

FIAP

Arrays

Caio Oliveira
Web Development

JS



Checkpoint (CP)

Calendário **1º semestre:**

CP1 – 30/03

CP2 – 27/04

CP3 – 25/05

FIAP

1° Checkpoint

Objetivo

Replicar em Javascript alguns dos elementos dinâmicos do *AirnBnb*

O que deve ser entregue?

- Menu funcionando
- Botão de Salvar
- Avaliações dos clientes dinâmica
- Galeria de fotos

Que recursos devem ser utilizados?

- Manipulação do DOM (Ex: `classList.toggle` e `classList.add`)
- Botão de Salvar (`localStorage`)
- Consumo de arrays de objetos
- Utilizar arrow functions e template strings
- Utilizar programação funcional

FIAP

**Data de
entrega**

**Prazo final
30/03**

Regras

- Grupos de no máximo 5 pessoas
- Entrega no Teams do link do repositório no GitHub

WEB DEVELOPMENT

Com o protótipo do projeto e já criado em estrutura html nas sprints anteriores, agora será feito a migração para o React utilizando componentes, imports (20 pontos)

Objetivo:

- Desenvolver um projeto em React que contenha estrutura de componentes como por exemplo (Cabeçalho, corpo e footer) sendo que deve ser utilizar a estrutura de pai para filho utilizando funções em javascript utilizando funcionalidades usando localStorage para tratamento de dados, operações usando Math (60 pontos)

O projeto:

- Ajuste do protótipo de seu projeto e utilizar fielmente para ser desenvolvido em React utilizando imagens, vídeos caso tenha, tipos de fontes entre outros. O projeto também precisa ser versionado em uma organização do github. (20 pontos)

WEB DEVELOPMENT

Com base no projeto e já criado em React utilizando estrutura de componentes e tratamento de estilização com bootstrap (20 pontos)

Objetivo:

- Projeto em React precisa consumir de uma API utilizando recursos de json(Local), revisão do DOM e criação de eventos nas partes de seu projeto , estilização com sass (60 pontos)

O projeto:

- Estilizar o projeto com Sass e bootstrap, dentro dos componentes criados utilizar um pelo menos para consumir de uma api. (20 pontos)

Regras:

- Pode utilizar a documentação do React como guia
- O projeto deve ser criado dentro das normas do W3C.
- O projeto precisa ter criação própria utilizando os conhecimentos adquiridos
- A estrutura deve ser semântica.
- Versionamento do projeto usando organização no github

Entrega:

- O formato da entrega deve ser um arquivo compactado sem os módulos.

FIAP

**Responder o forms o
quantos antes!**

[https://forms.gle/6yZg
QqNhAGJk1hLn7](https://forms.gle/6yZgQqNhAGJk1hLn7)







FIAP






.map(=>) => 

.filter() => 



.every() => false

.some() => true

.fill(, 1) => 

.findIndex(el => el===) => 2

.find() => 

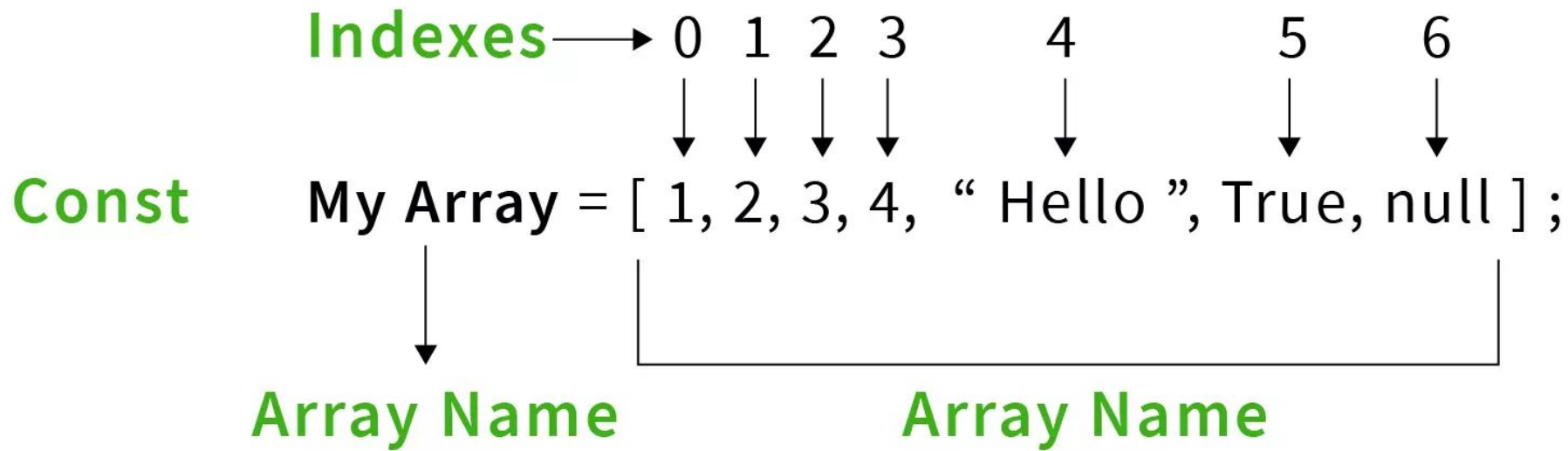
.reduce((acc, cur)=>acc+cur)=> 

FIAP

O que é um Array?

**O que são
arrays?**

É uma estrutura de dados
que armazena uma
coleção de elementos.



Index

Cada elemento dentro de um array tem um index, que é a posição do elemento, como se estivesse em uma lista numerada.

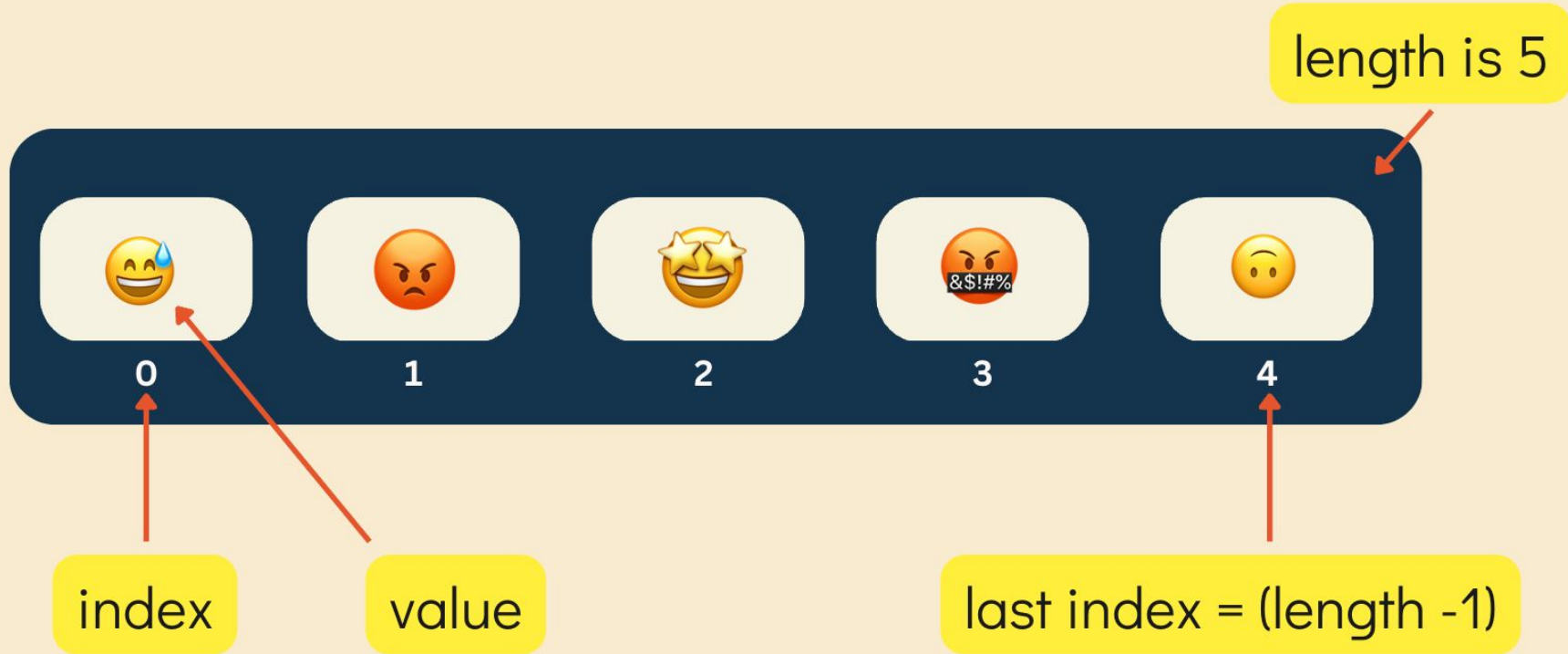
FIAP

```
let array = [1, 12, 2.5, null, 'John', true, 100]
```

	int	int	float	Null	string	bool	number
Elements: →	1	12	2.5	null	'John'	true	100
Index : → (position)	0	1	2	3	4	5	6

Javascript Array

```
const emotions = ['😓', '😡', '😄', '😡', '😓']
```



FIAP

**Você também pode
acessar caracteres
em uma string
utilizando []**

FIAP

```
const hello = 'Hello World';  
console.log(hello[6]);  
// Output: W
```

FIAP

**Como atualizar
um array?**

FIAP

```
let seasons = ['Winter', 'Spring', 'Summer',  
  'Fall'];
```

```
seasons[3] = 'Autumn';  
console.log(seasons);  
//Output: ['Winter', 'Spring', 'Summer', 'Autumn']
```

FIAP

Desestruturação

Array Destructuring: Skip Items

Leave a blank space to skip destructuring that item

```
const breakfast = ["Coffee", "Croissant", "Sunny side up"];
```



```
const [drink, , eggs] = breakfast;
```

```
// drink => 'Coffee'
```

```
// eggs  => 'Sunny side up'
```



FIAP

Spread e Rest

Array Destructuring: Rest Array

Destructure some items, assign all others to an array

```
const breakfast = ["Coffee", "Croissant", "Sunny side up", "08:38",  
"Sally", 9.15, 7];
```

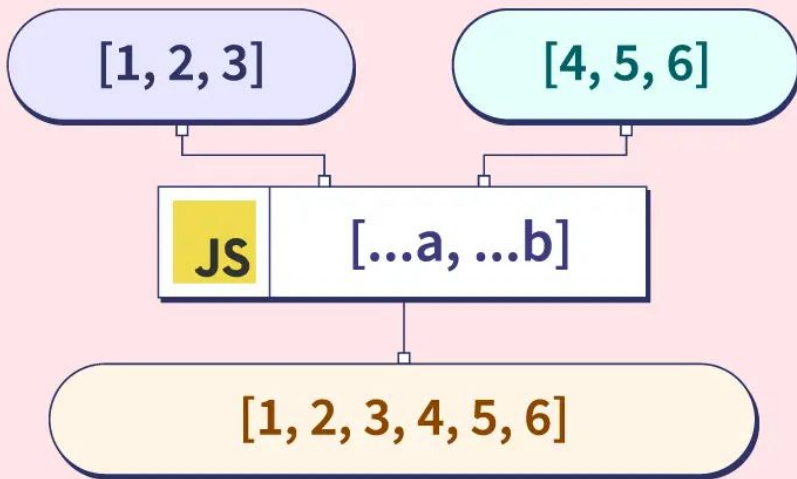


```
const [drink, bread, eggs, ...meta] = breakfast;
```

```
// drink => 'Coffee'  
// bread => 'Croissant'  
// eggs  => 'Sunny side up'  
// meta => ['08:38', 'Sally', 9.15, 7]
```

FIAP

Spread



FIAP

**Como saber o
tamanho de um
array?**

FIAP

```
const newYearsResolutions = ['Keep a  
journal', 'Take a falconry class'];
```

```
console.log(newYearsResolutions.length);  
// Output: 2
```

FIAP

**Como adicionar
itens à um
array?**

FIAP

```
const itemTracker = ['item 0', 'item 1', 'item 2'];
```

```
itemTracker.push('item 3', 'item 4');
```

```
console.log(itemTracker);
```

```
// Output: ['item 0', 'item 1', 'item 2', 'item 3', 'item 4'];
```

FIAP

**Como remover
itens de um
array?**

FIAP

```
const newItemTracker = ['item 0', 'item 1', 'item 2'];
```

```
const removed = newItemTracker.pop();
```

```
console.log(newItemTracker);  
// Output: [ 'item 0', 'item 1' ]  
console.log(removed);  
// Output: item 2
```

FIAP

Array

JS

Array Methods

- toString()
- join()
- concat()
- splice()
- slice()
- indexOf()
- lastIndexOf()
- flat()
- forEach()
- map()
- filter()
- reduce()
- some()
- every()
- find()
- findIndex()
- sort()




FIAP






.map(=>) => 

.filter() => 

.every() => false

.some() => true

.fill(, 1) => 

.findIndex(el => el===) => 2

.find() => 

.reduce((acc, cur)=>acc+cur)=> 

FIAP

forEach()

Irá realizar a mesma ação para cada elemento do array.

FIAP

```
const groceries = ['brown sugar', 'salt',  
                  'cranberries', 'walnuts'];
```

IDENTIFIER

ARRAY

```
groceries.forEach(function(groceryItem){  
  console.log(' - ' + groceryItem);  
});
```

KEY

● Iterator

● Callback function

map()

Ele irá criar um novo array baseado na interação entre os elementos e a iteração.

FIAP

#Javascript
.....

[●, ●].map(● => ● + ▲)

[●▲, ●▲]



@UbahTheBuilder

filter()

Ele irá criar um novo array somente com os elementos que atenderem a condição estabelecida.

#javascript
.....

Shapes = [▲, ●, ▲]

Shapes.filter(shape => shape ==▲)

[▲, ▲]



@UbahTheBuilder

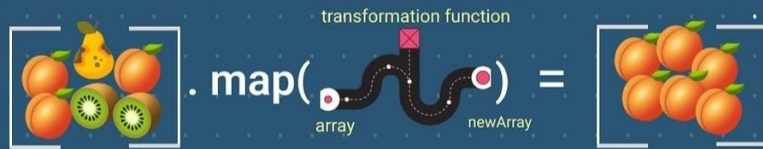
reduce()

Ele soma o parâmetro accumulator com cada elemento do array cumulativamente e retorna um único resultado.

How to Use Array Reduce Method in JavaScript



Map(), filter() & reduce() in JS



Have anything to say? Drop in the comments.