



upcomillas *es*

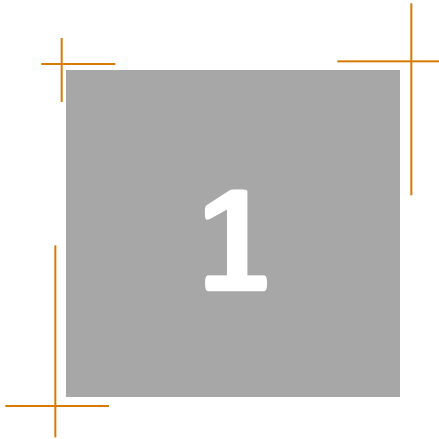
upcomillas *es*

2-Formularios y JavaScript

Course: Developing web-based applications

Cristina Puente, Rafael Palacios

2010-2011



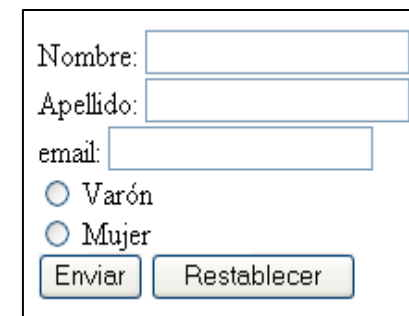
Creación de formularios



Formularios

- Un formulario HTML es una sección de un documento que contiene contenido normal, código, elementos especiales llamados *controles* (casillas de verificación (checkboxes), radiobotones (radio buttons), menús, 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

Formularios

- El elemento *form* actúa como contenedor de controles dentro de una página web.
- Atributos:

- Action: Programa que manejará el formulario completado y enviado (el atributo).
- Name: Nombre con el que se referirá al formulario en acciones posteriores. No es obligatorio pero conviene ponerlo!!!.
- Method: El método por el cual se enviarán los datos del usuario al servidor (el atributo). (Get por URL, Post por entrada estándar stdio)
- Accept-charset: Una codificación de caracteres que debe ser aceptada por el servidor para poder manejar este formulario

Formularios

- Métodos de envío de formularios:
 - get: Con el método HTTP "get", el conjunto de datos del formulario se agrega al URI especificado por el atributo action (con un signo de interrogación ("?") como separador) y este nuevo URI se envía al agente procesador.
 - post: Con el método HTTP "post", el conjunto de datos del formulario se incluye en el cuerpo del formulario y se envía al agente procesador.
 - El método "get" cuando no tiene efectos secundarios.
 - Si el servicio asociado con el procesamiento de un formulario causa efectos secundario (si el formulario modifica una base de datos), debería usarse el método "post".

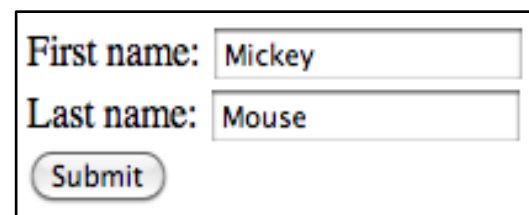
Formularios

```
<html>
<head>
  <title>Ejemplo de PHP</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>
```

Formularios

- Controles: 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 rounded rectangular button labeled "Submit".

- La etiqueta `<input>` define la introducción de variables.
- Junto a esta etiqueta solemos encontrar los siguientes atributos:
 - `type=""` Indicará el tipo de variable a introducir.
 - `maxlength=""` Seguido de un valor que limitará el número máximo de caracteres a introducir en ese campo.
 - `size=""` Seguido de un valor que limitará el número de caracteres a mostrar en pantalla.
 - `value=""` Indica que no hay valor inicial del campo.
 - `name=""` Indicará el nombre que se asigna a un determinado campo.

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

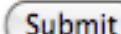
Formularios

- Controles: text inputs especiales
 - Estos tipos especiales son útiles para teléfonos móviles



Formularios

- Controles, botones

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

- Se definen mediante la tag `<input>` a la que le acompañan los atributos:
- `type`, Seguido de *submit* para enviar los datos del formulario y seguido de *reset* para borrar los datos que se han introducido, o *button* para ejecutar alguna acción.
- `value`, Indica el texto que incorporaran los botones. Normalmente, enviar y borrar
- `name`, Para identificar al botón

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

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

Formularios

- Controles, checkbox
 - El campo se elegirá marcando de entre varias opciones una casilla cuadrada. (Principal diferencia con radiobutton).
 - Puede haber una opción seleccionada por defecto (checked)
 - Se puede seleccionar más de una opción.

```
<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

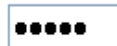


http://www.w3schools.com/tags/tag_input.asp

Formularios

- Controles, input especiales
 - Tipos especiales de inputs:
 - **Password** Indica que el campo a introducir será una palabra de paso. Mostrará asteriscos en lugar de letras escritas.
 - **hidden** El visitante no puede modificar su valor ya que no está visible. Se manda siempre junto al atributo value= seguido de su valor entre comillas.
 - Sus atributos son los mismos que para text.

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



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

Formularios

- Controles, radio
 - El campo se elegirá marcando de entre varias opciones una casilla circular.
 - Sólo puede tener una casilla activa.

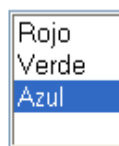
```
<input type="radio" name="g1" value="Leche"> Leche<br>  
<input type="radio" name="g1" value="Mant"> Mantequilla<br>  
<input type="radio" name="g1" value="Queso" check="checked"> Queso
```

- Leche
- Mantequilla
- Queso

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

Formularios

- Controles, listas desplegables
 - Las etiquetas `<select>....</select>` encierran los valores que podremos elegir a partir de una lista. Los atributos que acompañan a la etiqueta de apertura son:
 - `name=""` Indicará el nombre del campo de selección.
 - `Size=""` Indicará el número de opciones visibles. Si le asignamos 1, la selección se presentará como un menú desplegable. Si le asignamos un valor mayor se presentará como una lista con barra de desplazamiento.
 - `múltiple` indica si se pueden realizar múltiples selecciones.

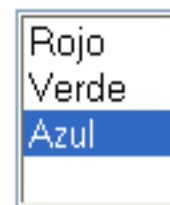


http://www.w3schools.com/tags/tag_select.asp

Formularios

- Controles, listas desplegables
 - Las diferentes opciones de la lista se indicarán mediante la etiqueta `<option>`
 - Puede acompañarse del atributo *selected* para indicar cual es la opción que aparecerá por defecto.
 - Si no lo especificamos, siempre será la primera de la lista

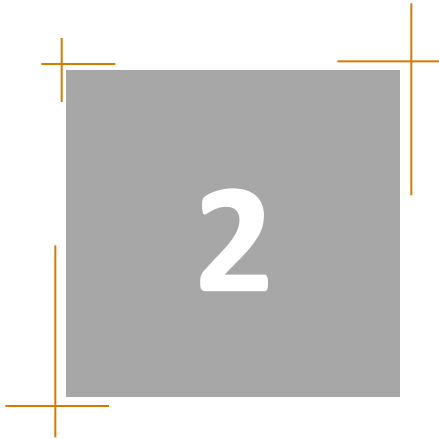
```
<SELECT NAME="Colores" MULTIPLE>  
  <OPTION VALUE="r">Rojo</OPTION>  
  <OPTION VALUE="g">Verde</OPTION>  
  <OPTION VALUE="b">Azul</OPTION>  
</SELECT>  
<BR><BR>  
<SELECT NAME="Colores" SIZE="1">  
  <OPTION VALUE="r">Rojo</OPTION>  
  <OPTION VALUE="g">Verde</OPTION>  
  <OPTION VALUE="b">Azul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



A screenshot of a web form showing a multiple-choice select box. The box contains three options: 'Rojo', 'Verde', and 'Azul'. The 'Azul' option is highlighted with a blue background, indicating it is the selected option.



A screenshot of a web form showing a single-choice dropdown menu. The box contains the text 'Rojo' and a small downward-pointing arrow on the right side, indicating it is a dropdown menu.



Acceso desde JavaScript a los objetos de los formularios



Localizar el objeto (utilizando *this*)

- Nombres en la creación del objeto:

```
<select name="nombre_var"  
  OnChange="nombre_funcion(this)">
```

```
</select>
```

- Obtener el objeto dentro de la función JavaScript:

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


Localizar el objeto (utilizando getElementById)

- Nombres en la creación del objeto:

```
<select name="nombre_var"  
        id="nombre_objeto"  
        OnChange="nombre_funcion()">
```

```
</select>
```

- Obtener el objeto dentro de la función JavaScript:

```
var elem;  
elem=document.getElementById("nombre_objeto");
```

Localizar el objeto (utilizando document)

- Nombres en la creación del objeto:

```
<form action="save.php" name="form_name" >  
  <select name="nombre_var"  
    OnChange="nombre_funcion()">  
  </select>  
</form>
```

- Obtener el objeto dentro de la función JavaScript:

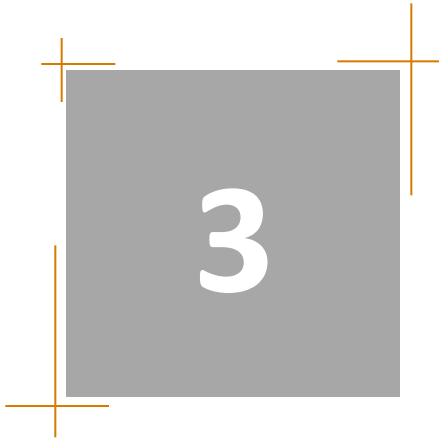
```
var elem;  
elem=document.form_name.nombre_var;
```

Contenido de diferentes tipos de objetos

- Generalmente los siguientes campos son válidos:
 - elem.name → nombre de la variable
 - elem.value → valor introducido por el usuario
- Radio y checkbox:

```
if (elem.checked) {  
    alert(elem.value); //sólo se utiliza el valor si está seleccionado  
}
```
- Select-one:

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



Ejemplos prácticos



Ejemplo de campo de texto numérico

- Código JavaScript:

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

- Código HTML:

```
Cantidad: <input onChange="formatea(this)" name="cantidad"
value="1.00">&euro;
```

Confirmación de envío (<form> event)

- Código JavaScript

```
function PedirConfirmacion()
{
    var agree;
    agree=confirm("Enviar ahora?");
    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="miform" OnSubmit="return PedirConfirmacion ()" >
    Nombre: <INPUT NAME="nombre" SIZE="15"><BR>
    <INPUT TYPE="SUBMIT" VALUE="Enviar datos">
</FORM>
```

Confirmación de envío (evento de botón)

- Código JavaScript

```
function PedirConfirmacion()
{
    var agree;
    agree=confirm("Enviar ahora?");
    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="miform">
    Nombre: <INPUT NAME="nombre" SIZE="15"><BR>
    <INPUT TYPE="SUBMIT" VALUE="Enviar datos"
        onClick="return PedirConfirmacion ()" >
</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

