

MATERIAL EXTRA

Variáveis CSS



growdev

VARIÁVEIS CSS

As variáveis CSS, também chamadas de **Custom Properties**, são uma funcionalidade moderna do CSS que permite definir valores reutilizáveis para propriedades estilísticas. Elas funcionam como "variáveis" em linguagens de programação, mas são definidas e manipuladas diretamente no CSS. A sintaxe para criar uma variável é simples: você usa dois traços -- no início do nome da variável e a declara dentro de um seletor CSS (geralmente :root para aplicação global).

```
:root {  
  --primary-color: #3498db;  
  --font-size: 16px;  
}
```

Aqui, --primary-color e --font-size são variáveis CSS.

Para que servem?

As variáveis CSS servem para centralizar e reaproveitar valores estilísticos no código. Com elas, você pode alterar um valor em um único lugar, e a mudança será refletida em todas as partes do CSS onde a variável foi usada. Elas ajudam a manter o código mais consistente, legível e fácil de atualizar, especialmente em projetos maiores onde cores, tamanhos ou outros valores se repetem frequentemente.

```
button {  
  background-color: var(--primary-color);  
  font-size: var(--font-size);  
}
```

Caso você altere o valor de `--primary-color` ou `--font-size`, todos os botões serão atualizados automaticamente.

Por que utilizar?

Utilizar variáveis CSS traz várias vantagens, como:

1. **Reutilização:** Você pode definir um valor uma vez e utilizá-lo em múltiplos lugares.
2. **Manutenção facilitada:** Alterar um único valor em uma variável atualiza todas as ocorrências automaticamente, reduzindo a necessidade de buscas manuais no código.
3. **Consistência visual:** Garantem que estilos como cores ou tamanhos sejam consistentes em toda a aplicação.
4. **Customização dinâmica:** As variáveis CSS podem ser alteradas dinamicamente via JavaScript, permitindo temas personalizados ou ajustes em tempo real.
5. **Escopo:** Elas respeitam o escopo do CSS, o que significa que você pode definir variáveis específicas para diferentes seletores, além de ter variáveis globais.

Variáveis CSS na prática

```
:root {
  --background-color: #f4f4f4;
  --text-color: #333;
  --primary-color: #3498db;
  --secondary-color: #2ecc71;
  --padding: 10px;
}

body {
  background-color: var(--background-color);
  color: var(--text-color);
  padding: var(--padding);
}

button {
  background-color: var(--primary-color);
  color: white;
  padding: var(--padding);
  border: none;
  border-radius: 5px;
  cursor: pointer;
}

button.secondary {
  background-color: var(--secondary-color);
}
```

Neste exemplo, as variáveis definidas em `:root` tornam o estilo da página mais modular e fácil de manter. Se o cliente decidir trocar a cor principal do site, basta alterar o valor de `--primary-color` no `:root` e todas as partes da interface que usam essa variável serão atualizadas automaticamente.

Você pode também combiná-las com JavaScript para criar um sistema de temas dinâmicos:

```
const toggleTheme = () => {  
  const root = document.documentElement;  
  root.style.setProperty("--background-color", "#333");  
  root.style.setProperty("--text-color", "#f4f4f4");  
};  
  
// Adicionando um evento ao botão  
document.getElementById("theme-button").addEventListener("click",  
toggleTheme);
```

Com isso, ao clicar no botão, as variáveis CSS são alteradas dinamicamente, modificando o estilo da página sem necessidade de recarregá-la.

Conclusão

As variáveis CSS representam um avanço significativo na escrita de estilos mais flexíveis, consistentes e fáceis de manter. Elas ajudam a resolver problemas comuns como repetição de valores, dificuldade de manutenção em projetos grandes e a necessidade de customização dinâmica. Incorporá-las ao seu fluxo de trabalho é uma boa prática que agrega eficiência e clareza ao desenvolvimento do front-end.

REFERÊNCIAS

1. [Utilizando propriedades CSS personalizadas \(variáveis\) - CSS | MDN](#)
2. [Use variáveis CSS como uma pessoa sênior com essas 5 dicas 🔥 - DEV Community](#)
3. [var\(\) - CSS | MDN](#)
4. [CSS Custom Properties Guide | CSS-Tricks](#)
5. [CSS Variables - The var\(\) function](#)

BONS ESTUDOS

