

# Desenvolvimento Web



Prof. Wesley Freitas Souza

# JavaScript - Output

- JavaScript **não tem** uma função para exibir, semelhantes ao `echo()` ou `print()` do PHP;
- As possibilidades de exibição no Javascript:
  - Caixa de Alerta - **`window.alert()`**
  - Escrita no documento HTML - **`document.write()`**
  - Escrita dentro de um elemento do HTML - **`innerHTML()`**
  - Escrita no console do navegador - **`console.log()`**

# JavaScript - window.alert()

- Abre uma caixa de alerta

```
<!DOCTYPE html>
```

```
<html>
```

```
  <body>
```

```
    <h1>My First Web Page</h1>
```

```
    <p>My first paragraph.</p>
```

```
    <script>
```

```
      window.alert(5 + 6);
```

```
    </script>
```

```
  </body>
```

```
</html>
```

[http://www.w3schools.com/js/tryit.asp?filename=tryjs\\_output\\_alert](http://www.w3schools.com/js/tryit.asp?filename=tryjs_output_alert)

# JavaScript - document.write()

- Escreve no documento;
- Deve ser usado para testes;

```
<!DOCTYPE html>
```

```
<html>
```

```
  <body>
```

```
    <h1>My First Web Page</h1>
```

```
    <p>My first paragraph.</p>
```

```
    <script>
```

```
      document.write(5 + 6);
```

```
    </script>
```

```
  </body>
```

```
</html>
```

[http://www.w3schools.com/js/tryit.asp?filename=tryjs\\_output\\_write](http://www.w3schools.com/js/tryit.asp?filename=tryjs_output_write)

# JavaScript - innerHTML()

- Escreve no interior de um elemento HTML;
- Substitui todo o seu conteúdo

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>My First Web Page</h1>
```

```
<p>My First Paragraph</p>
```

```
<p id="demo"></p>
```

```
<script>
```

```
document.getElementById("demo").innerHTML = 5 + 6;
```

```
</script>
```

```
</body>
```

```
</html>
```

[http://www.w3schools.com/js/tryit.asp?filename=tryjs\\_output\\_dom](http://www.w3schools.com/js/tryit.asp?filename=tryjs_output_dom)

# JavaScript - console.log()

- Escreve no console do navegador; (tecla F12)

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>My First Web Page</h1>
```

```
<p>My first paragraph.</p>
```

```
<script>
```

```
console.log(5 + 6);
```

```
</script>
```

```
</body>
```

```
</html>
```

[http://www.w3schools.com/js/tryit.asp?filename=tryjs\\_output\\_console](http://www.w3schools.com/js/tryit.asp?filename=tryjs_output_console)

# JavaScript - sintaxe

- São as regras de como escrever os códigos;
- As sentenças no JS são separadas por ponto e vírgula;

```
var x = 5;  
var y = 6;  
var z = x + y;
```

Também funciona se estiverem na mesma linha

```
var x = 5; var y = 6; var z = x + y;
```

[http://www.w3schools.com/js/tryit.asp?filename=tryjs\\_syntax\\_statements](http://www.w3schools.com/js/tryit.asp?filename=tryjs_syntax_statements)

# JavaScript - sintaxe

- As sentenças JS são compostas por:
  - Valores;
  - Operadores;
  - Expressões;
  - Palavras-chave;
  - Comentários;



# JavaScript - sintaxe - Valores

- Valores podem ser fixos (**literais**) ou **variáveis**:

- **Literais:**

- Numeros (com ou sem decimais)

`10.50`

`1001`

- Strings (entre aspas simples ou duplas)

`"John Doe"`

`'John Doe'`

# JavaScript - sintaxe - Valores

## ● Variáveis:

- Utilizadas para armazenar valores de dados;
- Palavra-chave **var** para definir variáveis;
- Sinal de igual para atribuir valor às variáveis;

```
var x;
```

```
x = 6;
```

# JavaScript - sintaxe - Comentários

- São ignorados na execução;
- Servem para explicação;

- Comentário simples //

```
var x = 5;    // Isso é um comentário
```

```
// var x = 6;    A linha inteira está comentada
```

- /\* Comentário de múltiplas linhas: \*/

```
The code below will change  
the heading with id = "myH"  
and the paragraph with id = "myP"  
in my web page:  
*/
```

## JavaScript - sintaxe - Case sensitive

- O JS diferencia maiúsculas de minúsculas
- lastName e lastname são variáveis diferentes;

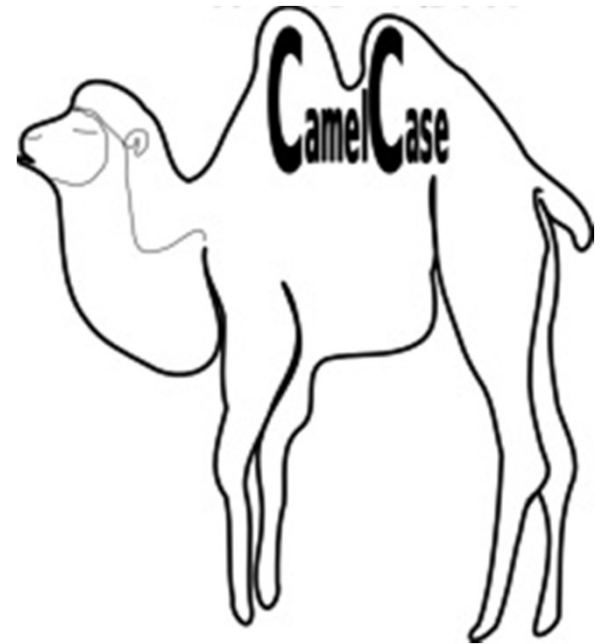
```
lastName = "Doe";  
lastname = "Peterson";
```

# JavaScript - sintaxe - Camel Case

- Variáveis com mais de uma palavra, escritas com a primeira letra das palavras Maiúscula;
- No JS, iniciar com minúscula

`primeiroNome`

`dataDeNascimento`



# JavaScript - Palavras-chave

## Algumas palavras-chave do JS

Keyword	Description
break	Terminates a switch or a loop
continue	Jumps out of a loop and starts at the top
debugger	Stops the execution of JavaScript, and calls (if available) the debugging function
do ... while	Executes a block of statements, and repeats the block, while a condition is true
for	Marks a block of statements to be executed, as long as a condition is true
function	Declares a function
if ... else	Marks a block of statements to be executed, depending on a condition
return	Exits a function
switch	Marks a block of statements to be executed, depending on different cases
try ... catch	Implements error handling to a block of statements
var	Declares a variable

# JavaScript - Operadores Aritméticos

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement

# JavaScript - Operadores Aritméticos

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement



# JavaScript - Operadores de Atribuição

Operator	Example	Same As
=	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

[exemplo](#)

# JavaScript - Operadores de Strings

- O operador + também serve para concatenar strings:

```
txt1 = "Santa";  
txt2 = "Catarina";  
txt3 = txt1 + " " + txt2;
```

txt3 tem valor "Santa Catarina"

# JavaScript - Operadores de Strings

- O operador += também pode ser usado para strings:

```
txt = "Santa ";  
txt += "Catarina";
```

txt tem valor "Santa Catarina"