



# SELETORES PERSONALIZADOS EM CSS



M02C15

# SELETORES PERSONALIZADOS EM CSS

Agora que nós já passamos pelos elementos mais importantes do design, que são as cores, as imagens e as fontes, vamos falar um pouco sobre formatações especiais que vão permitir organizar esses conteúdos na tela, indicando seus tamanhos, espaçamentos, sombras e tudo mais. Eu te convido a aprender maneiras que vão facilitar no desenvolvimento web de sites eficientes e bonitos.



Você tem todo o direito de usar esse material para seu próprio aprendizado. Professores também podem ter acesso a todo o conteúdo e usá-los com seus alunos. Porém todos o que usarem esse material - seja para qual for a finalidade - deverão manter a referência ao material original, criado pelo **Prof. Gustavo Guanabara** e disponível no endereço do seu repositório público [https:// github.com/gustavoguanabara/](https://github.com/gustavoguanabara/). Este material não poderá ser utilizado em nenhuma hipótese para ser replicada - integral ou parcialmente - por autores/editoras para criar livros ou apostilas, com finalidade de obter ganho financeiro com ele.

# Personalizando Seletores

Para começar a dar mais poder às CSS, criando estilos personalizados, precisamos aprender a utilizar os **seletores de id (#)** e de **class (.)** de maneira eficiente. Ao criar nosso conteúdo em HTML, podemos **identificar** um determinado **elemento único** com um **id**, ou **agrupar** elementos múltiplos que tenham **características semelhantes** com um **class**. Vamos olhar um exemplo simples de documento HTML, com propriedades identificadoras:



```
22 <body>
23   <h1 id="titulo-principal">Aprenda Desenvolvimento Web</h1>
24   <h1>Aprenda HTML</h1>
25   <h2 class="topico">Semântica Web</h2>
26   <p class="texto">Lorem ipsum dolor sit, amet consectetur adipisicing elit.
    Optio, quibusdam? Tempore sequi commodi natus, vitae enim sed molestias. Rem,
    voluptas illum. Ut, debitis error quas explicabo corporis officia fugit.
    Doloribus.</p>
27   <h1>Aprenda CSS</h1>
28   <h2 class="topico">Estilos Web</h2>
29   <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Dolores, repellat
    dicta. In distinctio quasi nesciunt blanditiis. Quas expedita et architecto
    alias laudantium, facere neque quo facilis! Ut est perferendis sed!</p>
30 </body>
```

Olhando atentamente para a **linha 23**, vimos que o `<h1>` principal se diferencia dos demais que estão nas **linhas 24 e 27**, pois ele possui um `id`.

Agora foque sua atenção nas **linhas 25 e 28** e perceba que os dois títulos `<h2>` possuem o mesmo `class`.

Um `id` vai **identificar** um **elemento único** dentro da página atual. Essa identificação vai nos permitir criar um estilo especial para um elemento isolado. Já um `class` vai **identificar** uma **classe** à qual um ou mais elementos pertençam, compartilhando características em comum a todos os que façam parte desse grupo.



**SE LIGA NA REGRA:** Em um **mesmo documento HTML**, só podemos **usar um id** para **um único elemento**, o que significa que não podemos ter dois elementos com um mesmo `id` dentro de uma mesma página. Porém, podemos atribuir um mesmo `class` para vários elementos que possuam essa mesma característica.

Em seguida, baseado no código HTML visto anteriormente, vamos criar um estilo local criando uma área `<style>` dentro da seção `<head>` do documento atual.

```

7      <style>
8      body {
9          font: normal 1em Arial, Helvetica, sans-serif;
10     }
11
12     h1 {
13         font-size: 2em;
14         color: darkgreen;
15     }
16
17     h2 {
18         color: green;
19     }
20     </style>

```

O que fizemos foi criar configurações simples de estilo baseadas nos elementos `body`, `h1` e `h2` do nosso HTML. Até o momento, não fizemos nenhuma configuração personalizada para nenhum seletor. O resultado é bem previsível:



Perceba no resultado ao lado que os dois primeiros títulos possuem apresentações idênticas, mesmo que o primeiro deles possua um `id`.

Isso aconteceu porque não fizemos nenhuma configuração específica para ele. Não adianta colocar apenas `id` ou `class` em um elemento e não personalizar o resultado visual usando seletores personalizados.

Agora vamos adicionar algumas declarações e seletores personalizados, ainda dentro da área `<style>` que criamos:

No código ao lado, criamos **seletores** com **simbologias novas**. Isso porque não são seletores para elementos HTML, e sim **seletores** para **id** e **class**. Na **linha 21**, criamos um seletor personalizado para o identificador **titulo-principal**, que está atribuído ao `h1`. Você deve ter percebido que colocamos um símbolo **#** antes do `id`.

Já as **linhas 27 e 31** são seletores que receberão declarações para as classes `topico` e `texto`. Todos os seletores personalizados para uma `class` são identificados por um ponto que vem antes do nome da classe.

```

21     #titulo-principal {
22         background-color: darkgreen;
23         color: white;
24         text-align: center;
25     }
26
27     .topico {
28         text-align: right;
29     }
30
31     .texto {
32         text-align: justify;
33     }

```



**MAIS UMA REGRA:** Todo `id` em HTML é identificado nas CSS por um símbolo `#`. Toda `class` em HTML é identificada nas CSS por um ponto.

Ao adicionar os seletores personalizados ao nosso estilo local, o resultado fica bem diferente:

ANTES...	...DEPOIS
<h1>Aprenda Desenvolvimento Web</h1> <h2>Aprenda HTML</h2> <h3>Semântica Web</h3> <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Optio, quibusdam? Tempore sequi commodi natus, vitae enim sed molestias. Rem, voluptas illum. Ut, debitis error quas explicabo corporis officia fugit. Doloribus.</p> <h2>Aprenda CSS</h2> <h3>Estilos Web</h3> <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Dolores, repellat dicta. In distinctio quasi nesciunt blanditiis. Quas expedita et architecto alias laudantium, facere neque quo facilis! Ut est perferendis sed!</p>	<h1>Aprenda Desenvolvimento Web</h1> <h2>Aprenda HTML</h2> <h3>Semântica Web</h3> <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Optio, quibusdam? Tempore sequi commodi natus, vitae enim sed molestias. Rem, voluptas illum. Ut, debitis error quas explicabo corporis officia fugit. Doloribus.</p> <h2>Aprenda CSS</h2> <h3>Estilos Web</h3> <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Dolores, repellat dicta. In distinctio quasi nesciunt blanditiis. Quas expedita et architecto alias laudantium, facere neque quo facilis! Ut est perferendis sed!</p>

Analise os dois resultados acima e perceba as diferenças. Note que apenas o título `h1` identificado como `titulo-principal` ganhou uma formatação diferente e os demais títulos de mesmo nível se mantiveram como antes. Inclusive, compare o resultado diferente obtido entre o primeiro e o segundo parágrafos. Apenas o primeiro ficou justificado. Isso foi por conta da aplicação de uma `class` apenas ao primeiro (volte lá no código HTML inicial e confira que o segundo parágrafo não possui `class`).

Soluções digitais  
para negócios

hostnet



# Propriedades herdadas



No **Capítulo 12**, discutimos três técnicas usadas para aplicar estilos aos nossos documentos HTML: **inline**, **local** e **externa**. Durante os nossos vídeos de acompanhamento do material em PDF, demonstramos que as três técnicas podem ser usadas em conjunto, e os estilos de cada serão combinados nessa ordem:

1º lugar: CSS externo

2º lugar: CSS interno/local

3º lugar: CSS inline

Sendo assim, caso haja duplicidade nas configurações, o que fica valendo no final é a propriedade especificada nas configurações inline. Cada camada de aplicação de estilos vai adicionando propriedades novas e sobrescrevendo as configurações do nível anterior.

Quando nós aplicamos configurações personalizadas de `id` ou de `class`, também vai existir uma combinação de configurações, como você já deve ter percebido ao analisar o exemplo que apresentamos nas páginas anteriores.

Voltando ao exemplo dado anteriormente (que vou replicar aqui para facilitar o entendimento), logo no início do corpo, tínhamos dois elementos `h1`:

```
<body>
  <h1 id="titulo-principal">Aprenda Desenvolvimento Web</h1>
  <h1>Aprenda HTML</h1>
```

Logo em seguida, na área de estilo, criamos uma configuração especial para os elementos `h1`:

```
h1 {
  font-size: 2em;
  color: darkgreen;
}
```

Essas duas configurações de tamanho e cor da fonte serão aplicadas aos dois títulos, já que ambos são do tipo `h1`. Por fim, adicionamos uma configuração específica para o elemento que tem um `id` personalizado:

```
#titulo-principal {
  background-color: darkgreen;
  color: white;
  text-align: center;
}
```

Dessa maneira, o primeiro `h1` do código HTML (aquele que tem um `id`) ganhará mais três configurações - cor de fundo, cor da fonte e alinhamento do texto - graças a esse seletor extra. Note que na primeira configuração CSS dos elementos `h1`, dissemos que a cor era `darkgreen`.

Porém, quando aplicamos a segunda configuração de `color` no seletor `#titulo-principal`, ela vai sobrescrever o valor da propriedade para `white`. As outras duas



propriedades (background-color e text-align) serão adicionadas às configurações desse primeiro título. O segundo título não sofrerá alterações, já que apenas o primeiro possui o id referido no seletor.

**Aprenda Desenvolvimento Web**

**Aprenda HTML**

## Pseudo-classes e pseudo-elementos

Uma pseudo-classes CSS é uma palavra-chave adicionada às declarações de um seletor após um sinal de dois pontos e especificam um estado especial de um elemento.

Existem várias pseudo-classes para estilos, podemos

citar :hover, :visited, :active, :checked, :empty e :focus.

Já um pseudo-elemento CSS é uma palavra-chave adicionada às declarações de um seletor após dois sinais de dois pontos e permitem que você formate um pedaço específico do elemento referenciado. Os principais pseudo-elementos usados nas CSS são ::before, ::after, ::first-letter, ::first-line.

Vamos começar criando um exemplo bem simples e bastante usado para exemplificar o uso de pseudo-classes e pseudo-elementos: a personalização de links.

```
43 <body>
44   <h1>Personalizando um link</h1>
45   <p>
46     Para aprender HTML5 e CSS3, acesse o
47     <a href="https://gustavoguanabara.github.io">
48       |   GitHub do Guanabara
49     </a>
50   </p>
51   <p>
52     Se quiser conhecer mais sobre os padrões, visite o site do
53     <a href="https://www.w3c.br">Escritório do W3C no Brasil</a>
54   </p>
55   <p>Para acompanhar os cursos, acesse o
56     <a href="https://www.youtube.com/cursoemvideo/" class="especial">
57       |   canal do CursoemVideo
58     </a>
59   </p>
60 </body>
```

Começando pelo corpo do documento, contendo só o HTML, vamos criar um texto com um título e três parágrafos, com três links. Note que o último link recebeu a atribuição de uma classe específica. O resultado visual está sendo apresentado a seguir:

# Personalizando um link

Para aprender HTML5 e CSS3, acesse o [GitHub do Guanabara](#)

Se quiser conhecer mais sobre os padrões, visite o site do [Escritório do W3C no Brasil](#)

Para acompanhar os cursos, acesse o [canal do CursoemVideo](#)

Na imagem acima, o primeiro link apareceu com a cor violeta pois já tínhamos visitado o perfil do GitHub anteriormente. Por padrão, os navegadores mostram links inéditos em azul e os visitados em violeta. Todos os links também aparecem sublinhados por padrão.

Vamos alterar essa apresentação padrão com nossas configurações de estilo. Crie a área `<style>` dentro de `<head>` e coloque os seguintes seletores.

```
8      body {
9          font: normal 1em Arial, Helvetica, sans-serif;
10     }
11
12     a {
13         font-weight: bold;
14         text-decoration: none;
15         color: red;
16     }
17
18     a:visited {
19         color: darkred;
20     }
21
22     a:hover {
23         text-decoration: underline;
24     }
```

Nas declarações acima, criamos três configurações para os links: a primeira (**linha 12**) para links inéditos (não visitados), colocando o texto em negrito, removendo o sublinhado e colocando-os em cor vermelha.

Cursos que vão te levar  
ao próximo nível

 estudonauta





No segundo seletor para links (**linha 18**), configuramos as âncoras já visitadas (com a pseudo-classe `:visited`). Nele, apenas mudamos a cor para vermelho escuro.

Já na terceira declaração (**linha 22**), fizemos configurações para todos os links, quando passarmos o mouse por cima (pseudo-classe `:hover`) e o sublinhado volta a aparecer.

O resultado visual está apresentado abaixo, compare com o resultado anterior e veja as diferenças:

## Personalizando um link

Para aprender HTML5 e CSS3, acesse o [GitHub do Guanabara](#)

Se quiser conhecer mais sobre os padrões, visite o site do [Escritório do W3C no Brasil](#)

Para acompanhar os cursos, acesse o [canal do CursoemVideo](#)

Por fim, vamos adicionar algumas configurações relacionadas à classe especial, criada no documento HTML, para o terceiro link.

```
26     .especial:hover {
27         color: white;
28         background-color: black;
29         text-decoration: none;
30     }
31
32     .especial::before {
33         content: '»';
34         font-weight: lighter;
35     }
36
37     .especial::after {
38         content: '«';
39         font-weight: lighter;
40     }
```

Na primeira declaração do código acima (**linha 26**), dizemos que o link da classe especial vai ter letra branca, fundo preto e perderá o sublinhado apenas quando movermos o mouse sobre ele.

Nas próximas declarações (**linhas 32 e 37**), vamos adicionar um símbolo » antes e outro símbolo « depois usando os pseudo-elementos `::before` e `::after`, respectivamente. O resultado dessas declarações está apresentado a seguir:

# Personalizando um link

Para aprender HTML5 e CSS3, acesse o [GitHub do Guanabara](#)

Se quiser conhecer mais sobre os padrões, visite o site do [Escritório do W3C no Brasil](#)

Para acompanhar os cursos, acesse o [canal do CursoemVideo](#)

## Mais um exemplo

Vamos criar mais um exemplo para o uso de pseudo-classes. Nosso HTML vai ter o seguinte corpo:

```
18 <body>
19   <h1>Conteúdo especial</h1>
20   <div>
21     Passe o mouse aqui
22     <p>SURPRESAAAA!</p>
23   </div>
24   <p>Veja o conteúdo oculto aparecendo!</p>
25 </body>
```

Na **linha 20**, criamos um bloco especial com a tag `<div>`. Uma das grandes vantagens em usar divs é que elas podem ter outras tags dentro dela, assim como o parágrafo interno que criamos na **linha 22**.

Vamos criar nosso estilo agora, dentro da área `<head>`:

```
7   <style>
8     div > p {
9       display: none;
10    }
11
12    div:hover > p {
13      display: block;
14      background-color: yellow;
15    }
16  </style>
```

O primeiro seletor, na **linha 8**, vai esconder o parágrafo que está dentro da div (representado em CSS como `div > p`) através da propriedade `display` com o valor `none`.

O segundo seletor, na **linha 12**, vai fazer o parágrafo escondido reaparecer, com o fundo pintado de amarelo apenas quando passarmos o mouse sobre ele.



**MAIS UM SÍMBOLO:** Toda vez que usamos o símbolo > em um seletor, indicamos os filhos (*children*) imediatos de um elemento. No exemplo HTML que criamos, dizemos que o primeiro parágrafo é *direct children* da div.

SEM MOUSE	COM MOUSE
<h2>Conteúdo especial</h2> <p>Passe o mouse aqui</p> <p>Veja o conteúdo oculto aparecendo!</p>	<h2>Conteúdo especial</h2> <p>Passe o mouse aqui</p> <p><b>SURPRESAAAA!</b></p> <p>Veja o conteúdo oculto aparecendo!</p>

## Hora de exercitar

Chegou a hora de acessar o endereço do nosso repositório público em <https://gustavoguanabara.github.io/html-css/exercicios/> e executar o **exercício 017** no seu computador. Agora tente atingir esse mesmo resultado em casa, sem copiar o código que eu criei. Nesse momento, a prática é algo que você mais precisa. Se por acaso ficar difícil, pode acessar o repositório público de HTML e CSS e dar uma olhada nos comandos, mas **EVITE COPIAR**.



— □ ×

Cursos grátis de tecnologia  
que te preparam para o  
mercado de trabalho

RECODE

# Quer acompanhar tudo em vídeo?

Eu sei que às vezes as pessoas gostam mais de assistir vídeos do que ler livros, e é por isso que eu lanço há anos materiais no canal Curso em Vídeo no YouTube. O link que vou compartilhar contigo faz parte da playlist completa onde você encontra os **Módulos 1 e 2** do **Curso de HTML5 e CSS3**, completamente gravado com base nesse material.



Módulo 1 do curso: [https://www.youtube.com/playlist?list=PLHz\\_AreHm4dkZ9-atkcmBaMZdmLHft8n](https://www.youtube.com/playlist?list=PLHz_AreHm4dkZ9-atkcmBaMZdmLHft8n)

Módulo 2 do curso: [https://www.youtube.com/playlist?list=PLHz\\_AreHm4dIUPEXkY1AyVLQGcpSgVF8s](https://www.youtube.com/playlist?list=PLHz_AreHm4dIUPEXkY1AyVLQGcpSgVF8s)

## Teste seus conhecimentos



Terminou de ler esse capítulo e já acompanhou todos os vídeos e referências externas que indicamos? Pois agora, responda a essas 10 perguntas objetivas e marque em cada uma delas a única opção verdadeira. Aí sim, você vai poder comprovar que realmente entendeu o conteúdo.

1. Em um mesmo documento HTML, vários elementos podem possuir o mesmo \_\_\_\_\_ para criarmos formatações em grupo com CSS, mas só podemos ter um elemento com um \_\_\_\_\_ único para criar formatações individuais.

- ☐ A id / class
- ☐ B class / id
- ☐ C style / class
- ☐ D code / id

2. Toda configuração de folhas de estilo que fizemos para um determinado id, deve ser feita iniciando pelo símbolo \_\_\_\_\_ antes do identificador. Já as configurações para um determinado class serão feitas usando o símbolo \_\_\_\_\_ antes do nome da classe.

- ☐ A # e .
- ☐ B . e #
- ☐ C # e \$
- ☐ D \$ e .

3. Como já sabemos, existem três técnicas para inserir configurações CSS a um código HTML5: inline, local e externa. Se uma página usa as três técnicas ao mesmo tempo e cria uma configuração duplicada para uma determinada propriedade, qual delas vai se sobrepôr às demais?

- ☐ A CSS local
- ☐ B CSS externo
- ☐ C CSS inline
- ☐ D Não existe ordem de prioridades para casos de duplicidade

4. Uma pseudo-classe CSS é uma palavra-chave adicionada às declarações de um seletor após um sinal de

- ☐ A dois pontos (:)
- ☐ B ponto-e-vírgula (;)
- ☐ C porcentagem (%)
- ☐ D arroba (@)

5. Qual dos itens a seguir é o único que não se refere a uma pseudo-classe CSS?

- ☐ A hover
- ☐ B before
- ☐ C checked
- ☐ D empty

6. Para realizar configurações de formatação em um pedaço específico do elemento HTML referenciado (a primeira letra de um texto, por exemplo), devemos aprender a criar seletores para \_\_\_\_\_.

- ☐ A pseudo-classes
- ☐ B pseudo-textos
- ☐ C pseudo-formatos
- ☐ D pseudo-elementos

7. Qual dos itens a seguir é o único que não se refere a um pseudo-elemento CSS?

- ☐ A first-line
- ☐ B first-letter
- ☐ C active
- ☐ D before

8. Considere que temos o seguinte link em uma mesma página: `<a href="pag002.html" class="link">Página 2</a>`. Qual será o nome do seletor para configurarmos o que vai acontecer quando o usuário mover o cursor do mouse por cima do link?

- ☐ A `a#link:hover`
- ☐ B `a.hover:link`
- ☐ C `a.hover#link`
- ☐ D `a.link:hover`

9. Se quisermos que todos os links de uma página tenham um símbolo » depois do texto, devemos usar a formatação:

- ☐ A `a::after { content: '»'; }`
- ☐ B `a { content: '»'; }::after`
- ☐ C `after::a { content: '»'; }`
- ☐ D `::after { content: '»'; } a`

10. Podemos definir seletores com o símbolo de maior que (>) para indicar o que chamamos de:

- ☐ A larger tag
- ☐ B next tag
- ☐ C direct children
- ☐ D alternative element

# Suas anotações



*Não guarde conhecimento. Ele é livre. Compartilhe o seu e veja ele se espalhando pelo mundo 🌍*