



## Arrays e Coleções

Vamos fortalecer nosso conhecimento sobre arrays explorando diferentes ações. Veremos se eles se comportam como esperamos ou se ocorrem resultados inesperados.

Em primeiro lugar, vamos praticar as ações mais comuns e necessárias

- A. Acessar elementos específicos de um array.
- B. Modificar cada um dos elementos de um array e imprimi-los no console.
- C. Adicionar elementos a um array.
- D. Extrair elementos de um array.
- E. Comparar elementos de um array com os elementos de outro.

## O que esses códigos retornam?

Neste exercício, você terá que pensar sobre o que esses códigos retornam sem testá-lo no console.

1. 

```
let numbers =[22, 33, 54, 66, 72]  
console.log(numbers[numbers.length])
```
2. 

```
let grupoDeAmigos = [ ["Harry", "Ron", "Hermione"], ["Spiderman", "Hulk",  
"Ironman"], ["Penélope Glamour", "Pierre Nodoyuna", "Patán"] ]
```



# Certified Tech Developer

The Ultimate Degree

```
console.log(grupoDeAmigos[1][0])
```

3. `let str = "uma string qualquer"`

```
let grupoDeAmigos = [ [45, 89, 0], ["Digital", "House", true], ["string",  
"123", "false", 54, true, str] ]
```

```
console.log(grupoDeAmigos[2][grupoDeAmigos[2].length - 1])
```

## Array Invertido

Neste exercício, você terá que criar uma função que retorne um array com seus elementos invertidos, sem modificá-lo. Então, você deve fazer uma função que o modifique e inverta a ordem de seus elementos.

1. Crie a função *imprimirInverso* que pega um Array como um argumento e imprime cada elemento em ordem reversa no console (você não precisa inverter o Array).
2. Crie a função *inverter* que recebe um Array como argumento e retorna um novo invertido.

## somarArray()

Neste exercício, você criará uma função *somarArray()* que aceita um array de números e retorna a soma de todos eles.

Exemplo:

- `somarArray([1,2,3])` // 6
- `somarArray([10, 3, 10, 4])` // 27



# Certified Tech Developer

The Ultimate Degree

- `somarArray([-5,100])` // 95

## Simulação `Array.join()`

Neste exercício, você criará uma função chamada *join* que recebe um array e simula o comportamento do método `Array.join()`.

**Importante:** Não poderá utilizar o método `Array.join()` original.

*Por exemplo:*

*`join(["o", "l", "á"])` deve retornar a string "olá".*

*`join(["t", "c", "h", "a", "u"])` deve retornar a string "tchau".*

## Coleções de Filmes (e mais...)

O tech leader da equipe precisa conhecer a estrutura de dados do novo projeto. Para isso, devemos:

1. Criar a estrutura apropriada para armazenar os seguintes filmes e séries:

*"star wars", "matrix", "mr. robot", "o preço do amanhã", "a vida é bela".*

**Importante:** verifique se tudo funciona corretamente acessando qualquer um dos filmes uma vez que a estrutura correspondente tenha sido criada.

2. Os filmes devem estar todos em letras maiúsculas. Para isso, crie uma função que recebe um array por parâmetro e converta o conteúdo de cada elemento



# Certified Tech Developer

The Ultimate Degree

em letras maiúsculas.

Dica: revise o que faz o método de strings `.toUpperCase()`.

3. Crie outra estrutura semelhante à primeira, mas com os seguintes filmes de animação:

*"Toy Story", "Procurando Nemo", "Kung-fu Panda", "Wally", "Fortnite"*

Em seguida, crie uma função que receba dois arrays como parâmetros, para poder adicionar os elementos contidos no segundo array, dentro do primeiro, a fim de retornar um único array com todos os filmes como seus elementos.

4. Durante o processo, percebemos que o último elemento na série de filmes animados é, na verdade, um game. Agora devemos editar nosso código e modificá-lo para que possamos remover o último elemento antes de migrar o conteúdo para o array que contém todos os filmes.

PS: por precaução, salve o elemento que você vai deletar em uma variável.

5. Finalmente, recebemos dois arrays com classificações feitas por diferentes usuários do mundo nos filmes com o seguinte formato:

```
//const asiaScores = [8, 10, 6, 9, 10, 6, 6, 8, 4];
```

```
//const euroScores = [8, 10, 6, 8, 10, 6, 7, 9, 5];
```

Crie uma função que compare as notas e nos diga se elas são iguais ou diferentes.



# Certified Tech Developer

The Ultimate Degree

Para verificar se tudo está bem até agora, recomendamos testar cada uma das funções, a fim de verificar seu correto funcionamento.

Se chegarmos a este ponto e tudo estiver bem, o tech leader da equipe deve estar extremamente feliz com nosso trabalho e desempenho.

Bom trabalho!