



JAVASCRIPT INTERMEDIÁRIO

Conhecendo mais a fundo a linguagem.

CENTEC
INSTITUTO CENTRO DE ENSINO TECNOLÓGICO



**GOVERNO DO
ESTADO DO CEARÁ**
*Secretaria da Ciência, Tecnologia
e Educação Superior*

Conhecendo o DOM

- Document Object Model é a interface que vai dizer como os documentos html e xml serão lidos pelo browser.
- Depois que o navegador faz a leitura dos documentos ele cria um objeto que faz uma representação estruturada do seu documento e define meios de como essa estrutura pode ser acessada.
- Nós podemos acessar e manipular o DOM com JavaScript, é a forma mais fácil e usada.



Conhecendo o DOM.

- Com o domínio do DOM as possibilidades são imensas. Você consegue criar e modificar componentes.
- Fazer com que o usuário modifique a interface gráfica do seu projeto.



Método de busca de elementos da Dom.

- **document.getElementById**: Retorna um objeto representando o elemento cujo id foi igual ao o especificado pela String.
- **document.getElementsByTagName**: Retorna o conjunto de todos os elementos que usam determinada tag.
- **document.getElementsByClassName** busca todos os elementos que usem aquela classe.



Universidade
do Trabalho
Digital

Usando o querySelector("")

Permite que você vasculhe todo o documento e retorne o elemento que desejamos.

```
document.querySelector('div#lista');
```

```
document.querySelector("input[name]");
```

O query selector só vai retornar um elemento.



Eventos em Javascript.

Onclick	Dispara o evento quando o elemento é clicado.
OnMouseOver	Evento ocorre quando passa o mouse em cima.
OnLoad	Quando a página carrega
OnSubmit	Alção que ocorre no envio do formulário
OnFocus	Quando o elemento é focado.



Funções js

```
<script>
    function soma(numero1, numero2) {
        return numero1 + numero2;
    }
</script>
```

- Para declarar uma função em javascript, utilizamos a palavra reservada **function** junto com seu nome (a não ser que seja função anônima), parâmetros e o corpo da mesma.



Universidade
do Trabalho
Digital

ECMASCRIPT-6

ECMAScript 6

JS

- Também conhecida como ES-6 é uma importante atualização da linguagem javascript porque trouxe funcionalidades que ao mesmo tempo resolveram problemas passados como melhoraram o desempenho da linguagem de programação. Algumas dessas mudanças foram:
 - variáveis LET e CONST
 - arrow function
 - desctructuring



Valores padrão

```
<script>
  function soma(numero1=10, numero2=2) {
    return numero1 + numero2;
  }

  console.log(soma);
</script>
```

Podemos usar essa nova funcionalidade para setar valores as variáveis, caso não seja passado nenhum. Vamos empregar esse recurso no nosso exemplo passado de função.



Async e await

```
<script>
  const minhaPromise = () => new Promise((resolve, reject)=>{
    setTimeout(()=>{resolve("OK")},3000);
  });

  minhaPromise().then(response=>{
    console.log(response);
  }).catch(err=>{
    console.log("Não executou");
  });
}
```

Exemplo de promise comum. Ela é utilizada para requisições assíncronas, ou seja, que não tenham o mesmo fluxo que o nosso script como por exemplo as requisições AJAX.

Async e await

```
//método async await
async function executaPromise() {
    const promise = await minhaPromise();
}

executaPromise();

const minhaPromise = () => new Promise((resolve, reject)=>{
    setTimeout(()=>{resolve("OK")},3000);
});

minhaPromise().then(response=>{
    console.log(response);
}).catch(err=>{
    console.log("Não executou");
});
```



Universidade
do Trabalho
Digital

Métodos getters and setters

```
const al = new aluno("Bianca", 25, "bianca@email.com")  
  
al.nome = "Laura";  
  
al.chamada(true);  
  
console.log(al.nome)  
console.log(al.idade)  
console.log(al.email)  
console.log(al.presenca);
```

