

## DOM

### Exemplos e Explicação:

O Document Object Model (DOM) é uma interface de programação para os documentos HTML e XML. Representa a página de forma que os programas possam alterar a estrutura do documento, alterar o estilo e conteúdo. O DOM representa o documento com nós e objetos, dessa forma, as linguagens de programação podem se conectar à página.

Uma página da Web é um documento. Este documento pode ser exibido na janela do navegador ou como a fonte HTML. Mas é o mesmo documento nos dois casos. O DOM (Document Object Model) representa o mesmo documento para que possa ser manipulado. O DOM é uma representação orientada a objetos da página da web, que pode ser modificada com uma linguagem de script como JavaScript.

Os padrões [W3C DOM](#) e [WHATWG DOM](#) são implementados na maioria dos navegadores modernos. Muitos navegadores estendem o padrão; portanto, é necessário ter cuidado ao usá-los na Web, onde os documentos podem ser acessados por vários navegadores com diferentes DOMs.

Por exemplo, o DOM padrão especifica que o método `getElementsByTagName` no código abaixo deve retornar uma lista de todos os elementos `<p>` no documento:

#### Exemplo 1: Alterar o texto de um elemento

Imagine um botão que, ao ser clicado, muda o texto de um parágrafo.

```
<p id="meuParagrafo">Texto original</p>
<button onclick="mudarTexto()">Clique aqui</button>
```

JavaScript:

```
function mudarTexto() {
    document.getElementById("meuParagrafo").innerText = "Texto alterado!";
```

### Exemplo 2: Adicionar novos elementos à página

Um formulário que insere itens em uma lista.

```
<ul id="minhaLista">
  <li>Item 1</li>
</ul>
<input id="novoItem" type="text" placeholder="Digite um novo item">
<button onclick="adicionarItem()">Adicionar</button>
```

### Exemplo 3: Alterar estilos dinamicamente

Um botão que muda a cor de fundo de um elemento.

```
<div id="caixa" style="width: 100px; height: 100px; background-color: blue;"></div>
<button onclick="mudarCor()">Mudar cor</button>
```

### Exemplo 4: Responder a eventos de teclado

Exibir o que o usuário está digitando em tempo real.

```
<input id="campoTexto" type="text" oninput="exibirTexto()" placeholder="Digite algo">
<p id="textoDigitado"></p>
```

### Exemplo 5: Manipular classes

Um botão que alterna o destaque em um parágrafo.

```
<p id="meuTexto" class="normal">Texto para destacar.</p>
<button onclick="alternarClasse()">Alternar destaque</button>
```

## **Aplicações do DOM no mundo real**

Páginas interativas: Como em SPAs (Single Page Applications).

Validação de formulários: Verificar campos antes de enviar.

Construção de jogos: Como manipular elementos gráficos para interação.

Atualização em tempo real: Notícias, dados financeiros ou chats.

Automação de testes: Ferramentas como Selenium interagem com o DOM para simular ações do usuário. O DOM é a base para tornar sites mais dinâmicos e responsivos à interação do usuário!

## **Conclusão:**

No meu entendimento o Dom serve para estruturar os objetos podendo fornecer funções para facilitar o desenvolvimento da página web, permitindo que desenvolvedores acessem e manipulem a estrutura, o conteúdo e o estilo de documentos HTML e XML de forma programática. Utilizei o dom para alterar o texto clicando no botão, onde facilita a dinâmica do texto.

