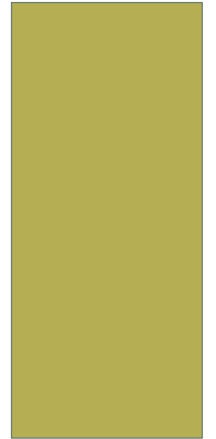


# Módulo 5

Desenvolvimento de páginas web dinâmicas

Prof: Paulo Quaresma



# O que é o Javascript?

.O Javascript é uma linguagem de script que incorporado nas tag's Html, permite incrementar a apresentação e interactividade das páginas Web.

.Com estes scripts é possível tornar as páginas mais ativas, possibilitando:

- ▣ A entrada de dados pelo utilizador, através de formulários;
- ▣ A realização de cálculos em tempo real;
- ▣ O controlo de janelas e botões;
- ▣ Alguma interacção com o utilizador;
- ▣ A integração com outros scripts.

# Javascript VS Java!

.Javascript é completamente diferente do Java. Mesmo que os dois sejam utilizados para criar páginas Web.

## JAVASCRIPT

Código integrado na página Html

Código interpretado pelo browser  
no momento da execução

Códigos de programação simples  
mas para aplicações limitadas

Permite aceder aos objectos do  
browser

Confidencialidade do código é nula  
(Código é visível)

## JAVA

Módulo (applet) distinto da página  
Html

O código é compilado antes da sua  
execução

Linguagem de programação muito  
mais complexa mas mais poderosa

Não tem acesso aos objectos do  
browser

Segurança (Código compilado)

# Código HTML e Scripts

- ✓ Os scripts podem ser combinados com a linguagem HTML de duas formas:
  - Declarando o código Javascript internamente (juntamente com o código HTML)
  - Recorrendo a um ficheiro externo de extensão **.js**

# Declaração interna

✓ Podem ser inseridas no cabeçalho <head> "exemplo 1" ou dentro do corpo <body> "exemplo 2"

✓ Exemplo 1:

▫ Este tipo de declarações é usada quando queremos carregar os scripts antes de ser possível a sua utilização.

▫ Usado para eventos ou para declaração de funções

```
✓ E <head>  
  <script language="javascript"> (código JS aqui)</script>  
</head>
```

```
<body>  
  <script language="javascript"> (código JS aqui)</script>  
</body>
```

# Declaração externa

Se pretendermos declarar os scripts externamente, isto é, num ficheiro à parte, basta para isso incluir uma referência no código html sobre a localização desse ficheiro

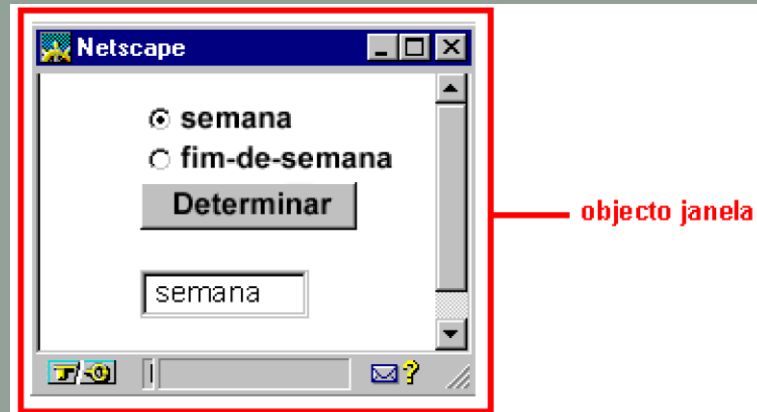
```
<body>  
  <script src="script.js"></script>  
</body>
```

# Os objectos e as suas hierarquias

.Javascript divide uma página em objectos e vai permitir aceder e manipular estes objectos.



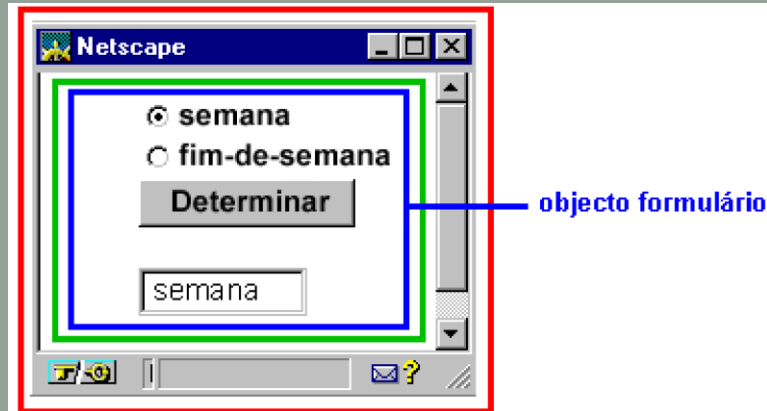
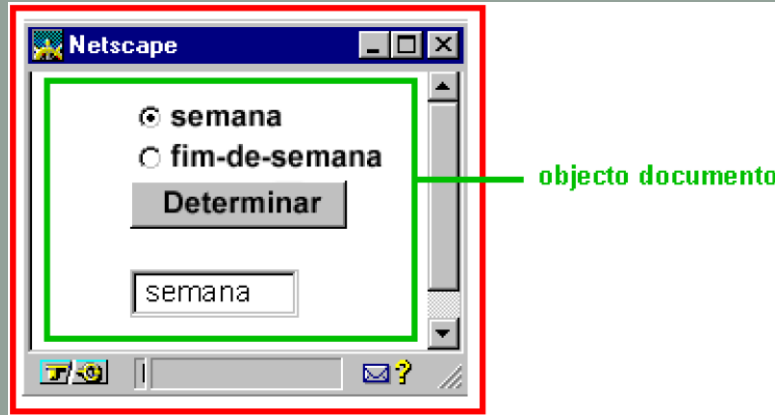
Esta página aparece numa janela.  
É o **objecto janela**.



**objecto janela**

Nesta janela, há um documento Html,  
é o **objecto documento**. Quer isto dizer  
o objecto janela contém o objecto  
documento

# Objecto documento



Neste documento, temos um formulário. É o **objecto formulário**. Dito de outra maneira, o objecto janela contém um objecto documento que contém por sua vez um objecto formulário.



# Objecto formulário

Neste documento encontramos 3 objectos. Os **botões radio**, um **botão clássico** e **uma zona de texto**. Que são respectivamente o **objecto radio**, o **objecto botão**, e o **objecto texto**. Por outras palavras o **objecto janela** contém o **objecto documento** que contém o **objecto formulário** que contém por sua vez os objectos radio, botão, e texto.

A hierarquia deste exemplo é então:

```
janela > documento > formulário > radio
                                     > botão
                                     > texto
```

# Objectos

Para aceder a um objecto, é necessário dar o caminho completo do objecto começando pelo objecto mais externo até ao objecto de referência.

.Seja por exemplo para o botão radio "semana":

`.(window).document.form.nome_do_objecto`

.Nota: Pode-se omitir o `window.document`

# Exemplo 1

```
<html>
  <head>
    <title>Escrever na página com Javascript</title>
  </head>
  <body>
    <script type="text/javascript">
      document.write("<b>Olá Mundo!</b>");
    </script>
  </body>
</html>
```

# Exemplo 2

## **.Mensagens escritas em HTML e em Javascript**

```
<html>
<head>
<title>Alternar entre HTML e Javascript</title>
</head>
<body>
Este texto está escrito utilizando HTML.
<br>
<script type="text/javascript">
document.write("Texto escrito em Javascript.");
</script>
<br>
Este texto foi escrito utilizando novamente o HTML.
</body>
</html>
```

# Exemplo 3

## .Escrita de mensagens formatadas (com tags HTML) em javascript

```
<html>
<head>
  <title>Gerar código HTML em Javascript</title>
</head>
<body>
  <script type="text/javascript">
    document.write("<b></b>Lista com dois items:<br>");
    document.write("<ul><li>item um</li><li>item
dois</li></ul>");
  </script>
</body>
</html>
```

# Exemplo 4

```
<html>
<head>
  <title>A Minha Página com JavaScript</title>
  <script type="text/javascript">
    alert("Seja bem vindo(a) à minha página!");
  </script>
</head>
<body>
  Aqui colocamos o conteúdo da página em HTML
</body>
</html>
```

# Operadores aritméticos

Operador	Descrição	Exemplo	Resultado
+	Adição	2+2	4
-	Subtracção	5-2	3
*	Multiplicação	4*5	20
/	Divisão	15/5 5/2	3 2.5
%	Modulus (resto da divisão)	5%2 10%8 10%2	1 2 0
++	Incrementar (aumentar uma unidade)	X=5 X++	X=6
--	Decrementar (diminuir uma unidade)	X=5 X--	X=4

# Operadores de comparação

Operador	Descrição	Exemplo	Resultado
==	é igual a	5==8	false
!=	não é igual a	5!=8	true
>	é maior do que	5>8	false
<	é menor do que	5<8	true
>=	é maior ou igual a	5>=8	false
<=	é menor ou igual a	5<=8	true



# Operadores lógicos

Operador	Descrição	Exemplo
&&	e (and)	x=6 y=3 (x < 10 && y > 1) dá true
	ou (or)	x=6 y=3 (x==4    y==4) dá false
!	negação (not)	x=6 y=3 !(x==y) dá true

# Declaração de variáveis

- ✓ As variáveis podem ser declaradas de duas formas:
  - Procedidas da palavra var...
    - `var idade=18;`
    - `var nome="Joana";`
  - ...ou diretamente
    - `idade=18;`
    - `nome="Joana";`
- ✓ A palavra `var` não é obrigatória, com exceção da situação em que a variável não toma qualquer valor
  - `var idade`

# Estruturas de controlo

## ✓Estrutura de decisão **if-else**

```
if (condição)
{<bloco de instruções>;}
else
{<bloco de instruções>;}
```

## ✓Estrutura de decisão **switch-case**

```
switch (expressão)
{
case valor1:
<bloco de instruções>; break
case valor2:
<bloco de instruções>; break
(...)
case valorN:
<bloco de instruções>; break
default:<bloco de instruções>;
}
```

# Exemplo 5

```
<html>
<body>
<script type="text/javascript">
var nome = prompt("Qual o seu nome?");
var hora = prompt("Que horas são");
if(hora < 12)
    document.write("Bom dia. Seja bem vindo senhor(a) " + nome);
else {
    if(hora >= 13)
        document.write("Boa tarde. Seja bem vindo senhor(a) " + nome);
    else
        document.write("Seja bem vindo! Almoça conosco?");
}
</script>
</body>
</html>
```

# Ciclos

## ✓ **For – While - do-while**

```
for(inicialização; condição_de_paragem; incremento)
{
    < bloco de instruções >
}
```

```
while (condição)
{
    < bloco de instruções >
}
```

```
do
{
    < bloco de instruções >
} while (condição);
```

# Exemplo 6

```
<html>
<head>
  <title></title>
</head>
<body>
  <script type="text/javascript">
    for (var i = 0; i <= 5; i++)
    {
      document.write("o número é " + i);
      document.write("<br>");
    }
    document.write("O ciclo terminou<br>");
  </script>
</body>
```

# Funções

- ✓ São definidas pela palavra function podendo, receber ou não parâmetros, bem como, retornar ou não valores
- ✓ A estrutura é a seguinte:

```
function nome_da_funcao(parametro1, parametro2, ...)  
{  
    <bloco de instruções>;  
    return [valor ou expressão];  
}
```

# Exemplo

```
<html>
<head>
<title></title>
<script type="text/javascript">
  function somar(n)
  {
    var soma = n + 2;
    return soma;
  }
</script>
</head>
<body>
  <p>
    O valor devolvido pela função é:
    <script type="text/javascript">
      document.write("somar(5) = "+somar(5));
    </script>
  </p>
</body>
</html>
```



# Eventos

- ✓A linguagem HTML evoluiu de forma a aceitar a inclusão de eventos em Java Script.
- ✓Assim, passou a permitir a possibilidade da programação interativa em páginas web
- ✓Alguns dos principais eventos encontram-se na tabela seguinte:

onClick	onSubmit	MouseOver	onUnload
onDragDrop	KeyUp	MouseUp	onChange
onKeyDown	MouseDown	Move	onReset
KeyPress	MouseMove	Resize	
onSelect	MouseOut	onLoad	

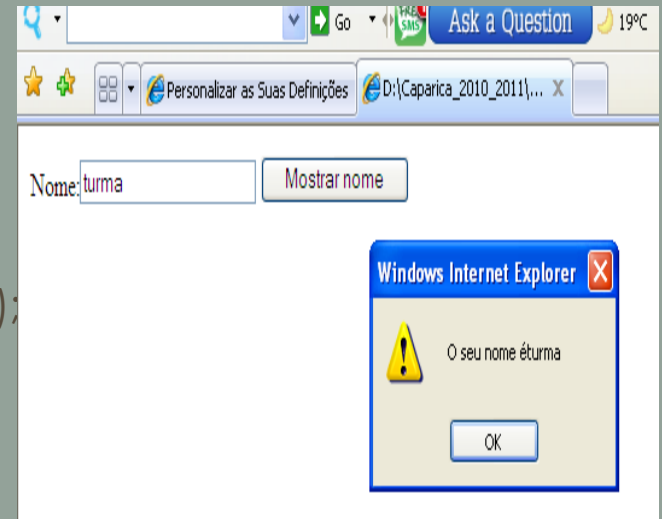
# Exemplo 6

```
<html>
<head>
<title></title>
<script type="text/javascript">
function funcao1 ()
{
  alert("Ola");
}
</script>
</head>
<body>
<p>
<b>Nota:</b>
  Em JavaScript tudo são objetos, funções e variáveis incluídas
</p>
<form>
Clique nos botões para executar as funções:<br>
<input type="button" onclick="funcao1 ()" value="Executar funcao1 ()">
</form></body></html>
```

# JavaScript e formulários

# Exemplo 7

```
<head>
  <script language="javascript">
    function testResultd()
    {
      if(form1.nome.value=="")
        alert("A caixa encontra-se em branco");
      else
      {
        var testar=form1.nome.value;
        alert("O seu nome é" +testar);
      }
    }
  </script>
</head>
<body>
  <form name="form1">
    Nome:<input type="text" name="nome">
    <input type="button" value="Mostrar nome" onClick="testResultd()">
  </form></body>
```



# Exemplo – Validar os campos de um formulário

```
1 <html>
2 <head>
3   <titleValidação de formulário</title>
4 </head>
5 <script language="javascript">
6   function testResultd(){
7     if(form1.nome.value=="")
8       alert("A caixa encontra-se em branco");
9     else
10      {
11        if ((form1.idade.value=="") || (isNaN(form1.idade.value)))
12          alert("Campo idade Vazio ou valor inválido!!");
13      }
14  }
15 </script>
16
17 <body>
18 <form name="form1">
19   Nome:<input type="text" name="nome">
20   Idade<input type="text" name="idade">
21   <input type="button" value="Mostrar nome" onClick="testResultd()">
22 </form>
23
24 </body>
25 </html>
```

## Função isNaN(...)

abreviatura de  
“is not a number” –  
Verifica se o que se  
Encontra dentro de  
parênteses não é  
um número

# Exercício prático

.Validar todos os campos do formulário da ficha de revisões sobre HTML  
(Ficheiro registo.html)

# Propriedades dos objetos

## *text e password*

### **.name**

▫ nome do campo. Este nome não pode estar duplicado num formulário.

### **.value**

▫ valor do campo

### **.disabled**

▫ se seu valor for "true", bloqueia o campo de texto e não permite introduzir nenhum dado no campo

### **.ReadOnly**

▫ O valor do campo só pode ser lido. A diferença do *disabled* - esta propriedade, não bloqueia o campo, simplesmente não permite modificar seu conteúdo.

### **.Size**

▫ A largura "visível" da caixa de texto. NÃO limita a quantidade de caracteres.

### **.length**

▫ conta a quantidade de caracteres introduzidos no campo.

### **.maxlength**

▫ É a quantidade máxima de caracteres permitidos dentro do campo.

# Propriedades dos objetos

## Checkbox e radio

### **.name**

É o nome que identifica a *checkbox* / *radio buttons*.

### **.value**

É o valor associado a cada *checkbox* ou *radio*.

### **.disabled**

bloqueia o *checkbox* ou *radio*.

### **.checked**

comprova se está marcado um *checkbox* ou um *radio*.

É uma propriedade *booleana* (que só admite *verdadeiro* ou *falso*) na qual seu valor será "*true*" (*verdadeiro*) se o objeto estiver marcado e "*false*" (*falso*) no caso em que não esteja.

### **.length**

é a quantidade de *radio buttons* que existe em um grupo determinado com o mesmo *name*.



# Principais eventos

## Checkbok e radio

### **.onFocus**

▪ Permite realizar uma ação ao colocar o foco no objeto.

### **.onBlur**

▪ Permite realizar uma ação quando o foco já não se encontra no objecto.

### **.onClick**

▪ Permite realizar uma ação quando se clica sobre o objeto.

### **.onChange**

▪ Evento que se produz ao marcar/desmarcar um *checkbox* ou um *radio button*.

# Exemplo Checkbox

```
1 <html>
2 <head>
3   <title>Exemplo CheckBox</title>
4
5   <script language="javascript">
6     function testResultd()
7     {
8       if((form1.d1.checked == false) && (form1.d2.checked == false) && (form1.d3.checked == false) &&(form1.d4.checked == false))
9         alert("Deve seleccionar pelo menos uma opção");
10    }
11  </script>
12
13 </head>
14 <body>
15 <form name="form1">
16   Qual o desporto de que mais gosta?<br>
17   <input type = "checkbox" name="d1" value="d1"> Futebol<br>
18   <input type = "checkbox" name="d2" value="d2"> Ténis<br>
19   <input type = "checkbox" name="d3" value="d3"> Basquetebol<br>
20   <input type = "checkbox" name="d4" value="d4"> Não gosto de desporto<br>
21   <input type="button" value="Responder" onClick="testResultd()">
22 </form>
23
24
25 </body>
26 </html>
```

# SELECT: Propriedades principais

## **.name**

Nome do select

## **.size**

Estabelece a quantidade de opções que à primeira vista se verão no select

## **.option**

É a principal propriedade já que permite aceder a cada opção do select.

## **.disabled**

Se seu valor for "true", bloqueia o select e não permite realizar nenhum tipo de seleção.

## **.multiple**

Permite ao utilizador seleccionar mais de uma opção (mediante a tecla **Ctrl**) **type** Informa se o select é do tipo *múltiplo* ou *simples*.

# OPTION: Propriedades principais

## **value**

▪ Valor associado a cada opção do select. É invisível ao utilizador, porém é o valor que se envia junto ao formulário.

## **.Text**

▪ É o texto que o utilizador vê que se encontra contido entre as etiquetas `<option>` e `</option>`

## **.selected**

▪ Indica se uma opção está selecionada.

## **.selectedIndex**

▪ Dá acesso à opção (*option*) que selecionou o utilizador.

## **.length**

▪ Conta a quantidade de opções (*option*) que conformam um select

# Exemplo select

```
1 <html>
2 <head>
3   <title>Exemplo validação select</title>
4
5   <script language="javascript">
6     function testResultd()
7     {
8       if(form1.escolha.selectedIndex == 0)
9         alert("Deve escolher pelo menos uma pergunta!");
10    }
11  </script>
12 </head>
13 <body>
14   <form name="form1">
15     <select name="escolha">
16       Pergunta:
17       <option>Escolha uma pergunta ....</option>
18       <option>Qual o nome da minha escola?</option>
19       <option>Qual a minha banda favorita?</option>
20       <option>Qual o meu filme favorito?</option>
21     </select>
22     <input type="button" value="Mostrar nome" onClick="testResultd()">
23   </form>
24
25 </body>
26 </html>
```

# Exemplo prático 2

```
3 <head>
4 <script language="JavaScript">
5
6 function enviardados()
7 {
8 if(document.dados.tx_nome.value==" " || document.dados.tx_nome.value.length < 8)
9
10
11     alert( "Preencha campo NOME corretamente!" );
12     document.dados.tx_nome.focus();
13     return false;
14 }
15 if( document.dados.tx_email.value==" " || document.dados.tx_email.value.indexOf('@')== -1 || document.dados.tx_email.value.indexOf('.')== -1 )
16 {
17     alert( "Preencha campo E-MAIL corretamente!" );
18     document.dados.tx_email.focus();
19     return false;
20 }
21 if (document.dados.tx_mensagem.value=="")
22 {
23     alert( "Preencha o campo MENSAGEM!" );
24     document.dados.tx_mensagem.focus();
25     return false;
26 }
27 if (document.dados.tx_mensagem.value.length < 50 )
28 {
29     alert( "É necessario preencher o campo MENSAGEM com mais de 50 caracteres!" );
30     document.dados.tx_mensagem.focus();
31     return false;
32 }
33 return true;
34 }
35 </script>
36
37 </head>
```

Nome completo:

E-mail:

Mensagem:

Utilização do evento focus

Utilização da propriedade length