

Curso de

JavaScript

Links importantes

Escaneie esses QR Codes para acessar os links importantes do curso

GRUPO DO WHATSAPP



SERVIDOR DO DISCORD



Respostas Atividade de Nivelamento

O que é o DOM e qual sua função no JavaScript?

De forma resumida, DOM é a forma que permite que o Javascript possa interagir com o HTML e o CSS

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>3a-aula</title>
7   <script src="script.js"></script>
8 </head>
9 <body>
10  <button id="btn-hello" onclick="mostrarMensagem()">Clique Aqui!</button>
11
12  <h1 id="hello"></h1>
13 </body>
14 </html>
```

```
1 function mostrarMensagem() {
2   const msg = document.getElementById('btn-hello');
3
4   const hello = document.getElementById('hello')
5
6   hello.innerHTML =
7     '<h1>Hello World!</h1>'
8
9 }
```

Respostas Atividade de Nivelamento

Escreva um código que altere o texto de um elemento `<p id="mensagem">` para "Mensagem alterada".

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>3a-aula</title>
7   <script src="script.js"></script>
8 </head>
9 <body>
10  <p id="mensagem">Mensagem</p>
11  <button id="btn-message" onclick="mudarMensagem()">Mudar mensagem</button>
12 </body>
13 </html>
```

```
1 function mudarMensagem() {
2   const change = document.getElementById('btn-message');
3
4   const msg = document.getElementById('mensagem');
5
6   msg.innerText = 'Mensagem Alterada';
7 }
```

Respostas Atividade de Nivelamento

Crie um botão que, ao ser clicado, exiba um alerta "Botão clicado!".

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>3a-aula</title>
7   <script src="script.js"></script>
8 </head>
9 <body>
10  <button id="btn-message" onclick="mensagemClicado()">
11    Mudar mensagem
12  </button>
13 </body>
14 </html>
```

```
1 function mensagemClicado() {
2   const msg = document.getElementById('btn-message');
3
4   alert('Botão Clicado!');
5
6 }
```

Respostas Atividade de Nivelamento

O que é **AJAX** e por que é importante no desenvolvimento web?

AJAX é uma forma de o Javascript conversar com o servidor sem precisar carregar a página inteira.

Tornando a página mais rápida, dinâmica e interativa.

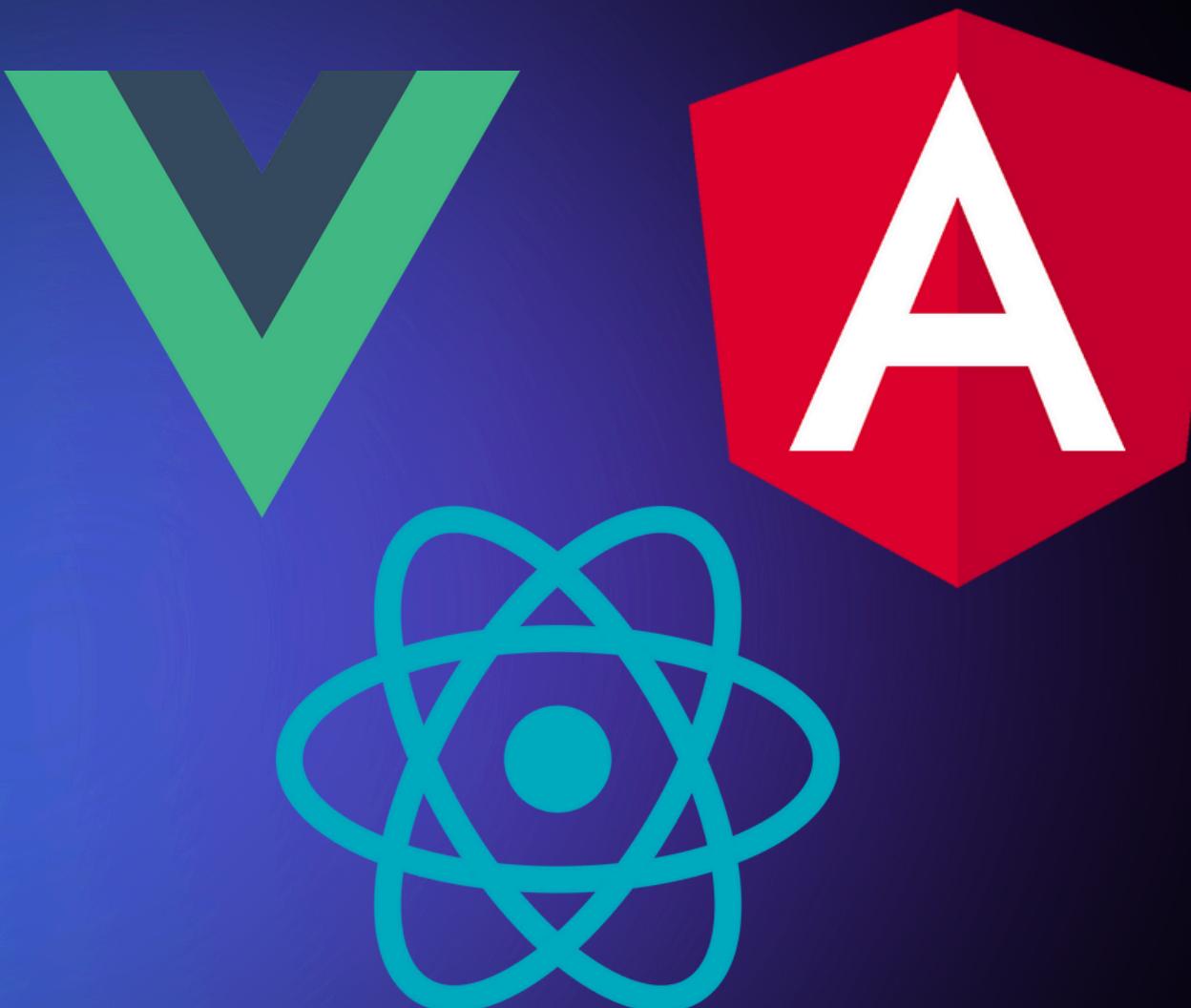
Exemplo:



Respostas Atividade de Nivelamento

Cite uma vantagem e uma desvantagem de usar frameworks como React, Angular ou Vue.

Frameworks auxiliam na construção e organização do código, também permitindo uma construção muito mais rápida, porém, em projetos pequenos pode se tornar desnecessário



Respostas Atividade de Nivelamento

Explique o que significa “componentização” em frameworks front-end.

Componentização se refere em dividir a interface de um site ou de aplicativo em certas partes independentes e reutilizáveis, chamados de componentes.

Um botão de ação pode ser reutilizado diversas vezes durante o funcionamento de um app/site, podendo mudar só a ação, mas o componente é o mesmo

Arrays

Arrays

Um array nada mais é que uma maneira de armazenar diversos valores diferentes dentro de uma mesma variável, se comporta como uma lista e seu tamanho pode ser atualizado de maneira dinâmica



JS

```
1 const array = [1,2,3,4,5,6,7,8]
```



JS

```
1 const array2 = ["a","b","c"]
```



JS

```
1 const array3 = ["palavra 1", "palavra 2", "palavra 3"]
```

Declarando e chamando uma item de um array

Um array é declarado da mesma forma que uma variável, com o prefixo **const**, **let** ou **var**, seus valores iniciais são ditados entre **colchetes** ([]) e seus valores são separados por vírgula.



```
JS  
1 const frutas = ["Maçã", "Banana", "Kiwi", "Limão"]
```



JS

```
1 console.log(frutas[2]);  
2 // Kiwi
```



JS

```
1 console.log(`Lista de compras: ${frutas}`)  
2 // Lista de compras: Maçã,Banana,Kiwi,Limão
```

Alterando a estrutura de um array

Arrays em Javascript são dinâmicos, isso é, ao contrário de outras linguagens seu tamanho **não precisa ser delimitado na hora de sua criação**, podendo ser atualizado a qualquer momento.

A diferença entre Arrays, **const** e **let** estão na reatribuição de do próprio array!



JS

```
1 const array = [1,2,3,4]
2 //Log: 1,2,3,4
```



JS

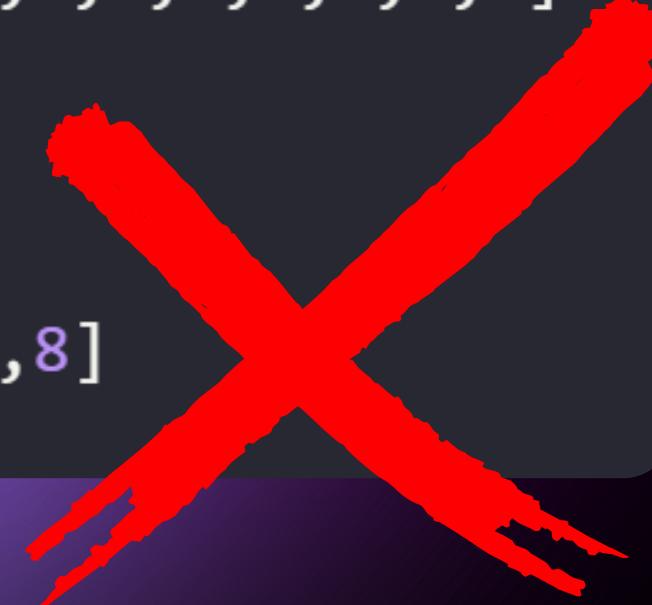
```
1 array.push(54) ← Adiciona um item no final
2 //Log: 1,2,3,4,54
3
4 array[0] = 99 ← Seta o valor de um item
5 //Log: 99,2,3,4,54
6
7 array.pop(); ← Retira o último item do
8 //Log: 99,2,3,4
9
10 array.splice(1,2); ← Retira todos os itens
11 //Log: 9,4 dentro de um intervalo
```

Diferença entre Array Const e Let



JS

```
1 const arrayConst = [1,2,3,4,6,7,8,9]
2
3
4
5 arrayConst = [9,6,3,4,8]
```



JS

```
1 let arrayLet = [1,2,3,4,5,6,7,8,9]
2
3
4
5 arrayLet = [9,6,3,4,8]
```



Exemplo

Usando arrays em uma lista de compras

Estruturas de Repetição

Estruturas de Repetição

Estruturas de repetição são usadas onde você precisa realizar alguma ação repetidas vezes, até determinada condição ser satisfeita



JS

```
1 let i = 0
2 while(i<10){
3     console.log(i)
4     // Printará números de 0 a 9
5     i++
6 }
```



JS

```
1 for(let i = 0; i<10; i++){
2     console.log(i); // Printará números de 0 a 9
3 }
```



JS

```
1 const frutas = ["maçã", "banana", "uva"];
2 for (const fruta of frutas) {
3     console.log(fruta);
4     // Printará maçã, banana, uva
5 }
```

Estruturas de repetição em Arrays

Em Javascript existem alguns tipos de métodos de repetição.

Cada um deles é usado em alguma situação, podendo ser usado para percorrer um array, ou criar arrays novos



JS

```
1 // Percorre cada item
2 [1, 2, 3].forEach(num => console.log(num));
3 // Resultado: 1,2,3
4
5 // Cria um novo array transformado
6 const dobrados = [1, 2, 3].map(num => num * 2);
7 console.log(dobrados); // Resultado: 2, 4, 6
8
9 // Cria um novo array só com os que passam no teste
10 const pares = [1, 2, 3, 4].filter(num => num % 2 === 0);
11 console.log(pares); // Resultado: 2, 4
```

Funções dentro de uma estrutura de repetição



```
1 const palavras = ["ovo", "madeira", "asa"];
2
3 palavras.forEach((palavra) => {
4     const contrario = palavra.split("").reverse().join("");
5     if(contrario == palavra){
6         console.log(`#${palavra} é um palíndromo`)
7     }else{
8         console.log(`#${palavra} não é um palíndromo`)
9     }
10 })
```

ovo é um palíndromo

madeira não é um palíndromo

asa é um palíndromo



Exemplo

Melhorando a lista de compras

Objetos

Objetos

Em JavaScript, objetos são estruturas de dados que permitem armazenar múltiplos valores relacionados em uma única variável. Pense neles como "caixas" que contêm propriedades (características) e métodos (ações).



JS

```
1 // Exemplo básico de um objeto
2 const usuario = {
3     nome: "Maria",
4     idade: 25,
5     email: "maria@email.com"
6 };
```

Formas de declarar um objeto em JavaScript

Declaração simples



JS

```
1 const produto = {  
2     nome: "Notebook",  
3     preco: 2500.00,  
4     categoria: "eletrônicos",  
5     disponivel: true  
6 };
```

Construtor



JS

```
1 const prod = new Produto();  
2 prod.nome = "Notebook";  
3 prod.preco = 2500.0;  
4 prod.categoria = "eletrônicos";  
5 prod.disponivel = true;
```

Formas de declarar um objeto em JavaScript

Função construtora



JS

```
1  function Produto(nome, preco, categoria, disponivel) {  
2      this.nome = nome;  
3      this.preco = preco;  
4      this.categoria = categoria;  
5      this.disponivel = disponivel;  
6      this.vender = function() {  
7          if (this.disponivel === true) {  
8              return `O produto ${this.nome} foi vendido por R$ ${this.preco}`;  
9          } else {  
10              return `O produto ${this.nome} não está mais disponível`;  
11          }  
12      };  
13  }  
14  
15 const prod = new Produto("Notebook", 2500.0, "eletrônicos", true);  
16
```

Acessar e Modificar Propriedades

JavaScript suporta duas formas de acessar as propriedades de objetos, sendo elas a **Notação de Ponto** (mais comum) e a **Notação de Colchetes**.

Notação de ponto



JS

```
1 console.log(produto.nome); // "Notebook"
2 produto.disponivel = false; // Modificando
```

Notação de colchetes



JS

```
1 console.log(produto["categoria"]); // "eletrônicos"
2 produto["marca"] = "dy"; // Adicionando nova propriedade
```

Metodos em Objetos



JS

```
1 const calculadora = {  
2     somar: function(a, b) {  
3         return a + b;  
4     },  
5     multiplicar(a, b) {  
6         return a * b;  
7     },  
8     dividir: (a, b) => a / b  
9 };  
10  
11 console.log(calculadora.somar(5, 3)); // 8
```

JSON

JSON (JavaScript Object Notation) é um formato de texto usado para organizar e armazenar dados de forma simples e legível. Ele funciona como uma lista de pares chave: valor. Muito usado em sistemas e APIs, facilita a troca de informações entre diferentes linguagens e aplicações.



JS

```
1 // Convertendo objeto para JSON
2 const dados = { nome: "Carlos", idade: 35 };
3 const json = JSON.stringify(dados);
4
5 // Convertendo JSON para objeto
6 const objetoNovo = JSON.parse(json);
7
8 // Exemplo com fetch
9 fetch('/api/dados')
10    .then(response => response.json())
11    .then(objeto => {
12      console.log(objeto.propriedade);
13    });
14
```

Exercício

**Exercício já disponível na pasta “Aula 4” na branch
master**

Links importantes

Escaneie esses QR Codes para acessar os links importantes do curso

GIT HUB DO CURSO



COMANDOS GITHUB



Chamada

