

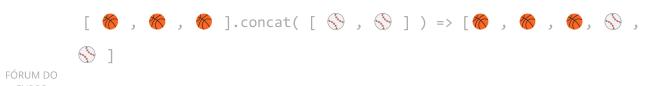
### **Outros métodos especiais**



# **Guia do Mochileiro JavaScript**

ATIVIDADES 5 DE 9

# Método .concat()

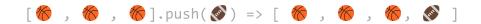


O método concat junta dois (ou mais) arrays em um novo array, sem alterar os já existentes.

# Método.pop()

O método pop remove o último elemento de um array.

### Método.push()



O método push adiciona um novo elemento no final do array, aumentando seu tamanho.

### Método .includes()

```
[ ﴿ , ﴿ , ﴿ ) .includes( ♠ ) ⇒ true
```

O método includes verifica se um elemento está contido em um array e retorna true se o elemento estiver contido ou false caso contrário.



FÓRUM DO

#### Método .fill()

```
[\begin{tabular}{ll} \begin{tabular}{ll} \be
```

O método fill preenche os elementos especificados em um array com um determinado valor.

# Método .indexOf()

```
[    ,         ].indexOf(<math>    }) => 2
```

O método indexof retorna o primeiro índice encontrado de um valor especificado. Se o valor não for encontrado o método retorna -1.

# Método .reverse()

```
[ ∅ , ∅ , ಠ , ಠ ].reverse( ) ⇒ [ ፟ , ಠ , ∅ , ∅ ]
```

O método reverse inverte a ordem dos elementos de um array e substitui o array original.

### Método .slice()

```
[\ \odot\ ,\ \varnothing\ ,\ \odot\ ,\ \odot\ ].slice(1,3) => [\ \varnothing\ ,\ \odot\ ]
```

O método slice retorna elementos de um array, selecionados de determinada posição de início até determinada posição final. O elemento na posição final não é incluso.

# Método .some()



```
[ \textcircled{\S} , \textcircled{\S} , \textcircled{\S} , \textcircled{\S} ].some( ( \textcircled{\S} ) \Rightarrow { return bola === \textcircled{\S} }
```

ATIVIDADES 5 DE 9

teste. Esse teste é feito através de uma função callback. O método executa a função de callback para cada elemento uma vez e retorna true se o teste for true para um dos elementos, e false se o teste for false para todos os elementos. Além disso, o método não executa a função callback para arrays vazios e não altera o array.

O método some verifica se algum elemento do array passa em um

FÓRUM DO CURSO

# Método .join()

O método join puxa elementos de um array e lista no formato de string, o resultado da operação puxou as propriedades do array e as listou de acordo com o que foi determinado.

### Método .shift()

$$[\mathfrak{S}, \mathfrak{S}, \mathfrak{S}, \mathfrak{S}]. \mathsf{shift}() \Rightarrow [\mathfrak{S}, \mathfrak{S}, \mathfrak{S}]$$

O método shift é parecido com o método .pop() mas ao invés de remover o último elemento do array, ele é usado para remover o

primeiro elemento do array.

### Método .unshift()





O método unshift é parecido com o que método .push() realiza, mas ao invés de adicionar no final do array, ele é utilizado para adicionar um elemento no início de um array.

ATIVIDADES 5 DE 9

### Método .splice()

```
FÓRUM DO [ \diamondsuit , \textcircled{o} , \textcircled{o} , \textcircled{o} ].splice( 1, 2, \textcircled{o} ) \Rightarrow [\diamondsuit , \textcircled{o} , \textcircled{o} , \textcircled{o} ] [\diamondsuit , \textcircled{o} , \textcircled{o} ].splice( 2, 3) \Rightarrow [\diamondsuit , \textcircled{o} ]
```

Com o método splice conseguimos escolher um índice inicial e final para substituirmos valores no lugar deles. E também podemos remover itens, no segundo exemplo, foram removidos três elementos a partir da posição dois.

### Método .length()

```
[\diamondsuit, \diamondsuit, \textcircled{o}, \textcircled{o}].length(3) \Rightarrow \diamondsuit, \textcircled{o}, \textcircled{o} [\diamondsuit, \textcircled{o}, \textcircled{o}].
```

O método length define ou retorna o número de elementos em um array.

### Método .sort()

O método sort ordena os elementos do próprio array e retorna o array. A ordenação padrão é de acordo com a pontuação de código unicode.

# Método .toString()



O método toString retorna uma string com todos os valores do array

ATIVIDADES separados por vírgulas.

### Método .findIndex()

FÓRUM DO CURSO

```
[ \odot , \odot , \odot , \odot ].findIndex(emoji => emoji === "\odot") \Rightarrow
```

O método findIndex retorna o índice no array do primeiro elemento que satisfizer a função de teste provida. Caso contrário, retorna -1, indicando que nenhum elemento passou no teste.

### Método .find()

```
[ \diamondsuit , \diamondsuit , \diamondsuit , \diamondsuit ].find( bola \Rightarrow bola.formato === "redonda" ) => \diamondsuit
```

O método find é utilizado para procurar um elemento dentro do array que atenda a condição atribuída a ele, que retornará o primeiro elemento encontrado. Ele percorre todo o array buscando o elemento que atenda a condição e retorna o primeiro que foi encontrado, caso não encontre retorna undefined. Este método não executa caso o array esteja vazio e não altera o array original

### Método .at()

$$[\ \odot\ ,\ \odot\ ,\ \odot\ ,\ \odot\ ].at(-1)\ \Rightarrow\ \odot\ /\ [\ \odot\ ,\ \odot\ ,\ \odot\ ,\ \odot\ ].at(1)\ \Rightarrow\ \odot$$



O at acessar os elementos de um array usando um índice inteiro positivo e negativo, sendo que o índice negativo enumera os itens de trás para frente, portando o sestá posicionado no índice -1.

```
ATIVIDADES
5 DE 9
```

### Método .isArray()

```
var bolas = [  \odot  ,  \odot  ,  \odot  ,  \odot  ] Array.isArray(bolas) \Rightarrow true
```

O método Array.isArray verifica se a variável é um array e retorna true ou false.

# Método .every()

```
[ \textcircled{\$} , \textcircled{\$} , \textcircled{\$} , \textcircled{\$} ].every( (bola ) \Rightarrow { return bola === \textcircled{\$}} } ) => false
```

O método every verifica se cada elemento do array passa em um teste. Esse teste é feito através de uma função callback. O método executa a função de callback para cada elemento uma vez e retorna true se o teste for true para todos os elementos, e false se o teste for false para pelo menos um elemento. Além disso, o método não executa a função callback para arrays vazios e não altera o array.

### Método .filter()

O método filter usa uma função callback de teste e executa ela para cada elemento do array. Ele retorna um novo array com os elementos que passarem no teste.



### Método .map()

```
ATIVIDADES [\%, \%, \%, \%].map(console.log((bola) \Rightarrow {return \varnothing}))

5 DE 9

=> [\varnothing, \varnothing, \varnothing, \varnothing]
```

FÓRUM DO CURSO O método map usa uma função callback de teste e executa ela para cada elemento do array, retornando um novo array modificado. Não executa a função de callback para arrays vazios.

### Método .forEach()

```
[ \mbox{$\lozenge$} , \
```

O método forEach usa uma função callback e executa ela para cada elemento do array. Não executa a função de callback para arrays vazios e não retorna um novo array, diferente do método map.

### Método.copyWithin()

$$[\ \odot\ ,\ \varnothing\ ,\ \odot\ ,\ \odot\ ]$$
.copyWithin(2,0)  $\Rightarrow$   $[\ \odot\ ,\ \varnothing\ ,\ \odot\ ,\ \varnothing\ ]$ 

O método copyWithin copia os elementos do array para outra posição no array, ele não adiciona itens apenas substitui os valores existentes.

## Método .lastIndexOf()

```
[\bigcirc, \bigcirc, \bigcirc, \bigcirc, \bigcirc].lastIndexOf(\bigcirc) \Rightarrow 2
```

O método lastIndexOf retorna o valor do último índice especificado, se o valor não for encontrado ele retorna -1.



#### Método .valueOf()

```
ATIVIDADES [ \textcircled{5}, \textcircled{5}, \textcircled{6}, \textcircled{5}].valueOf() \Rightarrow [ \textcircled{5}, \textcircled{5}, \textcircled{6}, \textcircled{5}]
```

O método valueOf é usado para retornar o array. É um método padrão do objeto Array. Este método retorna todos os itens na mesma matriz. Ele não altera o conteúdo original da matriz e não contém nenhum valor de parâmetro.

# Método .reduce()

O método reduce executa uma função redutora para o elemento array, ele retorna um único valor que é o resultado acumulado da função. Ele não executa a função para elementos de array vazios e não altera a matriz original.

### Método .keys()

```
[ 📎 , 🚳 , 🍪 , 🏀 , ⑤ ].keys() => ["0","1","2","3"]
```

O método keys retorna um novo array composto pelas chaves (posições) do array o qual ele foi aplicado.