

JAVASCRIPT

E A CÂMARA DOS ALGORITMOS 🧙‍♂️



GIULIANE CAMILO

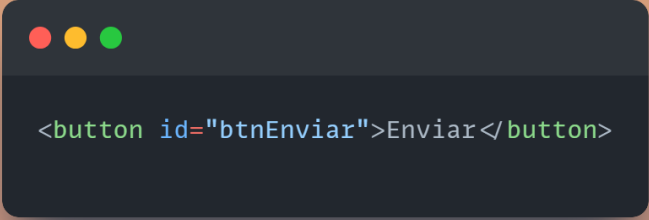


01

PEGANDO UM ÚNICO ELEMENTO PELO ID


getElementById()

Esse é o seletor mais direto: retorna apenas um elemento com o ID informado.



```
<button id="btnEnviar">Enviar</button>
```

snappify.com



```
const botao = document.getElementById("btnEnviar");  
botao.addEventListener("click", () => {  
  alert("Formulário enviado!");  
});
```

snappify.com

Quando usar:

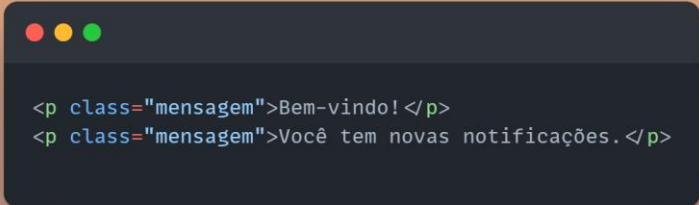
Quando você sabe exatamente qual elemento quer acessar e ele tem um ID único.

02

SELECIONANDO VÁRIOS ELEMENTOS PELA CLASSE

getElementsByClassName()

Retorna uma coleção de todos os elementos com a classe informada.



```
<p class="mensagem">Bem-vindo!</p>
<p class="mensagem">Você tem novas notificações.</p>
```

snappify.com



```
const mensagens = document.getElementsByClassName("mensagem");

for (let msg of mensagens) {
  msg.style.color = "green";
}
```

snappify.com


Dica: O retorno é uma lista, então use `for...of` ou transforme em array com `Array.from()`.

03

**BUSCANDO POR TIPO DE
ELEMENTO**

getElementsByTagName()


Seleciona todos os elementos de uma mesma tag HTML.



```
<ul>
  <li>Item A</li>
  <li>Item B</li>
</ul>
```

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. It displays HTML code for an unordered list with two items, 'Item A' and 'Item B'.

snappify.com



```
const itens = document.getElementsByTagName("li");

for (let item of itens) {
  item.style.fontWeight = "bold";
}
```

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. It displays JavaScript code that selects all 'li' elements and sets their font weight to 'bold'.

snappify.com


Uso comum: Alterar estilos ou conteúdos de listas, parágrafos ou botões.

04

O SELETOR MODERNO E FLEXÍVEL


querySelector()

Permite buscar qualquer elemento, usando sintaxe de CSS (ID, classe, tag, etc). Retorna apenas o primeiro elemento encontrado.



```
<div class="card destaque">Produto 1</div>  
<div class="card">Produto 2</div>
```

snappify.com



```
const destaque = document.querySelector(".card.destaque");  
destaque.style.backgroundColor = "#ffd700";
```

snappify.com

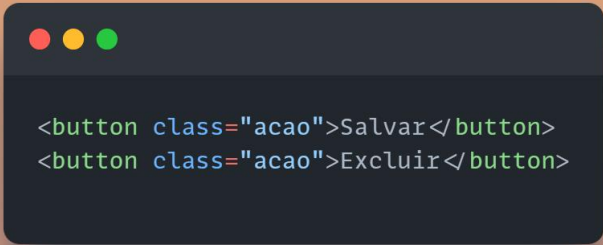
Vantagem: Combina o poder dos seletores CSS com a simplicidade do JavaScript.

05

SELECCIONANDO MÚLTIPLOS ELEMENTOS COM CSS

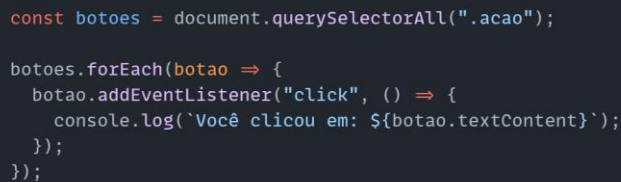
querySelectorAll

Semelhante ao anterior, mas retorna todos os elementos que correspondem ao seletor.



```
<button class="acao">Salvar</button>  
<button class="acao">Excluir</button>
```

snappify.com



```
const botoes = document.querySelectorAll(".acao");  
  
botoes.forEach(botao => {  
  botao.addEventListener("click", () => {  
    console.log(`Você clicou em: ${botao.textContent}`);  
  });  
});
```

snappify.com

Dica: Ideal para aplicar eventos ou estilos em grupo.

AGRADECIMENTOS

Obrigado por ler até aqui!

Este E-Book foi gerado por IA e diagramado por Humano.

Este conteúdo foi gerado com fins educativos de comunicação, não foi realizada uma curadoria detalhada.

