

Exercícios Aula 04

Exercício: `push()`

1. Crie um novo array vazio.
2. Utilize o método "push" para adicionar um elemento ao final do array.
3. Utilize o método "push" novamente para adicionar outro elemento ao final do array.
4. Verifique o comprimento (length) do array para confirmar que os elementos foram adicionados corretamente.

Boa sorte!

Exercício: `pop()`

1. Temos a lista:

```
const numeros = [1, 2, 3, 4];
```

1. Utilize o método "pop" para remover o último elemento do array.
2. Verifique o comprimento (length) do array para confirmar que o elemento foi removido corretamente.
3. Utilize o método "pop" novamente para remover outro elemento do array.
4. Verifique o comprimento (length) do array novamente para confirmar que o segundo elemento foi removido corretamente.

Boa sorte!

Exercício: `shift()`

1. Temos a lista:

```
const linguagens = ['javascript', 'python', 'golang', 'java']
```

1. Utilize o método "shift" para remover o primeiro elemento do array.
2. Verifique o comprimento (length) do array para confirmar que o elemento foi removido corretamente.
3. Utilize o método "shift" novamente para remover outro elemento do array.
4. Verifique o comprimento (length) do array novamente para confirmar que o segundo elemento foi removido corretamente.

Boa sorte!

Exercício: `at()`

1. Temos a lista:

```
const produtos = ['sabao', 'detergente', 'amaciante', 'alvejante', 'sapolio']
```

1. Utilize o método "at" para acessar o primeiro elemento do array. Lembre-se de que o índice do primeiro elemento é 0.
2. Utilize o método "at" novamente para acessar o terceiro elemento do array.
3. Utilize o método "at" mais uma vez para acessar o último elemento do array. Lembre-se de que o índice do último elemento é o comprimento do array menos 1.

4. Verifique se você conseguiu acessar os elementos corretamente.

Boa sorte!

Exercício: `concat()`

1. Temos essa duas listas:

```
const lista1 = ['banana', 'pera', 'melancia'];
const lista2 = ['alface', 'tomate', 'rucula'];
```

- 1. Utilize o método "concat" para mesclar a lista1 e lista2 em um novo array.
- 2. Verifique o comprimento (length) do novo array para confirmar que os elementos foram mesclados corretamente.
- 3. Tente utilizar o método "concat" com mais algumas listas, mesclando todas em uma só.

```
const lista3 = ['limao', 'laranja', 'acerola'];
const lista4 = ['pimenta', 'pimentao', 'alho']
```

Boa sorte!

Exercício: `every()`

1. Temos essa lista de pessoas:

```
var pessoas = [
  { nome: "João", idade: 22 },
  { nome: "Maria", idade: 17 },
  { nome: "Pedro", idade: 18 },
  { nome: "Ana", idade: 22 },
  { nome: "Carlos", idade: 16 }
];
```

- 1. Utilize o método "every" para verificar se todos os alunos têm idade maior ou igual a 18 anos.
- 2. Utilize o método "every" novamente para verificar se todos os alunos têm nome com mais de 3 caracteres.
- 3. Utilize o método "every" mais uma vez para verificar se todos os alunos têm idade e nome definidos.
- 4. Verifique se você conseguiu utilizar o método "every" corretamente.

Boa sorte!

Exercício: `filter()`

1. Temos a lista de alunos:

```
const alunos = [
  { nome: 'Ana', idade: 20 },
  { nome: 'Pedro', idade: 18 },
  { nome: 'Maria', idade: 22 },
  { nome: 'João', idade: 17 },
  { nome: 'Lucas', idade: 19 },
  { nome: 'Julia', idade: 21 }
]
```

- 1. Utilize o método "filter" para criar um novo array com os alunos que têm idade maior ou igual a 18 anos.
- 2. Utilize o método "filter" novamente para criar um novo array com os alunos que têm idade menor que 20 anos.
- 3. Utilize o método "filter" mais uma vez para criar um novo array com os alunos cujo nome começa com a letra "J".
- 4. Verifique se você conseguiu utilizar o método "filter" corretamente.

Boa sorte!

Exercício: `find()`

1. Temos a lista de lugares:

```
const lugares = [
  { nome: 'Sala de Reuniões', capacidade: 8 },
  { nome: 'Auditório', capacidade: 50 },
  { nome: 'Sala de Treinamento', capacidade: 20 },
  { nome: 'Sala de Conferências', capacidade: 30 },
  { nome: 'Sala de Estudos', capacidade: 12 }
]
```

1. Utilize o método "find" para encontrar o primeiro lugar que tenha capacidade máxima de pelo menos 10 pessoas.
2. Utilize o método "find" novamente para encontrar o primeiro lugar que tenha capacidade máxima de pelo menos 25 pessoas.
3. Utilize o método "find" mais uma vez para encontrar o primeiro lugar que tenha capacidade máxima de pelo menos 40 pessoas.
4. Verifique se você conseguiu utilizar o método "find" corretamente.

Boa sorte!

Exercício: `forEach()`

1. Temos uma lista de sabores de pizza e seus valores:

```
const pizzas = [
  { sabor: 'Mussarela', valor: 20 },
  { sabor: 'Calabresa', valor: 25 },
  { sabor: 'Marguerita', valor: 28 },
  { sabor: 'Frango com Catupiry', valor: 32 },
  { sabor: 'Portuguesa', valor: 30 }
]
```

1. Utilize o método "forEach" para percorrer a lista de pizzas e imprimir no console o nome do sabor da pizza e seu valor.
2. Verifique se você conseguiu utilizar o método "forEach" corretamente.

Boa sorte!

Exercício: `map()`

1. Temos uma lista de pessoas com seus respectivos nomes, alturas e pesos:

```
const pessoas = [
  { nome: 'João', altura: 1.75, peso: 80 },
  { nome: 'Maria', altura: 1.68, peso: 60 },
  { nome: 'Pedro', altura: 1.80, peso: 70 },
  { nome: 'Ana', altura: 1.65, peso: 55 },
  { nome: 'Carlos', altura: 1.90, peso: 100 }
]
```

1. Utilize o método "map" para criar um novo array de objetos que contenha somente o nome e o IMC (índice de massa corporal) de cada pessoa, sendo que o IMC é calculado pela fórmula `peso / altura²`.
2. Verifique se você conseguiu utilizar o método "map" corretamente.

Boa sorte!

Exercício: `reduce()`

1. Temos a lista de alunos com suas notas:

```
const alunos = [
  { nome: 'Ana', notas: [7, 8, 9] },
  { nome: 'Pedro', notas: [5, 6, 7] },
  { nome: 'Maria', notas: [9, 8, 10] },
  { nome: 'João', notas: [6, 7, 8] },
  { nome: 'Lucas', notas: [8, 9, 7] },
  { nome: 'Julia', notas: [10, 8, 9] }
]
```

1. Utilize o método "map" para criar um novo objeto com o nome e a média de cada aluno.
2. A média deve ser calculada utilizando o método "reduce" para somar as notas e dividi-las pelo número de notas.
3. Verifique se você conseguiu utilizar o método "reduce" corretamente.

Boa sorte!

Exercício: `reverse()`

1. Temos a lista:

```
const numeros = [1, 2, 3, 4, 5]
```

1. Utilize o método "reverse" para inverter a ordem dos elementos do array.
2. Verifique se a lista ficou com os elementos na ordem inversa: 5, 4, 3, 2, 1.

Boa sorte!

Exercício: `some()`

1. Temos a lista de empregados com seus respectivos nomes e salários:

```
const empregados = [
  { nome: 'João', salario: 1200 },
  { nome: 'Maria', salario: 1500 },
  { nome: 'Pedro', salario: 1800 },
  { nome: 'Ana', salario: 1400 },
  { nome: 'Carlos', salario: 2000 }
]
```

1. Utilize o método "some" para verificar se pelo menos um dos empregados tem salário maior ou igual a R\$ 1.500,00.
2. Utilize o método "some" novamente para verificar se pelo menos um dos empregados tem salário menor ou igual a R\$ 1.000,00.
3. Verifique se você conseguiu utilizar o método "some" corretamente.

Boa sorte!

Exercício: `sort()`

1. Temos uma lista de atletas com seus respectivos nomes, alturas e pesos:

```
const atletas = [
  { nome: 'João', altura: 1.75, peso: 80 },
  { nome: 'Maria', altura: 1.68, peso: 60 },
  { nome: 'Pedro', altura: 1.80, peso: 70 },
  { nome: 'Ana', altura: 1.65, peso: 55 },
  { nome: 'Carlos', altura: 1.90, peso: 100 }
]
```

1. Utilize o método "sort" para ordenar a lista de atletas em ordem crescente de peso.

2. Utilize o método "sort" novamente para ordenar a lista de atletas em ordem decrescente de altura.
3. Utilize o método "sort" mais uma vez para ordenar a lista de atletas em ordem alfabética de nome.
4. Verifique se você conseguiu utilizar o método "sort" corretamente.

Boa sorte!