

Javascript

O que vamos ver hoje?

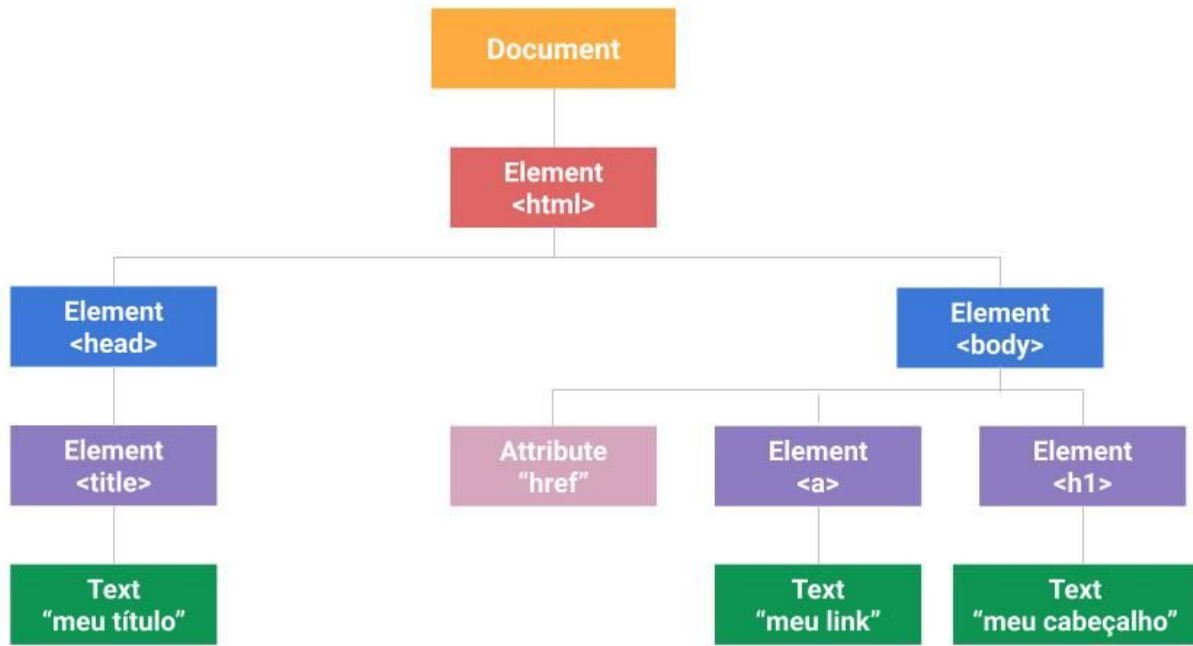
- DOM
- Eventos
- Seletores

DOM

DOM

- Document Object Model (Modelo de Documento por Objetos).
- Representa o documento com nós e objetos.
- Se estrutura em formato de árvore, com vários galhos que indicam diferentes elementos da página.

DOM



Eventos

Eventos

- Os eventos são ações que são realizadas em um determinado elemento da página, seja ele um texto ou uma imagem, por exemplo.
- Muitas das interações do usuário que está visitando sua página com o conteúdo do seu site podem ser consideradas eventos.

Eventos

- onload: Quando o documento é carregado.
- onsubmit: Ao enviar formulário.
- onchange: Quando elemento perde o foco ou é modificado.
- onclick: Clique em botão.
- onmouseover: Disparado ao passar o mouse por cima.
- onkeydown: Quando uma tecla é pressionada.

Eventos

- Podemos aplicar eventos de duas maneiras principais:
 - Inline: na própria tag html
 - Em arquivo externo.

Eventos

- Podemos aplicar eventos de duas maneiras principais:
 - Inline: na própria tag html

```
1 <button onclick="alert('Oi')">  
2   Sou um evento inline  
3 </button>
```

- Em arquivo externo.

```
1 document.querySelector("button").addEventListener(  
2   "click", () => {  
3     alert("oi");  
4   });
```

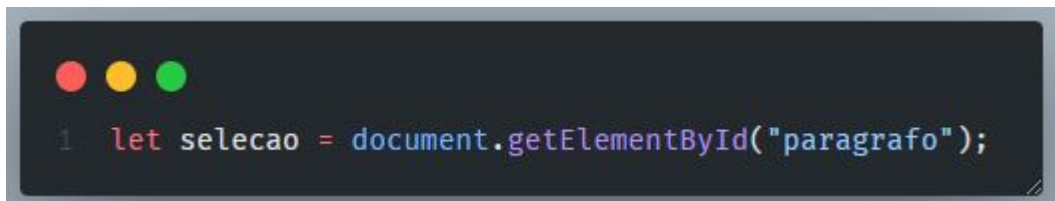
Seletores

Seletores

- `getElementById()`
- `getElementsByClassName()`
- `querySelector()`
- `querySelectorAll()`

Seletores

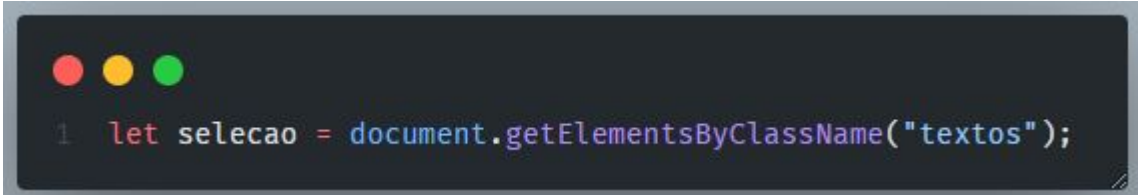
- getElementById()
 - Retorna o elemento correspondente ao id passado como parâmetro:



```
1 let selecao = document.getElementById("paragrafo");
```

Seletores

- `getElementsByClassName()`
 - Retorna os elementos que possuem a classe passada no parâmetro:



```
1 let selecao = document.getElementsByClassName("textos");
```

Seletores

- `querySelector()`
 - Retorna o primeiro elemento do id, classe ou tag passada como parâmetro:

```
1 let teste1 = document.querySelector("#souId");  
2 let teste2 = document.querySelector(".souClass");  
3 let teste3 = document.querySelector("div");
```

Seletores

- `querySelectorAll()`
 - Retorna uma `NodeList` dos elementos que correspondem ao parâmetro:

```
1 let teste = document.querySelectorAll("div");  
2 // Irá retornar todas as divs da página
```


Listener

addEventListener

- permite configurar funções a serem chamadas quando um evento acontece.

```
1 alvo.addEventListener(evento, função);
```

- alvo: é o elemento HTML ao qual você deseja adicionar os eventos
- evento: é onde você especifica qual é o tipo de evento que irá disparar a ação/função.
- função: o que vai ser executado.

addEventListener

```
1 var nomeElemento = document.getElementById("div");  
2  
3 nomeElemento.addEventListener("click", () => {  
4     ...  
5 });
```

Modificando o texto

Modificando o texto



```
1 <p id="paragrafo">Teste</p>
2   <button onclick="trocarTexto()">Trocar Texto</button>
```

Teste

Trocar Texto



```
1 const trocarTexto = () => {
2   let selecao = document.getElementById("paragrafo");
3   let novoTexto = selecao.firstChild.nodeValue;
4   novoTexto = novoTexto.replace("Teste", "Teste2");
5   selecao.firstChild.nodeValue = novoTexto;
6 };
```

Teste2

Trocar Texto

Inserindo elementos



```
1 <div class="options"></div>
2   <button onclick="inserirTexto()">Inserir texto</button>
```



```
1 const inserirTexto = () => {
2   let optionsHtml = `<div><p>Este texto foi inserido via Javascript</p></div>`;
3   document.querySelector(".options").innerHTML = optionsHtml;
4 };
```

Modificando o CSS

Estilo

- A propriedade style em JavaScript retorna o estilo de um elemento na forma de um objeto CSSStyleDeclaration que contém uma lista de todas as propriedades de estilos.

```
1 var button = document.querySelector("button");  
2  
3 button.addEventListener("click", () => {  
4   button.style.backgroundColor = "red";  
5 });
```


Classes

- Utilizando a propriedade `classList` você pode obter, definir ou remover classes CSS facilmente de um elemento.



```
1 let elementoDiv = document.getElementById("divPai");  
2 elementoDiv.addEventListener("click", () => {  
3     elementoDiv.classList.toggle("divColorida");  
4 });
```

Resumo

Resumo



- Podemos inserir, alterar e remover elementos, classes e estilos utilizando Javascript.

Dúvidas?

Programa
3000 TALENTOS TI
Obrigado(a)!