



upcomillas *es*

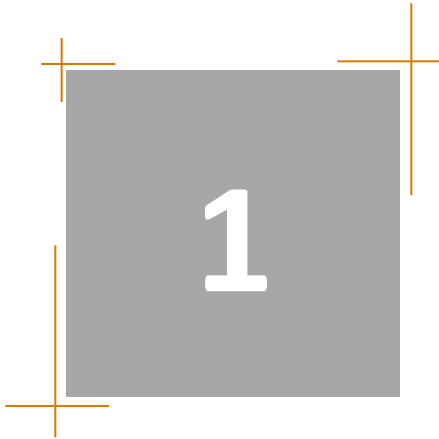
upcomillas *es*

2-Forms and JavaScript

Course: Developing web-based applications

Cristina Puente, Rafael Palacios

2010-2011



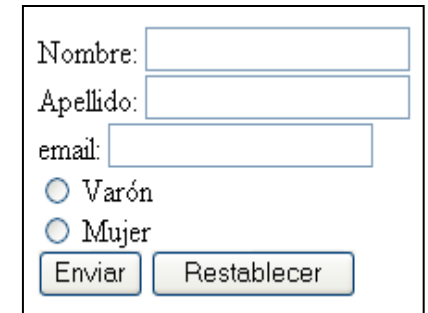
Creating forms



Forms

- An HTML form is a special section of a document which gathers the usual content plus codes, special elements called controls (checkboxes, radiobuttons, menus), etc.

```
<FORM action="http://algunsitio.com/prog/usuarionuevo.php"
method="post">
Nombre:<INPUT type="text" id="nombre"><BR>
Apellido:<INPUT type="text" id="apellido"><BR>
email:<INPUT type="text" id="email"><BR>
<INPUT type="radio" name="sexo" value="Varón"> Varón<BR>
<INPUT type="radio" name="sexo" value="Mujer"> Mujer<BR>
<INPUT type="submit" value="Enviar"><INPUT type="reset">
</FORM>
```



Nombre:
Apellido:
email:
 Varón
 Mujer

Forms

- The *form* element is something like a container for all the controls inside a web page.
- Attributes:
 - Action: refers to the program which will process the content of the form once sent.
 - Name: name of the form to be referred to. It's optative, but it is highly recommended to include.
 - Method: methods of sending data to the server. (Get by URL, Post by standard input stdio)
 - Accept-charset: set of accepted characters.

Forms

- Data sending methods:

- get: with this method, the set of data is linked to the URL specified by the action attribute (using a question mark as a separator ("?")) all this information is sent to the program in charge of processing it.
- post: with this method, the set of data is included and sent to the program in charge of processing it by the standard input/output via.
- The *post* method is more secure than *get*.

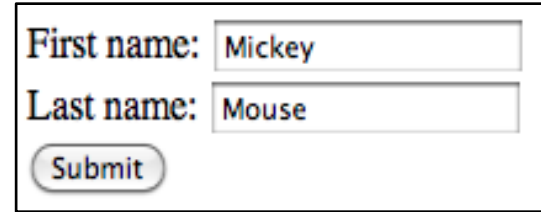
Forms

```
<html>
  <head>
    <title>PHP example</title>
  </head>
  <body>
    <H1>Ejemplo de procesado de formularios</H1>

    <FORM ACTION="procesa2.php" METHOD="GET">
      Introduzca su nombre:<INPUT TYPE="text" NAME="nombre"><BR>
      Introduzca sus apellidos:<INPUT TYPE="text" NAME="apellidos"><BR>
      <INPUT TYPE="submit" VALUE="Enviar">
    </FORM>
  </body>
</html>
```

Forms

- Controls: text input



A screenshot of a web form. It contains two text input fields. The first field is labeled "First name:" and contains the text "Mickey". The second field is labeled "Last name:" and contains the text "Mouse". Below the fields is a "Submit" button.

- The `<input>` tag defines the input fields in the form.
- Attributes:
 - `type=""` type of field.
 - `maxlength=""` indicates the maximum number of characters.
 - `size=""` maximum number of characters displayed on screen.
 - `value=""` initial value on this field.
 - `name=""` name to be referenced.

http://www.w3schools.com/tags/tag_input.asp

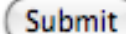
Forms

- Controls: special text inputs
 - These special types are useful for mobile devices



Forms

- Controls, buttons

A rounded rectangular button with a light gray background and a thin border, containing the word "Submit" in a dark gray sans-serif font.

- Defined by the `<input>` tag plus some attributes:

- type, followed by *submit* to send the data, or followed by *reset* to erase the data and leave the fields blank, or *button* to perform some action.

- value, to indicate the text inside a button (usually send or reset).

- name, to identify a button.

```
<input type="button" value="Cerrar esta  
ventana" onclick="window.close();">
```

http://www.w3schools.com/tags/tag_input.asp

Forms

- Controls, checkbox
 - To select one or some options from several choices. (main difference with radiobutton).
 - One option may be preselected (checked) when the page loads.

```
<INPUT TYPE="label" value="correcto">
```

```
<INPUT TYPE="checkbox" name="c1" value="1" onClick="if(this.checked == true){alert('verdadero!'); this.checked=false;}">
```

```
<INPUT TYPE="label" value="falso">
```

```
<INPUT TYPE="checkbox" name="c2" value="0" onClick="if(this.checked == true){alert('falso!'); this.checked=false;}">
```

correcto falso



http://www.w3schools.com/tags/tag_input.asp

Forms

- Controls, special inputs
 - There are some special inputs like:
 - **Password:** this field won't display the characters while typing. It will display asterisks instead.
 - **Hidden:** the value of this field can't be modified because the user can't see this field. It usually has a fixed value defined by the value attribute.
 - Their attributes are the same as for text.

```
<input type="password" name="clave" maxlength="5" size="6">
```



http://www.w3schools.com/tags/tag_input.asp

Forms

- Controls, radio

- Similar to checkbox, to select an option from some choices.
- Only one active field allowed.

```
<input type="radio" name="g1" value="Leche"> Leche<br>  
<input type="radio" name="g1" value="Mant"> Mantequilla<br>  
<input type="radio" name="g1" value="Queso" check="checked"> Queso
```

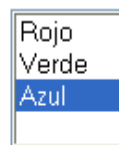
- Leche
- Mantequilla
- Queso

http://www.w3schools.com/tags/tag_input.asp

Forms

- Controls, select lists

- The tags `<select>.....</select>` hold the values to be selected from a list of choices. The attributes of the opening tag are:
 - `name=""` identifies the selection tag.
 - `Size=""` indicates the number of visible options. If it is 1, the selection will be a menu. If the value is greater than 1, the list will have a scroll bar.
 - `Multiple`: indicates multiple selection (the user can select more than one option).

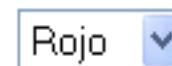


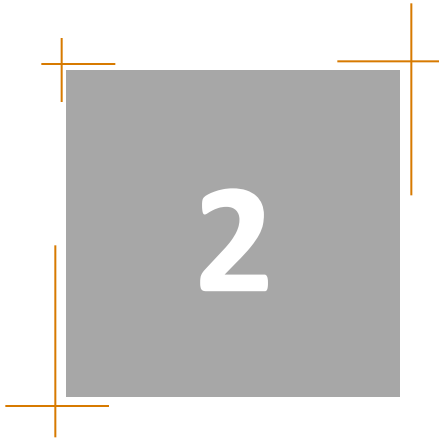
http://www.w3schools.com/tags/tag_select.asp

Forms

- Controls, select lists
 - The `<option>` tag indicates the different options.
 - The *selected* attribute, indicates the default option.
 - If it is not specified, it will be the first element on the list.

```
<SELECT NAME="Colores" MULTIPLE>  
<OPTION VALUE="r">Rojo</OPTION>  
<OPTION VALUE="g">Verde</OPTION>  
<OPTION VALUE="b">Azul</OPTION>  
</SELECT>  
<BR><BR>  
<SELECT NAME="Colores" SIZE="1">  
<OPTION VALUE="r">Rojo</OPTION>  
<OPTION VALUE="g">Verde</OPTION>  
<OPTION VALUE="b">Azul</OPTION>  
</SELECT>
```





Accessing form objects from JavaScript



Finding the object (using *this*)

- Names while creating the object:

```
<select name="var_name"  
  OnChange="function_name(this)">
```

```
</select>
```

- Obtaining the object within a JavaScript function:

```
function function_name(elem)  
{  
  alert(elem.value);  
}
```


Finding the object (using getElementById)

- Names while creating the object:

```
<select name="var_name"  
  id="object_name"  
  OnChange="function_name()">
```

```
</select>
```

- Finding any object within a JavaScript function:
var elem;
elem=document.getElementById("object_name");

Finding the object (using document)

- Names while creating the object:

```
<form action="save.php" name="form_name" >  
  <select name="var_name"  
    OnChange="function_name">  
  </select>  
</form>
```

- Accessing any object within a JavaScript function:

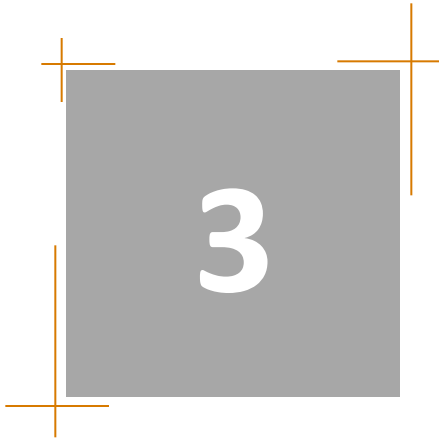
```
var elem=document.form_name.var_name;
```

Content of different object types

- In general the following are always valid:
 - elem.name → name of the variable
 - elem.value → value typed in by the user
- Radio and checkbox:

```
if (elem.checked) {  
    alert(elem.value); //use the value only if checked  
}
```
- Select-one:

```
selectValue = elem.options[elem.selectedIndex].text;
```



Some practical examples



Numeric Text field example

- JavaScript code:

```
function format(elem)
{
    var my_value;
    my_value=parseFloat(elem.value);
    elem.value=my_value.toFixed(2);
}
```

- HTML code:

Amount:

Submit confirmation (<form> event)

- JavaScript code

```
function AskConfirmation ()
{
    var agree;
    agree=confirm("Send data now?");
    if (agree) {
        //code to check data goes here
        return true;
    } else {
        return false ;
    }
}
```

- HTML code

```
<FORM
ACTION="http://www.iit.upcomillas.es/cgi-bin/test-cgi"
NAME="myform" OnSubmit="return AskConfirmation()" >
    Name: <INPUT NAME="first_name" SIZE="15"><BR>
    <INPUT TYPE="SUBMIT" VALUE="Submit Form">
</FORM>
```

Submit confirmation (button event)

- JavaScript code

```
function AskConfirmation()
{
    var agree;
    agree=confirm("Send data now?");
    if (agree) {
        //code to check data goes here
        return true;
    } else {
        return false ;
    }
}
```

- HTML code

```
<FORM
ACTION="http://www.iit.upcomillas.es/cgi-bin/test-cgi"
NAME="myform">
    Name: <INPUT NAME="first_name" SIZE="15"><BR>
    <INPUT TYPE="SUBMIT" VALUE="Submit Form"
        onClick="return AskConfirmation()" >
</FORM>
```



Escuela Técnica Superior de Ingeniería ICAI

Alberto Aguilera 25

28015 Madrid

Tel +34 91 542 28 00

Fax + 34 91 542 31 76

lwww.icaí.upcomillas.es

www.upcomillas.es

