**Métodos e Atributos de Arrays**

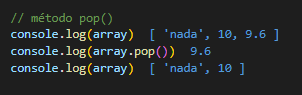
**Atributos**

**Length:** atributo que retorna o número de elementos dentro de um array, incluindo os elementos vazios;

**Ex.:**

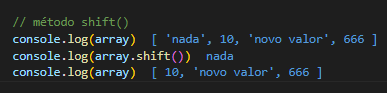
**Métodos para adicionar e remover elementos**

**POP(index):** remove e retorna o último elemento;

**Ex.:**

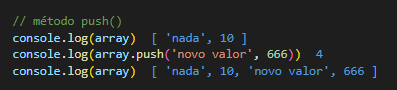
**SHIFT():** remove o primeiro elemento de um array e ajeita os index dos outros elementos.

* Retorna o elemento que foi removido

**Ex.:**

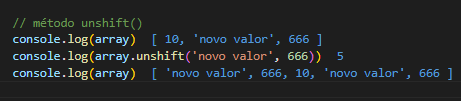
**PUSH(valor1, valor2, ...):** adiciona um ou vários elementos no final do array.

* Retorna o **comprimento do novo array;**
* A ordem que ficarão depende da ordem que foram passados como argumento;

**Ex.:**

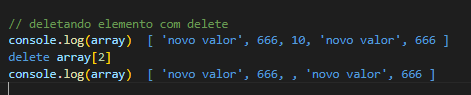
**UNSHIFT(valor1, valor2, valor3, ...):** adiciona um ou vários valores no começo do array

* Retorna o comprimento do novo array

**Ex.:**

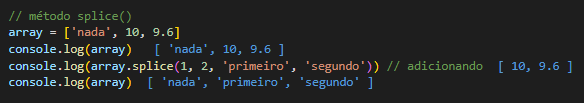
**Delete array[index]:** remove o elemento naquele index.

* O elemento deletado será substituído por um **undefined**
* Essa forma deixa buracos no array

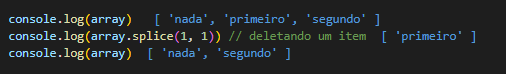
**Ex.:**

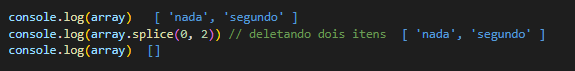
**Splice(index, delete, element1, element2, …):** é usado para adicionar ou remover elementos.

* **index(obrigatório):** o index onde será adicionado/excluído o elemento
* **Delete(obrigatório):** quantos elementos serão deletados a partir do index especificado no primeiro parâmetro
* **Element1, element2(opcional):** quais elementos serão adicionados
* Retorna um array com os elemento(s) deletado(s)
* Os indexes serão reajustados, ao contrário do delete array[index]

**Ex. 1 – adicionando elementos:** caso exista algum elemento no index espcificado no primeiro parâmetro e nos index dos elementos adicionados, será substituído ou excluído

**Ex. 2 – excluindo 1 elemento:** deletando o elemento com index 1(‘primeiro’)



**Ex. 3 – exluindo vários elementos:** deletando 2 elemento, começando do index 0

**Substituindo elementos dentro de um array**

**Fill(item, start, end):** substitui ou insere novos elementos em um array

* **Item:** o item que queremos usar para preencher o array
* **Start(opcional):** index inicial. Valor padrão é 0
* **End(opcional):** index final. Valor padrão é o final do array

**Ex. 1 – substituindo todos os valores**

**Ex. 2 – substituindo apenas um valor:** o index no parâmetro indica em qual index a substituição deve parar

**copyWithin(target, start, end):** faz um cópia dos elementos de um array e os insere em posições específicas dentro do mesmo array.

* **Target:** diz o index que as cópias começarão a ser inseridas
* **Start:** o index do elemento que será copiado. O padrão é o começo do array.
* **End:** o index que a cópia irá terminar. O padrão é o final do array.

**Ex. 1**: estamos dizendo para copiar o elemento no index 1 e inserir a partir do index 0 até o index 2, sem incluí-lo

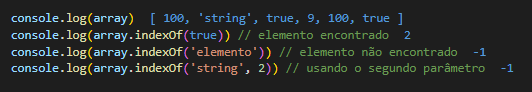
**Ex. 2:** pegando o elemento no index 3(9.7) e copiando para o elemento de index 0(100)



**Verificando elementos dentro de um array**

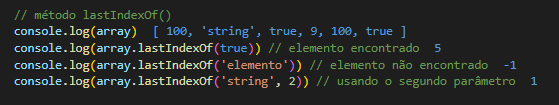
**indexOf(item, start):** retorna o index da primeira ocorrência do elemento procurado

* **Item:** o elemento que queremos procurar
* **Start:** diz por qual index começar a procurar
* Se o elemento não for encontrado, **retorna -1**.

**Ex.:**

**lastIndexOf(item, start):** retorna o index da última ocorrência do elemento procurado

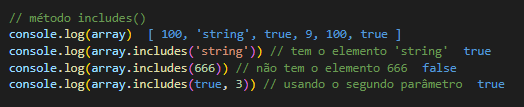
* **Item:** o elemento que queremos procurar
* **Start:** diz por qual index começar
* Se o elemento não for encontrado, **retorna -1**.

**Ex.:**

**Includes(item, start):** verifica se um determinando elemento está no array

* **Item:** o item que queremos procurar
* **Start:** por qual index começarmos a procura
* Se o item estiver no array, retorna **true**; caso contrário, retorna **false**.

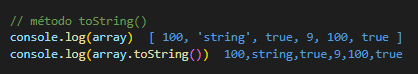
**Ex.:**



**Método para transformar um Array em uma String**

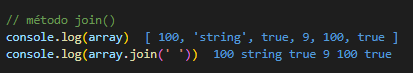
**toString():** transforma todos os elementos de um array em uma string;

* Ficam separados por vírgula
* Não é possível especificar o separador.

**Ex.:**

**Join(separador):** faz o mesmo que o método toString(), mas:

* É possível escolher o separador
* Se não for informado, o separador será a vírgula

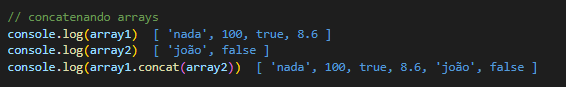
**Ex.:**

**Métodos para juntar Arrays**

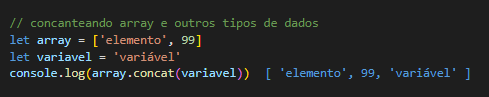
**Concat(valor ou array):** é possível juntar os elementos de vários arrays ou strings, números, objetos etc

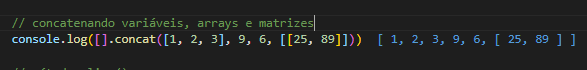
* Retorna um **novo array** com todos os elementos adicionados

**Ex. 1:** juntando dois arrays



**Ex. 2**: juntando arrays, string e números



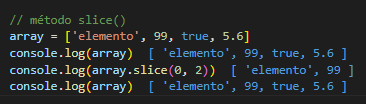
**Ex. 3:** juntando variáveis, arrays e matrizes(array miltidimensionais)

**Método para Copiar Arrays**

**Slice(start, length):** separa um array

* **Start:** o index inicial
* **Length(opcional):** quantos elementos serão copiados. Se não for informado, copia todos os elementos desde do index **start**
* Retorna um array com a cópia dos elementos

**Ex.:** copiando 2 elementos a partir do index 0



**Ex. 2:** copiando todo array

