

JavaScript: Fundamentos para Desenvolvimento Web Interativo - Turma 2025B

2.6 Concatenação

Em JavaScript, é possível concatenar (juntar) tipos diferentes de dados em uma expressão, como strings, números e até mesmo objetos. Quando você concatena tipos diferentes, o JavaScript automaticamente converte esses valores para um tipo comum, a fim de realizar a operação de concatenação.

Existem várias maneiras de realizar concatenação em JavaScript, mas veremos aqui os mais usados.

Operador + (operador de soma)

O operador `+` é amplamente utilizado para concatenar strings em JavaScript. Quando usado entre duas strings, ele une essas strings em uma única string. Veja um exemplo no quadro abaixo:

```
let saudacao = "Olá";  
let nome = "Maria";  
let mensagem = saudacao + ", " + nome + "!";  
console.log(mensagem); // Saída: "Olá, Maria!"
```

Template literals

Os template literals, introduzidos no ECMAScript 6, oferecem uma sintaxe mais legível e poderosa para concatenar strings, permitindo interpolação de variáveis e expressões dentro de strings. Veja um exemplo no quadro abaixo:

```
let saudacao = "Olá";  
let nome = "Maria";  
let mensagem = `${saudacao}, ${nome}!`;  
console.log(mensagem); // Saída: "Olá, Maria!"
```

Método concat()

O método **concat()** é usado para unir duas ou mais strings, retornando uma nova string resultante da concatenação. Veja um exemplo no quadro abaixo:

```
let saudacao = "Olá";  
let nome = "Maria";  
let mensagem = saudacao.concat(", ", nome, "!");  
console.log(mensagem); // Saída: "Olá, Maria!"
```

Spread Operator (...)

O **Spread Operator (...)** é uma característica poderosa introduzida no ES6 que permite expandir iteráveis, como arrays, em lugares onde múltiplos argumentos ou elementos são esperados. No contexto de concatenação, o Spread Operator pode ser utilizado para combinar facilmente os elementos de múltiplos arrays em um novo array.

A sintaxe do Spread Operator é simples: basta usar o operador ... seguido do nome do array que você deseja expandir. Veja um exemplo no quadro abaixo:

```
let array1 = [1, 2, 3];  
let array2 = [4, 5, 6];  
let newArray = [...array1, ...array2];  
console.log(newArray); // Saída: [1, 2, 3, 4, 5, 6]
```

Neste exemplo, o Spread Operator (...) é utilizado para expandir os elementos de **array1** e **array2**, que são então concatenados em **newArray**.

◀ 2.5 Conversão de tipos de dados

Seguir para...

2.7 Expressões ▶

[Baixar o aplicativo móvel.](#)