

Introdução às funções externas Java Script

Othon Oliveira

SENAC - PROA



Sumário

1 Entrada de dados externo

2 Desafio



O conceito de **Módulo** em JavaScript

Em JavaScript, um módulo é um arquivo que encapsula e organiza funcionalidades relacionadas. Ele pode conter variáveis, funções, classes e outros elementos. Através do `import`, você pode acessar esses elementos de outros arquivos.



Exemplo 1: Importar uma função específica de um módulo

Suponha que você tenha um módulo `meuModulo.js`:

```
// meuModulo.js
export function minhaFuncao() {
  console.log("Executando minha função!");
}
```

No seu arquivo principal:

```
// Arquivo principal
import { minhaFuncao } from './meuModulo';
minhaFuncao();
```



Exemplo 2: Importar várias funcionalidades de um módulo

No módulo meuModulo.js:

```
// meuModulo.js
export function funcao1() { /* ... */ }
export function funcao2() { /* ... */ }
export function funcao3() { /* ... */ }
```

No seu arquivo principal:

```
// Arquivo principal
import { funcao1, funcao2, funcao3 } from './meuModulo';
funcao1();
funcao2();
funcao3();
```



Exemplo 3: Importar todo o módulo para usar suas funcionalidades

No módulo meuModulo.js:

```
// meuModulo.js  
export function minhaFuncao() { /* ... */ }  
export function outraFuncao() { /* ... */ }
```

No seu arquivo principal:

```
// Arquivo principal  
import * as meuModulo from './meuModulo';  
meuModulo.minhaFuncao();  
meuModulo.outraFuncao();
```



Chamando Função JavaScript de um Arquivo Externo

Considere a estrutura de pastas:

```
projeto/  
|-- index.html  
|-- js/  
    |-- funcoes.js
```

No arquivo funcoes.js (dentro da pasta js/):

```
// funcoes.js  
function mostrarMensagem() {  
    alert("Função chamada!");  
}
```



Chamando Função JavaScript de um Arquivo Externo

No arquivo index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Chamando Função JavaScript</title>
  <script src="js/funcoes.js"></script>
</head>
<body>
  <button onclick="mostrarMensagem()">Clique-me</button>
</body>
</html>
```

Ao clicar no botão, a função `mostrarMensagem()` do arquivo `funcoes.js` será chamada.



Importação e Exportação de Funções

Considere a estrutura de pastas:

```
projeto/  
|-- index.html  
|-- js/  
    |-- funcoes.js  
    |-- modulo.js
```

No arquivo modulo.js (dentro da pasta js/):

```
// modulo.js  
export function mostrarMensagem() {  
    alert("Função chamada!");  
}
```



Exemplo: Importação e Exportação de Funções

No arquivo `funcoes.js` (dentro da pasta `js/`):

```
// funcoes.js
import { mostrarMensagem } from './modulo.js';

export function outraFuncao() {
  console.log("Outra função chamada!");
  mostrarMensagem();
}
```



Exemplo: Importação e Exportação de Funções

No arquivo index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">\\

  <title>Chamando Função JavaScript</title>
  <script type="module" src="js/funcoes.js"></script>
</head>
<body>
  <button onclick="outraFuncao()">Clique-me</button>
</body>
</html>
```

Ao clicar no botão, a função `outraFuncao()` do arquivo `funcoes.js` será chamada e ela por sua vez chamará a função `mostrarMensagem()` do

Exercícios de Estruturas de Repetição

Exercício 1: Imprima os números de 1 a 10.

Exercício 2: Imprima os números pares de 2 a 20.

Exercício 3: Calcule a soma dos números de 1 a 100.

Exercício 4: Imprima a tabuada do 5 (de 1 a 10).

Exercício 5: Conte quantas vogais há em uma palavra.

Exercício 6: Peça ao usuário um número e imprima os números de 1 até esse número.

Exercício 7: Peça ao usuário para digitar números até que ele digite zero. Imprima a soma dos números.

Exercício 8: Peça ao usuário para adivinhar um número entre 1 e 10. Continue pedindo até ele acertar.

Exercício 9: Peça ao usuário para digitar 5 números e imprima a média deles (utilize estrutura de repetição).

Exercício 10: Peça ao usuário para digitar um número e calcule seu fatorial.



Exemplo de formulário para apresentação da solução

```
<!DOCTYPE html>
<html>
<head>
  <title>Exercício 1</title>
  <script src="../../scripts/script.js"></script>
</head>
<body>
  <h1>Exercício 1</h1>
  <button onclick="exercicio1()">Clique para Executar</button>
</body>
</html>
```



Exemplo de formulário para apresentação da solução

```
<!DOCTYPE html>
<html>
<head>
  <title>Exercício 1</title>
  <script src="./script.js"></script>
</head>
<body>
  <h1>Exercício 1</h1>
  <button onclick="exercicio1()">Clique para Executar</button>
</body>
</html>
```



Resolva os exercícios que abordam o uso de funções e estrutura de repetição

1. **Soma dos número pares:** Crie uma função que receba um número inteiro positivo como parâmetro e retorne a soma de todos os números pares de 1 até esse número
2. **Sequência de Fibonacci:** Crie uma função que gere e retorne os primeiros n números da sequência de Fibonacci. A sequência começa com 0 e 1, e cada número subsequente é a soma dos dois números anteriores.
3. **Contagem de dígitos:** Crie uma função que conte e retorne quantos dígitos um número inteiro possui. Por exemplo, o número 12345 possui 5 dígitos.
4. **Soma de dígitos:** Crie uma função que some os dígitos de um número e verifique se ele é divisível por 3

