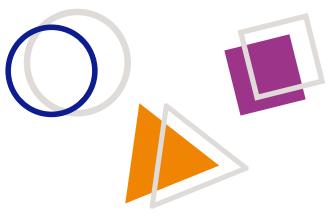




Mundo Tech



■ ■ ■

# PROGRAMAÇÃO FRONT-END FOLHAS DE ESTILO EM CASCATA

**SENAI** <LAB365>

# SUMÁRIO

Folhas de Estilo em Cascata.....	3
Introdução ao CSS 3 .....	3
Seleção de elementos.....	6
Formas de inclusão no HTML .....	11
Usando a tag <link> .....	11
Usando a tag de estilo ( <i>style</i> ) .....	12
Usando estilos em linha ( <i>inline styles</i> ) ou embutidos .....	13
Pseudoclasses e pseudoelementos.....	14
Pseudoclasses.....	14
Pseudoelementos .....	15
Cores e background .....	17
Cores .....	18
Backgrounds .....	20
Texto, fontes e web fonts.....	21
Texto.....	22
Fontes.....	24
Web fonts .....	26
Referências.....	27



# FOLHAS DE ESTILO EM CASCATA

O CSS controla todos os aspectos visuais de um site, de cores a layouts. Mesmo o desenvolvedor mais experiente pode aprender algo novo quando se trata de usar e entender como o navegador interpreta o CSS. Assim, neste estudo você confere os fundamentos básicos do CSS e seus principais recursos. Na sequência, serão apresentadas as formas de seleção de elementos. Você conhecerá as principais formas de inclusão do CSS no HTML. Acompanhará a definição de pseudoclasses e pseudoelementos. Aprenderá a realizar a alteração de cores e background de elementos. E, por fim, receberá dicas para aplicar a formatação de textos, a utilização de fontes e a inclusão de web fonts.



O CSS (abreviação de *Cascading Style Sheets* ou Folhas de Estilo em Cascata) é a linguagem que usamos para dar estilo a um arquivo HTML, e dizer ao navegador como ele deve renderizar os elementos da página. Nestes estudos você aprenderá sobre estilo de documentos HTML, embora o CSS também possa ser usado para dar estilo a outras coisas.

Para aprender as especificidades da codificação em CSS continue avançando em seus estudos!

## Introdução ao CSS 3

Antes de prosseguir, você confere uma breve recapitulação da história do CSS. O CSS foi criado a partir da necessidade de criar páginas web com estilo. Antes da introdução do CSS, as pessoas queriam uma maneira de estilizar suas páginas web, que eram todas muito parecidas na época. Não se podia fazer muito em termos de personalização.



O HTML 3.2 introduziu a opção de definir cores em linha (*inline*) como atributos do elemento HTML, e tags de apresentação, mas isso se transformou rapidamente em uma situação longe do ideal.

O CSS nos permitiu mover tudo relacionado à apresentação do HTML para o CSS, para que o HTML pudesse voltar a ser o formato que define a estrutura do documento, em vez de como as coisas devem ficar no navegador. O CSS está evoluindo continuamente, permitindo que novas técnicas idiomáticas de CSS sejam incorporadas.



### CURIOSIDADE .....

É difícil imaginar os tempos em que o CSS nasceu e como a web era diferente. Na época, tínhamos vários navegadores concorrentes, sendo os principais o Internet Explorer e o Netscape Navigator. As páginas eram estilizadas usando HTML. Isso significava que você tinha uma quantidade limitada de possibilidades de personalização. A maior parte das decisões de estilo foi deixada para o navegador. Além disso, na época, era necessário construir um site especificamente para cada navegador, porque cada um introduzia tags diferentes que estavam fora do padrão. Posteriormente, os desenvolvedores perceberam a necessidade de uma maneira de estilizar as páginas, de uma forma que funcionasse em todos os navegadores.



Após a ideia inicial proposta em 1994, o CSS recebeu seu primeiro lançamento, em 1996, quando a recomendação CSS Nível 1 (CSS 1) foi publicada. O CSS Nível 2 (CSS 2) foi publicado em 1998. Desde então, começaram os trabalhos sobre o CSS Nível 3. O grupo de trabalho do CSS decidiu dividir cada recurso e trabalhar sobre ele separadamente, em módulos. Os navegadores não foram especialmente rápidos na implementação do CSS. Tivemos que esperar até 2002 para que o primeiro navegador implementasse a especificação completa do CSS: o Internet Explorer para Mac (CSS..., 2021; REFERÊNCIA..., [s.d.] ).



### Dica .....

Quer saber mais sobre o histórico do CSS, confira a descrição neste post do site CSS Tricks (Truques do CSS), publicado por Jay Hoffmann, clicando no link a seguir ou aproximando seu celular do QR Code:

<https://css-tricks.com/look-back-history-css>



O Internet Explorer implementou o recurso box model incorretamente desde o início, o que levou a anos de desafios ao tentar ter o mesmo estilo aplicado de forma consistente em todos os navegadores. Era necessário usar vários truques e ajustes para fazer os navegadores renderizarem as coisas como eram desejadas.

Após o CSS 2.1, o próprio CSS foi modularizado, de modo que cada módulo pode ser desenvolvido independentemente do resto. Isso permite não apenas que os módulos existentes sejam nivelados de forma independente, mas que novos módulos sejam criados a qualquer momento, definindo conjuntos completamente novos de recursos ou estendendo-se de recursos CSS 2.1 existentes (CSS..., 2021; REFERÊNCIA..., [s.d.]).

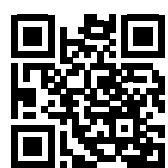
O CSS Nível 3 tem mais de 60 módulos que definem diferentes capacidades, algumas já fazem parte do padrão, outras ainda em desenvolvimento. O termo CSS 3 refere-se a tudo publicado após o CSS 2.1.



### Dica .....

Caso deseje navegar por um guia interativo gratuito que descreve os principais recursos do CSS acompanhados de exemplos, acesse a plataforma CSS Reference clicando no link a seguir ou aproximando seu celular do QR Code:

<https://cssreference.io/>





## Dica .....

+

+

+

Você ficará surpreso com as dicas e os exemplos de código disponibilizados! Como complemento, caso você deseje revisar os principais conceitos de HTML, acesse o guia HTML Reference, também interativo e ilustrado, clicando no link a seguir ou aproximando seu celular do QR Code:

<https://htmlreference.io/>



Com essas ferramentas em mãos, você dominará o assunto rapidamente!

## Seleção de elementos

Um arquivo CSS contém várias regras CSS. Cada regra é composta por duas partes: o seletor e o bloco de declaração. Perceba na imagem seguinte os componentes que fazem parte de uma regra CSS.



Figura 1 - Componentes de uma regra em CSS

Fonte: do Autor (2022)

O seletor é um recurso que identifica um ou mais elementos na página web, seguindo uma sintaxe especial sobre a qual discutiremos em breve. O bloco de declaração contém uma ou mais declarações, por sua vez compostas por uma propriedade e um valor de propriedade.

Neste exemplo, `<p>` é o seletor, e aplica uma regra que define o valor `20px` para a propriedade “`font-size`” (tamanho da fonte):

```
p {  
    font-size: 20px;  
}
```

Múltiplas regras podem ser empilhadas uma após a outra, entenda com outro exemplo:

```
p {  
    font-size: 20px;  
}  
  
a {  
    color: blue;  
}
```

Um seletor pode ter como alvo um ou mais itens, confira no exemplo:

```
p,  
a {  
    font-size: 20px;  
}
```



## Dica .....

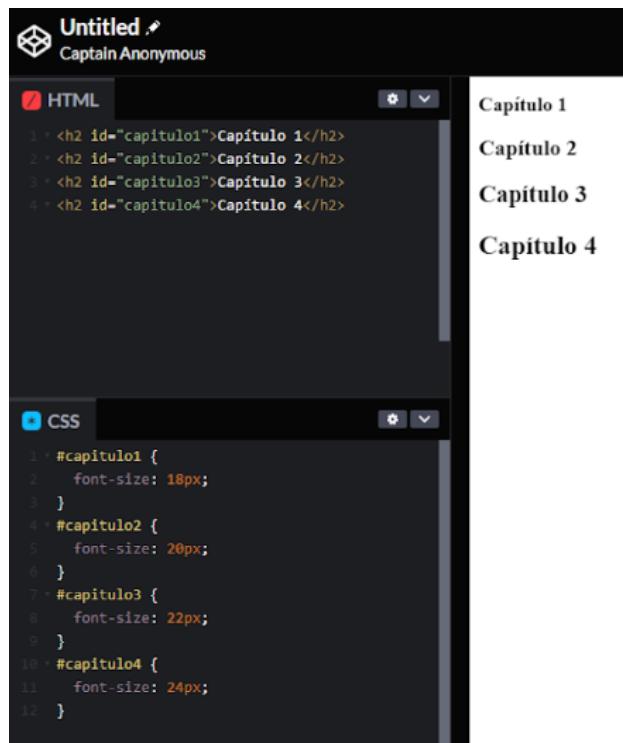
O seletor universal é utilizado para aplicar uma determinada estilização a todos os elementos apresentados em um documento HTML. O símbolo do seletor universal é “\*”.

É possível direcionar tags HTML a partir de elementos HTML que contenham um determinado atributo de classe, como “.minha-classe”, ou elementos HTML que tenham um atributo com um “id” específico, como “#meu-id”. Entenda com um exemplo:

```
•••  
.minha-classe {  
    color: orange;  
    font-weight: bold;  
}  
  
#meu-id {  
    font-size: 18px;  
}
```

Perceba que uma classe CSS é um atributo usado para definir um grupo de elementos HTML para aplicar estilo e formatação exclusivos a esses elementos com CSS. Já um seletor de “id” CSS usa o atributo “id” de um elemento HTML para selecionar um elemento exclusivo em uma página. Para usar um seletor de “id” em CSS, basta escrever uma hashtag (#) seguida do “id” do elemento. Em seguida, colocar as propriedades de estilo que deseja aplicar ao elemento.

A primeira regra a ter em mente ao usar o atributo “id” é que ele deve conter pelo menos um caractere e não pode começar com um número. Digamos que temos vários títulos <h2> em uma página web e cada um marca o início de um novo capítulo. Então, poderíamos dar a cada título <h2> um nome de “id”. A segunda regra a ser lembrada é que, se um elemento recebe um nome de “id”, ele deve ser exclusivo em uma página. Dessa forma, um seletor de “id” seleciona apenas um elemento exclusivo (CSS..., 2021; REFERÊNCIA..., [s.d.]). Para isso, confira um exemplo de um código CSS que alteraria o tamanho da fonte para cada título <h2>:



The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <h2 id="capítulo1">Capítulo 1</h2>
2 <h2 id="capítulo2">Capítulo 2</h2>
3 <h2 id="capítulo3">Capítulo 3</h2>
4 <h2 id="capítulo4">Capítulo 4</h2>
```

The 'CSS' tab contains the following code:

```
1 #capítulo1 {
2   font-size: 18px;
3 }
4 #capítulo2 {
5   font-size: 20px;
6 }
7 #capítulo3 {
8   font-size: 22px;
9 }
10 #capítulo4 {
11   font-size: 24px;
12 }
```

To the right of the editor, there is a vertical list of chapter titles corresponding to the IDs in the HTML code:

- Capítulo 1
- Capítulo 2
- Capítulo 3
- Capítulo 4

Figura 2 - Tipos de seletores em HTML

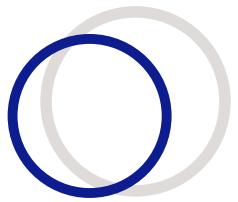
Fonte: do Autor (2022)



### Reflita .....

Não há uma regra fixa para a formatação. Embora seja recomendado que você siga algumas convenções, como as que viu nos exemplos acima: os seletores e as chaves de fechamento alinhados à esquerda, recuo de dois espaços para cada regra, inclusão da chave de abertura na mesma linha do seletor, separados por um espaço. O uso correto e consistente do espaçamento e do recuo é um auxílio visual na compreensão de seu código.

Até agora, você viu como destacar um elemento, uma classe ou uma identificação. Vamos introduzir seletores mais avançados. Você pode selecionar um elemento específico que tenha uma classe ou identificador vinculado. Confira um exemplo usando uma classe:



The screenshot shows a code editor window titled "Untitled" with the subtitle "Captain Anonymous". The top status bar displays "Carlos". The interface includes tabs for "HTML" and "CSS".

**HTML:**

```
<p class="nome-pessoa">Carlos</p>
```

**CSS:**

```
p.nome-pessoa {  
    color: blue;  
}
```

Figura 3 - Seleção de elemento a partir de uma classe

Fonte: do Autor (2022)

Agora, confira um exemplo utilizando um identificador:

The screenshot shows a code editor window titled "Untitled" with the subtitle "Captain Anonymous". The top status bar displays "Carlos". The interface includes tabs for "HTML" and "CSS".

**HTML:**

```
<p id="nome-pessoa">Carlos</p>
```

**CSS:**

```
p#nome-pessoa {  
    color: blue;  
}
```

Figura 4 - Seleção de elemento a partir de um identificador

Fonte: do Autor (2022)

Você pode acessar um elemento com duas ou mais classes, combinando os nomes das classes separados com um ponto, sem espaços. Por exemplo:

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. In the 'HTML' tab, there is a single line of code: 

```
<p class="nome-pessoa cadastro">Carlos</p>
```

. In the 'CSS' tab, there is a single rule: 

```
.nome-pessoa.cadastro { color: blue; }
```

. The word 'Carlos' in the HTML is colored blue, demonstrating the effect of the CSS rule.

Figura 5 - Selecionando elemento com duas ou mais classes  
Fonte: do Autor (2022)



### Atenção .....

Cada regra do CSS termina com um ponto e vírgula. Os pontos e vírgulas não são opcionais, exceto após a última regra. Mas sugerimos que sejam sempre usados para manter a consistência e para evitar erros, caso sejam acrescentadas outras propriedades e você esqueça de adicionar o ponto e vírgula na linha anterior.

## Formas de inclusão no HTML

O CSS é anexado a uma página HTML de diferentes maneiras. Confira algumas formas.

### Usando a tag <link>

A tag <link> é a maneira mais simples de incluir um arquivo CSS. Essa é a forma preferida de usar o CSS como ele se destina a ser usado: um arquivo CSS é incluído por todas as páginas de seu site, e mudar uma linha nesse arquivo afeta a apresentação de todas as páginas do site.

Para usar esse método, você adiciona uma tag de <link> com o atributo “href” apontando para o arquivo CSS que deseja incluir. Você a adiciona dentro da tag <head> do site. Entenda com um exemplo:

```
<!DOCTYPE html>
<html>

<head>
    <link rel="stylesheet" href="estilo.css">
</head>

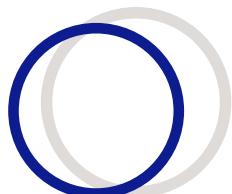
<body>
</body>

</html>
```

Os atributos “rel” e “type” também são necessários, pois eles dizem ao navegador qual tipo de arquivo estão se ligando.

## Usando a tag de estilo (**style**)

Em vez de usar a tag de <link> para apontar para uma folha de estilo separada contendo o arquivo CSS, podemos adicionar o CSS diretamente dentro de uma tag de estilo. Esta é a sintaxe:





The screenshot shows a code editor window titled "Untitled" with the subtitle "Captain Anonymous". The left pane is labeled "HTML" and contains the following code:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-color: linen;
}
h1 {
    color: maroon;
}
</style>
</head>
<body>
<h1>Este é o título</h1>
<p>Este é o parágrafo.</p>
</body>
</html>
```

The right pane displays the rendered HTML output:

Este é o título  
Este é o parágrafo.

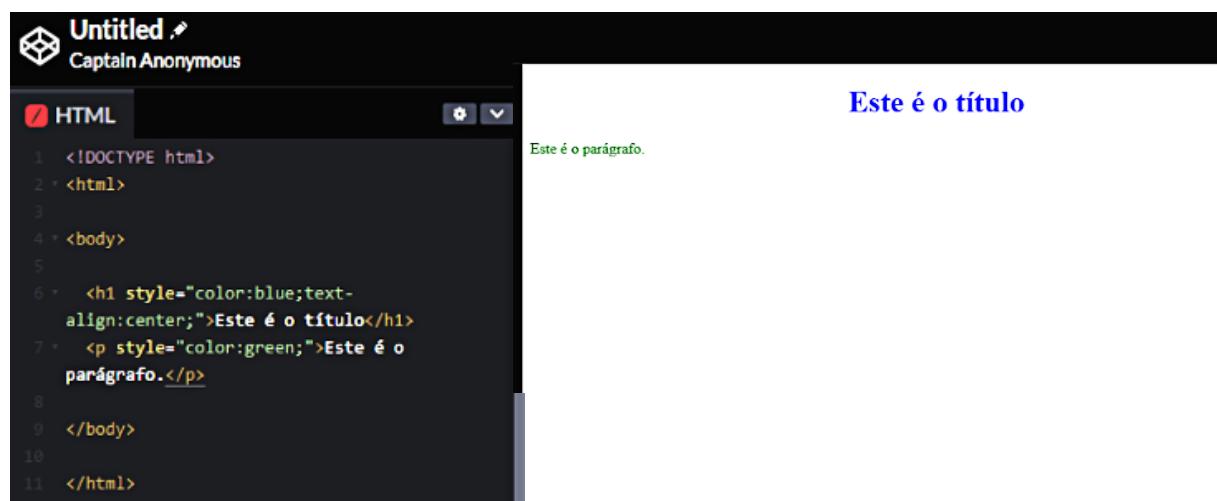
Figura 6 - Incluindo CSS pela tag de estilo

Fonte: do Autor (2022)

Usando esse método, podemos evitar criar um arquivo CSS separado. Esta é uma boa maneira de experimentar antes de “formalizar” o CSS em um arquivo separado ou adicionar uma linha especial de CSS apenas a um arquivo.

## Usando estilos em linha (*inline styles*) ou embutidos

Os estilos em linha são a terceira maneira de adicionar CSS a uma página. Podemos adicionar um atributo de estilo a qualquer tag HTML e adicionar CSS a ela. Perceba com um exemplo:



The screenshot shows a code editor window titled "Untitled" with the subtitle "Captain Anonymous". The left pane is labeled "HTML" and contains the following code:

```
<!DOCTYPE html>
<html>
<body>
<h1 style="color:blue;text-align:center;">Este é o título</h1>
<p style="color:green;">Este é o parágrafo.</p>
</body>
</html>
```

The right pane displays the rendered HTML output:

Este é o título  
Este é o parágrafo.

Figura 7 - Incluindo CSS pelos estilos em linha

Fonte: do Autor (2022)

# Pseudoclasses e pseudoelementos

Com a introdução do CSS 3, a diferença entre pseudoclasses e pseudoelementos é muito mais clara, pois agora é o padrão usar dois pontos (:) em pseudoelementos. Confira a seguir as principais diferenças entre esses recursos do CSS.

## Pseudoclasses

Pseudoclasses são palavras-chave predefinidas, utilizadas para selecionar um elemento com base em seu estado ou para selecionar um elemento-filho específico. Elas começam com um único sinal de dois pontos (:).

Podem ser usadas como parte de um seletor, e são muito úteis para criar links ativos (active) ou visitados (visited), por exemplo, mudar o estilo quando o mouse está próximo do elemento (hover) ou manter o foco no elemento (focus). Muito úteis em diversos casos. Confira na sequência as pseudoclasses mais populares que você provavelmente irá usar (CSS..., 2021; REFERÊNCIA..., [s.d.]):

Pseudoclasse	Descrição
:active	Um elemento sendo ativado pelo usuário (por exemplo, clicado). Mais utilizado em links ou botões.
:checked	Uma caixa de seleção (checkbox), opção (option) ou tipos de entrada de rádio (radio) que são habilitados.
:default	O padrão em um conjunto de escolhas (por exemplo, opção em um botão de seleção (option) ou rádio (radio)).
:disabled	Um elemento desabilitado.
:enabled	Um elemento que é habilitado.
:focus	Um elemento com foco.
:hover	Um elemento posicionado pelo mouse.
:link	Um link que não foi visitado.
:required	Um elemento do formulário com o conjunto de campos necessários.
:visited	Um link que foi visitado.

Quadro 1 - Pseudoclasses

Fonte: do Autor (2022)

Na sequência, confira um exemplo comum. Você quer estilizar um link, então cria uma regra CSS para direcionar o elemento:

```
a {  
    color: red;  
}
```

Tudo parece funcionar bem, até você clicar em um link. O link que era vermelho volta para a cor predefinida (azul) quando recebe um clique. Então, quando você abre o link e volta à página, percebe que o link ainda continua azul e não vermelho, conforme foi estilizado. Por que isso acontece? Porque o link, quando clicado, muda de estado, indo para o “:active”. E quando é visitado, fica no estado “:visited” para sempre, até que o usuário limpe o histórico de navegação. Então, para tornar o link vermelho em todos os estados, você precisa escrever o seguinte código:

```
a,  
a:visited,  
a:active {  
    color: red;  
}
```

## Pseudoelementos

Pseudoelementos são usados para estilizar uma parte específica de um elemento. Eles começam com duas sequências de dois pontos (::). Às vezes, você os identifica em outros códigos com um único símbolo de dois pontos (:), mas essa é apenas uma sintaxe suportada por razões de compatibilidade com o passado de alguns navegadores. Assim, você deve usar dois símbolos de dois pontos (::) para distingui-los das pseudoclasses.





Os pseudoelementos mais usados, provavelmente, são “::before” e “::after”. Eles são usados para adicionar conteúdo antes ou depois de um elemento, como ícones, por exemplo. Estude, a seguir, uma pequena lista dos pseudoelementos (CSS..., 2021; REFERÊNCIA..., [s.d.]):

Pseudoelemento	Objetivo
::after	Cria um pseudoelemento após o elemento.
::before	Cria um pseudoelemento antes do elemento.
::first-letter	Pode ser usado para estilizar a primeira letra de um bloco de texto.
::first-line	Pode ser usado para estilizar a primeira linha de um bloco de texto.
::selection	Destaca o texto selecionado pelo usuário.

Quadro 2 - Pseudoelementos

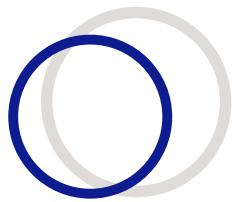
Fonte: do Autor (2022)

Vamos fazer um exemplo. Nele, você verá que a primeira linha de um parágrafo ficará ligeiramente maior em tamanho de fonte, algo comum em tipografia:

```
p::first-line {  
    font-size: 3rem;  
}
```

Nesse outro exemplo, você perceberá que a primeira letra da expressão ficará negritada:

```
p::first-letter {  
    font-weight: bolder;  
}
```



## Cores e background

O CSS é uma linguagem que permite criar belas páginas da web. Graças à sintaxe de cor de texto CSS, você pode determinar a cor exata do texto em sua página da web. Essa sintaxe é essencial porque aumenta a usabilidade e a acessibilidade do seu site.



A propriedade CSS “color” é usada para selecionar a cor do texto, a cor do plano de fundo da página da web e a cor das bordas. Por outro lado, a propriedade “background-color” especifica a cor de fundo de um elemento. Essa propriedade abrange todo o tamanho do elemento, incluindo preenchimento e borda. No entanto, não inclui margem.

# Cores

Por padrão, uma página HTML é renderizada pelos navegadores da web com muita carência em termos das cores utilizadas. Temos um fundo branco, textos na cor preta e links azuis. Felizmente, o CSS nos dá a capacidade de adicionar cores aos nossos designs. Nós temos estas propriedades: "color", "background-color" e "border-color". Todas elas aceitam um valor de cor, que pode ser de diferentes formas. Você confere esses tipos na sequência.

## Cores nomeadas

Primeiro, temos as palavras-chave CSS que definem as cores. O CSS começou com 16, mas hoje existe um grande número de nomes de cores. São insensíveis a maiúsculas e minúsculas. Confira um exemplo da relação de cores clicando no link a seguir ou aproximando seu celular do QR Code:



<https://htmlcolorcodes.com/color-names/>

## RGB

RGB é outro modelo de cores baseado na combinação das cores primárias vermelho, verde e azul. Composto por três números separados por vírgulas, cada um representa a intensidade da respectiva cor primária como um número inteiro entre zero e 255. Preto é `rgb(0, 0, 0)`, vermelho é `rgb(255, 0, 0)` e azul é `rgb(0, 0, 255)`. Você pode usar a função "rgb()" para calcular uma cor a partir de sua notação RGB, entenda com o código:



```
p {  
    color: rgb(255, 255, 255); /* branco */  
    background-color: rgb(0, 0, 0); /* preto */  
}
```

## Notação hexadecimal

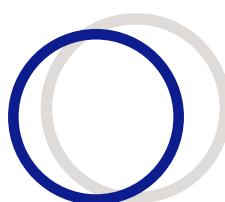
Outra opção é expressar as partes RGB das cores na notação hexadecimal, que é composta por três blocos. Esses códigos são compostos por três pares de caracteres que representam a intensidade das três cores primárias. Os valores possíveis variam de 00 (a menor intensidade de uma cor primária) a FF (a maior intensidade de uma cor primária). O código de cor hexadecimal para preto é #000000, vermelho é #FF0000 e azul é #0000FF. O preto, que é `rgb(0, 0, 0)` é expresso como #000000. Branco, `rgb(255, 255, 255)`, pode ser expresso como #FFFFFF. Perceba pela imagem seguinte um exemplo contendo uma tabela de cores nomeadas e seus valores correspondentes em hexadecimal.

#000000	#330000	#660000	#990000	#CC0000	#FF0000
#003300	#333300	#663300	#993300	#CC3300	#FF3300
#006600	#336600	#666600	#996600	#CC6600	#FF6600
#009900	#339900	#669900	#999900	#CC9900	#FF9900
#00CC00	#33CC00	#66CC00	#99CC00	#CCCC00	#FFCC00
#00FF00	#33FF00	#66FF00	#99FF00	#CCFF00	#FFFF00
#000033	#330033	#660033	#990033	#CC0033	#FF0033
#003333	#333333	#663333	#993333	#CC3333	#FF3333
#006633	#336633	#666633	#996633	#CC6633	#FF6633
#009933	#339933	#669933	#999933	#CC9933	#FF9933
#00CC33	#33CC33	#66CC33	#99CC33	#CCCC33	#FFCC33
#00FF33	#33FF33	#66FF33	#99FF33	#CCFF33	#FFFF33
#000066	#330066	#660066	#990066	#CC0066	#FF0066
#003366	#333366	#663366	#993366	#CC3366	#FF3366
#006666	#336666	#666666	#996666	#CC6666	#FF6666
#009966	#339966	#669966	#999966	#CC9966	#FF9966
#00CC66	#33CC66	#66CC66	#99CC66	#CCCC66	#FFCC66
#00FF66	#33FF66	#66FF66	#99FF66	#CCFF66	#FFFF66
#000099	#330099	#660099	#990099	#CC0099	#FF0099
#003399	#333399	#663399	#993399	#CC3399	#FF3399
#006699	#336699	#666699	#996699	#CC6699	#FF6699
#009999	#339999	#669999	#999999	#CC9999	#FF9999
#00CC99	#33CC99	#66CC99	#99CC99	#CCCC99	#FFCC99
#00FF99	#33FF99	#66FF99	#99FF99	#CCFF99	#FFFF99
#0000CC	#3300CC	#6600CC	#9900CC	#CC00CC	#FF00CC
#0033CC	#3333CC	#6633CC	#9933CC	#CC33CC	#FF33CC
#0066CC	#3366CC	#6666CC	#9966CC	#CC66CC	#FF66CC
#0099CC	#3399CC	#6699CC	#9999CC	#CC99CC	#FF99CC
#00CCCC	#33CCCC	#66CCCC	#99CCCC	#CCCCCC	#FFCCCC
#00FFCC	#33FFCC	#66FFCC	#99FFCC	#CCFFCC	#FFFFCC
#0000FF	#3300FF	#6600FF	#9900FF	#CC00FF	#FF00FF
#0033FF	#3333FF	#6633FF	#9933FF	#CC33FF	#FF33FF
#0066FF	#3366FF	#6666FF	#9966FF	#CC66FF	#FF66FF
#0099FF	#3399FF	#6699FF	#9999FF	#CC99FF	#FF99FF
#00CCFF	#33CCFF	#66CCFF	#99CCFF	#CCCCFF	#FFCCFF
#00FFFF	#33FFFF	#66FFFF	#99FFFF	#CCFFFF	#FFFFFF

Figura 8 - Tabela de cores hexadecimais

Fonte: Linhadecódigo ([s.d.])

Embora você possa usar qualquer um desses valores, nomes de cores não são recomendados. Eles são difíceis de lembrar, mas também introduzem imprecisão. Use códigos de cores hexadecimais ou valores RGB para garantir que o esquema de cores do seu site tenha a aparência desejada. Eles permitem que você escolha o tom exato da cor desejada.



# Backgrounds

O fundo de um elemento pode ser alterado utilizando várias propriedades do CSS, confira alguns exemplos (CSS..., 2021; REFERÊNCIA..., [s.d.]):

Propriedade	Descrição
background-color	Define a cor do plano de fundo do elemento.
background-image	Define uma imagem como plano de fundo do elemento.
background-clip	Define até onde o plano de fundo deve se estender dentro do elemento.
background-position	Define a posição da imagem de fundo.
background-origin	Define a origem da imagem de fundo.
background-repeat	Define como a imagem de fundo se repete no fundo do elemento, a partir da posição do fundo.
background-attachment	Define como a imagem de fundo se comportará ao rolar a página.
background-size	Define o tamanho da imagem de fundo.

Quadro 3 - Backgrounds

Fonte: do Autor (2022)

A propriedade “background-color” aceita um valor de cor, que pode ser uma das palavras-chave de cor ou um valor RGB:

```
● ● ●  
p {  
    background-color: red;  
}  
  
div {  
    background-color: #333;  
}
```



Ao invés de usar uma cor, você pode usar uma imagem como fundo para um elemento, especificando o URL de localização da imagem a partir da propriedade "background-image":

```
div {  
    background-image: url(imagem.png);  
}
```

O CSS lhe dá a possibilidade de escrever comentários em um arquivo CSS ou na tag de estilo (style), no cabeçalho da página. A sintaxe é: /\* Este é um comentário \*/. Confira um exemplo:

```
#nome {  
    display: block;  
} /* Aqui vem a descrição do comentário! */  
/* #nome { display: block; } */  
#nome {  
    display: block;  
/* color: blue; */  
}
```

## Texto, fontes e web fonts

O CSS fornece várias propriedades que permitem definir vários estilos de texto como cor, alinhamento, espaçamento, decoração, transformação etc. de forma muito fácil e eficaz. Além disso, permite definir diversas propriedades relacionadas à fonte e como os recursos de fonte são carregados. Ele permite definir o estilo de uma fonte, como sua família, tamanho e peso, entre outras características. Por fim, o CSS permite que fontes personalizadas sejam carregadas em uma página da web. Uma vez adicionada a uma folha de estilo, a regra instrui o navegador a baixar a fonte de onde ela está hospedada e exibi-la conforme especificado no CSS.

Outro detalhe é que uma das coisas que você vai usar todos os dias no CSS são unidades. Elas são usadas para definir comprimentos, espaçamentos, margens, elementos de alinhamento e assim por diante. Coisas como “px”, “em”, “rem” ou porcentagens. Elas estão em toda parte. Vamos conhecer algumas delas.

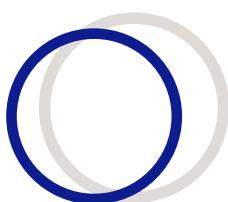
Pixels é a unidade de medição mais amplamente utilizada. Um pixel não se correlaciona realmente com um pixel físico em sua tela, pois isso varia muito por dispositivo. Há uma convenção que faz com que essa unidade funcione consistentemente em todos os dispositivos. Outra medida muito útil são as porcentagens, que permitem especificar valores em porcentagens da propriedade correspondente daquele elemento pai.

A unidade “em” é o valor atribuído ao tamanho da fonte desse elemento, portanto seu valor exato muda entre os elementos. Ele não muda dependendo da fonte utilizada, apenas do tamanho da fonte. Na tipografia, isso mede a largura da letra “m”. Já a unidade “rem” é semelhante à unidade “em”, mas em vez de variar no tamanho da fonte do elemento atual, ele usa o tamanho da fonte do elemento raiz (HTML). Você define o tamanho da fonte uma vez, e o “rem” será uma medida consistente em toda a página. Conheça alguns exemplos:

```
.setor {  
    margin: 10%;  
    font-size: 0.5em;  
}  
  
.cabecalho {  
    font-size: 2rem;  
}
```

## Texto

Nesta seção vamos falar sobre as seguintes propriedades relacionadas à formatação de texto utilizando o CSS (CSS..., 2021; REFERÊNCIA..., [s.d.]):



Propriedade	Descrição	Exemplo
text-transform	Essa propriedade pode transformar o texto de um elemento. Existem quatro valores válidos: "capitalize", "uppercase", "lowercase" e "none".	p { text-transform: uppercase; }
text-decoration	Essa propriedade é usada para adicionar decorações ao texto, incluindo, por exemplo: "underline", "overline" e "none".	p { text-decoration: underline; }
text-align	Por padrão, essa propriedade tem o valor "start", ou seja, o texto começa na origem 0 da caixa que o contém. Os valores possíveis são: "start", "end", "left", "right", "center" e "justify".	p { text-align: right; }
vertical-align	Determina como os elementos embutidos são alinhados verticalmente. Podemos atribuir um valor de comprimento ou porcentagem. Além disso, tem-se as palavras-chave: "baseline", "sub", "super", "top", "text-top", "middle", "bottom" e "text-bottom".	p { vertical-align: text-bottom; }
line-height	Isso permite que você altere a altura de uma linha. Cada linha de texto tem uma certa altura de fonte, mas há espaçamento vertical adicional entre as linhas.	p { line-height: 0.9rem; }
text-indent	Recuar a primeira linha de um parágrafo por um comprimento definido ou uma porcentagem da largura do parágrafo.	p { text-indent: -10px; }
text-align-last	Por padrão, a última linha de um parágrafo é alinhada seguindo o valor "text-align".	p { text-align-last: right; }
word-spacing	Modifica o espaçamento entre cada palavra.	p { word-spacing: 30px; }

letter-spacing	Modifica o espaçamento entre cada letra.	<pre>p {     letter-spacing: 0.2px; }</pre>
text-shadow	Aplica uma sombra ao texto.	<pre>span {     text-shadow: red 0.2px     2px 3px; }</pre>
white-space	Define como o CSS lida com o espaço em branco, novas linhas e guias dentro de um elemento. Os valores válidos que reduzem o espaço em branco são: "normal", "nowrap" e "preline".	<pre>span {     white-space: preline; }</pre>
tab-size	Define a largura do caractere de tabulação. Por padrão, é 8, e você pode determinar um valor inteiro que define os espaços de caracteres necessários ou um valor de comprimento.	<pre>p {     tab-size: 2; }</pre>
direction	Define a direção do texto. Os valores válidos são "ltr" e "rtl".	<pre>p {     direction: rtl; }</pre>

Quadro 4 - Propriedades de texto

Fonte: do Autor (2022)

## Fontes

No início da web, você tinha poucas possibilidades de fontes que podia aplicar em seus projetos. Felizmente, hoje você pode carregar qualquer tipo de fonte em suas páginas. O CSS ganhou muitas e boas capacidades ao longo dos anos no que diz respeito às fontes. A propriedade das fontes é a abreviação de uma série de propriedades, confira (CSS..., 2021; REFERÊNCIA..., [s.d.]):



Propriedade	Descrição	Exemplo
font-family	Define a família de fontes que o elemento irá usar. Nós as dividimos em fontes Serif, Sans-Serif e Monospace. Aqui está uma lista de algumas das mais populares: Serif: Georgia, Times New Roman; Sans-Serif: Arial, Helvetica, Verdana, Tahoma; Monospace: Courier New, Lucida Console.	body { font-family: Helvetica; }
font-weight	Essa propriedade define a largura de uma fonte. Você pode usar esses valores predefinidos: "normal", "bold", "bolder" e "lighter" as palavras-chave numéricas variando de 100 até 900, em que 100 é a fonte mais fina e 900 é a mais espessa.	p { font-weight: bold; }
font-style	Permite aplicar um estilo itálico (italic) a uma fonte. Essa propriedade também permite os valores oblíquos (oblique) e normal (normal).	p { font-style: italic; }
font-size	Essa propriedade é usada para determinar o tamanho das fontes. Você pode passar dois tipos de valores: um valor de comprimento, como "px", "em", "rem" etc. ou uma porcentagem; uma palavra-chave de valor predefinido como, por exemplo: "xx-small", "x-small", "small" e "large".	p { font-size: 20px; }  li { font-size: large; }
font	A propriedade da fonte permite aplicar diferentes propriedades de fonte em uma única, reduzindo a desordem. Devemos definir pelo menos duas propriedades, "font-size" e "font-family", as outras são opcionais.	body { font: 20px Helvetica; }

Quadro 5 - Propriedades de fontes

Fonte: do Autor (2022)

## Web fonts

Tecnicamente falando, você pode adicionar praticamente qualquer fonte a qualquer site. No entanto, ao pesquisar uma fonte on-line, você sempre deve ter em mente o preço, o contrato de licença e o método de instalação.

A biblioteca mais conhecida é do Google, denominada de Google Fonts. Conheça clicando no link a seguir ou aproximando seu celular do QR Code:



<https://fonts.google.com>

As fontes do Google possuem uma biblioteca impressionante com diversas fontes de licença livre que podem ser navegadas por meio de seu diretório da web interativo. Por causa de sua plataforma, as fontes do Google são as mais fáceis de adicionar ao seu site. Para usar uma fonte do Google em seu projeto web, é necessário que você copie o código de importação relacionado à fonte desejada, insira no seu arquivo CSS e aplique a fonte escolhida a partir das regras CSS já apresentadas.



### Dica .....



Quer saber mais detalhes de como utilizar fontes web em seus projetos? Acesse o site clicando no link a seguir ou aproximando seu celular do QR Code e confira um guia completo que irá lhe auxiliar nesta demanda.

<https://kinsta.com/pt/blog/html-fontes/>



Durante seus estudos, você pôde acompanhar que o CSS é uma linguagem de marcação utilizada para apresentar páginas web, envolvendo formatações, cores, layout e fontes, tornando nossas páginas da web apresentáveis aos usuários. Assim, o CSS é o que dá ao seu site todo o seu estilo. Essa linguagem afeta toda a aparência de uma página web, fazendo com que seja uma ferramenta importante e uma habilidade requisitada entre os desenvolvedores web.



Assim, você acompanhou os fundamentos básicos do CSS. Aprendeu sobre quais são as formas de seleção de elementos e dicas de como incluir arquivos CSS no HTML. Você conheceu os principais tipos de pseudoclasses e pseudoelementos. Viu vários modos de realizar a alteração de cores e background de elementos. E, por fim, compreendeu como realizar a formatação de textos e fontes e como incluir web fonts em seus projetos. Dessa forma, avance em seus estudos, pois existem diversos conteúdos complementares disponibilizados na sequência! Bons estudos!

## REFERÊNCIAS

CSS Snapshot 2021. **W3**, 2021. Disponível em: <https://www.w3.org/TR/css-2021/>. Acesso em: 12 jul. 2022.

REFERÊNCIA de CSS. **MDN**, [s.d.]. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/CSS/Reference>. Acesso em: 12 jul. 2022.



