

## **Programação para WEB**

### **PHP Básico**

**Prof. Bruno Camargo Ribeiro**

#### **Formulários**

Assuntos que serão abordados:

1.1 Elemento form .....	2
1.2 Propriedade action .....	2
1.3 Propriedade name .....	3
1.4 Elemento input e a propriedade type .....	4
1.4.1 Caixa de texto (uma linha) .....	4
1.4.2 Entradas de senha .....	4
1.4.3 Limitando a entrada de dados .....	5
1.4.4 Valores padrão em caixa de texto .....	6
1.4.5 Os elementos submit e reset .....	6
1.4.6 Caixas de seleção .....	7
1.4.7 Botões de opção .....	8
1.4.8 Campos ocultos .....	9
1.4.9 Substituindo botões submit e reset por imagens .....	9
1.4.10 Personalizando botões .....	11
2 Botões fora do formulário .....	11
3 Menus suspensos (combo) .....	13
4 Elemento textarea .....	14
5 Agrupando campos .....	15
6 Uso de label .....	17
7 Campos somente para leitura .....	18
8 Desabilitar e habilitar controles .....	19

## 1 Formulário

O formulário é a base para páginas que oferecem maior interatividade com o usuário, servindo como um repositório para vários controles que forma a sua estrutura.

### 1.1 Elemento *form*

O elemento *form* identifica os limites de um formulário dentro de uma página, cada *form* representa uma estrutura única. Podemos ter vários *forms* em uma única página, mas devidamente agrupados.

```
exemplo01.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form>
7          <input type="text" />
8          <input type="submit" value="Enviar" />
9      </form>
10 </body>
11 </html>
```

### 1.2 Propriedade *action*


Para que o conteúdo de um formulário seja processado as informações deverão ser repassadas para outra página, geralmente esta página é escrita em Java, PHP, ASP entre outras linguagens. Na chamada do *action* deverá ser identificado o nome do arquivo que receberá estes dados, em tempos de desenvolvimento a propriedade poderá ficar em branco.

```
exemplo02.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form action="cadastro.php">
7          <input type="text" />
8          <input type="submit" value="Enviar" />
9      </form>
10 </body>
11 </html>
```

Ao clicarmos no botão "Enviar" teremos um erro de página caso o arquivo **cadastro.php** não exista.

### 1.3 Propriedade *name*

A propriedade *name* é bastante útil para validação e controle dos dados, procure deixar sempre o formulário identificado para que posteriormente o *JavaScript* e demais linguagens tenham acesso aos dados do formulário.



```
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" action="cadastro.php">
7     <input type="text" />
8     <input type="submit" value="Enviar" />
9   </form>
10 </body>
11 </html>
```

### 1.3 Propriedade *method*

Para envio do conteúdo é preciso definir o método *http* que será utilizado para a transação dos dados. A propriedade *method* aceita dois valores:

**get:** é utilizado quando os dados do formulário não tem a necessidade de ficar oculto ao usuário, pois as informações são visualizadas na URL quando o *submit* é acionado. É bastante utilizado quando o usuário deseja recuperar URL com informações que serão tratadas por outras páginas, um bom exemplo são os sites de compras, botões de pesquisa, etc.



```
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="get" action="cadastro.php">
7     <input type="text" />
8     <input type="submit" value="Enviar" />
9   </form>
10 </body>
11 </html>
```

**post:** usado quando o formulário remete dados que devem ser ocultos e principalmente enviados para banco de dados, sendo assim, as informações não serão visualizadas na URL.

```
exemplo05.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          <input type="text" />
8          <input type="submit" value="Enviar" />
9      </form>
10 </body>
11 </html>
```

### 1.4 Elemento *input* e a propriedade *type*

O elemento *input* é utilizado para inserir controles no formulário, por meio da propriedade *type* definimos qual o tipo de controle será utilizado, são eles: ***text***, ***password***, ***checkbox***, ***radio***, ***submit***, ***reset***, ***hidden***, ***image*** e ***button***.

#### 1.4.1 Caixa de texto (uma linha)

São os controles mais utilizados, permite a entrada de dados em uma única linha.

```
exemplo06.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Nome:
8          <input type="text" name="nome"/> <br><br>
9          E-mail:
10         <input type="text" name="email"/> <br><br>
11         <input type="submit" value="Enviar" />
12     </form>
13 </body>
14 </html>
```

Observe o uso da propriedade ***name***, é através deste identificador que teremos os dados recuperados posteriormente.

#### 1.4.2 Entradas de senha

Utilize o tipo *password*, para entradas do tipo senha, ou para qualquer outra entrada onde os caracteres de entrada deverão permanecer ocultos.

```
exemplo07.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Login:
8     <input type="text" name="nome"/> <br><br>
9     Senha:
10    <input type="password" name="senha"/> <br><br>
11    <input type="submit" value="Enviar" />
12  </form>
13 </body>
14 </html>
```

### 1.4.3 Limitando a entrada de dados

As propriedades **size** e **maxlength** geralmente são utilizadas em conjunto, onde *size* define o tamanho da janela e *maxlength* a quantidade de caracteres a serem digitados.

```
exemplo08.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Nome:
8     <input type="text" name="nome" size="50" maxlength="45"/> <br><br>
9     <input type="submit" value="Enviar" />
10  </form>
11 </body>
12 </html>
```

Neste caso a janela de entrada (caixa de texto), possui 50 caracteres de largura e deverá receber no máximo 45 caracteres.

Caso a sua janela de entrada seja menor que a quantidade de caracteres a serem digitados, não se preocupe, pois o rolamento de caracteres deverá ocorrer normalmente.

```
Nome:
<input type="text" name="nome" size="10" maxlength="45"/> <br><br>
```

Alguns navegadores podem não interpretar corretamente a instrução *size*, esta forma vamos ao CSS.



```
exemplo10.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Nome:
8          <input type="text" name="nome" style="width:250px" maxlength="45"/> <br><br>
9          <input type="submit" value="Enviar" />
10     </form>
11 </body>
12 </html>
```

Ao utilizarmos a propriedade **width**, a tamanho deverá estar em pixel.

#### 1.4.4 Valores padrão em caixa de texto

A propriedade **value**, permite definir um valor padrão para entrada na caixa de texto.

```
exemplo11.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Nome:
8          <input type="text" name="nome" style="width:250px" maxlength="45"/> <br><br>
9          Estado:
10         <input type="text" name="estado" style="width:25px" maxlength="2" value="SP"/> <br><br>
11         <input type="submit" value="Enviar" />
12     </form>
13 </body>
14 </html>
```

#### 1.4.5 Os elementos **submit** e **reset**

O elemento **submit** é o botão que permite que os dados do formulário sejam enviados e o elemento **reset** permite limpar os campos ou retorná-lo ao valor padrão.

```
exemplo12.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Login:
8     <input type="text" name="nome" style="width:150px" maxlength="10"/> <br><br>
9     Senha:
10    <input type="password" name="senha" style="width:150px" maxlength="10" /> <br><br>
11    <input type="reset" value="Limpar" />
12    <input type="submit" value="Enviar" />
13  </form>
14 </body>
15 </html>
```

#### 1.4.6 Caixas de seleção

O tipo *checkbox*, permite criar caixas de seleção.

```
exemplo13.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Qual linguagem de programação você conhece?<br>
8     <input type="checkbox" name="asp"> ASP <br>
9     <input type="checkbox" name="php"> PHP <br>
10    <input type="checkbox" name="java"> JAVA <br>
11    <br><input type="submit" value="Enviar">
12  </form>
13 </body>
14 </html>
```

Não se esqueça de utilizar a propriedade **name** para posteriormente recuperar os dados. Para ajudar no processo de recuperação, valores poderão ser indicados nos elementos *checkbox* através da propriedade *value*, assim a recuperação fica mais fácil.

Durante o envio do formulário, somente as caixas selecionadas terão os valores enviados.

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Qual linguagem de programação você conhece?<br>
8          <input type="checkbox" name="asp" value="sim"> ASP <br>
9          <input type="checkbox" name="php" value="sim"> PHP <br>
10         <input type="checkbox" name="java" value="sim"> JAVA <br>
11         <br><input type="submit" value="Enviar">
12     </form>
13 </body>
14 </html>

```

Neste exemplo será considerado **sim**, somente para as opções selecionadas.

Para deixar uma caixa já selecionada utilize a propriedade *checked*.

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Qual linguagem de programação você conhece?<br>
8          <input type="checkbox" name="todas" value="sim" checked="true"> Todas as linguagens <br>
9          <br><input type="submit" value="Enviar">
10     </form>
11 </body>
12 </html>

```

#### 1.4.7 Botões de opção

Semelhante *checkbox* em sua estrutura, o *radio* permite que o usuário selecione uma opção em uma coleção, mas para que isso ocorra à propriedade *name* deverá possuir o mesmo identificador.

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Estado Civil<br>
8          <input type="radio" name="estadocivil"> Casado <br>
9          <input type="radio" name="estadocivil"> Solteiro <br>
10         <br><input type="submit" value="Enviar">
11     </form>
12 </body>
13 </html>

```

Assim como no *checkbox* as propriedades *value* e *checked* são aceitas.



```
exemplo17.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Renda familiar:<br>
8     <input type="radio" name="renda" value="nivel01"> até 01 salario mínimo <br>
9     <input type="radio" name="renda" value="nivel02"> de 02 até 05 salários mínimos <br>
10    <input type="radio" name="renda" value="nivel03"> acima até 15 salários mínimos <br>
11    <input type="radio" name="renda" value="naodeclarou" checked="true"> não vou declarar <br>
12    <br><input type="submit" value="Enviar">
13  </form>
14 </body>
15 </html>
```

#### 1.4.8 Campos ocultos

Os campos ocultos permitem a passagem de valores para a página que irá processá-los sem a necessidade de apresentá-los ao usuário auxiliando no processo de desenvolvimento. Um bom exemplo é passar o ID de um elemento do banco de dados para recuperá-lo posteriormente, é importante que a propriedade *value* e *name* esteja devidamente preenchida.

```
exemplo18.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     <input type="hidden" name="user" value="admin">
8     <input type="hidden" name="nivel" value="2">
9     Confirme a sua senha:<br><br>
10    <input type="password"> <input type="submit" value="OK">
11  </form>
12 </body>
13 </html>
```

#### 1.4.9 Substituindo botões submit e reset por imagens

O processo é bem simples, substitua o tipo para *image*, veja o exemplo:

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="get" action="pesquisa.php">
7          Pesquisas:<br><br>
8          <input type="text" name="pesquisar">
9          <input type="image" src="btn_ok.png">
10     </form>
11 </body>
12 </html>

```

O tipo *image* reconhece somente a função *submit*. Mas agora como *image* para os botões **submit** e **reset** ao mesmo tempo? Vamos contornar a situação fazendo uma simulação para um dos botões.

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="get" action="pesquisa.php">
7          Pesquisas:<br><br>
8          <input type="text" name="pesquisar"><br><br>
9          <input type="image" src="btn_enviar.png"/>
10         
11     </form>
12 </body>
13 </html>

```

Note que no evento *onclick*, deverá ocorrer à indicação do nome do formulário.

Outra forma para resolver o problema e criar um link sem redirecionamento.

```

1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="get" action="pesquisa.php">
7          Pesquisas:<br><br>
8          <input type="text" name="pesquisar"><br><br>
9          <input type="image" src="btn_enviar.png"/>
10         <a href="#" onclick="document.cadastro.reset()"></a>
11     </form>
12 </body>
13 </html>

```

#### 1.4.10 Personalizando botões

Podemos definir botões do tipo *button*, o único cuidado com este tipo de elemento é a responsabilidade em realizar a programação do mesmo.

```
exemplo22.html x
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="get" action="pesquisa.php">
7          Pesquisas:<br><br>
8          <input type="text" name="pesquisar"><br><br>
9          <input type="button" value="Voltar" onclick="window.history.back()">
10         <input type="reset" value="Limpar">
11         <input type="submit" value="Enviar">
12     </form>
13 </body>
14 </html>
```

**onclick="window.history.back()",** somente terá efeito se o navegador abriu outras páginas antes desta.

## 2 Botões fora do formulário

O elemento *button*, permite diferentes tipos de aplicações fora do formulário, mas não podemos deixar de programá-los, veja o exemplo:

```
exemplo23.html x
1  <html>
2  <head>
3      <title> Minha página web</title>
4  </head>
5  <body>
6      <button> Home </button>
7      <button> Dicas </button>
8      <button> Truques </button>
9  </body>
10 </html>
```

Bonitos, mas não funcionais. Vamos ao que interessa:

Programando link no botão.

**Etec "Sales Gomes" – 101 – Tatuí**

```
exemplo24.html x
1  <html>
2  <head>
3      <title> Minha página web</title>
4  </head>
5  <body>
6      <button onclick="window.location.href='http://www.google.com.br'"> Google </button>
7  </body>
8  </html>
```

Programando o botão de atualizar página.

```
exemplo25.html x
1  <html>
2  <head>
3      <title> Minha página web</title>
4  </head>
5  <body>
6      <button onclick="window.location.reload()"> Atualizar </button>
7  </body>
8  </html>
```

Programado o botão de impressão.

```
exemplo26.html x
1  <html>
2  <head>
3      <title> Minha página web</title>
4  </head>
5  <body>
6      <button onclick="window.print()"> Imprimir </button>
7  </body>
8  </html>
```

Criatividade com o uso do botão.

```
exemplo27.html x
1  <html>
2  <head>
3      <title> Minha página web</title>
4  </head>
5  <body>
6      <button onclick="window.history.back()"></button>
7  </body>
8  </html>
```

Ou ...



```
exemplo28.html
1 <html>
2 <head>
3   <title> Minha página web</title>
4 </head>
5 <body>
6   <button onclick="window.history.back()"><br>Voltar</button>
7 </body>
8 </html>
```

### 3 Menus suspensos (combo)

O elemento *select* permite criar uma lista de opção.

```
exemplo29.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Estados:
8     <select name="estados">
9       <option> São Paulo </option>
10      <option> Rio de Janeiro </option>
11      <option> Minas Gerais </option>
12      <option> Bahia </option>
13    </select><br><br>
14    <input type="submit" value="Enviar">
15  </form>
16 </body>
17 </html>
```

Caso seja necessário podemos indicar um elemento pré-selecionado, usando a propriedade *selected*.



```
exemplo30.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Estados:
8     <select name="estados">
9       <option> São Paulo </option>
10      <option> Rio de Janeiro </option>
11      <option> Minas Gerais </option>
12      <option selected="true"> Bahia </option>
13    </select><br><br>
14    <input type="submit" value="Enviar">
15  </form>
16 </body>
17 </html>
```

O importante é controlar os valores de retorno para as opções (*option*) indicadas pelo usuário, para isso fazemos o uso da propriedade *value*.

```
exemplo31.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Estados:
8     <select name="estados">
9       <option value="SP"> São Paulo </option>
10      <option value="RJ"> Rio de Janeiro </option>
11      <option value="MG"> Minas Gerais </option>
12      <option selected="true" value="BA"> Bahia </option>
13    </select><br><br>
14    <input type="submit" value="Enviar">
15  </form>
16 </body>
17 </html>
```

#### 4 Elemento *textarea*

O elemento *textarea* permite a criação de caixas de texto de mais de uma linha, sua criação é bem simples.

```
exemplo32.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Mensagem:<br>
8              <textarea name="mensagem">
9              </textarea>
10         <br><br>
11         <input type="submit" value="Enviar">
12     </form>
13 </body>
14 </html>
```

Para ajustar o tamanho da janela utilize as propriedades *col* e *rows*.

```
exemplo33.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          Mensagem:<br>
8              <textarea name="mensagem" cols="80" rows="7">
9              </textarea>
10         <br><br>
11         <input type="submit" value="Enviar">
12     </form>
13 </body>
14 </html>
```

## 5 Agrupando campos

O elemento *fieldset*, permite agrupar os campos de um formulário em uma janela, dando uma aparência mais agradável.

```
exemplo34.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Estado Civil<br>
8     <fieldset>
9       <input type="radio" name="estadocivil"> Casado <br>
10      <input type="radio" name="estadocivil"> Solteiro <br>
11    </fieldset>
12    <br><input type="submit" value="Enviar">
13  </form>
14 </body>
15 </html>
```

Ajustando o tamanho da janela do *fieldset* via CSS.

```
exemplo35.html
1 <html>
2 <head>
3   <title> Minha página web - Formulário</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     Estado Civil<br>
8     <fieldset style="width:200px;">
9       <input type="radio" name="estadocivil"> Casado <br>
10      <input type="radio" name="estadocivil"> Solteiro <br>
11    </fieldset>
12    <br><input type="submit" value="Enviar">
13  </form>
14 </body>
15 </html>
```

Inserindo título para a janela *fieldset*.

```
exemplo36.html
1  <html>
2  <head>
3      <title> Minha página web - Formulário</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          <fieldset style="width:200px;">
8              <legend> Estado Civil </legend>
9              <input type="radio" name="estadocivil"> Casado <br>
10             <input type="radio" name="estadocivil"> Solteiro <br>
11          </fieldset>
12          <br><input type="submit" value="Enviar">
13      </form>
14  </body>
15  </html>
```

## 6 Uso de *label*

Permite criar um título para os elementos de entrada, o qual pode ser usado de duas formas:

A primeira seria como rótulo de um elemento de entrada.

```
exemplo37.html
1  <html>
2  <head>
3      <title> Minha página web - Label</title>
4  </head>
5  <body>
6      <form name="cadastro" method="post" action="cadastro.php">
7          <label>Nome:</label>
8          <input type="text" name="nome"/> <br><br>
9          <label>E-mail:</label>
10         <input type="text" name="email"/> <br><br>
11         <input type="submit" value="Enviar" />
12     </form>
13 </body>
14 </html>
```

Na segunda opção com mais recurso. Veja no código e o relacionamento que ocorreu entre o **label** e o elemento de entrada, agora com o identificar **id**.



```
exemplo38.html
1 <html>
2 <head>
3   <title> Minha página web - Label</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     <label for="txt_nome">Nome:</label>
8     <input type="text" name="nome" id="txt_nome"/> <br><br>
9     <label for="txt_email">E-mail:</label>
10    <input type="text" name="email" id="txt_email"/> <br><br>
11    <input type="submit" value="Enviar" />
12  </form>
13 </body>
14 </html>
```

Melhorando o código.

```
exemplo39.html
1 <html>
2 <head>
3   <title> Minha página web - Label</title>
4   <style type="text/css">
5     b { color:red;}
6   </style>
7 </head>
8 <body>
9   <form name="cadastro" method="post" action="cadastro.php">
10    <label for="txt_nome" accesskey="n"><b>N</b>ome:</label>
11    <input type="text" name="nome" id="txt_nome"/> <br><br>
12    <label for="txt_email" accesskey="m">E-<b>m</b>ail:</label>
13    <input type="text" name="email" id="txt_email"/> <br><br>
14    <input type="submit" value="Enviar" />
15  </form>
16 </body>
17 </html>
```

Use as teclas ALT+N e ALT+M, para testar o efeito. Cuidado com as teclas de atalho do navegador.

## 7 Campos somente para leitura

A propriedade *readonly*, através dos valores **true** e **false**, permite que informações sejam visualizadas, mas não alteradas.



```
exemplo40.html
1 <html>
2 <head>
3   <title> Minha página web</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     <label>Nome:</label><br>
8     <input type="text" name="nome" readonly="true" value="digite seu nome"><br><br>
9     <label>Mensagem:</label><br>
10    <textarea name="mensagem" col="80" rows="7" readonly="true"> digite a sua mensagem </textarea><br><br>
11    <input type="submit" value="Enviar">
12  </form>
13 </body>
14 </html>
```

## 8 Desabilitar e habilitar controles

A propriedade **disabled**, permite habilitar e desabilitar um controle através dos valores *true* e *false*.

```
exemplo41.html
1 <html>
2 <head>
3   <title> Minha página web</title>
4 </head>
5 <body>
6   <form name="cadastro" method="post" action="cadastro.php">
7     <label>Nome:</label><br>
8     <input type="text" name="nome"><br><br>
9     <label>Mensagem:</label><br>
10    <textarea name="mensagem" col="80" rows="7" disabled="true"></textarea><br><br>
11    <input type="submit" value="Enviar">
12  </form>
13 </body>
14 </html>
```

### IMPORTANTE:

Alguns navegadores não reconhecem o valor **false** para algumas propriedades, desta forma omite completamente.