

Funções

Uma função é um grupo de linha(s) de código de programação destinado a uma tarefa específica e que podemos, se necessário, utilizar várias vezes. A utilização de funções melhora bastante a leitura do script.

Em Javascript, existem dois tipos de funções:

- * As funções próprias do Javascript. Que chamamos de "métodos". Elas são associadas a um objecto em particular como o caso do método `Alert()` com o objecto `window`.
- * As funções escritas por vós para executar o vosso script. É a estas que nos vamos interessar agora.

Declaração de funções

Para declarar ou definir uma função, utiliza-se a palavra `function`. A sintaxe de uma declaração de função é a seguinte:

function nome_da_função(argumentos)

```
{  
... código de instruções ...  
}
```

O nome da função segue as mesmas regras das variáveis (número de caracteres indefinido, começado por uma letra pode incluir números...). Volto a lembrar que o Javascript é case sensitive. Assim `função()` não será igual a `Função()`. Todos os nomes de funções num script devem ser únicos.

A menção dos argumentos é facultativo mas não haja argumentos os parêntese devem ficar. É alias graças aos parêntese que o interpretador Javascript distingue as variáveis das funções.

O facto de definir uma função não vai executar os comandos que ela contém. Só é executada quando chamamos a função.

Chamar uma função

A invocação de uma função faz-se o mais simples possível, pelo nome da função com parêntese.

Seja por exemplo **nome_da_função();**

Convém verificar (porque o browser lê o script de cima a baixo) que a vossa função deve estar bem definida antes de a chamar.

As funções dentro `<HEAD>...</HEAD>`

É assim aconselhado de inserir todas as declarações de funções no cabeçalho da página, isto é entre os tags `<HEAD>...</HEAD>`. Assim terão a certeza que as funções já estarão interpretadas antes de serem invocadas no `<BODY>`.

Exemplos 01

Neste exemplo, definimos dentro dos tags HEAD, uma função chamada mensagem() que insere um texto "Bem vindo a minha página". Esta função será chamada no carregamento da página, daí o onLoad dentro do tag <BODY>.

```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="Javascript">

function mensagem() {
document.write("Bem vindo a minha página");
}

</SCRIPT>
</HEAD>
<BODY onLoad='mensagem()' '>
</BODY>
</HTML>
```

Eventos

Geral

Com os eventos e sobretudo sua gestão, abordamos o lado "mágico" do Javascript.

Em Html clássico, há um evento que conhecem bem. É o clique do rato sobre um link que vai abrir outra página Web. Infelizmente, é praticamente o único. Felizmente, o Javascript vai acrescentar uma boa dezena.

Os eventos Javascript, associados às funções, aos métodos e aos formulários, abrem uma grande porta para uma verdadeira **interactividade** das páginas.

Eventos

Vamos ver os diferentes eventos implementados em Javascript.

EVENTOS	DESCRIÇÃO
Clik	Quando o utilizador clica sobre um botão, um link ou outro elementos.
Load	Quando a página é carregada pelo browser.
Unload	Quando o utilizador saia da página.
MouseOver	Quando o utilizador coloca o ponteiro do rato sobre um link ou outro elemento.
MouseOut	Quando o ponteiro do rato não está sobre um link ou outro elemento.
Focus	Quando um elemento de formulário tem o focus, isto é, que está activo.
Blur	Quando um elemento de formulário perde o focus, isto é, quando o deixa de estar activo.
Change	Quando o valor de um campo de formulário é modificado.
Select	Quando o utilizador selecciona um campo dentro de elemento de formulário.
Submit	Quando o utilizador clica sobre o botão <i>Submit</i> para enviar um formulário.

Gestão de eventos

Para ser eficaz, é necessário que a estes eventos sejam associados as acções previstas por ti. É o papel de gestão de eventos. A sintaxe é: `onevento="function() "`

Exemplos 02	onClick
<p>Por exemplo, <code>onClick="alert('Bem vindo à página')"</code>. De maneira literária, no clicar do utilizador, abrir uma caixa de aviso com a mensagem indicada.</p>	
<p>onClick</p>	
<p>Evento mais clássico em informática, o clique do rato.</p> <p>O código deste botão é:</p> <pre><FORM> <INPUT TYPE="button" VALUE="Clicar" onClick="alert('Acabas-te de clicar no botão')"> </FORM></pre>	
<pre><HTML> <HEAD> <TITLE>Função</TITLE> </HEAD> <BODY BGCOLOR="WHITE"> <H2>Os três grandes</H2> <HR> <FORM> <INPUT TYPE="BUTTON" VALUE="clicar" onClick="alert('Acabas-te de clicar no botão ')"> </FORM> </BODY> </HTML></pre>	

Exemplos 03	onLoad e onUnload
<p>onLoad e onUnload</p>	
<p>O evento Load aparece quando a página acaba de se carregar. O inverso, Unload aparece quando o utilizador sai da página.</p>	
<p>Os eventos onLoad e onUnload são utilizados sob forma de atributos do tags <BODY> ou <FRAMESET>. Pode-se assim escrever um script para desejar as boas vindas na abertura de uma página e uma pequena palavra de adeus ao sair desta.</p>	
<pre><HTML> <HEAD> <SCRIPT LANGUAGE='Javascript '> function bemvindo() { alert("Bem vindo a esta página"); } function adeus() { alert("Adeus"); } </SCRIPT> </HEAD> <BODY onLoad='bemvindo()' onUnload='adeus() '> Html normal </BODY> </HTML></pre>	

Exemplos 04**onmouseover e onmouseout****onmouseover e onmouseout**

O evento **onmouseover** executa-se quando o cursor passa por cima (sem clicar) de um link ou de uma imagem. Este evento é bastante prático, por exemplo, para inserir explicações na barra de estado ou mesmo com uma pequena janela tipo info-objecto.

Passar o cursor do rato sobre a palavra exemplo (sem clicar no link).

```
<HTML>
<HEAD>
<SCRIPT language="Javascript">
function mensagem() {
alert ("www.aulastec.no.sapo.pt")
}
</SCRIPT>
</HEAD>
<BODY>
<A HREF="#" onmouseout="mensagem()">Evento onmouseout</A>
<br><br><br>
<A HREF="#" onmouseover="mensagem()">Evento onmouseover</A>
</BODY>
</HTML>
```

Exemplos 044 uma outra forma de fazer o mesmo que em cima onmouseover e onmouseout

```
<HTML>
<HEAD>
</HEAD>
<BODY>
<A HREF="#" onmouseover="alert('Aulas de tec')">mensagem
importante</A>
</BODY>
</HTML>
```

Exemplos 05**onFocus****onFocus**

O evento **onFocus** é quando um objecto se torna activo, isto é, sempre que activamos uma janela, ou uma textfiel que fica pronta para receber texto. Os objectos podem tornar-se activos com o clique do rato ou com o uso da tecla "Tab".

Se clicar na zona do texto, irá se efectuar-se um focus.

```
<HTML>
<HEAD>
<TITLE>Exemplos</TITLE>
</HEAD>
<BODY BGCOLOR="WHITE">
<H2>Digita qualquer coisa</H2>
<HR>
<FORM>
<INPUT TYPE=text onFocus="alert('Isto é um onFocus!')">
</FORM>
</BODY>
</HTML>
```

Exemplos 055**onBlur**

onBlur

O evento onBlur executa-se quando por exemplo uma text field de um formulário perde o focus. Isto acontece quando o utilizador desactiva a text field clicando fora dela ou utilizando a tecla "Tab".

Se depois de clicar e/ou escrever na zona do texto, clica-se fora do documento, produz-se o evento Blur.

O código é:

```
<FORM>
<INPUT TYPE=text onBlur="alert('Isto é um Blur!')">
</FORM>
```

Numéricos caracteres matrizes

```
<HTML>
<HEAD>
  <TITLE>Exemplos</TITLE>
</HEAD>
<BODY BGCOLOR="WHITE">
<H2> Digita qualquer coisa </H2>
<HR>
<FORM>

<INPUT TYPE=text onBlur="alert('Isto é um Blur!')">
</FORM>
</BODY>
</HTML>
```

Exemplos 06**onchange****onchange**

Este evento é bastante semelhante ao do onBlur mas com uma pequena diferença. Não só a área de texto deve ter perdido o focus mas também o seu conteúdo deve ter sido alterado pelo utilizador.

```
<HTML>
<HEAD>
  <TITLE>Função</TITLE>
</HEAD>
<BODY BGCOLOR="WHITE">
<H2>Digita qualquer coisa</H2>
<HR>
<FORM>

<INPUT TYPE=text onchange="alert('Isto é um change!')">

</FORM>
</BODY>
</HTML>
```

Exemplos 07	onSelect
onSelect	
<p>Este evento executa-se quando o utilizador seleccionou toda ou parte de um texto numa text field.</p>	
<pre> <HTML> <HEAD> <TITLE>Função</TITLE> </HEAD> <BODY BGCOLOR="WHITE"> <H2>Os três grandes</H2> <HR> <FORM> <INPUT TYPE=text value="tenta seleccionar este texto" onSelect="alert('Não é permitida a copia !')"> </FORM> </BODY> </HTML> </pre>	

Exemplos 08	Troca de imagens
Troca de imagens	
<p>Com a gestão de eventos onMouseOver, podemos prever que depois de sobrevoar uma imagem pelo o utilizador, uma outra aparece (ambas as imagem devem ter o mesmo tamanho).</p> <p>O código é relativamente simples.</p>	
<pre> <HTML> <HEAD> <SCRIPT LANGUAGE="Javascript1.1"> function lightUp() { document.images["homeButton"].src="button_hot.gif" } function dimDown() { document.images["homeButton"].src="button_dim.gif" } </SCRIPT> </HEAD> <BODY> </BODY> </HTML> </pre>	
<p>Completem <u>sempre</u> em Javascript os atributos width=x height=y das tuas imagens.</p>	