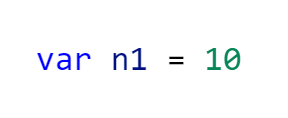
# Variáveis em JavaScript

Para declarar uma variável em JavaScript pode-se as palavras chaves **‘var’**, **‘let’** e **‘const’**. A sintaxe é a seguinte:

**var <identificador> = <valor da variável>;**

Por exemplo:



## Identificadores

O identificador de uma variável é o nome de referência da variável.

Os identificadores possuem algumas regras:

* Podem começar com uma letra, **‘$’**(cifrão) ou **‘\_’**(underline).
* Não podem conter espaços entre os caracteres.
* Não podem começar com números.
* Não podem ser palavras reservadas, como comandos em JS.
* Pode-se usar símbolos e acentos.
* Maiúsculas e minúsculas fazem diferença.

## *camelCase*

Assim é chamado a forma de escrever o nome das variáveis, onde o primeiro nome é escrito em minúsculo e os próximos em maiúsculo.

Por exemplo:

* javaScript
* firstName
* lastName

É comum também o uso de underlines (‘\_’) para a separação das palavras.

## Palavra-chave ‘let’

Declarar uma variável com a palavra-chave **‘let’** tornará possível um erro caso uma mesma variável seja declarada uma segunda vez. Ou seja, caso seja declarada uma variável existente, será executado um erro.

## Palavra-chave ‘const’

Variáveis que não devem ser reatribuídas devem ser declaradas com a palavra-chave **‘const’**, que as torna constante, não permitindo a alteração do valor.

Se caso houver a tentativa de reatribuir o valor da variável, ocorrerá um erro.

**NOTA**: É comum que o nome de variáveis imutáveis sejam MAIÚSCULAS.

# Tipos Primitivos

Em JavaScript, existem três tipos primitivos primordiais. São eles:

* Boolean (**true**, **false**).
* Number (**3**, **-2**, **0.2**, 5.9).
* String (**‘Olá, mundo’**, **“Olá, Mundo!”**, **`Olá, Mundo!`**)

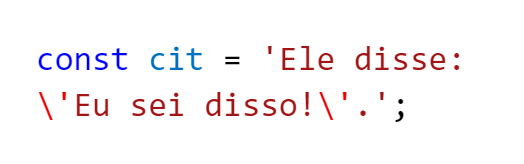
Com o comando **‘typeof’** é possível descobrir o tipo primitivo de uma variável, por exemplo. Irá retornar o tipo primitivo como valor.

## String

Em JavaScript, tudo entre aspas simples ou duplas são consideradas strings.

### Citações dentro da String

Para fazer uma citação dentro de uma string sem causar nenhum erro, use a barra invertida (**‘\’**) antes da citação da abertura de aspas da citação e antes do fechamento de aspas da mesma.



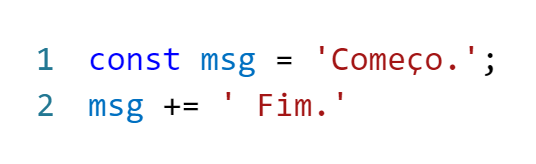
### Concatenação

O operador **‘+’** é usado também para concatenar strings.



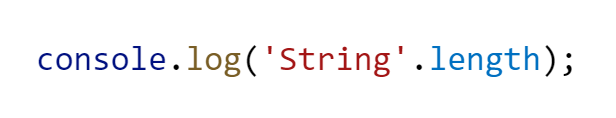
Perceba que na segunda string há um espaço. Esse espaço deve ser adicionado manualmente, ou o resultado será duas strings coladas uma à outra.

É possível usar também o sinal de mais junto ao de igual (**‘+=’**) para concatenar uma variável do tipo string com outra.



### Comprimento de uma String

Usar a propriedade **‘.length’** ao final da string retornará o comprimento do valor dessa string.

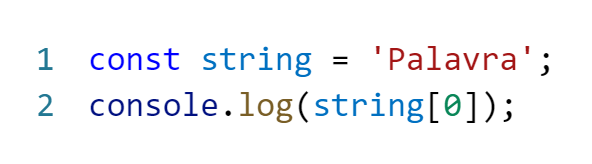


No caso acima, o resultado será 6.

### Fatiamento de String

Adicionando colchetes com algum número ao final de uma variável que tem uma string como tipo de valor, é possível especificar um caractere específico de uma string.

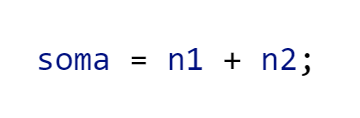
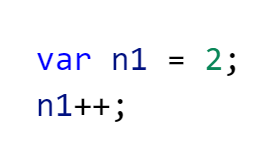
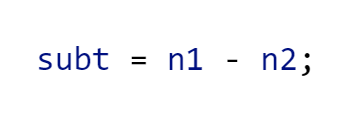
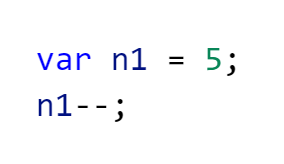
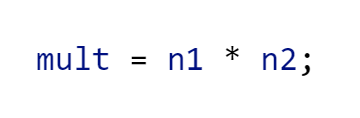
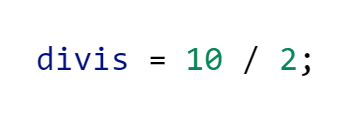
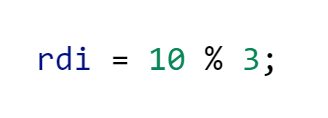
O primeiro caractere de uma string é zero, ou seja, o primeiro caractere de uma string seria: **‘string[0]’**.



## Number

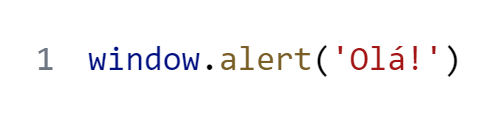
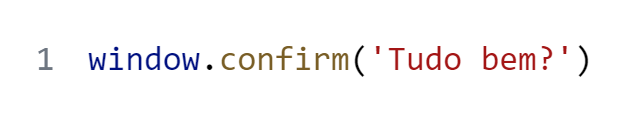
Os valores numéricos podem ser reais ou inteiros e com eles é possível fazer operações.

Os operadores são:

* **‘+’:** Adição.  
  
* **‘++’:** Soma mais um à variável.  
    
  O mesmo é equivalente a **‘n1 = n1 + 1’**. O resultado será 3.
* **‘-’:** Subtração.  
  
* **‘--’:** Subtrai um da variável.  
    
  O resultado será 4.
* **‘\*’:** Multiplicação.  
  
* **‘/’:** Divisão.  
  
* **‘%’:** Resto da divisão.  
  

# ‘window’

É um objeto em JavaScript que possui diversas funções. Algumas delas são:

* **‘alert()’:** Exibe uma caixa de texto.  
  
* **‘confirm()’:** Exibe uma caixa de texto com as opções “ok” e “cancelar”.  
  
* **‘prompt()’:** Cria uma caixa de texto que possui uma caixa de entrada. Para usar o valor escrito pelo usuário deve ser usado em conjunto a uma variável.  
  