Domine a força do Front-end



Alexandre A. Silva

Guia Prático dos Principais Seletores CSS

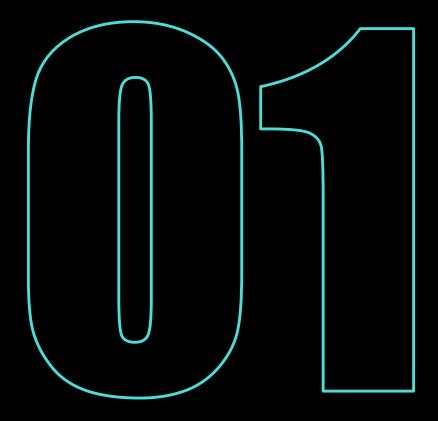
Aprenda a Estilizar com Precisão e Eficiência

CSS é a base para estilizar páginas web e criar experiências visuais atraentes. Conhecer os seletores é essencial para aplicar estilos de forma eficaz. Neste guia, você encontrará uma lista dos principais seletores CSS, explicados de maneira simples, com exemplos reais para facilitar sua compreensão e aplicação prática.

```
Introdução

a[href^="https"] {
   color: blue;
   font-weight: bold;
}

input[placeholder] {
   border: 2px solid green;
}
```



SELETORES BÁSICOS

Neste capítulo, abordaremos os seletores mais comuns e simples do CSS, ideais para iniciantes. Eles permitem estilizar elementos diretamente por suas tags, classes ou IDs.

Seletor de Elemento (Tag)

É utilizado para selecionar todos os elementos de um determinado tipo ou nome de tag na página, como <div>, , <h1>, entre outros. Ele aplica estilos a todos os elementos que correspondem a esse nome de tag, sem a necessidade de classes ou IDs específicos.

Resultado: Todos os <h1> ficam em um tom de verde, e os têm tamanho de fonte 16px.

Seletor de Classe

Usado para aplicar estilos a elementos com uma classe específica. É muito útil para reutilizar estilos em diferentes partes do site.

```
Seletor de Classe

<div class="card">Conteúdo da Card</div>
<div class="card">Outra Card</div>

.card {
  border: 1px solid #ddd;
  padding: 10px;
}
```

Resultado: Todos os elementos com a classe card recebem uma borda e espaçamento interno.

Seletor de ID

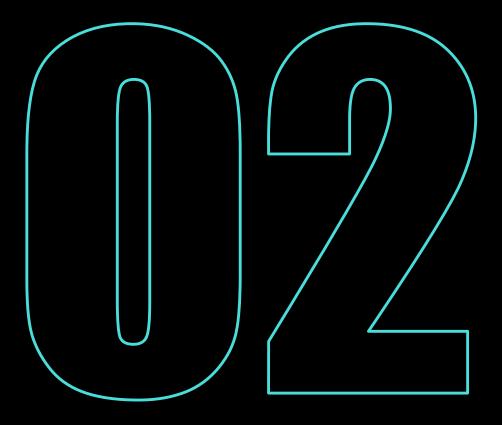
Aplica estilo a um elemento com um ID único. Deve ser usado com moderação para evitar dificuldades de manutenção do código.

```
Seletor de ID

<div id="cadapio"> Cárdápio Principal</div>

#cadapio {
  background-color: #333;
  color: white;
}
```

Resultado: O elemento com ID cardapio recebe um fundo escuro e texto branco.



SELETORES AVANÇADOS

Os seletores avançados permitem criar estilos mais específicos e precisos, ideais para projetos complexos. Eles ajudam a estilizar elementos com base em atributos, hierarquia e outros critérios.

Seletor Universal

O seletor universal em CSS é representado por um asterisco (*) e é utilizado para selecionar todos os elementos de uma página. Ele aplica estilos de forma global, sem distinção entre as tags.

Resultado: Remove espaçamentos de todos os elementos.

Seletor de Atributo

O seletor de atributo aplica estilos a elementos com um atributo específico, permitindo filtrar por valores exatos, parciais ou padrões. É útil para estilizar dinamicamente elementos com base em atributos como id, class, data-*, entre outros.

```
Seletor de Atributo

<input type="text" placeholder="Digite aqui">
<input type="password" placeholder="Senha">

input[type="text"] {
   border: 2px solid blue;
}
```

Resultado: Apenas campos de texto recebem borda azul.

Seletor de Descendente

O seletor descendente (espaço simples entre seletores) aplica estilos a todos os elementos que estão dentro de outro elemento, em qualquer nível de profundidade na hierarquia. Ele é ideal para estilizar estruturas com múltiplos níveis aninhados no DOM.

Resultado: Apenas parágrafos dentro de .container ficam verdes.

Seletor Filho Direto

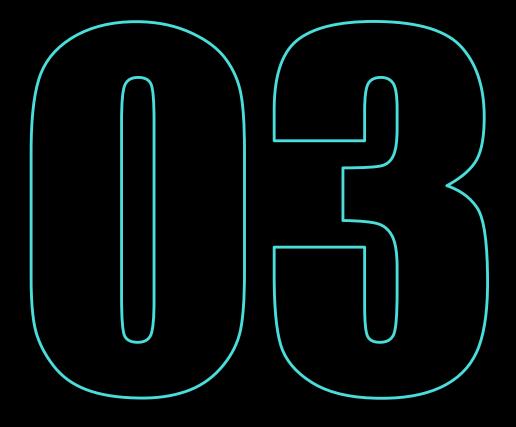
O seletor de filhos diretos (>) aplica estilos apenas aos elementos diretamente filhos de um elemento pai específico, ignorando descendentes em níveis mais profundos. Ele é útil para manter controle preciso sobre a hierarquia no DOM.

```
Seletor Filho Direto

    Item 1
    Item 2
    Subitem no pacote principal
    <lu>

ul > li {
    font-weight: bold;
}
```

Resultado: Somente os itens diretamente do ul>principal ficam em negrito.



SELETORES DE RELACIONAMENTO

Os seletores de relacionamento permitem estilizar elementos com base em sua posição relativa a outros elementos, como irmãos ou vizinhos.

Seletor de Irmão Adjacente

Aplica estilo ao primeiro elemento irmão que segue outro. Este seletor (+) é útil quando você deseja estilizar um elemento que aparece imediatamente após outro elemento no mesmo nível hierárquico (ou seja, ambos são filhos diretos do mesmo pai)

Resultado: O parágrafo logo após o <h1> fica em itálico.

Seletor de Irmãos Gerais

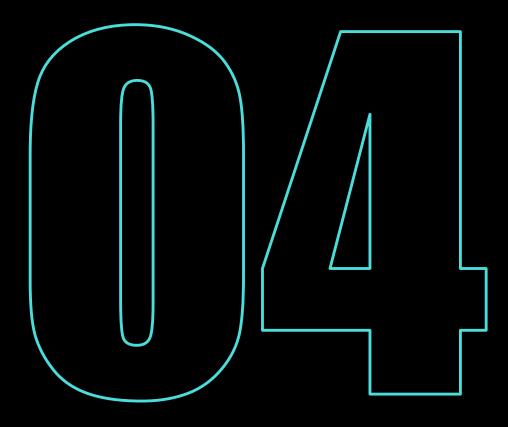
O seletor irmãos gerais (~) aplica estilos a todos os elementos no mesmo nível hierárquico que aparecem depois de um elemento específico. É útil para estilizar múltiplos elementos subsequentes em sequência, economizando o uso de classes extras e aproveitando a relação estrutural no DOM.

```
Seletor de Irmãos Gerais

<h1>Título</h1>
Parágrafo Primeiro da Constituição
Parágrafo Segundo da Constituição

h1 ~ p {
   color: gray;
}
```

Resultado: Todos os após o <h1> ficam cinza.



PSEUDO-CLASSES E PSEUDO-ELEMENTOS

As pseudo-classes e pseudo-elementos permitem aplicar estilos dinâmicos ou adicionar elementos virtuais ao DOM, sem alterar o HTML.

Seletor de Pseudo-Classe (Hover)

A pseudo-classe :hover é utilizada para aplicar estilos a um elemento quando o usuário passa o cursor do mouse sobre ele. Ela é comumente usada em links e botões para criar interatividade visual, como mudanças de cor ou efeitos de destaque.

```
Seletor de Pseudo-Classe (Hover)

<a href="#">CSS

<a href="#">Clique aqui</a>

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

Resultado: Links ficam vermelhos ao passar o mouse sobre eles.

Seletor de Pseudo-Elemento (::before e ::after)

O seletor de pseudo-elemento ::before é utilizado para inserir conteúdo antes do conteúdo de um elemento, enquanto ::after insere conteúdo após o conteúdo do elemento. Ambos permitem adicionar conteúdo gerado pelo CSS, como texto ou estilos, sem alterar o HTML.

```
Seletor de Pseudo-Elemento (::before e ::after) CSS

<h1>Capítulo 1 do 'Livro dos Passáros'</h1>

h1::before {
  content: "# ";
  color: gray;
}
```

Resultado: O texto "# " é adicionado antes do conteúdo do <h1>.

Agradecimentos

Sou grato por estarmos juntos até aqui

"Este Ebook foi gerado com o auxílio de inteligência artificial, mas a diagramação ficou a cargo de um ser humano.

Lembre-se de que este conteúdo foi criado com fins didáticos, para mostrar o processo de construção. Não houve uma validação cuidadosa por um humano, então podem existir erros gerados pela IA."

Deus seja Louvado!