***DOM (Document Object Model):***

* **Definição:**

É uma interface de programação para documentos HTML e XML. Ele fornece uma representação estruturada do documento como uma árvore, onde cada elemento, atributo ou texto é representado como um **nó**. Essa estrutura permite que linguagens de programação (como JavaScript) interajam dinamicamente com o conteúdo, estrutura e estilo de uma página web.

### **Exemplos de manipulação do DOM:**

#### **1. Selecionar elementos na página**

* Usamos métodos como getElementById, querySelector, e getElementsByClassName para acessar elementos.

javascript

// Selecionando um elemento pelo ID

const titulo = document.getElementById("titulo");

console.log(titulo.textContent);

// Selecionando um elemento pelo seletor CSS

const paragrafo = document.querySelector(".paragrafo");

console.log(paragrafo.innerHTML);

#### **2. Alterar conteúdo e estilos**

* Podemos modificar texto ou estilos diretamente.

javascript

// Alterando o texto de um elemento

titulo.textContent = "Bem-vindo ao DOM!";

// Alterando estilos

titulo.style.color = "blue";

titulo.style.fontSize = "2em";

#### **3. Criar e adicionar novos elementos**

* É possível criar elementos dinamicamente e adicioná-los à página.

javascript

// Criando um novo elemento

const novoParagrafo = document.createElement("p");

novoParagrafo.textContent = "Este parágrafo foi criado dinamicamente!";

document.body.appendChild(novoParagrafo);

#### **4. Manipular eventos**

* O DOM permite adicionar interatividade, como cliques em botões.

Javascript

// Adicionando um evento de clique

const botao = document.querySelector("#meuBotao");

botao.addEventListener("click", () => {

alert("Botão clicado!");

});

### **Aplicações do DOM:**

1. **Alterações dinâmicas no conteúdo**: O DOM é essencial para criar páginas dinâmicas, onde o conteúdo é alterado em resposta a ações do usuário sem precisar recarregar a página (por exemplo, jogos ou interfaces interativas).
2. **Validação de formulários**: Usando o DOM, você pode verificar se os campos de um formulário foram preenchidos corretamente antes de enviá-los ao servidor.
3. **Efeitos visuais e animações**: O DOM pode ser manipulado para criar animações simples, como alterar o tamanho ou a posição de um elemento ao longo do tempo.
4. **Interface com o usuário**: O DOM é utilizado para criar interações complexas com o usuário, como modais, menus suspensos e sliders.

**Tipos de nós no DOM:**

### **1. Nó de Documento (Document Node)**

O nó de documento é o topo da árvore do DOM. Ele representa o próprio documento HTML ou XML e permite que você acesse todos os outros nós abaixo dele.

* **Exemplo**: O document é o nó raiz, acessível em JavaScript com document.

*javascript*

console.log(document); // Exibe o nó do documento

### **2. Nó Elemento (Element Node)**

Os nós de elemento representam as tags HTML (ou XML) em um documento. Por exemplo, cada <div>, <p>, <h1>, etc., é um nó de tipo **Element**.

* **Exemplo**: O elemento <div> é um nó do tipo "elemento".

*html*

<div id="minhaDiv">Conteúdo aqui</div>

Em JavaScript, você pode acessar esse nó usando getElementById, querySelector, etc.

*javascript*

let div = document.getElementById('minhaDiv');

console.log(div); // Exibe o nó de elemento <div id="minhaDiv">

### **3. Nó de Texto (Text Node)**

Nó de texto é o conteúdo textual dentro de um elemento. Cada pedaço de texto entre as tags HTML é considerado um nó de texto no DOM.

* **Exemplo**: O texto "Conteúdo aqui" dentro de <div id="minhaDiv">Conteúdo aqui</div> é um nó de tipo **Texto**.

*javascript*

let texto = div.firstChild;

console.log(texto); // Exibe o nó de texto "Conteúdo aqui"

### **4. Nó de Atributo (Attribute Node)**

Nó de atributo representa os atributos de um elemento. Os atributos como id, class, src, href, entre outros, são nós de atributo.

* **Exemplo**: O atributo id="minhaDiv" no elemento <div id="minhaDiv"> é um nó de atributo.

*javascript*

let atributo = div.getAttributeNode('id');

console.log(atributo); // Exibe o nó de atributo { name: "id", value: "minhaDiv" }

### **5. Nó de Comentário (Comment Node)**

Nó de comentário representa os comentários HTML ou XML no documento. Por exemplo, <!-- Comentário -->.

* **Exemplo**: O comentário dentro de um HTML é um nó de tipo **Comentário**.

*html*

<!-- Este é um comentário -->

No JavaScript, você pode acessar e manipular esses nós.

*javascript*

let comentario = document.createComment("Este é um comentário");

document.body.appendChild(comentario);

### **6. Nó de Documento Tipo (DocumentType Node)**

Este nó representa o tipo do documento, como <!DOCTYPE html>, que especifica o tipo de documento e a versão do HTML.

* **Exemplo**: No começo de um arquivo HTML, a declaração <!DOCTYPE html> é um nó do tipo **Document Type**.

*javascript*

let doctype = document.doctype;

console.log(doctype); // Exibe o nó doctype

**Conclusão:**

O **Document Object Model (DOM)** é uma interface de programação essencial para manipulação de documentos HTML e XML. Ele organiza a estrutura do documento em uma árvore hierárquica, onde cada elemento do site é representado individualmente. Por meio de métodos de interações como “getElementById, querySelector, e getElementsByClassName” é possível selecionar elementos específicos para manipulação. Pelo DOM, também se pode alterar textos e estilos, criar novos elementos e adicionar interatividade por meio de eventos, como cliques em botões.

Além disso, o DOM possui nós/pilares, onde cada um deles desempenham um papel específico, como: os nós de documento ou “Document Node” representa o próprio documento em HTML em si; os nós de elemento “Element Node” referem-se às tags em HTML; os nós de texto “Text Node”representa o texto entre as tags; os nós de atributo ou “Attribute Node” que representam os atributos de elementos como id e classes; os nós de comentário ou “Comment Node” que representam os comentários da estrutura HTML de modo que não interfira na estrutura em si; e o nó de tipo de documento ou “DocumentType Node” que define a versão do HTML descrita no <!DOCTYPE html>.