

Resumo: JavaScript - Manipulação do DOM

Criando Novos Elementos

Podemos criar, inserir e excluir elementos da árvore DOM.

Usa-se a função `document.createElement('tag')` para criar o elemento.

```
var novoParagrafo = document.createElement("p");
```

Anexando um Elemento DOM

Para adicionar um novo elemento à página, usa-se:

`element.appendChild(novoElemento);`

```
document.body.appendChild(novoParagrafo);
```

Inserindo Texto ao Elemento

Para inserir texto antes de adicionar ao DOM:

```
var texto = document.createTextNode("Olá mundo!");  
novoParagrafo.appendChild(texto);
```

O Elemento `this`

Dentro de um manipulador de eventos, `'this'` se refere ao elemento que recebeu o evento.

```
<button onclick="mudarTexto(this)">Clique</button>
```

Localizando um Pai

Cada elemento (exceto o documento) possui um pai.

Usa-se a propriedade `parentNode` para acessá-lo.

```
var pai = elemento.parentNode;
```

Removendo um Filho

Para remover um filho, selecione o pai e use `removeChild()`.

Resumo: JavaScript - Manipulação do DOM

```
pai.removeChild(filho);
```

Localizando Filhos (childNodes)

Usa-se a propriedade `childNodes` para obter todos os filhos.

```
var filhos = elemento.childNodes;  
var primeiro = elemento.firstChild;  
var ultimo = elemento.lastChild;
```

Localizando Irmos (siblings)

Para acessar elementos no mesmo nível da árvore:

- `nextSibling`: próximo irmão
- `previousSibling`: irmão anterior

```
var proximo = elemento.nextSibling;  
var anterior = elemento.previousSibling;
```

Exercício Prático

Crie a página `historia.html` com uma lista de tópicos a serem estudados.

Use JavaScript para tornar a lista interativa, conforme funcionalidades descritas no enunciado.