



# IN

**Javascript**  
**Aula 03 - Introdução**

# ***Introdução***

## ***Aula 03***

**Os objetivos da aula de hoje são:**

- 1.Objetos**
- 2.Loops**
- 3.Arrays**

# Objetos

- Objetos são a base de uma metodologia de programação chamada POO (Programação Orientada a Objetos)
- Um objeto possui propriedades (dados) e métodos (procedimentos) embutidos
- **Uso:**

```
var animal = {  
    nome: "Espeto",  
    tipo: "gato",  
    idade: 8  
};
```

Exemplo: – [https://www.w3schools.com/js/tryit.asp?filename=tryjs\\_form\\_elements](https://www.w3schools.com/js/tryit.asp?filename=tryjs_form_elements)

## 1.1 Prática

- **Exercício 1:** Crie uma página que solicita (usando prompt ou form) ao usuário que ele informe sua bebida preferida e exibe a frase “Sua bebida preferida é (nome da bebida informada)”
- **Exercício 2:** Crie uma página que solicite (usando prompt ou form) o nome, o sobrenome e o ano de nascimento do usuário e armazene isso em um objeto chamado “dadosUsr”.
- Dica: – <https://github.com/EduSSXavier/Infinity-JS> ou <https://url.gratis/4aiX5J>

# Arrays

- Arrays são estruturas de dados que permitem o armazenamento de diversas ocorrências de dados em uma única variável.

idades: 

10	23	18	03	60	42
----	----	----	----	----	----

- Lembrete 1:** Arrays são objetos.
- Lembrete 2:** Arrays começam de zero (0).
- Uso:

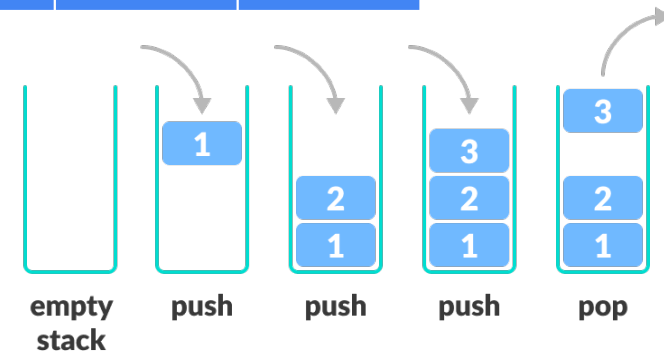
```
const cidades = ["Salvador", "Natal", "Recife", "Curitiba"];
```

# Arrays – Adicionando e removendo elementos

Para **adicionar** um elementos, podemos usar o comando **push**



Para **remover** elementos, podemos usar o comando **pop**



[https://www.w3schools.com/jsref/jsref\\_push.asp](https://www.w3schools.com/jsref/jsref_push.asp)

[https://www.w3schools.com/jsref/jsref\\_pop.asp](https://www.w3schools.com/jsref/jsref_pop.asp)

## 1.3 Prática

- **Exercício 1:** Altere a página criada no exercício 1 da prática anterior (informar a bebida preferida) para que o usuário informe 5 bebidas, armazene isso em um [array](#).
- **Exercício 2:** Crie uma lista de compras com 3 itens em um [array](#) e depois execute as seguintes operações:
  - Incluir mais 2 itens na lista de compras
  - Retirar um item da lista de compras
  - Substituir um item na lista de compras
- Dica: – <https://github.com/EduSSXavier/Infinity-JS> ou <https://url.gratis/4aiX5J>

/\* Outras formas de inserir e remover dados  
Shift, unshift e splice, lenght \*/

```
const frutas = ["Banana", "Laranja", "Maçã"];
```

```
let fruta = frutas.shift(); //"Banana"; Remove o 1º elemento  
// ["Laranja", "Maçã"];
```

```
frutas.unshift("Limão","Abacaxi"); //[ "Limão","Abacaxi", "Laranja", "Maçã"]; Add na 1ª posição
```

```
/* Splice - array.splice(index, howmany, item1, ....., itemX)  
Na posição 2, remove 1 elemento, depois adiciona dois nessa posição */
```

```
frutas.splice(2, 1, "Uva", "Kiwi"); //[ "Limão", "Abacaxi", "Uva", "Kiwi", "Maçã"];
```

```
//Tamanho do array - lenght (get)  
let tamanho = frutas.lenght; // 5
```

```
//Tamanho do array - lenght (set)  
frutas.lenght = 2; // [ "Limão","Abacaxi"];
```



# Prática 04

- Exercício 1: Crie um array contendo, pelo menos, cinco profissões.
  - Inclua “veterinário” no início do array
  - Retire a última profissão do array
  - Inclua “piloto” no final do array
  - Retire o primeira profissão do array
  - Inclua as profissões “designer” e “carpinteiro” a partir da terceira posição do array (lembre-se que começa a contar de zero)
  - Retire a segunda profissão do array
- Dica: – <https://github.com/EduSSXavier/Infinity-JS> ou <https://url.gratis/4aiX5J>

# Loops

- Loops – Algumas vezes a lógica de uma solução exige que se execute um bloco de códigos repetidas vezes. Para isso existem comandos de repetição (loops).
- **for** – executa um bloco de código diversas vezes

```
for (let i = 0; i < frutas.length; i++) {  
  text += frutas[i] + "<br>";  
}
```

[https://www.w3schools.com/js/js\\_loop\\_for.asp](https://www.w3schools.com/js/js_loop_for.asp)

# Loops

- **for/in** – percorre as propriedades de um objeto

```
const pessoa = {fname:"John", lname:"Doe", age:25};
```

```
let text = "";  
for (let prop in pessoa) {  
    text += pessoa[prop];  
}  
//JohnDoe25
```

[https://www.w3schools.com/js/js\\_loop\\_forin.asp](https://www.w3schools.com/js/js_loop_forin.asp)

# Loops

- **for/of** – percorre objetos usando iteração automática e executa um bloco de código para cada ocorrência
- `const carros = ["BMW", "Volvo", "Mini"];`

```
let text = "";  
for (let carro of carros) {  
    text += carro;  
}
```

[https://www.w3schools.com/js/js\\_loop\\_forof.asp](https://www.w3schools.com/js/js_loop_forof.asp)

# Loops

- **while** e **do/while** – executa um bloco de código controlando o número de execuções com um teste lógico.

```
while (i < 10) {  
    text += "The number is " + i;  
    i++;  
}
```

```
do {  
    text += "The number is " + i;  
    i++;  
}  
while (i < 10);
```

[https://www.w3schools.com/js/js\\_loop\\_while.asp](https://www.w3schools.com/js/js_loop_while.asp)

# Loops

- Break e continue – forçam a interrupção de um loop

```
let i = 0;
while (i < 10) {
  if (i == 5)
    break;
  text += "The number is " + i;
  i++;
}
//0 1 2 3 4 5
```

```
let i = 0;
while (i < 10) {
  if (i == 5)
    continue;
  text += "The number is " + i;
  i++;
}
//0 1 2 3 4 6 7 8 9 10
```

[https://www.w3schools.com/js/js\\_break.asp](https://www.w3schools.com/js/js_break.asp)

# Prática 05

- **Exercício 1:** Implemente uma função que receba um array como parâmetro e imprima (converta) os dados do array para o formato list (<ul>) do HTML.
- **Exercício 2:** Crie um loop para entrada de dados (via prompt) de diversos nomes de ferramentas, que devem ser armazenadas em um array. O loop só deve encerrar se o usuário informar uma ferramenta chama “fim”.
- **Exercício 3:** Crie um array contendo nomes de cores com, no mínimo, 5 ocorrências. Realize uma pesquisa, utilizando algum mecanismo de loop, para verificar se um valor informado pelo usuário (via prompt) existe no array e informe ao usuário o resultado da pesquisa (encontrou / não encontrou).

# Javascript e HTML

- Crie uma função que receba um array e o transforme em uma lista HTML
- Permita um parâmetro para escolher entre lista ordenada e lista não ordenada
- Crie um objeto que transforme um array multidimensional (de nº de colunas fixo) em uma tabela HTML



# “ Referências”

1. <https://www.w3schools.com>