

# IW-II

# INTERFACE WEB II

---

Prof. Márcio

# **Variáveis e Tipos de Dados**

# Variáveis e Tipos de Dados

Na linguagem JavaScript, toda a vez que desejarmos declarar novas variáveis devemos utilizar a seguinte sintaxe:

**var** *variavel* = *valor*;

# Variáveis e Tipos de Dados

Sabendo agora como declarar uma variável vamos declarar algumas variáveis em nosso arquivo JavaScript e apresentá-las no Console do navegador.

A linguagem JavaScript faz diferenciação entre caracteres maiúsculos e minúsculos, este conceito de análise tipográfica é chamado em inglês de **Case Sensitive**.

# Variáveis e Tipos de Dados

```
aula01 > <> exemplo_variavel.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Exemplo Variaveis</title>
7    <script src="variaveis.js"></script>
8  </head>
9  <body>
10   <h1>Exemplo Variáveis</h1>
11 </body>
12 </html>
```

```
aula01 > JS variaveis.js > ...
1  var nomeDaVariavel = "valor da variável";
2  var outraVariavel = 'valor da outra variável';
3  var variavelIdade = 22;
4  var variavelTenhoCabeloBranco = false;
5  var variavelIndefinida ;
6
7  console.log(nomeDaVariavel);
8  console.log(outraVariavel);
9  console.log(variavelIdade);
10 console.log(variavelTenhoCabeloBranco);
11 console.log(variavelIndefinida);
12 console.log(variavelNula);
```

▼ aula01

- <> exemplo\_variavel.html
- <> index.html
- JS script.js
- JS variaveis.js

valor da variável	<a href="#">variaveis.js:7</a>
valor da outra variável	<a href="#">variaveis.js:8</a>
22	<a href="#">variaveis.js:9</a>
false	<a href="#">variaveis.js:10</a>
undefined	<a href="#">variaveis.js:11</a>
✖ Uncaught ReferenceError: variavelNula is not defined	
at <a href="#">variaveis.js:12:13</a>	
>	

# Tipos de dados

Ao declararmos as variáveis em nosso exemplo é possível notar que os valores atribuídos as variáveis são diferentes, alguns com aspas, alguns sem aspas, números, etc. Isto ocorre, pois os diferentes tipos de dados necessitam que a sintaxe de atribuição seja diferente. Para melhor entender o assunto vamos dar uma olhada nos tipos de dados existentes na linguagem JavaScript.

# Tipos de dados - Primitivos

## **Numérico (Number)**

O tipo de dado numérico na linguagem de programação JavaScript será sempre decimal, mesmo que você o declare como 2, a linguagem de programação sempre o entenderá como 2.0, mesmo que você não veja isto.

Em algumas linguagens de programação, denominadas linguagens tipadas, há como definir se uma variável do tipo numérico será inteira ou decimal, mas isto não é necessário no JavaScript.

Para atribuir este tipo de dado a uma variável não é necessário utilizar aspas.

# Tipos de dados - Primitivos

## **Booleano (Boolean)**

Uma variável do tipo booleano pode assumir apenas dois valores: `true` e `false`. Os valores deste tipo são em geral usados pela linguagem como resultado de comparações e podem ser usados pelo usuário para valores de teste ou para atributos que possuam apenas dois estados. O JavaScript converte automaticamente `true` para `1` e `false` para `0` quando isso for necessário.



# Tipos de dados - Primitivos

## Booleano (Boolean) - EXEMPLO

```
var a = 14;
var b = 42;
var tr = (a == 14);
var fl = (a == b);
// Neste caso tr irá conter o valor true e fl o valor false.
var int1 = tr + 1;
var int2 = fl + 1;
// A variável int1 irá conter o valor 2 (true + 1), pois true é
// automaticamente convertido para 1 e int2 irá conter o valor 1
// (false + 1), pois false é convertido para 0.
```

# Tipos de dados - Primitivos

## **Texto (String)**

Tipo de dado texto são basicamente sequências de caracteres como por exemplo o nome de alguém ou uma mensagem de bom dia.

Para atribuir este tipo de dado a uma variável é necessário colocar o texto entre aspas ou aspas duplas, conforme o último exemplo.

# Tipos de dados - Primitivos

## Texto (String) - EXEMPLO

```
var str = "Eu sou uma string!";  
var str2 = 'Eu também sou uma string';  
// Declaração de strings primitivas  
var str3 = new String("Outra string");  
  
// Acima um objeto string declarado de forma explícita  
// não há diferença nenhuma entre esses dois tipos no que se  
refere a seu uso.
```

# Tipos de dados - Primitivos

## Indefenido (Undefined)

Tipo de dado indefinido é o valor que uma variável recebe automaticamente, pela linguagem JavaScript, ao ser declarada sem que seja definido um valor para a mesma, ou seja uma variável não populada é preenchida com o tipo de dado indefinido, entenderemos melhor isto no decorrer do curso.

Em nosso exemplo foi declarado uma variável chamada *variavelIndefinida* que ao apresentarmos ela no console o resultado será ***undefined***.

# Tipos de dados - Primitivos

## Nulo (Null)

O tipo de dado nulo é similar ao indefinido, porém este realmente não existe.

Em nosso exemplo estamos tentando apresentar no console uma variável chamada *variavelNula* , porém esta variável não foi declarada previamente, então ela não existe e consequentemente é preenchida com o valor ***null*** .

# Tipagem Dinâmica

Diferente de algumas linguagens de programação, onde ao definir uma variável é necessário especificar o tipo de dado desta, a linguagem JavaScript possui algo chamado tipagem dinâmica, significa que ao declarar uma variável não é necessário definir ou restringir o tipo de dado desta, pois ela irá se adaptar a cada novo tipo de dado atribuído a ela.

# Operadores Aritméticos

<b>Operador</b>	<b>Operação</b>	<b>Exemplo</b>
<b>+</b>	Adição	$x+y$
<b>-</b>	Subtração	$x-y$
<b>*</b>	Multiplicação	$x*y$
<b>/</b>	Divisão	$x/y$
<b>%</b>	Módulo (resto da divisão inteira)	$x\%y$
<b>-</b>	Inversão de sinal	$-x$
<b>++</b>	Incremento	$x++$ ou $++x$
<b>--</b>	Decremento	$x--$ ou $--x$

# Operadores Comparação

<b>Operador</b>	<b>Função</b>	<b>Exemplo</b>
<b>==</b>	Igual a	(x == y)
<b>!=</b>	Diferente de	(x != y)
<b>===</b>	Idêntico a (igual e do mesmo tipo)	(x === y)
<b>!==</b>	Não Idêntico a	(x !== y)
<b>&gt;</b>	Maior que	(x > y)
<b>&gt;=</b>	Maior ou igual a	(x >= y)
<b>&lt;</b>	Menor que	(x < y)
<b>&lt;=</b>	Menor ou igual a	(x <= y)



# Operadores Lógicos

<b>Operador</b>	<b>Função</b>	<b>Exemplo</b>
<b>&amp;&amp;</b>	E Lógico	(x <b>&amp;&amp;</b> y)
<b>  </b>	OU Lógico	(x <b>  </b> y)
<b>!</b>	Negação Lógica	<b>!</b> x

# Atribuição

Operador	Exemplo	Equivalente
=	x = 2	Não possui
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y
&=	x &= y	x = x & y
=	x  = y	x = x   y
^=	x ^= y	x = x ^ y
>>=	x >>= y	x = x >>= y
<<=	x <<= y	x = x <<= y
>>>=	x >>>= y	x = x >>>= y

# Exercícios

1. Dados os seguintes algoritmos, implemente-os utilizando javascript.

a)

$A \leftarrow 10$
$B \leftarrow 20$
Escrever B
$B \leftarrow 5$
Escrever A, B

b)

$A \leftarrow 30$
$B \leftarrow 20$
$C \leftarrow A + B$
Escrever C
$B \leftarrow 10$
Escrever B, C
$C \leftarrow A + B$
Escrever A, B, C

c)

$A \leftarrow 10$
$B \leftarrow 20$
$C \leftarrow A$
$B \leftarrow C$
$A \leftarrow B$
Escrever A, B, C

d)

$A \leftarrow 10$
$B \leftarrow A + 1$
$A \leftarrow B + 1$
$B \leftarrow A + 1$
Escrever A
$A \leftarrow B + 1$
Escrever A, B

# Exercícios

2. Crie um programa em javascript e defina três notas de um aluno, calcule e escreva a média final deste aluno. Considerar que a média é ponderada e que o peso das notas é 2, 3 e 5. Fórmula para o cálculo da média final é:

$$\text{mediafinal} = \frac{n1 * 2 + n2 * 3 + n3 * 5}{10}$$