

JavaScript Avançado

Prof. Bruno Torres

Revisão Rápida

- JavaScript é uma linguagem de programação para a web.
- Permite manipular o DOM e responder a eventos.
- Possui variáveis (`let` , `const`), funções e estruturas de controle.

Objetos em JavaScript

- Coleção de pares **chave-valor**
- Permite estruturar dados de forma organizada

Exemplo:

```
const pessoa = {  
  nome: "João",  
  idade: 30,  
  saudacao: function() {  
    return `Olá, meu nome é ${this.nome}`;  
  }  
};  
console.log(pessoa.saudacao());
```

Arrays e Métodos Úteis

- Estrutura de dados para armazenar listas

Métodos comuns:

```
let numeros = [1, 2, 3, 4, 5];  
numeros.push(6); // Adiciona ao final  
numeros.pop(); // Remove último elemento  
numeros.shift(); // Remove primeiro elemento  
numeros.unshift(0); // Adiciona ao início  
console.log(numeros);
```

Funções Avançadas

Funções de Alta Ordem

- Funções que recebem ou retornam outras funções

Exemplo:

```
function calcular(a, b, operacao) {  
    return operacao(a, b);  
}  
  
function soma(x, y) {  
    return x + y;  
}  
  
console.log(calcular(5, 3, soma));
```

Arrow Functions

- Sintaxe mais curta para funções anônimas

Exemplo:

```
const dobro = numero => numero * 2;  
console.log(dobro(4));
```

Promises e Async/Await

- Permitem trabalhar com código assíncrono
- Promises representam valores que estarão disponíveis no futuro

Exemplo:

```
const minhaPromise = new Promise((resolve, reject) => {  
  setTimeout(() => resolve("Dados carregados!"), 2000);  
});  
  
minhaPromise.then(console.log);
```

Async/Await

- Forma mais legível de trabalhar com promessas

Exemplo:

```
async function carregarDados() {  
  let resposta = await minhaPromise;  
  console.log(resposta);  
}  
carregarDados();
```


Manipulação do DOM Avançada

- Criando e removendo elementos

Exemplo:

```
const novoElemento = document.createElement("p");  
novoElemento.textContent = "Novo parágrafo!";  
document.body.appendChild(novoElemento);
```

Eventos Avançados

- Adicionando eventos dinamicamente

Exemplo:

```
document.getElementById("botao").addEventListener("click", () => {  
    alert("Botão clicado!");  
});
```

LocalStorage e SessionStorage

- Armazena dados no navegador

Exemplo:

```
localStorage.setItem("nome", "Bruno");  
console.log(localStorage.getItem("nome"));
```

Conclusão

- JavaScript permite interatividade poderosa
- Manipulação do DOM e eventos avançados
- Uso de Promises e Async/Await para assíncronismo

Obrigado!

Dúvidas?