



escola  
britânica de  
artes criativas  
& tecnologia

# Front-End



# Fundamentos do JavaScript



## GUIA DA AULA 3



# Explore os tipos de dados



Acompanhe aqui os temas que serão tratados na videoaula

- Sobre o tipo de dados
- Literais
- Lógicos
- Numéricos
- Objetos
- Sobre o JavaScript



## Sobre o tipo de dados

Os tipos de dados que temos disponíveis no JavaScript são:

- 1) **String**: tipo literal, usado para textos e caracteres;
- 2) **Number**: tipo numérico, usado tanto para números inteiros e reais;
- 3) **Boolean**: tipo lógico, que indica se determinada afirmação é verdadeira ou falsa;
- 4) **Object**: tipo mais básico do JavaScript, todos os tipos nasceram a partir do Object.



## Literais

Uma **string** é um tipo de dado literal usado quando trabalhamos com caracteres, até mesmo um espaço em branco é uma string.

Podem ser construídas ao envolver o valor atribuído a uma variável em aspas simples ou duplas.

**var nome = "gian"**

**Nome = 'gian'**



Podemos usar alguns recursos da string para contar o seu tamanho, selecionar apenas um pedaço ou procurar por um pedaço de texto ou uma letra.

Consideramos:

**var nome = "JavaScript"**

Contando o tamanho de uma string

**nome.length // 10**

Tornando todo o texto em caixa alta

**nome.toLocaleUpperCase() // JAVASCRIPT**



## Procurando por uma letra ou palavra:

**nome.search('v') // 2**

O 2 indica a posição da letra V em JavaScript, naturalmente seria 3, mas a posição inicial é 0 e não 1.

**nome.slice(0, 2) // Já**

Retorna um pedaço da string, 0 sendo a posição inicial e 2 o número de caracteres que queremos selecionar.



## Unindo strings:

Podemos unir Strings + Strings ou Strings + Números, com o operador +

**"Olá " + nome // Olá JavaScript**

**nome + " é " + 10 + "!" // JavaScript é 10!**

Chamamos essa união de **concatenação**.





## Lógicos

Os **booleanos** são valores do tipo lógico compostos por *true* ou *false*.  
 Através desses valores podemos mudar o fluxo de execução do *software*.

**var foiConvidado = true**

**var podeEntrar = foi Convidado**

Nesse exemplo temos uma situação na qual se pode entrar num local apenas se for convidado, a primeira variável diz se a pessoa foi convidada, e como o valor da segunda é condicionada a ela, podemos simplesmente repassar seu valor, logo:

**foiConvidado = true ENTÃO podeEntrar = true**

**foiConvidado = false ENTÃO podeEntrar = false**



## Numéricos

Para criar uma variável do tipo numérico em JavaScript, basta atribuir o número ao valor. Apesar de tanto o número inteiro como o fracionado serem considerados ambos do tipo **number**, existe uma separação interna entre o tipo *integer* (**inteiro**) e *float* (**fracionado**):

**var idade = 32 // Numbermas Integer**

**var altura = 1.72 // Numbermas Float**



Com os números podemos fazer cálculos aritméticos:

**var ano = 2022**

**var idade = 32**

**idade \* 2 = 64**

**idade / 2 = 16**

**var anoDeNascimento = ano - idade**

**Observação:** Podemos usar operações para atribuir o valor às variáveis.



Assim como nas strings, os elementos do tipo “number” possuem alguns recursos especiais, como:

**Verificar se um número é integer:**

`Number.isInteger(10) // true`

`Number.isInteger(10.5) // false`

**Converter uma string para float:**

`parseFloat(“10.5632”) // 10.5632`

`parseFloat(“10,5632”) // 10 – não devemos usar vírgula`



**Converter um float para integer:**

**`parseInt(10.43) // 10` – o restante do número é descartado**



## Objetos

O tipo “Object” é o mais primitivo do JavaScript, através dele outros tipos nasceram. Também são chamados de **arrays**.

Um ponto importante é que o JavaScript trata os vetores como objetos.

Vetores são conjuntos de dados. O conjunto é representado pelos símbolos de colchetes ([.....]), e seus itens são separados por vírgula.



Exemplo de um array de strings

```
var nomes = ["gian", "carlos", "maria"]
```

Por possuir uma fraca tipagem o JavaScript não nos obriga a adicionar itens do mesmo tipo em um conjunto, por exemplo:

```
nomes = ["gian", "carlos", "maria", "joão", 15, 32]
```

Para acessar um item do array usamos seu índice, no qual o primeiro item possui o índice 0.

```
nomes[0] // gian
```

```
nomes[4] // 15
```



## Sobre o JavaScript

O JavaScript não é uma linguagem fortemente “tipada”, isso quer dizer que na hora de declararmos uma variável, por exemplo, não precisamos informar que será um texto ou número. Mas isso pode gerar alguns problemas, como uma variável que originalmente era numérica e em determinado momento do programa pode se tornar um número. Por isso, é importante ter cuidado ao lidar com tipos de dados em JavaScript e garantir que você esteja realizando as conversões de tipo adequadas, quando necessário, para evitar problemas e garantir o correto funcionamento do seu código.

