

Entre no mundo

PARTE I

DA JAVASCRIPT

Iniciamos mais um tutorial, desta vez sobre a JavaScript que lhe vai permitir "artilhar" os seus *sites* e fazer alguns programas.

Javascript é uma linguagem *script* que vai permitir ao leitor adicionar códigos de programação às suas páginas de Internet. Esta linguagem permite criar alguns programas, tais como, uma calculadora ou até mesmo um pequeno jogo primitivo.

De qualquer modo, existem utilizações mais sérias a dar a esta linguagem como, por exemplo, a detecção do *browser* do utilizador que visita a página, a possibilidade de usar *cookies*, abrir páginas personalizadas e, ainda, a validação de campos de informação.

Antes do leitor começar a escrever os seus *scripts*, será melhor começar pelos aspectos básicos desta linguagem. Este tutorial cobre todos os aspectos essenciais que precisará de conhecer antes de entrar em pormenores mais complexos.

O princípio...

O JavaScript não é um código HTML e, sendo assim, o leitor necessita de dar a conhecer ao *browser* onde é que o código se inicia. Para isso, usa-se a "tag" `<script language="javascript">` e `</script>`.

Considere este pequeno exemplo:

```
<html>
<head>
<title>Página em Javascript</title>
</head>

<body>
<script language="javascript">
alert("Bem-vindo à minha página em
JavaScript!!!");
</script>
</body>
</html>
```

A palavra *alert* é um comando *standard* em JavaScript que faz aparecer uma caixa de alerta no ecrã. O leitor vai ter de pressionar o botão "OK" para

continuar.

Caso o leitor não introduzisse as tags `<script language="javascript">` e `</script>`, o *browser* não reconheceria as linhas de código nesta linguagem, imprimindo apenas essas linhas como simples texto.

O leitor poderá escrever o código em JavaScript, tanto na secção `<head>` como em `<body>`, sendo aconselhável escrever sempre que possível, em `<head>`.

Outros pormenores a saber antes de escrever o seu primeiro *script*:

As linhas em JavaScript acabam com ponto e vírgula

O leitor deverá ter reparado que a linha em JavaScript no exemplo anterior acaba com um ponto e vírgula.

Coloque sempre o texto entre " "

Quando o leitor desejar introduzir texto que irá ser tratado pelo JavaScript, deverá fazê-lo sempre entre duas aspas.

Se se esquecer disto, o JavaScript interpretará o texto como sendo variáveis.

As letras maiúsculas são diferentes das letras minúsculas

O leitor deverá lembrar-se que escrever em letras maiúsculas é diferente de escrever em letras minúsculas. Isto significa que quando se escreve um comando, deverá colocar as letras maiúsculas e minúsculas nos sítios correctos.

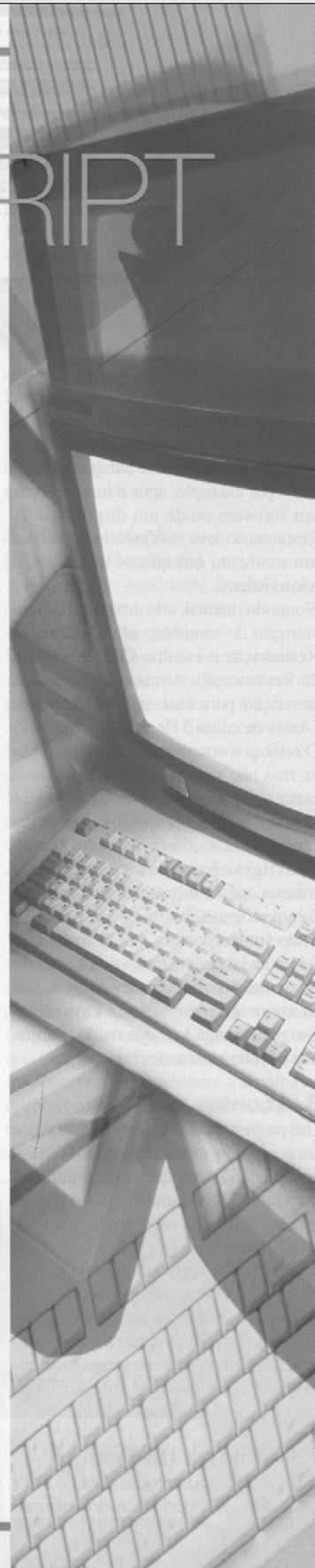
Por exemplo, um comando em JavaScript chama-se `onMouseOver`.

Se o leitor escrever `Onmouseover`, o JavaScript não o vai entender.

Considere agora o seguinte exemplo:

Em vez de escrever uma mensagem numa janela *pop up*, poderá escrever directamente no documento.

```
<html>
```



```
<head>
<title>Página em JavaScript</title>
</head>

<body>
<script>
document.write("Bem-vindo ao meu
mundo!!!");
</script>
</body>
</html>
```

O comando `document.write` diz ao *browser* para escrever no documento o texto entre parênteses.

Nota:

Quando escrever o texto em JavaScript faça-o sempre entre " ".

Este exemplo escreverá no documento a frase:

```
Bem-vindo ao meu mundo!!!
Considere o seguinte exemplo para
aprender onde o JavaScript coloca o
texto:
<html>
<head>
<title> Página em JavaScript</title>
</head>
```

```
<body>
Olá!!!<br>
<script>
document.write("Bem-vindo ao meu
mundo!!!<br>");
</script>
Aprecie a sua visita...<br>
</body>
</html>
```

O "output" para este código seria:
Olá!!!
Bem-vindo ao meu mundo!!!
Aprecie a sua visita...

Como o leitor pode observar, o JavaScript escreve o texto no sítio onde se encontra o código na estrutura HTML.

Um aspecto interessante é que o leitor poderá introduzir todos os tipos de *tags* em HTML dentro do método `document.write`.

Letras maiúsculas

É extremamente importante ter presente que o JavaScript faz uma distinção completa entre letras maiúsculas e minúsculas. Uma variável de nome valor é diferente da variável VALOR.

Considere os seguintes exemplos:

```
Exemplo 1:
<html>
<head>
<title>A minha página</title>
</head>
<body>
<script>
valor=2;
valor=5;
resultado=valor+valor;
document.write(resultado);
</script>
</body>
</html>
```

O *output* deste exemplo seria: 10 (5 + 5).

```
Exemplo 2:
<html>
<head>
<title>A minha página</title>
</head>
<body>
<script>
valor=2;
VALOR=5;
resultado=valor+VALOR;
document.write(resultado);
</script>
</body>
</html>
```

O "output" para este exemplo seria 7 (2 + 5).

Janelas "pop up"

Em JavaScript é possível criar três tipos de janelas *pop up*. A *alert box*, a *confirm box* e a *prompt box*.

Alert Box (caixa de alerta)

A sintaxe da *alert box* é: `alert ("texto");` O utilizador terá de carregar em seguida no botão "OK" para continuar. Um uso típico será o de criar certos avisos para o utilizador.

Confirm Box

(caixa de confirmação)

A sintaxe para esta caixa é: `confirm ("texto");` o utilizador terá de carregar em "OK" ou "Cancel" para continuar. Se o utilizador carregar em "OK" a caixa retornará ao valor *true*. Se carregar em "Cancel", retornará ao valor *false*.

```
Exemplo:
if (confirm("Aceita?")) {alert("Você
aceita")}
else {alert("Você não aceita")};
```

Prompt Box

A sintaxe da *Prompt Box* será: `prompt ("texto","valor_por_defeito");` O utilizador terá de pressionar "OK" ou "Cancel" para continuar, depois de introduzir o texto.

Se o utilizador pressionar em "OK" a *prompt box* devolve o texto inserido; se pressionar "Cancel" a caixa devolve o valor *null*.

Como o utilizador geralmente quer guardar o texto inserido na caixa, será normal guardar esse *input* numa variável, como mostra este exemplo: `nome=prompt("Introduza o seu nome","Introduza o seu nome aqui");`

Variáveis

Uma variável é, simplesmente, um lugar na memória do computador onde este armazena informação. Todas as variáveis são identificadas por um nome único que lhes é atribuído. Considere o exemplo seguinte:

```
<html>
<head>
<title>Página em JavaScript</title>
</head>

<body>
<script>
nome="Sérgio";
document.write(nome);
</script>
```

Operadores aritméticos

A tabela seguinte mostra os operadores aritméticos *a++* e *a--*:

Operador	Significado	Exemplo
++	incremento	a=5;a++ a variável a seria agora igual a 6.
--	decremento	a=5;a--;a variável a seria
%	retorna o resto da divisão entre dois números	a=8%3; a variável a seria agora igual a 2.

```
</body>
</html>
```

Este exemplo escreveria "Sérgio" no documento. Note que quando se pretende que o texto seja armazenado numa variável, é necessário escrever o texto entre " ".

A razão é de que o JavaScript usa os símbolos " " para diferenciar o texto das variáveis.

Veja o seguinte exemplo:

```
<html>
<head>
<title>Página em JavaScript</title>
</head>
```

```
<body>
<script>
sergio="Primeiro nome";
nome=sergio;
document.write(nome);
</script>
</body>
</html>
```

Na primeira linha do script, o texto "Primeiro nome" é guardado na variável "sergio".

Na segunda linha, a variável "sergio" é atribuída à variável "nome". Finalmente, na terceira linha, a variável "nome" é escrita no documento.

Atribuição de valores às variáveis

A maneira mais comum de atribuir um valor a uma variável é usar o sinal =. Considere os seguintes exemplos de como se pode atribuir valores a variáveis:

Comparando variáveis

Existem diversas maneiras para comparar variáveis. A maneira mais simples

Operadores de comparação

Operador	Significado	Exemplo
==	igual a	4==5 (falso) 5==5 (verdadeiro) 5==4 (falso)
!	diferente de	4!=5 (verdadeiro) 5!=5 (falso) 5!=4 (verdadeiro)
<	menor que	4<5 (verdadeiro) 5<5 (falso) 5<4 (falso)
>	maior que	4>5 (falso) 5>5 (falso) 5>4 (verdadeiro)
<=	menor ou igual	4<=5 (verdadeiro) 5<=5 (verdadeiro) 5<=4 (falso)
>=	maior ou igual	4>=5 (falso) 5>=5 (verdadeiro) 5>=4 (verdadeiro)

será a comparação de igualdade, que se pode fazer usando um duplo sinal de igual.

```
if (a==b) {alert("a igual a b");}
```

```
if (nome=="Sérgio") {alert("Bonito nome!!!");}
```

Nota:

A função if é explicada mais adiante.

Se se esquecer de usar um duplo sinal de igualdade quando comparar variáveis e usar um único sinal, o leitor não estará a comparar os dois valores mais sim a atribuir o valor da variável à direita do sinal, à variável da esquerda.

Um exemplo deste erro:

```
if (nome="Sérgio"){alert("Bonito nome!!!");}
```

Este é um erro muito comum, que estragará por completo o seu script. A tabela acima mostra os diferentes operadores de comparação.

IF e ELSE

Por vezes o javascript permite fazer distinções entre diferentes possibilidades. Por exemplo, o leitor poderá ter um script que verifique que browser é que o utilizador está a usar ao visitar determinada página. Se for o MSIE, uma página especialmente desenhada para esse browser será carregada, se for o Netscape outra página irá aparecer.

A sintaxe geral para esta função será:

```
if (condição){acção1}else{acção2};
```

Um exemplo:

```
if (browser=="MSIE") {alert("Está a usar o MSIE");}
else {alert("Está a usar o Netscape");}
```

Note que usa-se o "if" e não "IF". Caso contrário, daria um erro. Note-se ainda que se usa dois sinais de igualdade para a comparação "==".

Se escrevêssemos browser="MSIE" estaríamos a atribuir "MSIE" a uma variável chamada browser.

Para o mês que vem...

Acompanhe na próxima edição a segunda parte deste tutorial para a linguagem Java. Até lá, vá praticando.

SÉRGIO AZEREDO

Atribuição de valores às variáveis

Exemplo	Valor
a=2	a=2
a=2; a++;	a=3 (2+1)
a=2; a--;	a=1 (2-1)
a=2; b=3; c=a+b	c=5 (2+3)
a=2; d=a+6;	d=8 (2+6)
primeiro="Sérgio"; último="Azeredo";	primeiro=Sérgio último=Azeredo
completo=primeiro+" "+último;	completo=Sérgio Azeredo
a=2*7	a=14 (2*7)
b=20/5	b=4 (20/5)
c= (20/5) *2;	c=8 (4*2)
d=20/ (5*6);	d=2 (20/10)