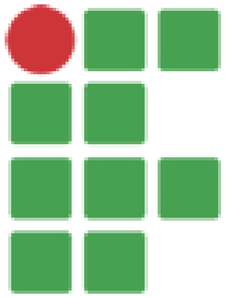

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

Disciplina

Programação Front-End I

PROF. FERNANDO TAMBERLINI ALVES

Iteradores em JavaScript

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- Iteradores são objetos que permitem percorrer (iterar) elementos um por um;
- Um iterador possui um método `.next()` que retorna:
  - `{value: <valor>, done: <boolean>}`
- São tipos iteráveis: Array, String, Map, Set, DOM Collections

```
const array = [1, 2, 3];
const iterator = array[Symbol.iterator]();
console.log(iterator.next());   //   {   value:   1,   done:   false   }
console.log(iterator.next());   //   {   value:   2,   done:   false   }
console.log(iterator.next());   //   {   value:   3,   done:   false   }
console.log(iterator.next());   //   {   value: undefined, done: true }
```

Tipos Iteráveis Nativos

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- Array

```
const frutas = ['maçã', 'banana', 'laranja’];
for (const fruta of frutas)
      {console.log(fruta); // maçã, banana, laranja }
```

- String

```
const texto = 'JavaScript’;
for (const letra of texto)
      { console.log(letra); // J, a, v, a, S, c, r, i, p, t }
```

Tipos Iteráveis Nativos

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- Map

```
const mapa = new Map([['nome', 'João'], ['idade', 30]]);
for (const [chave, valor] of mapa)
      { console.log(`${chave}: ${valor}`); // nome: João, idade: 30 }
```

- Set

```
const conjunto = new Set([1, 2, 3, 2, 1]);
for (const valor of conjunto)
      { console.log(valor); // 1, 2, 3 (valores únicos) }}
```

Comparação entre Coleções

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- Array
  - É uma lista ordenada de elementos indexados numericamente (começando do zero).
  - Permite valores duplicados
  - Acesso por índice: arr[0], arr[1], etc.

- Set
  - É uma coleção de valores únicos, não ordenada.
  - Não permite duplicatas
  - Não há índice, mas é iterável.

- Map
  - É uma coleção de pares chave/valor, semelhante a um objeto, mas com ordem garantida de inserção e qualquer tipo como chave (inclusive objetos e funções)

Laços sobre Iteráveis

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- for ... of
  - Itera valores de objetos iteráveis
  - Permite break e continue

```
for (let n of [1, 2, 3]) {
  if (n === 2) break;
  console.log(n); // 1
}
```

- forEach
  - Executa callback para cada item, sem interrupção possível

```
[1, 2, 3].forEach(n => {
  if (n === 2) return; // retorna do callback, não do loop
  console.log(n); // 1, 3
});
```

Métodos Fundamentais de Array

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- .map() : transforma cada item

```
const numeros = [1, 2, 3];
const dobrados = numeros.map(n => n * 2); // [2, 4, 6]
```

- .filter(): filtra conforme condição

```
const pares = numeros.filter(n => n % 2 === 0); // [2]
```

- .reduce(): reduz a um único valor

```
const soma = numeros.reduce((acc, n) => acc + n, 0); // 6
```

Métodos Fundamentais de Array

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- .find(): encontra o primeiro item que satisfaz condição

```
const encontrado = numeros.find(n => n > 1); // 2
```

- .some() / .every()

```
const temPar = numeros.some(n => n % 2 === 0); // true
const todosMaioresQueZero = numeros.every(n => n > 0); // true
```

- .includes(): verifica existência

```
[1, 2, 3].includes(2); // true
```

Outro exemplo

```
const usuarios = [
  { nome: "Ana", idade: 18 },
  { nome: "Carlos", idade: 25 },
  { nome: "Beatriz", idade: 30 }
];

// nomes em maiúsculas
const nomes = usuarios.map(u => u.nome.toUpperCase());
// ["ANA", "CARLOS", "BEATRIZ"]

// maiores de 20 anos
const adultos = usuarios.filter(u => u.idade > 20);
// [{ Carlos }, { Beatriz }]

// soma das idades
const totalIdades = usuarios.reduce((acc, u) => acc + u.idade, 0);
// 73
```

Métodos Fundamentais de Set

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- .add() : adiciona um novo elemento ao Set. Se o elemento já existir, ele não será adicionado novamente, pois Set só armazena valores únicos

```
const meuSet = new Set();
meuSet.add(1);
meuSet.add(2)
meuSet.add(1);
// meuSet é {1, 2}
```

- .delete(): Remove o elemento especificado do Set. Retorna true se o elemento existia e foi removido com sucesso, e false caso contrário.

```
meuSet.delete(1);
//retorna true e meuSet é {2}
```

Métodos Fundamentais de Set

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

- .has(value): Verifica se um elemento com o valor especificado existe no Set. Retorna um valor booleano (true ou false).

```
meuSet.has(2); // retorna true
meuSet.has(1); // retorna false
```

- clear(): Remove todos os elementos do Set. Não aceita parâmetros e não retorna valor.

```
meuSet.clear() //meuSet é {}
```

- size: Não é um método, mas uma propriedade que retorna a quantidade de elementos (o tamanho) do Set.

```
meuSet.size //retorna 0
```

Atividade Prática I

Exercício: Sorteador de Nomes

Objetivo: Exercício Prático sobre Iteradores

Adicionar participante e sortear

Digite o nome de um participante...

Limpar | Adicionar | Sortear

Lista de Participante(s) (0):

Atividade Prática I

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

Instruções do Exercício:

Objetivo: Praticar o uso de estrutura de dados com Javascript

O que você deve implementar:

1. **Botão Limpar:**
   - Deverá limpar o conteúdo do campo de entrada
   - Deverá limpar a lista de Participantes
2. **Botão Adicionar**
   - Deverá adicionar um participante no sorteio.
   - Não deverá permitir participante repetido.
   - Deverá tirar os espaços das extremidades e colocar em maiuscula.
   - Deverá ser informado o número de participantes.
3. **Botão Sortear**
   - Deverá sortear um participante aleatoriamente.
   - O participante sorteado deverá sair da lista de participantes após ser sorteado.

Dicas:

- Utilizar a estrutura mais adequada e os seus métodos.
- Percorrer a estrutura de dados com uso de iteradores.
- Crie uma função para cada funcionalidade, por exemplo: adicionar(), limpar() sortear().

Dúvidas?

INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Rio de Janeiro

Dúvidas?

fernando.alves@ifrj.edu.br

ftamberlini.dev.br