

# Java Script – Aula 02

Prof André Rabelo

# Operadores Aritméticos

- + Adição
- - Subtração
- \* Multiplicação
- / Divisão
- % Módulo (resto da divisão)
- ++ Incremento
- -- Decremento

# Operadores de atribuição

=	$x = y$	$x = y$
+=	$x += y$	$x = x + y$
-=	$x -= y$	$x = x - y$
*=	$x *= y$	$x = x * y$
/=	$x /= y$	$x = x / y$
%=	$x \% = y$	$x = x \% y$

# Operadores de Strings

- O operador + pode ser usado para unir (concatenar) strings
- Exemplo:

```
var txt1 = "John";  
var txt2 = "Doe";  
var txt3 = txt1 + " " + txt2;
```

# Operadores de Comparação

Operador	Descrição
==	Igual á
===	Valor igual e tipo igual
!=	diferente
!==	Tipo diferente e valor diferente
>	Maior que
<	Menor que
>=	Maior ou igual
<=	Menor ou igual
?	Operador ternário

# Operadores Lógicos

Operador	Descrição
&&	E lógico
	Ou lógico
!	Não lógico

# Operadores de tipo

Operador	Descrição
typeof	Retorna o tipo da variável
instanceof	Retorna verdadeiro se o objeto for instanciado

# JavaScript - Tipos de Dados

- Entre os tipos de dados mais conhecidos no JavaScript podemos destacar:

```
var length = 16;           // Númerico
var lastName = "Johnson";  // String
var x = {firstName:"John", lastName:"Doe"};  // Objeto
```



# JavaScript - Tipos de Dados

- Os tipos de dados no JavaScript são dinâmicos

```
var x;           // Agora x é undefined
x = 5;           // Agora x é um número
x = "John";      // Agora x é uma String
```

# JavaScript - Tipos de Dados

- Em Strings podemos usar aspas simples ou duplas para representar uma string

```
var carName = "Volvo XC60";    // Aspas duplas  
var carName = 'Volvo XC60';    // Aspas simples
```

```
var resposta = "It's alright";    // Aspas simples dentro de aspas duplas  
var resposta = "He is called 'Johnny'";    // Aspas simples dentro de aspas duplas  
var resposta = 'He is called "Johnny"';    // Aspas duplas dentro de aspas simples
```

# JavaScript - Tipos de Dados

- JavaScript tem apenas um formato de números
- Estes podem ser escritos com ou sem casas decimais:

```
var x1 = 34.00;  
var x2 = 34;
```

```
var y = 123e5;      // 12300000  
var z = 123e-5;     // 0.00123
```

# JavaScript - Tipos de Dados

- JavaScripts possui valores booleanos (verdadeiro ou falso):

```
var x = 5;  
var y = 5;  
var z = 6;  
(x == y)      // Retorna verdadeiro  
(x == z)      // Retorna falso
```

# JavaScript - Tipos de Dados

- Arranjos são escritos com colchetes e os itens separados por vírgula

```
var carros = ["Fiat", "Volvo", "BMW"];
```

# JavaScript - Tipos de Dados

- Objetos são variáveis mais complexas.
- Elas conseguem guardar mais informações
- Veja o exemplo:

```
var pessoa = {nome:"John", sobrenome:"Doe", age:50, corOlhos:"blue"};
```

# JavaScript - Tipos de Dados

- Operador typeof recupera o tipo de dado da variável

```
typeof ""           // Retorna "string"
typeof "John"       // Retorna "string"
typeof "John Doe"   // Retorna "string"

typeof 0            // Retorna "number"
typeof 314          // Retorna "number"
typeof 3.14         // Retorna "number"
typeof (3)          // Retorna "number"
typeof (3 + 4)      // Retorna "number"
```

# JavaScript - Funções

- Uma função é um bloco de código projetado para executar uma tarefa específica.
- Uma função é executada quando “algo” a chamada

```
function myFunction(p1, p2) {  
    return p1 * p2; // Retorna a multiplicação de p1 e p2  
}
```



# JavaScript - Funções

- Para declararmos funções utilizamos a seguinte sintaxe:

```
function nome(parametro1, parametro2, parametro3...) {  
    Código da função  
}
```

- Os parâmetros da função ajudam no seu funcionamento e são inseridos entre parênteses
- Os argumentos da função são os valores passados via parâmetros

# JavaScript - Funções

- Exemplo:

```
var x = myFunction(4, 3); // Chamada de função
```

```
function myFunction(a, b) {  
    return a * b; // a função retorna o produto de a e b  
}
```

- Por que utilizar funções?
- Por que economizamos código e podemos repetir as funções diversas vezes em situações diferentes.

# Atividades de Exemplo

- Desenvolva um script em JavaScript que realize a soma de dois números e mostre seu resultado no interior de uma tag h1
- Desenvolva um script em JavaScript que realize a divisão entre dois números e mostre seu resultado em uma caixa de mensagem

# Atividades de Exemplo

- Desenvolva um script em JavaScript que realize a soma de dois números e mostre seu resultado no interior de uma tag h1
- Desenvolva um script em JavaScript que realize a divisão entre dois números e mostre seu resultado em uma caixa de mensagem

# Atividades no GoogleClassRoom

- Dada a função em JavaScript abaixo realize uma chamada para esta função:
- ```
function minhaFuncao() {  
    document.getElementById("demo").innerHTML = "Alô Mundo!";  
}
```
- Monte uma função que retorne a soma de 5 + 5
- Monte uma função que receba dois parâmetros e retorne a multiplicação destes. Em seguida realize uma chamada para sua função