

# Fundamentos JavaScript

Kauê Luis Oliveira da Silva

# Pré-requisito:

Estar familiarizado com o básico da lógica de programação Web.

# Objetivo:

Reforçar os principais fundamentos do JavaScript

# Fontes:

- [MDN web Docs](#)
- [FreeCodeCamp youtube channel](#)
- [Jordan Hayashi JSBootcamp](#)
- [Cs50 Web programming](#)

(caso o hyperlink não funcionar ,urls estão no último slide)

# Como utilizar essa aula:

- Cada tópico da aula tem um código fonte exemplo correspondente.
- Leia o código e tente modificar com seus próprios dados.

# O que vamos cobrir (parte 1) :

- Variáveis
- Entrada e saída de dados
- Tipo de dados: Numeros e Strings
- Funções
- Hoisting
- Expressões e operadores
- `== Vs ===`
- Operadores Ternários
- Switch
- `eval()`

# Variaveis:

Na ES6 temos mais 2  
jeitos de declarar  
variáveis:  
let e const.  
Veremos na aula  
focada em ES6

```
1  /*Variaveis*/  
2  var meuNome;  
3  meuNome = 'Kaue';  
4  
5  console.log(meuNome);  
6  
7  var meuSobrenome = 'Silva'  
8  
9  console.log(meuSobrenome);  
10  
11 meuSobrenome = 'Oliveira'  
12  
13 console.log(meuSobrenome);
```

# Variáveis:

- **Variáveis não devem:**
  - começar com \_ ou números
- **Boa Prática:**
  - lower camelCase
  - Dê nomes que sejam intuitivos

# Entrada e saída e leitura de dados:

- `prompt()`
- `alert()`
- `document.write()`
- `console.log()`
- `parseInt()`
- `parseFloat()`

# Tipo de dados:

## Tipo primitivos (sem métodos):

- Boolean
- null
- Number
- String
- Undefined
- (symbol) \*
- Objects (métodos associados)



# Números:

- Todas operações básicas: + , - , / , \* e %.
- Incrementar e decrementar variáveis.

# String:

- Sequencia de zero ou mais caracteres entre aspas
- Quando a backslash é usada para sinalizar algum 'comando'

| Codigo | Saída      |
|--------|------------|
| \'     | '          |
| \"     | "          |
| \n     | nova linha |
| \\     | \          |

# String:

- É possível concatenar strings com o operador mais (+)

```
var myName = 'Kaue';  
var fullName = 'Kaue ' + 'Silva';  
var sentence = 'Meu nome é ' + fullName;  
fullName += ' é meu nome.';
```

- String são imutáveis, ou seja não é possível mudar somente caractere da string.

# String:

- Para acessar um elemento específico dentro de uma string usa-se [bracket notation]

```
1  var meuNomeCompleto = 'Kaue Luis Oliveira da Silva'
2
3  console.log(meuNomeCompleto[0])
4  console.log(meuNomeCompleto[meuNomeCompleto.length - 2])
5
```

# String:

- **Alguns métodos:**
  - **charAt**
  - **charCodeAt**
  - **concat**
  - **endsWith**
  - **fromCharCode**
  - **includes**
  - **indexOf**
  - **lastIndexOf**
  - **Match**
  - **toUpperCase**
  - **repeat**
  - **replace**
  - **search**
  - **slice**
  - **split**
  - **startsWith**
  - **substr**
  - **toLowerCase**
  - **trim**

# Funções:

- São um dos blocos fundamentais em JavaScript.
- Funções são sequencias de comandos que realizam uma tarefa.
- Para usar uma função você deve declara-la dentro do escopo que você deseja chama-la.

# Expressões e operadores:

- Javascript vários tipos de operadores e , nessa aula vamos focar nos operadores de comparação, lógicos e expressões if,else.
- if / else if / else
- Operadores de comparação : >= , < , > ,<= , == , !=, === , !==

**== Vs === :**

- **== :** Retorna verdadeiro caso os elementos sejam iguais. (converte valores para o mesmo tipo 'type coercion')

```
○ var x = 42;  
○ var explicit = String(x);    // explicit === "42"  
○ var implicit = x + "";      // implicit === "42"
```

- **===:** retorna verdadeiro somente se os elementos foram iguais e do mesmo tipo.



[illegible]

# Hoisting:

- Declarações são processadas antes que qualquer código seja executado, por isso declarar uma variável em qualquer lugar, é o mesmo que declara-la no topo.
- Isso significa que uma variável pode ser usada antes de ser declarada.

# Hoisting:

- Por essa razão, recomenda-se sempre declarar variáveis na parte superior do seu escopo de aplicação (o topo do código global e a parte superior do código da função).
- `null !== undefined`

# Hoisting:

```
1  function do_something() {  
2      console.log(bar); // undefined  
3      var bar = 111;  
4      console.log(bar); // 111  
5  }  
6  
7  // is implicitly understood as:  
8  function do_something() {  
9      var bar;  
10     console.log(bar); // undefined  
11     bar = 111;  
12     console.log(bar); // 111  
13 }
```

# Operadores ternários:

- O operador condicional (ternário) é o único operador JavaScript que possui três operandos. Este operador é frequentemente usado como um atalho para a instrução if.

```
if (idade >= 18) {  
    console.log("Você é um adulto!");  
} else {  
    console.log("Você é uma criança!");  
};
```

```
console.log((age >= 18) ? "Você é um adulto!" : "Você é uma criança.");
```

# Função eval(string):

- O método eval() código JavaScript representado como uma string.

```
1 console.log(eval('2 + 2'));
2 // expected output: 4
3
4 console.log(eval(new String('2 + 2')));
5 // expected output: 2 + 2
6
7 console.log(eval('2 + 2') === eval('4'));
8 // expected output: true
9
10 console.log(eval('2 + 2') === eval(new String('2 + 2')));
11 // expected output: false
12
```

# Switch:

- A condicional switch avalia uma expressão, combinando o valor da expressão para um cláusula case, e executa as instruções associadas ao case.

```
3  var day;  
4  switch (new Date().getDay()) {  
5      case 0:  
6          day = "Sunday";  
7          break;  
8      case 1:  
9          day = "Monday";  
10         break;  
11     case 2:  
12         day = "Tuesday";  
13         break;  
14     case 3:  
15         day = "Wednesday";  
16         break;  
17     case 4:  
18         day = "Thursday";  
19         break;
```

```
20         case 5:  
21             day = "Friday";  
22             break;  
23         case 6:  
24             day = "Saturday";  
25     }  
26     console.log(day)
```

# Exercícios:

- bemVindo
- Jordan Hayashi BootCamp day0
- Se souberem todos os conceitos , ToDo APP



# Fontes:

- <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide>
- [https://www.youtube.com/playlist?list=PLWKjhJtqVAbk2qRZtWSzCIN38JC\\_NdhW5](https://www.youtube.com/playlist?list=PLWKjhJtqVAbk2qRZtWSzCIN38JC_NdhW5)
- <https://prod.jordanhayashi.com/bootcamp>
- <https://cs50.github.io/web/>