

# DOM PARA INICIANTES

---

Transversing e Manipulação

## outerHTML, innerHTML e innerText

---

Propriedades que retornam uma string contendo o html ou texto. É possível atribuir um novo valor para as mesmas

```
element.innerText = 'Novo Texto'.
```

```
const menu = document.querySelector('.menu');
```

```
menu.outerHTML; // todo o html do elemento
```

```
menu.innerHTML; // html interno
```

```
menu.innerText; // texto, sem tags
```

```
menu.innerText = '<p>Texto</p>'; // a tag vai como texto
```

```
menu.innerHTML = '<p>Texto</p>'; // a tag é renderizada
```

## Transversing

---

Como navegar pelo DOM, utilizando suas propriedades e métodos.

```
const lista = document.querySelector('.animais-lista');

lista.parentElement; // pai
lista.parentElement.parentElement; // pai do pai
lista.previousElementSibling; // elemento acima
lista.nextElementSibling; // elemento abaixo

lista.children; // HTMLCollection com os filhos
lista.children[0]; // primeiro filho
lista.children[--lista.children.length]; // último filho

lista.querySelectorAll('li'); // todas as LI's
lista.querySelector('li:last-child'); // último filho
```

## Element vs Node

---

Element's representam um elemento html, ou seja, uma tag. Node representa um nó, e pode ser um elemento (Element), texto, comentário, quebra de linha e mais.

```
const lista = document.querySelector('.animais-lista');
```

```
lista.previousElementSibling; // elemento acima
```

```
lista.previousSibling; // node acima
```

```
lista.firstChild; // primeiro node child
```

```
lista.childNodes; // todos os node child
```

*Geralmente estamos atrás de um elemento e não de qualquer node em si.*

## Manipulando Elementos

---

É possível mover elementos no dom com métodos de Node.

```
const lista = document.querySelector('.animais-lista');  
const contato = document.querySelector('.contato');  
const titulo = contato.querySelector('.titulo');  
  
contato.appendChild(lista); // move lista para o final de  
contato  
contato.insertBefore(lista, titulo); // insere a lista antes de  
titulo  
contato.removeChild(titulo); // remove titulo de contato  
contato.replaceChild(lista, titulo); // substitui titulo por  
lista
```

## Novos Elementos

---

Podemos criar novos elementos com o método

`createElement()`

```
const animais = document.querySelector('.animais');

const novoH1 = document.createElement('h1');
novoH1.innerText = 'Novo Título';
novoH1.classList.add('titulo');

animais.appendChild(novoH1);
```

## Clonar Elementos

---

Todo elemento selecionado é único. Para criarmos um novo elemento baseado no anterior, é necessário utilizar o método

`cloneNode()`

```
const titulo = document.querySelector('h1');
const titulo2 = document.querySelector('h1');
const novoTitulo = titulo;
// titulo, titulo2 e novoTitulo são iguais

const cloneTitulo = titulo.cloneNode(true);
const contato = document.querySelector('.contato');
contato.appendChild(cloneTitulo);
```

`true` sinaliza para incluir os  
filhos

## Exercício

---

```
// Duplica o menu e adiciona ele em copy  
  
// Seleciona o primeiro DT da dl de Faq  
  
// Seleciona o DD referente ao primeiro DT  
  
// Substitua o conteúdo html de .faq pelo de .animais
```