

12 - Operador Spread

O operador spread é um recurso poderoso que pode simplificar e tornar seu código muito mais flexível. Prontos para expandir seus conhecimentos?

O que é o Operador Spread?

O Operador Spread, representado por três pontos (...), permite que você "espalhe" os elementos de um array ou de um objeto em outro lugar. Isso pode ser incrivelmente útil em várias situações.

Usando Spread com Arrays

```
let numeros = [1, 2, 3];  
let maisNumeros = [...numeros, 4, 5, 6];  
console.log(maisNumeros); // [1, 2, 3, 4, 5, 6]
```

Aqui, usamos o Spread para criar um novo array `maisNumeros`, que contém todos os elementos de `numeros` seguidos por 4, 5, 6. É uma maneira elegante e concisa de combinar arrays.

Usando Spread com Objetos

```
let pessoa = { nome: 'João', idade: 25 };  
let pessoaAtualizada = { ...pessoa, idade: 26 };  
console.log(pessoaAtualizada); // { nome: 'João', idade: 26 }
```

Neste exemplo, criamos um novo objeto `pessoaAtualizada` que possui todas as propriedades de `pessoa`, mas com a `idade` atualizada. O Spread aqui nos permite copiar facilmente propriedades entre objetos.

Aplicações Práticas do Spread

O Spread é útil em diversas situações, como na manipulação de dados e na passagem de argumentos para funções.

Copiando Arrays

```
let original = [1, 2, 3];  
let copia = [...original];  
console.log(copia); // [1, 2, 3]
```

Ao usar o Spread, você cria uma cópia superficial do array. Isso é particularmente útil quando você não quer que as mudanças no novo array afetem o original.

Concatenando Arrays

```
let inicio = [1, 2];  
let fim = [3, 4];  
let combinado = [...inicio, ...fim];  
console.log(combinado); // [1, 2, 3, 4]
```

O Spread torna a concatenação de arrays uma tarefa simples e limpa.

Espalhando Argumentos

```
function somar(a, b, c) {  
  return a + b + c;  
}  
  
let numeros = [1, 2, 3];  
console.log(somar(...numeros)); // 6
```

Aqui, espalhamos os elementos de **numeros** como argumentos individuais para a função **somar**.

Combinando Objetos

```
let objeto1 = { a: 1, b: 2 };  
let objeto2 = { c: 3, d: 4 };  
let combinado = { ...objeto1, ...objeto2 };  
console.log(combinado); // { a: 1, b: 2, c: 3, d: 4 }
```

O Spread também é eficaz para combinar ou clonar objetos.

Exercícios

1. Criando uma Lista de Compras: Use o operador spread para combinar dois arrays de itens de supermercado em uma lista de compras.
2. Atualizando um Perfil de Usuário: Crie um objeto `usuario` com propriedades como `nome` e `email`. Use o operador spread para criar um novo objeto com os dados do usuário e uma propriedade adicional `status`.
3. Festa de Aniversário: Crie uma função que aceita um número variável de nomes e os imprime como lista de convidados para uma festa de aniversário, usando o operador spread para coletar os nomes.
4. Clonando um Array de Músicas: Dado um array de músicas, crie uma cópia do array usando o operador spread. Altere um elemento no array copiado e verifique se o array original permanece inalterado.
5. Mesclando Objetos com Informações Complementares: Crie dois objetos, um com informações básicas de um filme (título e diretor) e outro com informações complementares (ano e gênero). Use o operador spread para mesclar esses objetos em um único objeto `filme`.
6. Criando um Menu de Restaurante Variável: Faça uma função `criarMenu` que aceita vários itens (entradas, pratos principais, sobremesas) como arrays e usa o operador spread para criar um único array de menu.