DOM

Document Object Model

O DOM HTML é um padrão de como acessar e modificar os elementos HTML de uma página.

Document -> <html> documento raiz/principal - > Varias tags filhas (head, body) cada um desses elementos é chamado de nó e eles também podem ter filhos ou não.

BOM

Browser Object Model

Tudo que está dentro do objeto window (arvore de dependências)

Window -> document, history, location, screen, navigator

Pode-se observar que o DOM está contido na arvore de dependências do BOM

Estrutura HTML

Tag, id e classe

Métodos

Selecionando os elementos de uma página

Document.elementById(‘id’) Id é único então ele volta com apenas um elemento diferente das duas outras formas a seguir

Document.elementsByTagName(‘tag’) retorna um array com todos as ocorrências dessa tag na pagina

Document.elementsByClassName(‘class’) retorna um array com todos as ocorrências dessa classe na pagina

Mesmo se houver apenas um elemento no texto ele ainda vai ter um retorno em Array

Document.querySelectorAll(‘.primeira-classe .segunda-classe’) Ele também retorna um array

Utilizado como forma de busca personalizado para mais de um elemento combinando tipos de buscas diferentes

|  |  |
| --- | --- |
| Método | Descrição |
| Document.creatElement(element) | Cria um novo elemento HTML |
| Document.removeChild(elemento) | Remove um elemento |
| Document.appendChild(elemento) | Adiciona um elemento |
| Document.replaceChild(new, old) | Substitui um elemento |

Trablhando com Estilos

Classes

Element.classList

classList é uma das propriedades desse elemento

Element.classList.add(“classe”) – Adiciona uma classe

Element.classList.remove(“classe”) – remove uma classe

Element.classList.toggle(“dark-mode”) – Vai olhar para lista se na lista não constar o que esta sendo inserido “dark-mode” ele vai adicionar a lista, caso o “dark-mode” conste na lista ele ira removelo

CSS

Acessando diretamente o CSS de um elemento

Document.elementsByTagName(“p”).style.color = “blue”;

Faz com a cor de todos os parágrafos “p” seja azul

Eventos

Qualquer tipo de interação do usuário com a pagina

Eventos de mouse – mouseover, mouseout

Eventos de click – click, dbclick

Eventos de atualização – change, load

Event listener

Diretamente no javascript, cria um evento que vai ser acionado no momento em que o usuário realizar determinada ação.

Const botão = document.elementById(‘meuBotao’);

Botão.addEventListener(“click”, outraFuncao)

Dentro do HTML

Especifica a função a ser chamada diretamente no elemento HTML

<h1 onclick=”mudaTexto”>Clique aqui</h1>