

## 8 - Estrutura de Dados Set

### O que é um Set?

Um Set é como um array, só que cada valor nele é único. Nada de repetição aqui, galera! Isso é super útil quando você quer garantir que cada elemento seja único, tipo quando tá lidando com uma lista de IDs ou algo assim.

Criando um Set

```
let meuSet = new Set();
```

Aqui, estamos criando um novo Set vazio. É como abrir uma nova caixa para guardar coisas, mas só uma de cada tipo.

### Adicionando Valores ao Set

Para adicionar coisas no nosso Set, usamos o método **add**.

Exemplo de Adição

```
meuSet.add(1);  
meuSet.add(2);  
meuSet.add(2); // Esse cara não vai entrar, porque 2 já está no Set
```

Nós adicionamos 1 e 2 ao Set. Tentamos adicionar 2 de novo, mas o Set já tem um 2, então ele ignora. Sets são tipo aquele amigo que não gosta de repetição.

### Verificando a Presença de Valores

Quer saber se algo tá no Set? Usa o **has**.

Exemplo de Verificação

```
console.log(meuSet.has(1)); // true  
console.log(meuSet.has(3)); // false
```

Aqui, estamos checando se 1 e 3 estão no Set. O 1 tá lá, mas o 3 não.

## Contando os Valores

Pra saber quantos itens tem no Set, dá uma olhada no **size**.

### Exemplo de Tamanho

```
console.log(meuSet.size); // 2
```

Como só temos 1 e 2 no Set, o **size** mostra 2.

## Dando um Rolê pelo Set

Quer ver tudo que tem no Set? É só iterar sobre ele.

### Exemplo de Iteração

```
for (let valor of meuSet) {  
  console.log(valor);  
}
```

Aqui, estamos passando por cada valor do Set e dando um print. Bem prático!

## Limpando a Casa

Sets têm uns métodos maneiros tipo **delete** e **clear**.

### Exemplo de Métodos

```
meuSet.delete(1); // Tchau, 1!  
meuSet.clear();   // Limpou tudo!
```

Primeiro, a gente tira o 1 do Set. Depois, limpa tudo, deixando o Set zerado.

## Sets no Dia a Dia

Sets são ótimos pra quando você precisa de uma coleção sem repetição. Tipo, sabe quando você tem um monte de dados e precisa garantir que não tem nada duplicado? Então, Sets são perfeitos pra isso.

## Exemplo Prático

```
let numeros = [1, 2, 3, 2, 1];  
let numerosUnicos = new Set(numeros);  
console.log([...numerosUnicos]); // [1, 2, 3]
```

Pegamos um array com uns números repetidos e transformamos num Set. Depois, convertemos de volta pra array, mas agora só com valores únicos. Bem legal, né?

## Exercícios para Praticar Sets:

1. Criar e Adicionar Valores ao Set: Crie um **Set** chamado **frutas** e adicione nele três frutas diferentes. Depois, tente adicionar uma fruta que já está no **Set**.
2. Verificar Valores no Set: Use o método **has** para verificar se uma fruta específica está no seu **Set frutas**.
3. Remover Valores do Set: Use o método **delete** para remover uma fruta do seu **Set frutas**.
4. Limpar o Set: Use o método **clear** para limpar todo o seu **Set frutas**.
5. Tamanho do Set: Crie um **Set** com vários valores e use a propriedade **size** para imprimir o número de elementos no **Set**.
6. Iterar Sobre um Set: Use um loop **for...of** para iterar sobre o seu **Set** e imprimir cada valor.
7. Criar um Set a Partir de um Array: Crie um **array** com alguns valores duplicados. Em seguida, crie um **Set** a partir desse **array** para remover as duplicatas.