

# Introdução ao JavaScript - aula 03

## 1. Arrays

Arrays são estruturas de dados utilizadas para armazenar múltiplos valores em uma única variável. No JavaScript, um array pode conter diferentes tipos de dados e possui métodos poderosos para a manipulação desses arrays.

Os principais métodos de manipulação são:

- **push():** adiciona um elemento ao **final** do array;
- **unshift():** adiciona um elemento ao **início** do array.
- **pop():** **remove** e **retorna** o último elemento do array;
- **shift():** **remove** e **retorna** o primeiro elemento do array;
- **map():** retorna um **novo array** com a transformação aplicada a cada elemento;
- **filter():** retorna um **novo array** contendo apenas os elementos que atendem a condição;
- **reverse:** **inverte** a ordem dos elementos no array;
- **at(index):** retorna o **elemento** na **posição especificada** ;
- **slice(inicio, fim):** retorna uma **cópia parcial** do array sem modificá-lo.

## Exemplos:

- Adicionando e removendo elementos

```
let frutas = ["maçã", "banana"];
```

```
// push - Adiciona no final
```

```

frutas.push("laranja");
console.log(frutas); // ["maçã", "banana", "laranja"]

// unshift - Adiciona no início
frutas.unshift("uva");
console.log(frutas); // ["uva", "maçã", "banana", "laranja"]

// pop - Remove do final
let ultimaFruta = frutas.pop();
console.log(ultimaFruta); // "laranja"
console.log(frutas); // ["uva", "maçã", "banana"]

// shift - Remove do início
let primeiraFruta = frutas.shift();
console.log(primeiraFruta); // "uva"
console.log(frutas); // ["maçã", "banana"]

```

- Transformação e filtragem

```

let numeros = [1, 2, 3, 4, 5];

// map - Multiplicando cada número por 2
let dobrados = numeros.map(num => num * 2);
console.log(dobrados); // [2, 4, 6, 8, 10]

// filter - Pegando apenas os números pares
let pares = numeros.filter(num => num % 2 === 0);
console.log(pares); // [2, 4]

```

- Invertendo e acessando elementos

```

let letras = ["a", "b", "c", "d", "e"];

// reverse - Inverte a ordem dos elementos
let invertido = [...letras].reverse();

```

```
console.log(invertido); // ["e", "d", "c", "b", "a"]

// at - Obtendo elementos por índice (suporta negativos)
console.log(letras.at(2)); // "c"
console.log(letras.at(-1)); // "e" (último elemento)
```

- Extraíndo parte de um array

```
let animais = ["cachorro", "gato", "elefante", "leão", "tigre"];

// slice - Pegando elementos do índice 1 ao 3 (não inclui o último)
let selecionados = animais.slice(1, 4);
console.log(selecionados); // ["gato", "elefante", "leão"]
```

## 2. Spread Operator

o spread operator (...) permite a cópia ou combinação de arrays de forma simples

### Exemplos:

```
let lista1 = [1, 2, 3];
let lista2 = [4, 5, 6];

// Juntando arrays
let uniao = [...lista1, ...lista2];
console.log(uniao); // [1, 2, 3, 4, 5, 6]

// Copiando um array
let copia = [...lista1];
copia.push(10);
console.log(lista1); // [1, 2, 3] (original inalterado)
console.log(copia); // [1, 2, 3, 10]
```