font-weight: bold; }</pre>
ITÁLICO ADICIONAR	Itálico é um estilo obtido com a propriedade font-style , utilizando o valor italic.	<pre>p { font-style: italic; }</pre>
ITÁLICO REMOVER	Removemos o itálico de um elemento.	<pre>p { font-style: normal; }</pre>
COR	A cor de um texto é customizada utilizando a propriedade color, que pode receber valores numéricos ou textuais.	<pre>p { color: red; } p { color: #ff0000; }</pre>
FUNÇÕES DE COR	Algumas propriedade podem receber o resultado de uma função como valor. No caso de color, essas funções são rgb, rgba e hsl.	<pre>p { color: rgb(81, 155, 0); }</pre>
	RGB é um padrão de cor familiar	<pre>p {</pre>

FUNÇÕES DE COR RGB	a designers, onde se usa red (vermelho), green (verde) e blue (azul) para gerar uma cor. Assim o RGB recebe como parâmetro três valores de 0 a 255 que correspondem a essas cores. Dessa forma, para produzir apenas vermelho	<pre>color: rgb(81, 155, 0); }</pre>
FUNÇÕES DE COR RGBA	. A diferença entre elas é que rgba recebe mais um parâmetro chamado alpha, que determina a opacidade da cor . Quando alpha é 0 a cor está totalmente transparente e quando ele é 1 ela está completamente opaca. Essa função é utilizada para fins mais específicos como deixar um elemento semitransparente ou fazer ajustes mais finos a uma cor .	<pre>p { color: rgba(81, 155, 0, 0.5); }</pre>
TAMANHO DA FONTE	O tamanho da fonte pode ser customizado com a propriedade font-size . A unidade de medida mais utilizada no CSS é o pixel (px).	<pre>p { font-size: 22px; }</pre>
CUSTOMIZAR A FONTE	Para customizar a fonte utilizada em um elemento de texto usamos a propriedade font-family que recebe como valor o nome de uma fonte. "web safe fonts", um conjunto de fontes que existe na maioria dos navegadores	<pre>p { font-family: "Arial Black", Gadget, sans-serif; }</pre>
ALINHAMENTO	Saindo agora da propriedade font, existem quatro formas de alinhar um texto utilizando a propriedade text-align . Elas são à direita (right) , à esquerda (left) , centralizado (center) e justificado (justify) .	<pre>p { text-align: center; }</pre>

FONTE

A tipografia é uma parte importante do design de um site/aplicação e, juntamente com cores e outros elementos, compõem sua identidade visual. **Normalmente, em projetos que requerem um design aprimorado, a utilização de fontes textuais específicas se faz necessária para obter os resultados esperados.**

Na versão mais recente das folhas de estilo, porém, **importar fontes externas se tornou uma tarefa bastante simples**. Para isso, **podemos utilizar o recurso @font-face, cuja sintaxe e funcionamento serão vistos neste documento.**

TIPOGRAFIA	é uma parte importante do design de um site/aplicação e, juntamente com cores e outros elementos, compõem sua identidade visual.	Podemos utilizar o recurso @font-face .
SINTAXE BASICA font-family = nome da fonte	font-family: [nome]; O nome definido para essa fonte será utilizado em todos os elementos em que essa fonte deva ser aplicada. Caso o nome escolhido possua espaços em branco, é necessário informá-lo entre aspas duplas	A sintaxe para uso dessa função é a seguinte: 01 @font-face { 02 font-family: [nome]; 03 src: url([caminho]); 04 }
SINTAXE BASICA	No atributo src devemos informar o caminho do arquivo de fonte que queremos	A sintaxe para uso dessa função é a seguinte:

src = importar fonte	importar. Esse arquivo pode estar hospedado localmente ou em um servidor externo.	01 @font-face { 02 font-family: [nome]; 03 src: url([caminho]); 04 }
SINTAXE BASICA Na pratica importando fonte	Definida a diretiva @font-face, podemos referenciar essa fonte na formatação dos elementos da página. Por exemplo, a folha de estilos abaixo importa a fonte do arquivo Barrio-Regular.ttf e a referencia pelo nome Barrio para formatar os elementos <h1>.	A sintaxe para uso dessa função é a seguinte: @font-face { font-family: Barrio; src: url(../fonts/Barrio-Regular.ttf) } h1 { font-family: Barrio; }
FUNÇÃO LOCAL Usar uma fonte local	Ao preencher o atributo src da diretiva @font-face podemos usar a função local() para informar ao browser que ele deve tentar usar uma fonte local	A sintaxe para uso dessa função é a seguinte: 01 @font-face { 02 font-family: [nome]; 03 src: local([nome da fonte local]), url([caminho]); 04 }
Na pratica da função local	podemos definir essa diretiva da seguinte forma:	@font-face { font-family: Barrio; src: local(Barrio), url(../fonts/Barrio-Regular.ttf) }
Aplicar fonte customizada apenas a elementos de mesmo valor	font-stretch Faz com que a fonte customizada seja aplicada apenas a elementos que possuem esse mesmo valor na propriedade font-stretch	Os valores possíveis são: normal (padrão), condensed, ultra-condensed, extra-condensed, semi-condensed, expanded, semi-expanded, extra-expanded, ultra-expanded;
Fonte aplicada apenas aos texto de memo valor	font-style Define que a fonte será aplicada apenas aos textos formatados com o mesmo valor de font-style	Os valores possíveis são: normal (padrão), italic oblique;
Fonte ser aplicada apenas a texto com espessura de texto iguais	font-weight Faz com que essa fonte seja aplicada apenas a textos formatados com essa espessura de texto.	Exemplo, se definirmos esse atributo como 400, apenas textos formatados com font-weight: 400 usarão essa fonte;
Em hexadecimal	unicode-range Faixa de caracteres (em hexadecimal, baseado na tabela ASCII) que serão formatados com essa fonte.	O valor padrão é U+0-10FFFF, que abrange todos os caracteres.
Compatibilidade entre os browsers	Atualmente diversos formatos de arquivo de fonte e é importante optar por aquele que apresenta maior compatibilidade entre os browsers.	

FORMATO DE FONTE	Formato	Chrome	IE/Edge	Firefox	Safari	C
Atualmente diversos formatos de arquivo de fonte e é importante optar por aquele que apresenta	TTF/OTF	4.0	9.0	3.5	3.1	1

maior compatibilidade entre os browsers.						
	WOFF	5.0	9.0	3.6	5.1	1
	WOFF2	36.0	Não suportado	35.0	Não suportado	2
	SVG	4.0	Não suportado	Não suportado	3.2	3
	EOT	Não suportado	6.0	Não suportado	Não suportado	4

EXEMPLO PRÁTICO

```

<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <link rel="stylesheet" href="css/estilo.css">
</head>
<body>
  <p id="p1">Texto normal.</p>
  <p id="p2">Texto com font-weight: 600</p>
  <p id="p3">Texto com font-weight: 800</p>
</body>
</html>

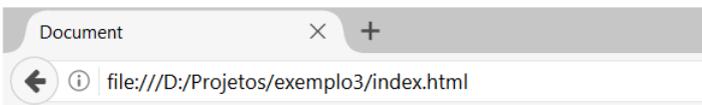
```

E logo abaixo podemos ver a folha de estilos:

```

01 @font-face {
02   font-family: "Raleway Especial";
03   src: local(Raleway), url(../fonts/Raleway-Normal.ttf);
04 }
05
06 @font-face {
07   font-family: "Raleway Especial";
08   src: local(Raleway), url(../fonts/Raleway-Bold.ttf);
09   font-weight: 600;
10 }
11
12 @font-face {
13   font-family: "Raleway Especial";
14   src: local(Raleway), url(../fonts/Raleway-ExtraBold.ttf);
15   font-weight: 800;
16 }
17
18 p {
19   font-family: "Raleway Especial"
20 }
21
22 #p2 {
23   font-weight: 600;
24 }
25
26 #p3 {
27   font-weight: 800;
28 }

```



Texto normal.

Texto com font-weight: 600

Texto com font-weight: 800

EDITANDO UM TEXTO

Como utilizar negritos, itálicos e sublinhados, alterar a fonte e aplicar efeitos sobre elementos de texto a definir o espaçamento entre as linhas e letras.

COR NO TEXTO	Ao utilizar essa propriedade, todo o texto do elemento será alterado, assim como decorações que ele contenha, por exemplo sublinhado.	p { color : blue; }
FONTE DA LETRA	Essa propriedade permite informar uma lista composta por uma ou mais fontes que, quando forem encontradas pelo navegador no computador do usuário do site, serão utilizadas	p { font-family : arial; }
FONT caso NÃO tenha a fonte DISPONÍVEL	No caso de a fonte escolhida pelo desenvolvedor não estar disponível no computador do usuário, o navegador pode decidir utilizar um fonte web safe em seu lugar.	p { font-family : arial, sans-serif; }
FONT exemplo de fontes genéricas	Sans-serif, serif, , cursive e fantasy também são exemplos de fontes genéricas que podemos utilizar como alternativa caso a fonte informada não seja encontrada no computador do usuário.	p { font-family : arial, cursive; }
FONT exemplo de fontes genéricas OUTRO RECURSO	Outro recurso que podemos utilizar é informar mais de uma fonte para que se caso a primeira não seja encontrada, o navegador tente utilizar a próxima, como uma pilha de fontes	p { font-family : "Trebuchet MS", Verdana, sans-serif; }
ALTERAR O TAMANHO DA FONTE	A propriedade font-size altera o tamanho do texto . Podemos utilizar diferentes unidades para determinar esse tamanho, sendo px e em as mais comuns.	p { font-size : 16px; }
ALTERAR O TAMANHO DA FONTE pai e filho	Quando utilizamos em como unidade, o tamanho da fonte utilizada no elemento pai afeta o elemento filho	<p>Texto em destaque no parágrafo</p> p { font-size: 16px; }
TRAÇADO	Recebem os valores lighter e bolder dependem do grau do peso que desejamos aplicar ao texto. Esse mesmo efeito pode ser obtido utilizando valores de 100 a 900 , onde em 100 o texto possui traços mais finos que vão ficando mais pesados ao avançarmos até 900 . Em 400, font-weight atribui a fonte seu traço padrão.	<p>Texto em destaque no parágrafo</p> p { font-style: italic; } span { font-weight: bold; }
LETRA PRIMEIRA MAIÚSCULA	Para aplicar transformações ao texto utilizando para isso a propriedade text-transform . Para deixar a primeira letra de cada palavra em maiúscula atribuindo a essa propriedade o valor capitalize .	p { text-transform: capitalize ; } o resultado do estilo acima aplicado será "Texto Do Parágrafo".
LETRA REMOVER	none : remove todas as transformações, o que é útil para anular algum estilo anteriormente aplicado pelo navegador	p { text-transform: none ; }

LETRA MAIUSCULA	uppercase: deixa todas as letras em maiúscula;	p { text-transform: uppercase ; }
LETRA MINUSCULA	lowercase: deixa todas as letras em minúscula.	p { text-transform: lowercase ; }
REMOVER O SUBINHADO DE LINKS	text-decoration. Com ela podemos, por exemplo, remover o sublinhado de links	a { text-decoration: none; }
LINHA ABAIXO DO TEXTO	underline: aplica uma linha abaixo do texto;	a { underline: none; }
LINHA SOBRE O TEXTO	overline: aplica uma linha sobre o texto;	a { overline: none; }
RISCA O TEXTO	line-through: risca o texto.	a { line-through: none; }

SOMBRAS

Nos dias atuais é muito comum aplicarmos efeitos de sombra ao texto. Quando utilizadas corretamente, as sombras facilitam a leitura e destacam o texto de imagens ou qualquer outro elemento que seja utilizado como fundo.

SOMBRAS text-shadow	A propriedade utilizada para esse fim é a text-shadow , que recebe quatro valores	p { text-shadow: 2px 4px 5px #eee; }
O PRIMEIRO VALOR (horizontal)	o primeiro valor corresponde a distância horizontal da sombra em relação ao texto no parágrafo	p { text-shadow: 2px 4px 5px #eee; }
O SEGUNDO VALOR (vertical)	segundo valor determina o deslocamento vertical .	p { text-shadow: 2px 4px 5px #eee; }
O TERCEIRO VALOR (borrado)	o terceiro valor indica o quanto desejamos que a sombra seja borrada , perdendo assim a sua solidez	p { text-shadow: 2px 4px 5px #eee; }
O QUARTO VALOR (cor da sombra)	definimos a cor da sombra , nesse caso cinza.	p { text-shadow: 2px 4px 5px #eee ; }
APLICAR MULTIPLAS SOMBRAS	aplicar múltiplas sombras ao texto com essa propriedade (apenas o último valor, separado do anterior por uma vírgula, recebe um ponto e vírgula no final)	p { text-shadow: 2px 4px 5px #eee, 0 0 1px red; }

ALINHAMENTO

A propriedade text-align permite controlar como o texto será alinhado dentro do elemento que o contém.

ALINHAMENTO	text-align permite controlar como o texto será alinhado dentro do elemento que o contém	p { text-align: center; }
ESQUERDA	left: justifica o texto a esquerda;	p { text-align: left ; }

DIREITA	right: justifica o texto a direita;	<code>p { text-align: right; }</code>
JUSTIFICADO	justify: ajusta o espaço entre as palavras para que o texto ocupe todo o espaço disponível em uma linha.	<code>p { text-align: justify; }</code>

ALTURA DE LINHA E ESPAÇAMENTO

ALTURA DE LINHA	Podemos controlar a altura da linha de um texto com a propriedade line-height . Geralmente utilizamos a unidade em como valor para essa propriedade para que a distância entre as linhas do texto acompanhe o tamanho da fonte .	<code>p { font-size: 14px; line-height: 1.6em; }</code>
ESPAÇAMENTO	modificar a aparência de um texto com as propriedades letter-spacing e word-spacing , que permitem controlar o espaço entre as letras e palavras de um texto .	<code>p { letter-spacing: 2px; word-spacing: 4px; }</code>

1.5

CORES POSSIBILIDADES

CORES		
nome da cor	red, blue, yellow, etc;	01 background-color: nome_da_cor;
código Hexadecimal	#ffffff, #ababab	02 background-color: #codigo_hexadecimal;
RGB	define uma cor de acordo com os valores de vermelho, verde e azul (Red, Green e Blue) que a compõem;	03 background-color: rgb(red, green, blue);
RGBA	define uma cor de acordo com os valores de Red, Green, Blue e Alpha-Opacity. O último parâmetro diz respeito à opacidade da cor e varia de 0 (totalmente transparente) a 1 (totalmente opaco);	04 background-color: rgba(red, green, blue, alpha);
HSL	função que define a cor de acordo com os valores de Hue, Saturation e Lightness. O valor de hue representa um ângulo na roda de cores, logo, varia de 0 a 360. O segundo argumento, saturation, varia de 0 a 100% e representa a saturação da cor, ou seja, o quão "viva" (valores maiores) ou "apagada" (valores menores) a cor será. Por fim, lightness indica a luminosidade aplicada sobre a cor. Seu valor também varia de 0 (totalmente escuro, ou preto) a 100% (totalmente iluminado, ou branco).	05 color: hsl(hue, saturation, lightness);
HSLA	funciona da mesma forma que a HSL,	06 color: hsla(hue,



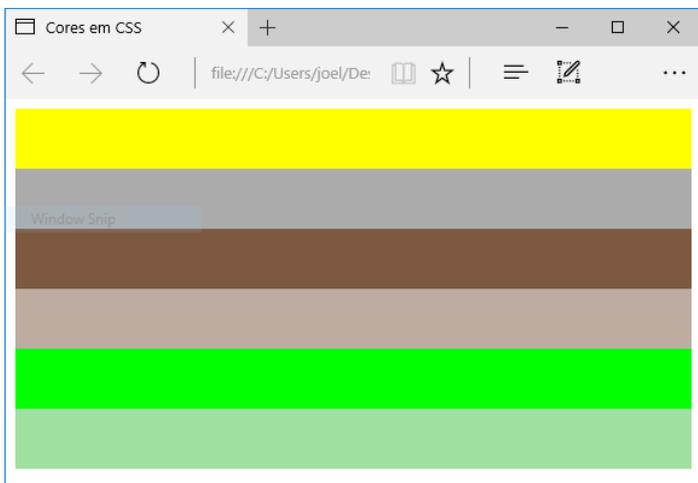
contando apenas com um argumento
alpha adicional, assim como na RGBA.

saturation, lightness,
alpha);

4- Estilizando o FUNDO

BACKGROUND

```
01 <!DOCTYPE html>
02 <html lang="pt-br">
03 <head>
04   <meta charset="UTF-8">
05   <meta name="viewport" content="width=device-width, initial-scale=1.0">
06   <title>Cores em CSS</title>
07   <style>
08     div { height: 50px }
09     #div1 { background-color: yellow; }
10     #div2 { background-color: #ababab; }
11     #div3 { background-color: rgb(125, 89, 65); }
12     #div4 { background-color: rgba(125, 89, 65, 0.5); }
13     #div5 { background-color: hsl(120, 100%, 50%); }
14     #div6 { background-color: hsla(120, 50%, 50%, 0.5);}
15   </style>
16 </head>
17 <body>
18   <div id="div1"></div>
19   <div id="div2"></div>
20   <div id="div3"></div>
21   <div id="div4"></div>
22   <div id="div5"></div>
23   <div id="div6"></div>
24 </body>
25 </html>
```



Um plano de fundo, ou background, é a área posicionada sob o conteúdo, preenchimento e borda de um elemento. Com a propriedade background podemos preencher essa área.

BACKGROUND

BACKGROUND	é a área posicionada sob o conteúdo, preenchimento e borda de um elemento.	
COR DO FUNDO	utilizamos a propriedade background-color para atribuir a cor amarela	div { background-color : yellow; }
ADICIONAR IMAGEM NO FUNDO	O CSS nos permite utilizar imagens como plano de fundo de um elemento. Para isso utilizamos a propriedade background-image , que pode receber como valor o retorno na função url(), para a qual informamos o caminho para a imagem	div { background-color: yellow; background-image : url(../imagem/plano-de-fundo.jpeg); }
REPETIR IMAGEM NO FUNDO	Caso a imagem utilizada seja menor que o elemento no qual ela foi inserida, o navegador levará em consideração o valor padrão da propriedade background-	div { background-repeat; }

	repeat , que costuma ser repeat e repetirá a imagem até que ela preencha todo o espaço disponível.	
NAO REPETIR IMAGEM NO FUNDO	Para anular esse comportamento podemos alterar o valor da propriedade background-repeat:	div { background-repeat: no-repeat; }
IMAGEM REPETIR NO EIXO X NO FUNDO	background-repeat pode ainda receber os valores repeat-x, para que a imagem se repita no eixo x,	
IMAGEM REPETIR NO EIXO Y NO FUNDO	repeat-y para que a imagem se repita no eixo y	
POSICIONAMENTO DA IMAGEM NO FUNDO	controlar o posicionamento da imagem , utilizando para isso a propriedade background-position . Por padrão, uma imagem que não se repete é posicionada no centro e a esquerda do elemento. Com background-position podemos determinar o quando essa imagem se deslocará em relação a essa posição inicial	div { background-position: 10px center; } No caso acima a imagem se deslocará 10 pixels no eixo x e estará centralizada no eixo y;
POSICIONAMENTO DA IMAGEM NO FUNDO	Podemos ainda utilizar palavras chave como top, left, bottom e center para determinar esses valores.	div { background: yellow url(../imagem/plano-de-fundo.jpeg) no-repeat 10px center; }

BACKGROUND - AJUSTAR O TAMANHO DA IMAGEM

AJUSTAR O TAMANHO DA IMAGEM	ajustar o tamanho da imagem para que ela caiba no elemento. A propriedade background-size nos permite realizar pequenas modificações na imagem de um elemento para que ela seja redimensionada de acordo com a necessidade do estilo.	
AJUSTAR A LARGURA DA IMAGEM	background-size: informamos um valor de 16 pixels para a largura da imagem e deixamos que sua altura seja ajustada automaticamente com base nesse valor	div { background-size: 16px auto; }
AJUSTAR A ALTURA DA IMAGEM		div { background-size: auto 16px; }
IMAGEM OCUPE TODO O ESPAÇO DISPONÍVEL	atribuir o valor especial contain para background-size, que faz com que a imagem ocupe todo o espaço disponível, sem que para isso ela seja cortada ou distorcida	div { background-size: contain; }
	cover , é muito utilizado, pois permite escalar a imagem de forma que ela se torna tão larga quanto o necessário	div { background-size: cover; }

<p>ESCALAR A IMAGEM</p>	<p>para cobrir todo o espaço interno do elemento. cover, é muito utilizado, pois permite escalar a imagem de forma que ela se torna tão larga quanto o necessário para cobrir todo o espaço interno do elemento.</p>	
<p>IMAGEM NAO ACOMPANHA A BARRA DE ROLAGEM</p>	<p>propriedade background-attachment permite controlar como uma imagem se comporta em relação a barra de rolagem da página ou do elemento no qual ela está contida.</p> <p>Por exemplo, é possível fazer com que a imagem não acompanhe a rolagem da página, mantendo-se estática em relação ao seu conteúdo</p>	<pre>div { background-attachment: fixed; }</pre>
<p>IMAGEM ACOMPANHAR A BARRA DE ROLAGEM</p>	<p>é local. Considerando que o elemento que contém a imagem possui uma barra de rolagem, a imagem acompanhará essa barra, ignorando a rolagem da página</p>	<pre>div { background-attachment: local; }</pre> <p>*Essa propriedade funciona apenas em conjunto com background-image, ou seja, quando atribuímos uma imagem ao elemento</p>

CONTACT



Brunna Croches

Developer Full Stack



brunnacroches.dev



linkedin.com/brunnacroches



github.com/brunnacroches



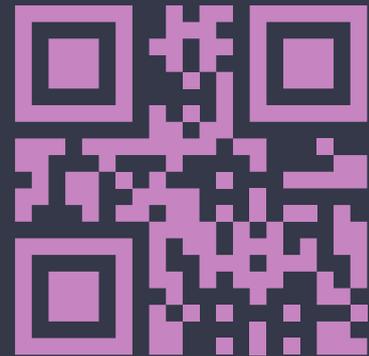
@brunnacroches.dev



discord.com/brunnacroches



brunnacroches@gmail.com



let's share