

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

Escreva um código que declare uma variável `idade` e exiba "Você tem [idade] anos" no console.



JS

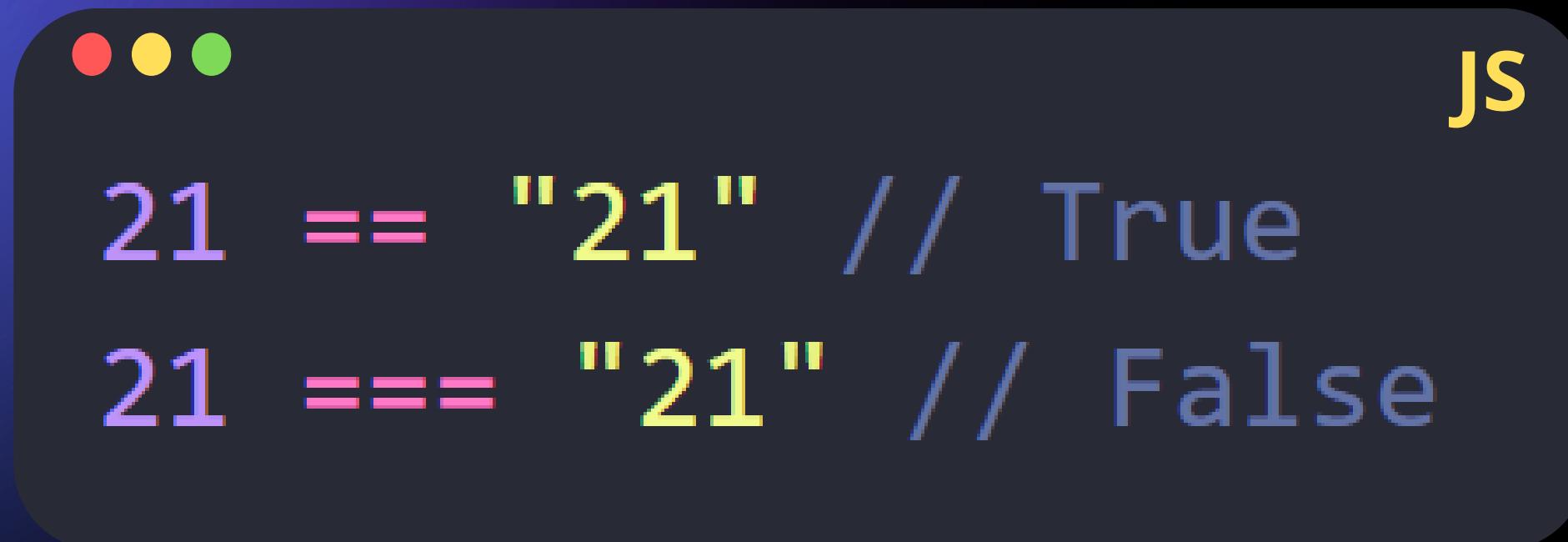
```
function exibirNome(){
    const idade = 21;
    console.log(`Você tem ${idade} anos`);
}

// Você tem 21 anos
```

Respostas Atividade de Nivelamento

Explique a diferença entre == e === no JavaScript.

" == " Igual fraco, considera apenas o conteúdo
" === " Igual forte, considera o tipo



```
JS
21 == "21" // True
21 === "21" // False
```

Respostas Atividade de Nivelamento

Complete o código para exibir todos os números ímpares de 1 a 9:

```
for (let i = 1; i <= 9; i++) { }
```



JS

```
for (let i = 1; i <= 9; i++) {
    if(i%2 != 0){
        console.log(i);
    }
}
```

Respostas Atividade de Nivelamento

Crie uma função que receba dois números e retorne a soma deles.



JS

```
function soma(valor1, valor2){  
    const somado = valor1+valor2;  
    return somado;  
}
```

Respostas Atividade de Nivelamento

Qual a diferença entre função declarada e função anônima? Dê um exemplo de cada.

Função Declarada = É declarada com "function", pode ser chamada de qualquer lugar do código
Função Anônima = É declarada em uma variável, só pode ser usada após sua declaração

● ● ● JS
console.log(soma(2, 3)); // 5

function soma(a, b) {
 return a + b;
}

● ● ● JS
console.log(multiplicar(2, 3)); // Erro

const multiplicar = function(a, b) {
 return a * b;
};

console.log(multiplicar(2, 3)); // 6

Respostas Atividade de Nivelamento

Dado o objeto:

let aluno = { nome: "Maria", idade: 21, curso: "Sistemas de Informação" };

Escreva o código para acessar a propriedade nome.



```
console.log(aluno.nome); // Maria  
console.log(`A aluna se chama ${aluno.nome}`);  
// A aluna se chama Maria
```

JS

Resolução --- Exercícios

Reposta Exercício 01

Exercício 1: Some o valor de dois campos

1. Criar uma função
2. Buscar os valores do HTML
3. Converter os valores em número
4. Somar os valores
5. Printar no console

Reposta Exercício 02

Exercício 2: Logar um número crescente no console

1. Criar uma variável fora da função
2. Criar uma função
3. Printar a variável no console
4. Aumentar o número

Reposta Exercício 03

Exercício 3: Imprima no console se o número do campo é par ou ímpar.

- 1. Criar uma função**
- 2. Buscar o valor do campo do HTML**
- 3. Verificar se é par ou ímpar**
- 4. Printar o resultado**

Reposta Exercício 04

Desafio: Imprima "Hello World" na tela, e a cada vez que ele for clicado a partir da primeira, alterar entre duas cores

1. Criar uma variável de controle fora da função
2. Criar uma função
3. Buscar o campo do HTML
4. Mudar seu texto
5. Utilizar a variável de controle para realizar a mudança de cor
6. Mudar a cor

Funções

Funções

Uma função em Javascript é nada mais nada menos que um conjunto de instruções que executa alguma tarefa, ou define algum valor, conforme o que for definido dentro dela.



JS

```
function somarNumeros() {  
    const campo1 = document.getElementById('valor1').value;  
  
    const campo2 = document.getElementById('valor2').value;  
  
    const numero1 = parseFloat(campo1);  
    const numero2 = parseFloat(campo2);  
  
    const soma = numero1 + numero2;  
  
    alert('A soma dos números é: ' + soma);  
}
```

Declarando e chamando uma função

Uma função é declara em JavaScript com o prefixo “function” seguido de seu nome e uma abertura e fechamento de parênteses, e abertura de chaves.

Para chamá-la, deve-se simplesmente declarar o nome da função seguido de abertura e fechamento de parênteses.



```
function helloworld(){
    console.log("Hello World");
}
```



```
<button onclick="HelloWorld()"></button>
```



```
function funcao(){
    helloworld();
}
```

Parâmetros e Retorno

Parâmetros são valores que a função recebe ao ser chamada e usa durante sua execução, são definidos e enviados entre parênteses.

Retorno é um valor que a função retornará para o local onde foi chamada.



JS

```
function maiusculo(texto){  
    const maiusculo = texto.toUpperCase();  
    return maiusculo;  
}
```



```
function puxarDoHtml(){  
    const valor = maiusculo(document.getElementById("input").value);  
    console.log(valor);  
}
```

JS



JS

```
function funcao(){  
    const texto = maiusculo("um texto qualquer");  
    console.log(texto);  
    // UM TEXTO QUALQUER  
}
```

Exemplo

Usar uma função para realizar uma soma

Funções Anônimas

Funções anônimas são funções sem um nome definido, geralmente criadas no exato lugar onde serão usadas e normalmente são atribuídas a variáveis, passadas como argumento para outras funções, ou em estruturas que exigem uma função rápida, sem precisar reutilizá-la em vários pontos do código.



JS

```
const soma = function(a, b) {  
    return a + b;  
};  
  
console.log(soma(2, 3));
```

Arrow Functions

Arrow Functions é basicamente um jeito mais curto e moderno de escrever funções em javascript. Substitui o “function” pela setinha (=>), e caso seja uma função simples, não se torna necessário utilizar o return, nem mesmo as chaves {}.



JS

```
const soma = (a, b) => a + b;  
console.log(soma(3, 4));
```



JS

```
const oi = () => "Olá!";  
console.log(oi());
```



JS

```
const criarPessoa = (nome, idade) => ({ nome, idade });  
console.log(criarPessoa("João", 25));
```

Exemplo

Utilizando funções anônimas e Arrow Functions

Operador ternário

O operador condicional (ternário) é o único operador JavaScript que possui três operandos. Este operador é frequentemente usado como um atalho para a instrução `if`.



JS

```
condição ? valorSeVerdadeiro : valorSeFalso;
```



JS

```
let idade = 20;
```

```
let podeBeber = idade >= 18 ? "Pode beber" : "Não pode beber";
```

```
console.log(podeBeber);
```

Condicionais

Serve para tomar decisões no seu código, "Se for isso, então faça uma coisa, se não, faça outra coisa.



```
let idade = 18;

if (idade >= 18) {
    console.log("Você é maior de idade!");
}
```



```
let idade = 16;

if (idade >= 18) {
    console.log("Você é maior de idade!");
} else {
    console.log("Você é menor de idade!");
}
```



JS

```
let nota = 7;

if(nota >= 9) {
    console.log("Excelente!");
} else if(nota >= 7) {
    console.log("Bom!");
} else {
    console.log("Precisa melhorar.");
}
```

Exercício

Exercício já disponível na pasta “Aula 3” na branch master

Links importantes

Escaneie esses QR Codes para acessar os links importantes do curso

GIT HUB DO CURSO



COMANDOS GITHUB



Chamada

