



# IW-II

# JavaScript – Variáveis Compostas

Prof. Anderson Vanin

```
let num = [5, 8, 4]
```

num

5	8	4
0	1	2

# Acrescentando valores ao array

```
let num = [5,8,4]
```

```
num[3] = 10
```

```
num.push(7)
```

JS exemplo.js

```
1 let num = [5, 8, 4]
2 num[3] = 10
3
4 console.log(num)
```

Info: Start process (08:43:19)  
[ 5, 8, 4, 10 ]  
Info: End process (08:43:19)

JS exemplo.js > ...

```
1 let num = [5, 8, 4]
2 //num[3] = 10
3
4 num.push(6)
5
6 console.log(num)
```

Info: Start process (08:44:56)  
[ 5, 8, 4, 6 ]  
Info: End process (08:44:56)

# Sabendo o tamanho do array

num.length

JS exemplo.js > ...

```
1  let num = [5, 8, 4]
2  //num[3] = 10
3
4  num.push(6)
5
6  console.log(num.length)
```

Info: Start process (08:47:00)

4

Info: End process (08:47:00)

# Ordenando o array

num.sort()

JS exemplo.js > ...

```
1 let num = [5, 8, 4]
2 console.log(num.sort())
```

Info: Start process (08:48:26)

[ 4, 5, 8 ]

Info: End process (08:48:26)

# Percorrendo os elementos de um array

JS exemplo.js > ...

```
1 let valores = [8, 1, 7, 4, 2, 9]
2
3 for(let pos=0; pos < valores.length; pos++){
4   |   console.log(`A posição ${pos} tem o valor ${valores[pos]}`)
5 }
```

```
for(let pos in valores){
  |   console.log(`A posição ${pos} tem o valor ${valores[pos]}`)
}
```

```
Info: Start process (08:52:49)
A posição 0 tem o valor 8
A posição 1 tem o valor 1
A posição 2 tem o valor 7
A posição 3 tem o valor 4
A posição 4 tem o valor 2
A posição 5 tem o valor 9
Info: End process (08:52:49)
```

# Recuperando uma chave de um array

JS exemplo.js > ...

```
1 let valores = [8, 1, 7, 4, 2, 9]
2
3 v /* for(let pos=0;pos < valores.length; pos++){
4 |   console.log(`A posição ${pos} tem o valor ${valores[pos]}`)
5 | } */
6
7 v /* for(let pos in valores){
8 |   console.log(`A posição ${pos} tem o valor ${valores[pos]}`)
9 | } */
10
11 console.log(valores.indexOf(7))
```

JS exemplo.js > ...

```
1 let valores = [8, 1, 7, 4, 2, 9]
2
3 console.log(valores.indexOf(3))
```

Info: Start process (08:59:18)

-1

Info: End process (08:59:18)

Info: Start process (08:57:47)

2

Info: End process (08:57:47)

# Exercícios

1. Crie um programa que solicite ao usuário quantos números ele deseja guardar. Em seguida solicite os números para o usuário e armazene-os em um vetor. Mostre os números ordenados de forma crescente.
2. Crie um programa que solicite um número inteiro para o usuário. O programa deve criar um vetor que comece pelo número informado pelo usuário e vá adicionando as próximas posições deste vetor o incremento de um em um até que o último número armazenado neste vetor seja o dobro do número digitado.