FORMULARIO

# 1. Introducción a los formularios

Los formularios son una de las herramientas de que disponemos a la hora de hacer nuestras páginas Web interactivas, en este sentido nos permiten por ejemplo:

* recopilar información de la persona que ve la pagina
* procesarla y responder a ella

De esta forma podremos responder adecuadamente a sus acciones o peticiones.

El proceso comienza con la creación en nuestra página de un formulario de entrada de datos, que va a contener diversos campos diferentes en su funcionalidad y que nos van a permitir recopilar toda aquella información que deseemos de la persona que ve la pagina. Estos datos, una vez completado el formulario, serán enviados normalmente *al servidor para su procesamiento o a nosotros directamente, mediante correo electrónico.*

Para poder procesar la información recibida mediante un formulario necesitaremos un programa especial, que normalmente se aloja en el servidor Web, llamado CGI (Common Gateway Interface), que suele estar escrito en Perl o en C++, de los cuales disponemos de gran cantidad de ellos en paginas especializadas de Internet. También es posible procesar esa información mediante páginas ASP (Active Server Pages), JSP (Java Server Pages) o PHP (PHP: Hypertext Pre-processor), que son páginas que contienen dentro de su código HTML trozos de código escrito en Visual Basic Script, Java o PHP respectivamente. Mediante alguno de estos métodos podremos incorporar los datos recibidos del formulario a una base de datos, crear un registro, enviar un mensaje automático de respuesta al usuario, validar un pedido, confirmar el acceso a un sistema remoto, etc.

Usando únicamente HTML sólo podremos enviar el formulario a un correo electrónico. La información se recibirá en bruto y tendrá que procesarse manualmente.

Una vez procesada la información recibida, podremos enviar una página de respuesta definida a nuestro visitante, con lo que podremos personalizar esta respuesta de antemano.

Los formularios están formados por una serie de controles distintos, cada uno de los cuales esta asociado a un tipo concreto de datos o una acción predeterminada.

## Inserción de formularios (<FORM>)

Para introducir un formulario en una página Web se utilizan las etiquetas **<FORM>** .. **</FORM>**. Estas etiquetas son las que delimitan el comienzo y el final de un formulario y se introducirán dentro del BODY de la página, en el lugar en que queramos que nos aparezcan los elementos del formulario.

Entre estas etiquetas se introducirán los elementos que sean necesarios para configurar el formulario, tales como campos para introducir texto, botones, menús desplegables, etc.

Los atributos que posee esta etiqueta son:

* **NAME**

Asigna un nombre al formulario de modo que se pueda hacer referencia a él desde hojas de estilo o scripts.

* **ACTION**

Indicará al navegador que hacer con la información que el usuario rellena una vez que éste pulse el botón de envío. Puede tomar los valores:

* **URL (ruta del programa)**

Esta URL indicará la dirección del programa que se va a encargar de procesar las variables que se envíen con los datos del formulario.

* **mailto:direccióndecorreo**

Esta dirección de correo será la que reciba los datos del formulario. Hay que tener en cuenta que los datos se recibirán como se indicó anteriormente.

Si se opta por esta opción, hay que añadir obligatoriamente el atributo del formulario **ENCTYPE** con el valor “**text/plain**” (SIN CODIFICACIÓN), para que resulte legible lo que recibamos en la dirección de correo electrónico

* **METHOD**

Especifica el método HTTP que se empleará para enviar los datos introducidos en el formulario. Los valores posibles (que no distinguen entre mayúsculas y minúsculas) son:

* GET

Envía los datos del formulario en la cabecera de la petición de la página, utilizando el método GET estamos limitados en la cantidad de datos a enviar. Los datos son visibles en la barra del navegador.

(Normalmente los servidores Web tienen limitada la longitud de las direcciones URL, por lo que con el método GET no puede enviarse una cantidad de datos exagerada.

POST

Envía los datos del formulario de forma independiente(los datos se envían con el cuerpo del formulario). Los datos se envían de forma transparente sin hacerla visible al usuario. Soporta más longitud.

* **ENCTYPE**

Especifica el mecanismo de codificación (tipo MIME) que se usará en la transferencia de los datos del formulario.

Sólo se aplicará si el atributo **METHOD** del formulario es igual a POST.

El contenido del formulario llega al servidor como un flujo de información (sea cual sea el método utilizado: GET o POST). El servidor es capaz de reconocer ciertos tipos de datos (como imágenes, vídeos, sonidos, etc.) gracias a que tiene configurados los tipos MIME (o tipos de medios).

MIME adjunta a cada fichero un archivo de cabecera donde se indica el tipo y el subtipo del contenido de los datos del mismo. Esta configuración de tipos MIME indica al servidor cómo debe tratar cada tipo de información.

Los principales tipos Mime soportados por los navegadores Web son:

|  |  |
| --- | --- |
| *Tipo MIME* | *Extensión* |
| ***Imagen*** | |
| image/bmp image/x-windows-bmp image/gif image/jpeg image/jpeg image/png | .bmp, .bm .bmp .gif .jpe .jpg .png |
| **Sonido** | |
| audio/basic audio/x-au audio/midi audio/x-midi audio/x-wav audio/mod audio/x-mod audio/mpeg3 audio/x-mpeg-3 audio/x-pn-realaudio audio/x-pn-realaudio | .au, .snd .au .mid, .midi .mid, .midi .wav .mod .mod .mp3 .mp3 .ra, .ram .ra, .ram |
| **Video** | |
| video/avi video/x-motion-jpeg video/quicktime video/mpeg application/x-shockwave-flash | .avi .mjpg .mov .mpg .swf |

**Mediante el atributo ENCTYPE indicamos al servidor qué tipo de datos le estamos enviando a través del formulario. Algunos valores posibles son:**

* application/x-www-form-urlencoded

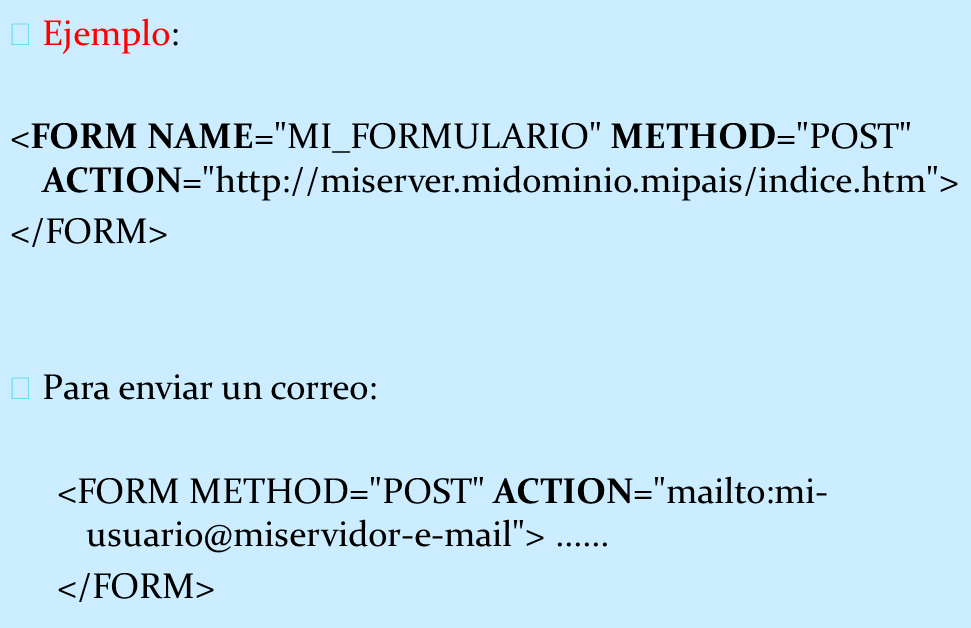
Es el valor que se aplica por defecto, pero no es válido cuando se envían archivos junto con el formulario.

* multipart/form-data

Indica al servidor que el formulario contiene distintos tipos de datos (como textos, imágenes, etc.) y le advierte para que los identifique mediante los tipos MIME que tenga configurados.

* text/plain

El contenido del formulario se envía al servidor como texto plano, sin ningún tipo de formato. Esta opción suele ser interesante cuando la acción del formulario es enviarse por correo electrónico en lugar de ser procesado por un agente de procesamiento.



## Inserción de controles en formularios

Los usuarios introducen datos en los formularios mediante los controles.

Una cualidad interesante de los formularios HTML es que están perfectamente integrados con ciertos lenguajes de programación, como JavaScript, de modo que no tenemos que esperar al final para procesar todo el formulario de una vez, sino que podemos crear formularios dinámicos que pueden interactuar con el usuario a medida que este los rellena (por ejemplo, indicándole que ha introducido un número de tarjeta de crédito no válido, o recordándole que se ha olvidado de rellenar algún campo).

Dentro de un formulario se pueden incluir tres tipos de controles para introducir los datos:

* **Campos de entrada**
* **Listas desplegables de opciones**
* **Áreas de texto multilínea**

### 1.2.1. Campos de entrada (<INPUT>)

Para la introducción de datos se utiliza la etiqueta **<INPUT>**, que permite definir gran variedad de tipos de campos de entrada. Los atributos de posee son:

* **NAME**

Asigna un nombre o identificador a la variable donde se guardará la información que introduzca el usuario, para poder ser tratada posteriormente por un programa o al recibirlo por correo electrónico.

También sirve para identificar o interactuar con el control adecuado desde los scripts de la propia página.

Es obligatorio excepto en los tipos **“submit” y “reset”.**

* **VALUE**

Asigna un valor por defecto o inicial, que aparecerá en campo de texto o selección. Es opcional en los campos de texto, pero es muy recomendable en los campos de selección

* **TYPE**

Indica el tipo de campo de entrada que crearemos. Según el valor asignado admitirá unos atributos adicionales.

#### 1.2.1.1. Tipo TEXTO SIMPLE (<INPUT TYPE=”TEXT”>)

Nos permite introducir un tipo de texto corto de una sola línea. Es el tipo por defecto del atributo TYPE.

Se muestra como un rectángulo y no se puede sobrepasar la longitud de una línea.

Los atributos adicionales son:

* **MAXLENGTH**

Permite indicar mediante un número el máximo de caracteres a introducir en la caja de texto. Si el número es mayor que el tamaño de la caja (especificado con **SIZE**), el texto se desplazará dentro del rectángulo.

* **SIZE**

Indica mediante un número la longitud visible (medida en caracteres) de la caja de texto en pantalla( longitud de la caja).

* **DISABLED**

Cuando se asigna a un control de un formulario deshabilita el control para la entrada de datos por parte del usuario.

Cuando se establece tiene los siguientes efectos sobre un control:

* + - No se puede dirigir el foco hacia controles deshabilitados
    - En el orden de tabulación, se salta por encima de los controles deshabilitados
    - Los controles deshabilitados no se envían al servidor
* **READONLY(yes/no)**

Cuando se asigna a un control de un formulario impide que haya cambios en el control. Especifica que el control no puede ser modificado por el usuario.

**Valores:**

**Readonly=”yes” es lo mismo readonly=”readonly”**

**Readonly=”no” o readonly**

Cuando se establece tiene los siguientes efectos sobre un control:

* + - Se puede dirigir el foco hacia el control, pero éste no pueden ser modificados por el usuario
    - Los controles de sólo lectura están incluidos en la navegación con tabulador
    - Los elementos de sólo lectura se envían al servidor, a diferencia de los deshabilitados

También pueden aparecer los argumentos **class e id que nos servirá para aplicar hojas de estilos**

Ejemplo:

<html>

<head></head>

<body>

<FORM>

Nombre:<INPUT name="nombre" type="text" size="20" maxlength= "5" >

</FORM>

</body>

</html>

|  |  |
| --- | --- |
| <**FORM**>  Nombre:<**INPUT** **name**="nombre" **type**="**text**">  <**/FORM**> | Nombre: |

#### 1.2.1.2. Tipo CONTRASEÑA (<INPUT TYPE=”PASSWORD”>)

Es un campo de tipo texto similar al anterior con los mismos atributos. La diferencia es que el texto introducido en la caja aparece oculto, mostrando un símbolo \* (asterisco) por cada carácter en el navegador. Como su nombre indica, sirve para introducir contraseñas y que nadie de nuestro alrededor pueda ver lo que se escribe.

Ejemplo:

<html>

<head></head>

<body>

<FORM>

Nombre:<INPUT name="nombre" type="text" size="20" maxlength= "5" >

Contraseña:<INPUT name="contraseña" type="password" maxlength= "3" size="5" >

</FORM>

</body>

</html>

|  |  |
| --- | --- |
| <**FORM**>  Contraseña:<**INPUT** **name**="contraseña" **type**="**password**">  <**/FORM**> | Contraseña: |

**EJEMPLO:**

**Los controles de tipo text y password pueden limitar la cantidad de caracteres que puede introducir el usuario a partir de la propiedad maxlength.** Debemos asignarle un valor entero que indica hasta cuantos caracteres está permitido ingresar. **No hay que confundir el objetivo de la propiedad size con maxlength.** **Con la propiedad size solo indicamos la cantidad máxima de caracteres a mostrar dentro del control antes de hacer scroll de los datos. Ejemplo:**formulario que introduzca el nombre de usuario y su clave y solo permita introducir nombres de usuarios de hasta 20 caracteres y claves de hasta 12.

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

**<form action="/registrardatos.php" method="post">**

Introduzca su nombre:

<input type="text" name="nombre" maxlength="20" size="20">

<br>

Introduzca su clave:

<input type="password" name="clave" maxlength="12" size="12">

<br>

<input type="submit" value="enviar">

</form>

</body>

</html>

Cuando se ejecute esta página no podremos cargar un nombre de usuario de más de 20 caracteres, el teclado se inhabilita cuando se han introducido 20 caracteres en el primer control.

#### EJEMPLO:

#### <html>

#### <head>

#### <title>Prueba de formulario</title>

#### </head>

#### <body>

#### <form action="mailto:usuario@hotmail.com method="post">

#### Introduzca su nombre:

#### <input type="text" name="nombre" size="12" >

#### <br>

#### Introduzca su año de nacimiento:

#### <input type="text" name="nombre" size="4" DISABLED>

#### <br>

#### Introduzca su clave:

#### <input type="password" name="clave" size="6">

#### <br>

#### <input type="submit" value="enviar">

#### <input type="reset" value="borrar">

#### </form>

#### </body>

#### </html>

**<KEYGEN> HTML5**

El <keygen> especifica un campo generador de par de claves utilizado para el formulario. Cuando se envía el formulario, la clave privada se almacena localmente, y la clave pública se envía al servidor.

El atributo **keytype** específica un tipo de clave a utilizar.  El soporte para este tipo de clave puede variar entre los navegadores.

**<keygen keytype="rsa|dsa|ec">**

<!DOCTYPE html>

<html>

<body>

<form action="demo\_keygen.asp" method="get">

usuario: <input type="text" name="usr\_name">

Tipo de encriptación: <keygen name="security" keytype="rsa">

<input type="submit">

</form>

<h1> prueba de keygen y su atributo keytype<h1>

</body>

</html>

#### 1.2.1.3. La etiqueta LABEL<LABEL>

El elemento <**labe**l> sirve para asociar un campo de texto con un objeto de edición.

Hay dos formas de realizar la asociación:

**La primera** consiste en anidar el elemento de edición dentro del contenido del elemento label.

**<label>Dirección:<br/><input type “text” name=”dirección” size=”80” maxlength=”120”/></label>**

**La segunda** consiste en identificar ambos objetos mediante el valor del argumento **for** del element **label** y el valor del elemento **id** del elemento de edición

**<label for =”textdir”>Dirección</label><br/><input type “text” name=”dirección” size=”80” maxlength=”120” id=”textdir”/>**

#### 1.2.1.4. Tipo CASILLA DE VERIFICACIÓN (<INPUT TYPE=”CHECKBOX”>)

Es un campo tipo casilla cuadrangular de verificación que puede tomar dos estados: activada (ON) o desactivada (OFF). Los controles de tipo checkbox sólo son enviados cuando están activados. En ese caso el valor por defecto es "On", pero se puede modificar empleando el atributo **value**, asignándole el valor que queramos que se envíe al estar seleccionado.

Los atributos adicionales que posee son:

* **CHECKED**

Indica que la casilla de verificación aparecerá activada se carga el formulario.

* **DISABLED**

Tiene la misma función que la explicada anteriormente.

**Ejemplo:**

<!--checkbox-->

Seleccione sus aficiones <br>

<INPUT name="MB" type="checkbox" value="Sí">Bicicleta de montaña <br>

<INPUT name="FO" type="checkbox" value="Sí">Fotografía<br>

<INPUT name="ES" type="checkbox" value="Sí" checked>Escalada<br>

|  |  |
| --- | --- |
| <**FORM**>  Seleccione sus aficiones  <**INPUT** **name**="MB" **type**="**checkbox**" **value**="Sí">Bicicleta de montaña  <**INPUT** **name**="FO" **type**="**checkbox**" **value**="Sí">Fotografía  <**/FORM**> | Seleccione sus aficiones  Bicicleta de montaña Fotografía |

EJEMPLO

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

<p>¿desearía más informacion sobre ?</p>

<blockquote>

<p>

<input type= "checkbox" name="opcion1" value ="on" > Material audiovisual<br>

<input type= "checkbox" name="opcion2" value ="on" > textos electronicos <br>

<input type= "checkbox" name="opcion3" value ="on" checked> libros de texto <br>

<input type= "checkbox" name="opcion4" value ="on" > diseños curriculares <br>

<input type= "checkbox" name="opcion5" value ="on" > Legislación educativa <br>

</p>

</blockquote>

</form>

</body>

</html>

#### 1.2.1.5. Tipo CASILLA DE BOTÓN DE OPCIÓN (<INPUT TYPE=”RADIO”>)

Es un campo tipo casilla circular que permite seleccionar **únicamente una** de las opciones posibles mostradas en el formulario. Se emplea cuando sólo una de las opciones es posible (p.e hombre/mujer). Al seleccionar la opción, pulsando en el interior del círculo asociado al texto, su contenido aparecerá seleccionado.

El **atributo name** debe tener el **mismo valor** para todos aquellos que estén relacionados y compongan las opciones posibles para una misma propiedad.

Los atributos adicionales que posee son:

* **CHECKED**

Indica cual del conjunto de radio-botones aparecerá seleccionada cuando se carga el formulario.

Si se incluye este atributo en más de uno del conjunto de radio-botones, sólo se mostrará seleccionado el último.

* **DISABLED**

Tiene la misma función que la explicada anteriormente.

Ejemplo:

<!--radio-->

Indique su profesión:<br<

<INPUT name="profesión" type="radio" value="ES" CHECKED>Estudiante<br>

<INPUT name="profesión" type="radio" value="FU">Funcionario<br>

<INPUT name="profesión" type="radio" value="DE">Desempleado<br>

|  |
| --- |
| <**FORM**>  Indique su profesión:  <**INPUT** **name**="profesión" **type**="**radio**" **value**="ES" **CHECKED**>Estudiante  <**INPUT** **name**="profesión" **type**="**radio**" **value**="FU">Funcionario  <**INPUT** **name**="profesión" **type**="**radio**" **value**="DE">Desempleado  <**/FORM**> |
| Indique su profesión:  Estudiante Funcionario Desempleado |

EJEMPLO:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Introduzca su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione el máximo nivel de estudios que tiene:

<br>

<input type="radio" name="estudios" value="1" DISABLED>Sin estudios<br>

<input type="radio" name="estudios" value="2">Primario<br>

<input type="radio" name="estudios" value="3" CHECKED>Secundario<br>

<input type="radio" name="estudios" value="4">Universitario<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

#### 1.2.1.6. Tipo OCULTO (<INPUT TYPE=”HIDDEN”>)

Es un campo del formulario que no se visualizará en el navegador, pero que si se envía con los datos del mismo (el valor que se envía es el asignado en el atributo **value**, y será asociado al campo cuyo nombre se indique en el atributo **name**).

Este tipo de campo se utiliza cuando es preciso enviar información en los formularios sin que el usuario tenga que introducir dicha información.

Ejemplo:

<**FORM**>

<**INPUT** **name**="sesion" **type**="**hidden**" **value**="A154">

<**/FORM**>

EJEMPLO:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

<input type="hidden" value="10:20" name="hora">

Introduzca su nombre:

<input type="text" name="nombre" size="30">

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Como vemos la sintaxis para definir un campo oculto es: **<input type="hidden" value="10:20" name="hora">** En el atributo value almacenamos el valor (este valor no se puede modificar desde el formulario). Podemos imaginar una utilidad, supongamos que queremos que un visitante reenvíe un formulario cargado con todos los datos en un tiempo limitado, podríamos almacenar en el campo oculto la primera vez que solicita la página que contiene el formulario donde se registrará dicha hora. Luego al subir el formulario ya cargado al servidor controlaremos mediante un lenguaje de servidor si la hora actual y la hora de pedido del formulario no hacen invalidar los datos.

#### 1.2.1.7. Tipo BOTÓN DE ENVIO DE FORMULARIO (<INPUT TYPE=”SUBMIT”>)

Es un campo que se utiliza para realizar el envío de datos del formulario, para su procesamiento por el programa o recurso especificado en el atributo **action**.

Mediante el atributo **value** se le puede asignar una leyenda al botón.

Los atributos adicionales que posee son:

* **DISABLED**

Tiene la misma función que la explicada anteriormente.

**Ejemplo:**

<**FORM action**="mailto:destino@url.com" **method**="post" **enctype**="text/plain">

Nombre:<**INPUT** **name**="nombre" **type**="**text**">

<**INPUT** **name**="botonenvio" **type**="**submit**" **value**="Enviar por e-mail">

<**/FORM**>



#### Otra forma de poner un botón:

#### <button type= ”submit” >enviar</button>

Ambas formas de ponerlo se comportan igual. La diferencia está en que el texto en la primera forma, aparece como valor del argumento value, Y no podemos aplicarle estilos. En la segunda forma al ser un contenido del elemento si se podría aplicar estilos

#### 1.2.1.8. Tipo BOTÓN DE REINICIO DE FORMULARIO (<INPUT TYPE=”RESET”>)

Es un campo que se utilizar para reiniciar el formulario, es decir, eliminar el contenido de todos los valores introducidos o seleccionados, volviendo a los valores por defecto que tenía el formulario al cargarse.

Mediante el atributo **value** se le puede asignar una leyenda al botón (por defecto tendrá el valor "Reset").

Ejemplo:

<**FORM action**="mailto:destino@url.com" **method**="post" **enctype**="text/plain">

Nombre:<**INPUT** **name**="nombre" **type**="**text**">

<**INPUT** **name**="borrar" **type**="**reset**" **value**="Limpiar datos">

<**/FORM**>

<html>

<head></head>

<body>

<FORM action="mailto:destino@url.com" method="post" enctype="text/plain">

<!--text-->

Nombre:<INPUT name="nombre" type="text" size="20" maxlength= "5" >

<!--password-->

Contraseña:<INPUT name="contraseña" type="password" maxlength= "3" size="5" ><br><br>

<!-- submit-->

<INPUT name="botonenvio" type="submit" value="Enviar por e-mail"><br>

<!--reset-->

<INPUT name="borrar" type="reset" value="Limpiar datos">

</form>

</html>

EJEMPLO:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="registrardatos.php" method="post">

Introduzca su nombre:

<input type="text" name="nombre" size="20">

<br>

<input type="submit" value="enviar">

</form>

</body>

</html>

Veamos detenidamente la estructura de un formulario elemental, lo primero la apertura de la marca form donde debemos definir dos propiedades (**action y method**):

<form action="/registrardatos.php" method="post">

La propiedad **action** **se inicializa con el nombre de la página que procesará los datos en el servidor.** Como no conocemos la sintaxis de un lenguaje de servidor almacené en la raíz de este sitio una página codificada en PHP que recibe los datos del formulario y los imprime en una nueva página. La segunda propiedad que debemos inicializar es **method**. Esta propiedad puede almacenar únicamente dos valores (**post o get**). **Normalmente un formulario se envía mediante post** (los datos se envían con el cuerpo del formulario) **En caso de utilizar get** los datos se envían en la cabecera de la petición de la página, utilizando el método get estamos limitados en la cantidad de datos a enviar, no así con el método post. Ahora veamos el cuadro de texto donde se introduce el nombre:

Introduce su nombre:

<input type="text" name="nombre" size="20"><br>

<input type=”text” name =”email” size “25”>

<br>

El mensaje "Introduzca su nombre:" es un texto fijo. El elemento **input permite definir un cuadro de texto (editor de línea) si asignamos a la propiedad type el valor "text".**

Todo cuadro de texto debe inicializar la propiedad **name (asigna un nombre a cada control) con un nombre para el cuadro de texto**. Este es un dato fundamental para poder recuperar el dato introducido en el servidor (por ejemplo mediante PHP)

Por último inicializamos la propiedad **size** con el valor 20, esto significa que el cuadro de texto se dimensiona para permitir mostrar 20 caracteres (no se limita la cantidad de caracteres a introducir por parte del visitante sino la cantidad de caracteres que se pueden visualizar). <input type="submit" value="enviar">

También mediante el elemento **input definimos un botón para el envío de datos al servidor. Debemos inicializar la propiedad type con el valor submit**, con esto ya tenemos un botón para el envío de datos.

La propiedad value almacena la etiqueta que debe mostrar el botón.

Finalmente cerramos el formulario:

</form>

#### 1.2.1.9. Tipo BOTÓN DE ACCIÓN GENERICO (<INPUT TYPE=”BUTTON”>)

Es un campo tipo botón pulsador de uso genérico al que se pueden asociar distintos eventos que son manejados desde los scripts. Un uso muy frecuente consiste en asociar a la pulsación del botón (evento OnClick) la llamada a una función que comprueba la coherencia y validez de los datos introducidos en el formulario.

Mediante el atributo **value** se le puede asignar una leyenda al botón.

Los atributos adicionales que posee son:

* **DISABLED**

Tiene la misma función que la explicada anteriormente.

**Ejemplo:**

<**FORM action**="mailto:destino@url.com" **method**="post" **enctype**="text/plain">

<**INPUT** **type**="**button**" **value**="Texto del botón" **onclick**="código de script">

<**/FORM**>

EJEMPLO:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

<h1>Cuál es la velocidad de la luz?</h1>

300000 Km por segundo.<input type="button" name="boton1" value="Correcta?">

<br>

200000 Km por segundo.<input type="button" name="boton2" value="Correcta?">

<br>

100000 Km por segundo.<input type="button" name="boton3" value="Correcta?">

</form>

</body>

</html>

EJEMPLO: CALCULADORA

El tipo de botón que podemos crear es mediante el **elemento input y fijando en la propiedad type el valor "button".** Este tipo de botón no se le puede hacer que actúe como los botones de tipo submit o reset, su actividad dependerá de un programa desarrollado generalmente en JavaScript. Si bien no podemos ver su funcionamiento ya que no conocemos JavaScript, si podemos implementar una página que muestre este control. Ejemplo página que muestre el teclado de una calculadora:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="mailto:destino@url.com" **method**="post" **enctype**="text/plain">

<h1>Resultado:<h1>

<input type="button" name="boton7" value=" 7 ">

<input type="button" name="boton8" value=" 8 ">

<input type="button" name="boton9" value=" 9 ">

<input type="button" name="botondiv" value=" / ">

<br>

<input type="button" name="boton4" value=" 4 ">

<input type="button" name="boton5" value=" 5 ">

<input type="button" name="boton6" value=" 6 ">

<input type="button" name="botondiv" value=" \* ">

<br>

<input type="button" name="boton1" value=" 1 ">

<input type="button" name="boton2" value=" 2 ">

<input type="button" name="boton3" value=" 3 ">

<input type="button" name="botonmas" value=" + ">

<br>

<input type="button" name="boton0" value=" 0 ">

<input type="button" name="botonigual" value=" = ">

<input type="button" name="botonmenos" value=" - ">

</form>

</body>

</html>

Vemos que cuando se presiona el botón no sucede nada. Esto es así porque no hemos asignado ninguna actividad cuando sea presionado. Recordemos que HTML solo tiene Contenido, si queremos funcionalidad deberemos definir los eventos para dichos botones.

#### EJEMPLO COMPLETO

#### <HTML>

#### <HEAD>

#### <TITLE>Libro de Visitas</TITLE>

#### </HEAD>

#### <BODY>

#### <H2>Libro de visitas</H2>

#### <FORM ACTION=“mailto:asir.dam\_smr@yahoo.es" METHOD=“post” ENCTYPE=“text/plain”>

#### Nombre

#### <BR><INPUT TYPE="text" NAME="Nombre" size="30" maxlength="50">

#### <p>

#### Contraseña

#### <BR><INPUT TYPE="password" NAME="Contraseña" size="10" maxlength="10">

#### <p>

#### ¿Registrado?

#### <INPUT TYPE="checkbox" NAME="Registrado">

#### <p>

#### Sexo:

#### Hombre<INPUT TYPE="radio" NAME="Sexo" value="Hombre">

#### Mujer<INPUT TYPE="radio" NAME="Sexo" value="Mujer">

#### <p>

#### Adjunta tu fichero<INPUT TYPE="file" NAME="fichero">

#### <BR>

#### Escribe tus comentarios:

#### <BR>

#### <TEXTAREA NAME="Comentarios" ROWS="6" COLS="40">

#### </TEXTAREA>

#### <p>

#### <INPUT TYPE="button" NAME="saludo" VALUE="Saludar" onClick="alert('!HOLA!')">

#### <INPUT TYPE="submit" NAME="enviar" VALUE="Enviar datos">

#### <INPUT TYPE="reset" NAME="borrar" VALUE="Borrar datos">

#### <INPUT TYPE="image" NAME="boton" SRC="" ALT="Imagen del Botón">

#### <INPUT TYPE="hidden" NAME="Opcion" VALUE="SI">

#### <BR>

#### <BUTTON name="reset" value="reset" type="reset"> Limpiar Formulario <IMG SRC="Imagenes/boton-03.gif" ALT="Limpiar Formulario" width="32" height="30" align="middle">

#### <BUTTON name="submit" value="submit" type="submit"> Enviar Datos <IMG SRC="Imagenes/boton-17.gif" ALT="Enviar Datos del Formulario" width="32" height="30" align="middle">

#### <BUTTON name="button" value="button" onClick="alert('!HOLA!')" type="button">Saludar <IMG SRC="Imagenes/boton-21.gif" ALT="Saludar" width="32" height="30" align="middle">

#### </FORM>

#### <P>

#### <HR>

#### </BODY>

#### </HTML>

#### 

#### 1.2.1.10. Tipo BOTÓN DE ENVÍO CON IMAGEN (<INPUT TYPE=”IMAGE”>)

Se utilizan para personalizar el aspecto visual del botón de envío de un formulario, generando un botón de envío basándose en una imagen.

Su funcionamiento es análogo al de submit, y únicamente añade información adicional relativa a las coordenadas ‘x’ e ‘y’ del clic del usuario dentro del botón.

Los atributos adicionales que posee son:

* **SRC**

Especifica la ruta de la imagen que decorará el botón

* **WIDTH**

Especifica el ancho en pixels de la imagen

* **HEIGHT**

Especifica el alto en pixels de la imagen

* **ALIGN**

Especifica la posición del texto que rodea el botón respecto a este. Sus posibles valores son: **left** / **center** / **right** / **top** / **bottom** / **middle**.

* **HSPACE**

Especifica el espacio horizontal entre el botón y el texto

* **VSPACE**

Especifica el espacio vertical entre el botón y el texto

* **ALT**

Especifica un texto alternativo para la imagen.

* **DISABLED**

Tiene la misma función que la explicada anteriormente.

Ejemplo:

<**FORM action**="mailto:destino@url.com" **method**="post" **enctype**="text/plain">

<**INPUT** **name**="BotonImagen" **type**="**IMAGE**" **src**="images1.jpg" width="50" height="20" hspace="10" align="middle">

<**/FORM**>

#### 1.2.1.11. Tipo ENVÍO DE FICHEROS (<INPUT TYPE=”FILE”>)

Define un elemento que permite al usuario seleccionar un fichero que será enviado al servidor junto con el resto del formulario. Su apariencia consiste en una caja de texto, igual a las de tipo text, y un botón con una leyenda (habitualmente "Examinar") que muestra un cuadro de diálogo para seleccionar el archivo local que se desea adjuntar. Para poder emplear este elemento, es preciso que el atributo enctype del formulario se defina con el valor "**multipart/form-data**".

Los atributos adicionales que posee son:

* **TITLE**

Especifica un texto que aparece al situar el cursor encima del botón de examinar (sólo en Internet Explorer)

* **ACCEPT**

Permite especificar el tipo de archivo que se admite para enviar

* **SIZE**

Especifica la anchura de la caja de texto asociada

* **DISABLED**

Desactiva tanto el botón como la caja de texto, impidiendo al usuario seleccionar un archivo.

EJEMPLO:

**El control de tipo file nos permite enviar un archivo al servidor**. HTML solo indica al navegador que debe enviar el archivo al servidor pero debe haber en el servidor un programa que lo almacene en una carpeta del servidor. **Sintaxis para disponer un control de tipo file:** **<input type="file" name="archi">** Se utiliza el elemento HTML input para definir este tipo de control. **En la propiedad type inicializamos con el valor file. Inicializar la propiedad name también es importante ya que mediante este nombre se recupera en el servidor.** Otra cosa muy importante a tener en cuenta cuando hacemos **upload de archivos al servidor es inicializar la propiedad enctype del elemento form: <form method="post" action="/registrardatos.php" enctype="multipart/form-data">** Con esto indicamos al navegador que el formulario almacena uno o más archivos que deben ser enviados al servidor. Ejemplo: página que solicite que introduzca un nombre y la foto de la persona:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post" enctype="multipart/form-data">

Intrdouzca su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione la foto:

<input type="file" name="foto">

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

**Recordemos siempre inicializar la propiedad enctype del elemento form:** <form action="/registrardatos.php" method="post" enctype="multipart/form-data"> Tenga en cuenta que si no se hace ningún programa en el servidor el archivo no se almacena.

EJEMPLO:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post" enctype="multipart/form-data">

Seleccione el archivo 1:<input type="file" name="archi1"><br>

Seleccione el archivo 2:<input type="file" name="archi2"><br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

### 1.2.2. Listas desplegables de opciones (<SELECT>)

Para definir una entrada de datos en forma de lista desplegable se utiliza la etiqueta

<**SELECT**> .. <**/SELECT**>.

Esta etiqueta nos permite presentar un menú con varias opciones de elección, de tal forma que el usuario pueda elegir una o varias de ellas, si así se especifica.

Los atributos que posee son:

* **NAME**

Específica un nombre para hacer referencia al elemento tipo lista del formulario.

* **SIZE**

Especifica el número de opciones visibles en la lista. Si ‘n’ es igual a 1, se mostrará como una lista desplegable. Si ‘n’ es igual a X, se mostrará como una caja de lista con barra de desplazamiento, especificando el número de opciones visibles.

* **MULTIPLE**

Si está presente este atributo booleano permite una selección múltiple de opciones. Si no está activado, sólo se permiten selecciones únicas.

Para realizar selecciones múltiples se debe usar la tecla CONTROL.

* **DISABLED**

Desactiva el elemento lista del formulario.

#### 1.2.2.1. Creación de elementos en la lista de opciones (<OPTION>, <OPTGROUP>)

Para introducir cada una de las opciones disponibles en un elemento tipo lista definido con <**SELECT**>, se utiliza la etiqueta

<**OPTION**> .. <**/OPTION**>

Todo elemento <**SELECT**> debe contener al menos un elemento <**OPTION**>.

Los atributos que posee son:

* **VALUE**

Especifica el valor que será asociado al parámetro name del <SELECT> cuando se envíe el formulario. Este valor **debe ser único** para cada opción.

* **SELECTED**

Permite especificar la opción que aparecerá seleccionada cuando se muestre el formulario. Si no se especifica se tomará la primera de ellas por defecto.

La anchura de la lista desplegable vendrá determinada por la opción que mayor número de caracteres tenga.

EJEMPLOS:

|  |  |
| --- | --- |
| EJEMPLO: select (cuadro de selección individual) |  |

El elemento select es un cuadro de selección. Este elemento HTML nos permite seleccionar una opción entre un conjunto. Veremos en el concepto próximo que según como la configuramos podemos seleccionar varias opciones.

Ejemplo como crear un control de tipo select. Realizar un formulario que solicite cargar el nombre de una persona y el país donde vive, este último elemento mediante un control de tipo select permitir seleccionar el país.

El archivo pagina1.html es:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Introduzca su nombre:

<input type="text" name="nombre" size="30"><br>

Seleccione su pais:

<select name="pais">

<option value="1">Argentina</option>

<option value="2" selected>España</option>

<option value="3">México</option>

<option value="4">Guatemala</option>

<option value="5">Honduras</option>

<option value="7">El Salvador</option>

<option value="8">Venezuela</option>

<option value="9">Colombia</option>

<option value="10">Cuba</option>

<option value="11">Bolivia</option>

<option value="13">Perú</option>

<option value="14">Ecuador</option>

<option value="15">Paraguay</option>

<option value="16">Uruguay</option>

<option value="17">Chile</option>

</select>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Veamos la sintaxis para crear un cuadro de selección, primero abrimos la marca select la cual tiene definido la propiedad name: **<select name="pais">** Luego sin cerrar el select definimos tantos elementos de **tipo option** como opciones tendrá el cuadro de selección: **<option value="1">Argentina</option> <option value="2">España</option> <option value="3">México</option>**

El elemento option define el texto a mostrar y en la **propiedad value** indica el valor a enviar al servidor en caso de estar seleccionada dicha opción.

Para definir todas las opciones de nuestro cuadro de selección cerramos la marca select: </select>. Una variante gráfica de este control es inicializar **la propiedad size** del elemento select con un valor distinto a uno, con esto creamos un cuadro de selección que muestra simultáneamente varios elementos (de todos modos solo uno se puede elegir)

Es decir que con la propiedad size solo logramos un cambio estético del control.

**EJEMPLO - select (cuadro de selección múltiple)**

Una variante del cuadro de selección que vimos en el concepto anterior es permitir que el visitante del sitio pueda seleccionar varias opciones. Supongamos que tenemos un cuadro de selección con una lista de colores y queremos que el usuario pueda elegir varios . Ejemplo:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Seleccione uno o varios colores (Presione Ctrl para seleccionar varios colores)<br>

<select name="colores[]" size="4" multiple>

<option value="1">Rojo</option>

<option value="2">Verde</option>

<option value="3">Azul</option>

<option value="4">Amarillo</option>

<option value="5">Blanco</option>

<option value="6">Negro</option>

<option value="7">Naranja</option>

<option value="8">Violeta</option>

</select>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Podemos observar la sintaxis para la definición de un cuadro de selección múltiple: **<select name="colores[]" size="4" multiple> Definimos una propiedad llamada multiple y no le asignamos valor, por otro lado al nombre definido en la propiedad name le añadimos al final los caracteres [] para que desde el servidor podamos identificar que el control retorna posiblemente muchos valores.Es también común inicializar la propiedad size con un valor mayor a 1 para que sea más fácil la selección.** La mayoría de los navegadores permiten seleccionar opciones que no se encuentran juntas mediante el mouse y presionando simultáneamente la tecla Ctrl.

Opcionalmente se puede usar también <**OPTGROUP**> .. <**/OPTGROUP**> que permite agrupar opciones. Esta etiqueta tiene un atributo **LABEL** que especificará el rótulo del grupo de opciones. Este rótulo no se puede seleccionar, no es una opción.

Ejemplo:

|  |  |
| --- | --- |
| <**FORM** **action**="mailto:url@url.com" **method**="post" **enctype**="text/plain">  Cuadro de lista desplegable  <B>Indique su edad:</B>  <**SELECT** **name**="desplegable" **size**="1">  <**OPTION** **value**="<15">&lt;15</OPTION>  <**OPTION** **value**="15a20">15-20</**OPTION**>  <**OPTION** **value**="21a30" **SELECTED**>21-30</**OPTION**>  <**OPTION** **value**="31a40" >31-40 </**OPTION**>  <**OPTION** **value**="41a50">41-50</**OPTION**>  <**OPTION** **value**="51a60">51-60</**OPTION**>  <**OPTION** **value**="61a70">61-70</**OPTION**>  <**OPTION** **value**="71a80">71-80</**OPTION**>  <**OPTION** **value**=">80">&gt;80</**OPTION**>  </**SELECT**>  Cuadro de lista desplazable  <B>Indique sus aficiones:</B>  <**SELECT** **name**="desplazable" size="5" **MULTIPLE**>  <**OPTGROUP** **label**="Naturaleza">  <**OPTION** **value**="Senderismo">  Senderismo</**OPTION**>  <**OPTION** **value**="Escalada"> Escalada</**OPTION**>  <**OPTION** **value**="Natación"> Natación</**OPTION**>  </**OPTGROUP**>  <**OPTGROUP** **label**="Tecnología">  <**OPTION** **value**="Fotografía"> Fotografía</**OPTION**>  <**OPTION** **value**="Informática"> Informática</**OPTION**>  <**OPTION** **value**="Imagen">Imagen</**OPTION**>  </**OPTGROUP**>  </**SELECT**>  <**INPUT** **type**="**SUBMIT**" **value**="Enviar">  </**FORM**> |  |

### EJEMPLO:

**Ejemplo - select (agrupamiento de opciones)**

Hemos visto que podemos crear cuadros de selección individual o de selección múltiple. Ahora veamos que podemos agrupar las opciones que tiene el cuadro de selección, esto tiene sentido si el cuadro de selección tiene muchos items. Se cuenta con un nuevo elemento **llamado optgroup que agrupa un conjunto de elementos option.**

Ejemplo agrupar una serie de opciones, agrupar una serie de frutas y verduras:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Seleccione una fruta o verdura:

<select name="articulo">

<optgroup label="Frutas">

<option value="1">Naranjas</option>

<option value="2">Manzanas</option>

<option value="3”>Sandia</option>

<option value="4">Frutilla</option>

<option value="5">Durazno</option>

<option value="6">Ciruela</option>

</optgroup>

<optgroup label="Verduras">

<option value="7">Lechuga</option>

<option value="8">Acelga</option>

<option value="9">Zapallo</option>

<option value="10">Papas</option>

<option value="11">Batatas</option>

<option value="13">Zanahorias</option>

<option value="14">Rabanitos</option>

<option value="15">Calabaza</option>

</optgroup>

</select>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Como podemos observar para agrupar una serie de opciones dentro de un select debemos encerrarlas con el elemento **optgroup:**

**<optgroup label="Frutas"> <option value="1">Naranjas</option> <option value="2">Manzanas</option> <option value="3>Sandia</option> <option value="4">Frutilla</option> <option value="5">Durazno</option> <option value="6">Ciruela</option> </optgroup>**

La propiedad **label del elemento optgroup aparece dentro del control select pero no se puede seleccionar, es un título.** La propiedad label del elemento optgroup es el texto que se debe mostrar dentro del select.

### 1.2.3. Áreas de texto multilínea

Para introducir en un formulario un campo de texto multilínea, donde se podrán introducir textos largos, como descripciones o sugerencias, se utiliza la etiqueta

<TEXTAREA> .. **</TEXTAREA>**

Consiste en una caja de filas y columnas dentro de la cual se introduce el texto. Al no disponer de límite de caracteres, como sucede en los “input tipo text”, las barras de desplazamiento de la caja (*scroll*) se activan cuando es necesario para permitir visualizar todo el texto.

Se puede incluir un contenido inicial por defecto que se muestra inicialmente al cargar el documento. Éste se incluye entre las etiquetas de apertura y fin.

Los atributos que posee son:

* **NAME**

Especifica un nombre para hacer referencia al elemento tipo área de texto del formulario.

* **COLS**

Permite definir el número de columnas visibles de la caja de texto multilínea.

* **ROWS**

Permite definir el número de filas visibles de la caja de texto multilínea.

* **DISABLED**

Permite desactivar la caja de texto.

* **READONLY**

Permite impedir que el contenido de la caja sea modificado por el usuario.

Ejemplo:

|  |  |
| --- | --- |
| <**form** **action**="XX" **method**="post" **enctype**="text/plain" **name**="miform">  introduce un comentario:  <**textarea** **name**="comentario" **cols**="20" **rows**="10">  <**/textarea**>  <**/form**> | introduce un comentario: |
| <**FORM** **action**="mailto:url@url.com" **method**="post" **enctype**="text/plain">  <P>Nombre:  <**INPUT** **name**="nombre" **type**="**text**"></P>  <P>Contraseña:  <**INPUT** **name**="contraseña" **type**="**passwor**d"></P>  <**TEXTAREA** **name**="consulta" **rows**="3" **cols**="30">--Escriba aquí su consulta--<**/TEXTAREA**>  <P><**INPUT** **type**="**submit**" **value**="Enviar por e-mail"></P>  <**/FORM**> |  |

**Ejemplo: textarea**

El elemento de tipo textarea nos permite introducir varias líneas a diferencia del cuadro de texto (input/text). Es muy utilizado cuando queremos introducir un comentario de una longitud de caracteres grande. Ejemplo: formulario para que un visitante pueda introducir su nombre, su mail y un comentario del sitio, empleando para este último dato un elemento de tipo textarea:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Introduzca su nombre:

<input type="text" name="nombre" size="30"><br>

Introduzca su mail:

<input type="text" name="mail" size="50"><br>

Comentarios:<br>

<textarea name="comentarios" rows="5" cols="60"></textarea>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

La sintaxis para definir un área de texto para la introducción de múltiples líneas es: **<textarea name="comentarios" rows="5" cols="60"></textarea>** Es un elemento que requiere una marca de comienzo y una de finalización. **Además de tener la propiedad name similar a los otros elementos relacionados a formularios tiene dos propiedades llamadas rows y cols. Estas dos propiedades indican la**

**cantidad de filas y columnas que visualiza el área de texto.**

**EJEMPLO:** FORMULARIO CON TABLA, RADIOS CHECKBOX, SELECT

<html>

<head>

<title>Prueba de formulario con select</title>

</head>

<body>

<CENTER>

<h1> FORMULARIO CON TABLA, RADIOS CHECKBOX, SELECT </H1></CENTER><BR>

<FORM ACTION="http://mecaniza.sestud.uv.es/cgi-bin/test2-cgi" METHOD=POST>

<CENTER>

<TABLE BORDER>

<TR>

<TD>Nombre:</TD>

<TD> <INPUT TYPE="text" NAME="nombre" SIZE=18 MAXLENGTH=18> F.Nac.:

<INPUT TYPE="text" NAME="edad" SIZE=8 MAXLENGTH=8> DNI:

<INPUT TYPE="text" NAME="dni" SIZE=8 MAXLENGTH=8></TD>

<TR>

<TD>Apellidos:</TD>

<TD> <INPUT TYPE="text" NAME="apellidos" SIZE=48 MAXLENGTH=48></TD>

<TR>

<TD>Calle y número:</TD>

<TD> <INPUT TYPE="text" NAME="domicilio" SIZE=48 MAXLENGTH=48></TD>

<TR>

<TD>Código Postal:</TD>

<TD> <INPUT TYPE="text" NAME="postal" SIZE=5 MAXLENGTH=5>Ciudad:

<INPUT TYPE="text" NAME="localidad" SIZE=36 MAXLENGTH=36></TD>

<TR>

<TD>Provincia: </TD>

<TD> <INPUT TYPE="text" NAME="provincia" SIZE=20 MAXLENGTH=20> Teléfono:

<INPUT TYPE="text" NAME="telefono" SIZE=19 MAXLENGTH=19></TD>

<TR>

<TD>Estado Civil:</TD>

<TD> <SELECT NAME="OPCION">

<OPTION>Soltero

<OPTION>Casado

<OPTION>Divorciado

<OPTION>Otro

</SELECT>

<INPUT TYPE="radio" NAME="sexo" VALUE="H"> Hombre

<INPUT TYPE="radio" NAME="sexo" VALUE="M"> Mujer </TD>

<TR>

<TD>Idiomas:</TD>

<TD><INPUT TYPE="checkbox" NAME="idiomas" VALUE="E"> Español

<INPUT TYPE="checkbox" NAME="idiomas" VALUE="A"> Alemán

<INPUT TYPE="checkbox" NAME="idiomas" VALUE="F"> Frances

<INPUT TYPE="checkbox" NAME="idiomas" VALUE="I"> Inglés </TD>

<TR>

<TD>Comentarios<BR> personales:</TD>

<TD><TEXTAREA NAME="coment" ROWS=2 COLS=48></TEXTAREA></TD>

<TR>

<TH>Pulse aquí:</TH>

<TD ALIGN=CENTER>

<INPUT TYPE="submit" VALUE="Enviar datos ">

<INPUT TYPE="reset" VALUE="Borrar los datos"></TD>

</TABLE>

</CENTER>

</FORM>

## 1.3. Diseño de Formularios

Cuando nuestros formularios son muy extensos puede resultar útil etiquetar y agrupar los controles que lo componen. Esto se consigue con un grupo de etiquetas que comentaremos a continuación.

### 1.3.1. Agrupar controles y etiquetas que estén relacionados

Hasta ahora, no disponíamos de ninguna manera de agrupar visualmente varios controles, si no echábamos mano de elementos que no son del formulario, como tablas o imágenes.

Ahora, si encerramos una parte de un formulario dentro de la etiqueta **<FIELDSET>** .. **</FIELDSET>**, se mostrará un rectángulo alrededor de los controles que incluya.

Si además, le indicamos un título por medio de la etiqueta **<LEGEND>** .. **</LEGEND>**, conseguiremos que nuestros formularios sean más comprensibles para los usuarios desde un punto de vista visual y conceptual. Esta etiqueta posee un atributo align con los posibles valores top, bottom, left y right.

**<form>  
  <fieldset>  
    <legend>*Rótulo primer grupo*</legend>  
    *- controles -*  
  </fieldset>  
  <fieldset>  
    <legend>*Rótulo segundo grupo*</legend>  
    *- controles -*  
  </fieldset>  
    ...  
</form>**

**Fieldset tiene por objetivo agrupar un conjunto de controles de un formulario.** Debemos encerrar todos los controles a agrupar entre las marcas **<fieldset> y </fieldset>.** Además para agregar un título a este recuadro debemos agregar otro elemento HTML llamado **legend. Ejemplo:** formulario que solicite los datos personales de una persona y los datos del lugar donde trabaja, cada grupo de datos los agrupamos en un fieldset:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

<fieldset>

<legend>Datos personales</legend>

Apellido y nombre:

<input type="text" name="nombre" size="30"><br>

Documento de identidad:

<input type="text" name="dni" size="8"><br>

Fecha de nacimiento:

<input type="text" name="fechanacimiento" size="12"><br>

Dirección:

<input type="text" name="direccionpersona" size="30"><br>

</fieldset>

<fieldset>

<legend>Datos Laborales</legend>

Nombre de la empresa:

<input type="text" name="nombreempresa" size="30"><br>

Actividad:

<input type="text" name="actividad" size="50"><br>

Dirección:

<input type="text" name="direccionempresa" size="30"><br>

</fieldset>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Podemos ver que cada grupo de controles está encerrado con el elemento **fieldset:**

<fieldset>

<legend>Datos personales</legend>

Apellido y nombre:

<input type="text" name="nombre" size="30"><br>

Documento de identidad:

<input type="text" name="dni" size="8"><br>

Fecha de nacimiento:

<input type="text" name="fechanacimiento" size="12"><br>

Dirección:

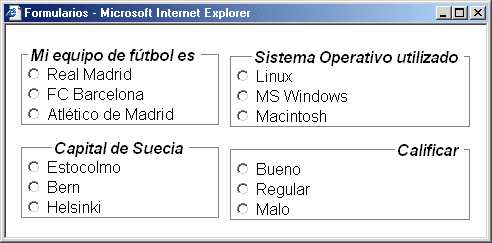
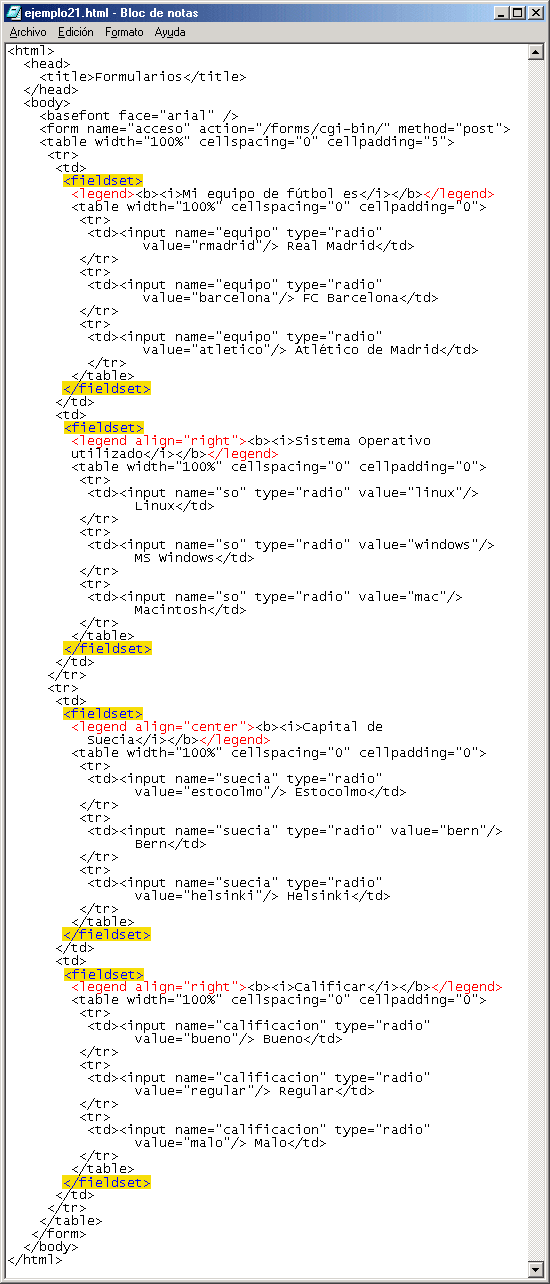
<input type="text" name="direccionpersona" size="30"><br>

</fieldset>

Luego el título de cada fieldset lo ponemos con: **<legend>Datos personales</legend>**

Un ejemplo con radio botones añadiendo estos nuevos elementos:

EJEMPLO:



<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

<fieldset>

<legend>Seleccione tipo y cantidad de pizzas</legend>

Pizza:

<select name="pizza1">

<option value="1">Mozzarella</option>

<option value="2">Jamon y Queso</option>

<option value="3">Napolitana</option>

</select>

Cantidad:

<select name="cantidad1">

<option value="0">0</option>

<option value="1">1</option>

<option value="2">2</option>

<option value="3">3</option>

<option value="4">4</option>

</select>

<br>

Pizza:

<select name="pizza2">

<option value="1">Mozzarella</option>

<option value="2">Jamon y Queso</option>

<option value="3">Napolitana</option>

</select>

Cantidad:

<select name="cantidad2">

<option value="0">0</option>

<option value="1">1</option>

<option value="2">2</option>

<option value="3">3</option>

<option value="4">4</option>

</select>

<br>

Pizza:

<select name="pizza3">

<option value="1">Mozzarella</option>

<option value="2">Jamon y Queso</option>

<option value="3">Napolitana</option>

</select>

Cantidad:

<select name="cantidad3">

<option value="0">0</option>

<option value="1">1</option>

<option value="2">2</option>

<option value="3">3</option>

<option value="4">4</option>

</select>

</fieldset>

<fieldset>

<legend>Datos</legend>

Nombre:

<input type="text" name="nombre" size="30"><br>

Dirección:

<input type="text" name="direccion" size="25"><br>

Teléfono:

<input type="text" name="telefono" size="10"><br>

</fieldset>

<input type="submit" value="Enviar">

</form>

</body>

</html>

**EJEMPLOS:**

Ejercicos1- controles con valores iniciales.

. Veamos como inicializar con valores por defecto para cada uno de los controles que hemos visto.

**Para inicializar un control de tipo text** debemos dar un valor a la propiedad value: **<input type="text" value="aquí su nombre" name="nombre" size="20">** El control aparece cargado con la cadena "aqui su nombre".

**Para inicializar un control de tipo textarea** debemos indicar el dato entre el comienzo y el fin de la marca: **<textarea rows="10" cols="40" name="curriculum">Introduzca aquí su currículo</textarea> El control textarea se inicializa con la cadena "Introduzca aquí su curriculum".**

**Para inicializar un control de tipo checkbox debemos disponer la propiedad checked sin asignar valor: <input type="checkbox" name="java" checked>Opcion 1<br>** Con esto logramos que el checkbox aparezca tildado apenas aparece el formulario.

**Para inicializar un control de tipo radio debemos definir la propiedad checked sin valor, igual que un checkbox, con la salvedad que solo un control de tipo radio puede tener definida esta propiedad: <input type="radio" name="estudios" value="1" checked>Opción 1<br>**

**Para inicializar un control de tipo select** **con selección individual debemos definir la propiedad selected de los elementos option:**

<select name="pais">

<option value="1">Argentina</option>

<option value="2" selected>España</option>

<option value="3">México</option>

<option value="4">Guatemala</option>

<option value="5">Honduras</option>

<option value="7">El Salvador</option>

<option value="8">Venezuela</option>

<option value="9">Colombia</option>

<option value="10">Cuba</option>

<option value="11">Bolivia</option>

<option value="13">Perú</option>

<option value="14">Ecuador</option>

<option value="15">Paraguay</option>

<option value="16">Uruguay</option>

<option value="17">Chile</option>

</select>

En este caso aparece seleccionado España, aunque sea el segundo option en la lista.

Para inicializar un control de tipo select con selección múltiple debemos definir la propiedad selected de varios elementos option:

<select name="colores[]" size="4" multiple="multiple">

<option value="1" selected>Rojo</option>

<option value="2">Verde</option>

<option value="3" selected>Azul</option>

<option value="4">Amarillo</option>

<option value="5" selected>Blanco</option>

<option value="6">Negro</option>

<option value="7">Naranja</option>

<option value="8">Violeta</option>

</select>

En este ejemplo los items Rojo,Azul y Blanco aparecen seleccionados desde un comienzo.

Ejemplo un formulario que pida el nombre de una persona. Luego que seleccione si es mayor de edad o no (por defecto inicializar en si), seguidamente el teléfono (cargar por defecto 949-) y por último en un textarea solicitar que introduzca comentarios.

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Apellido y nombre:

<input type="text" name="nombre" size="30"><br>

Es mayor de edad?:<br>

<input type="radio" name="radio1" checked

value="si">Si<br>

<input type="radio" name="radio1" value="no">No<br>

Telefono:

<input type="text" value="449-" name="telefono"

size="15"><br>

<textarea name="comentarios" rows="5" cols="40">Ingrese

aqui sus comentarios</textarea><br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

**Ejemplo2: Inhabilitar controles.**

Todos los controles que hemos visto podemos hacer que aparezcan inhabilitados. Supongamos que disponemos 3 controles de tipo radio para indicar que sección del sitio deseamos introducir. Nosotros queremos mostrar que tiene 3 secciones pero una no está disponible. Esto lo resolvemos deshabilitando un radio:

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Seleccione la sección donde desea ingresar:

<br>

<input type="radio" name="seccion" value="1"

disabled>Niños<br>

<input type="radio" name="seccion" value="2">Adolescentes<br>

<input type="radio" name="seccion" value="3">Mayores<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

Para deshabilitar el primer radio añadimos la propiedad disabled sin asignarle valor: **<input type="radio" name="seccion" value="1" disabled>Niños** Los siguientes elementos pueden inhabilitarse: button, input, optgroup, option, select y textarea. Esta propiedad tiene mucha aplicación si se aplica javascript en la página. Con javascript una vez cargado el documento se podría modificar el estado de los controles, habilitando y deshabilitando de acuerdo a los datos que carga el visitante al sitio.

**Ejemplo3 - text/password/textarea y readonly**

Otra propiedad que podemos asignarle a los controles que creamos con el elemento input y también el elemento **textarea es readonly**.

Si definimos la propiedad **readonly a un control el mismo es de solo lectura y no podemos modificar su contenido. Esta propiedad tiene uso cuando mediante un lenguaje de script (generalmente javascript) modificamos el control cambiándolo de estado ante ciertos eventos.**

**Cuando un control tiene la propiedad readonly el control tiene el foco, pero no se puede cambiar su contenido. La diferencia con la propiedad disabled es que con esta no toma el foco el control y generalmente aparece con un color que indica que el control está deshabilitado.**

Ejemplo: formulario que aparezca el nombre de una empresa en un text y el texto de un contrato en un textarea, ambos de solo lectura.

<html>

<head>

<title>Prueba de formulario</title>

</head>

<body>

<form action="/registrardatos.php" method="post">

Introduzca su nombre:

<input type="text" name="nombre" size="30" value="Interpolacion"

readonly><br>

Contrato:<br>

<textarea name="comentarios" rows="5" cols="60" readonly>

Por la presente..............................

</textarea>

<br>

<input type="submit" value="Enviar">

</form>

</body>

</html>

**MEJORAS DE HTML5 EN LOS FORMULARIOS. PARA LA ETIQUETA INPUT**

**Argumento required:** permite cambiar como obligatorio un campo de forma que no se pueda ejecutar la función submit.

**Argumento autofocus:** Debería usarse solo en un objeto de cada formulario. Selecciona el campo que tomará el foco de forma automática.

**Argumento autocomplete:** puede tomar dos valores: on/off. Indica al navegador si puede hacer sugerencias al usuario dependiendo de lo que haya escrito en ocasiones anteriores en ese mismo campo.

**Argumento placeholder:** válidos en todos los campos de texto, nos muestra un texto con instrucciones sobre que debemos de cumplimentar en el mismo.

**Argumento wrap:** Es válido en los input textarea y nos permite indicar si queremos que los saltos de línea que el usuario escriba en ellos se transmita (wrap=”hard”) o no (wrap=”soft”) que es la opción por defecto.

**Datalist:** asociada a los campos de texto, que nos permite mostrar al usuario una lista de sugerencias. (parecida a select). La identificación entre input y datalist se hace a través de los argumentos list y id cuyos valores deben ser iguales.

**<form action=**”recogida.php” **method=**”post”**>**  
 **<label for=**”gustos”**>**  
 Escribe en qué te gusta pasar tu tiempo de ocio  
 **</label>  
 <input type=**”text” **id=**”gustos” **name**=”gustos”

**list**=”listaGustos” **>**

**<datalist id=**”listaGustos”**>**

**<option label=**”deportes” **value=**”Deportes” **>**  
 **<option label=**”teatro” **value=**”Teatro”**>**  
 **<option label=**”cine” **value=**”Cine”**>**  
 **<option label=**”leer” **value=**”Leer”**>**  
 **</datalist>**  
 **<input type=**”submit” **value=**”enviar”**><br>**  
**</form>**

El usuario puede elegir cualquiera de las opciones de la lista, pero también se puede escribir lo que se quiera en el cuadro.

**Argumento pattern**: permite especificar una expresión regular que debe de cumplir lo que pongamos en el campo para que el formulario sea validado.

**<label>NIF: <input type="text" name="nif" pattern="[0-9]{8}[A-Za-z]{1}" /></label><br />**

**OTROS TIPOS DE ENTRADAS html5**

**Tipo de entrada color**

**<input type="color" name="color">**

<!DOCTYPE html>

<html>

<body>

<h1>color</h1>

<form action="/registrardatos.php" method="post">

Selecciona color:

<input type="color" name="color"><br>

<input type="submit">

</form>

</body>

</html>

**Tipo de entrada Fecha**

<input type=”date” name=”fecha”>

<!DOCTYPE html>

<html>

<body>

<h1>date</h1>

<form action="/registrar datos.php">

Cumpleaños:

<input type="date" name="cumpleday">

<input type="submit">

</form>

</body>

</html>

**Tipo de entrada Fecha y hora local**

<input type=”datetime-local” name=”fechahora”>

<input type=”datetime” name=”fechahora”>

<!DOCTYPE html>

<html>

<body>

<h1>datetime</h1>

<form action="/registrardatos.php">

aniversario:

<input type="datetime-local" name="fechahora">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada email**

<input type=”email” name=”email”>

<!DOCTYPE html>

<html>

<body>

<h1>email</h1>

<form action="/registrardatos.php">

email

<input type="email" name="email">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada mes.**

<input type=”month” name=”cumplemes”>

<!DOCTYPE html>

<html>

<body>

<h1>mes</h1>

<form action="/registrardatos.php">

email

<input type="month" name="mes">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada número**

Alguno de los atributos

<input type=”number” max=”5” min=”2”>

<!DOCTYPE html>

<html>

<body>

<h1>numero</h1>

<form action="/registrardatos.php">

numero

<input type="number" min="2" max="5" name="number">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada rango**

<input type=”range” name=”rango” min=”1” max=”4”>

<!DOCTYPE html>

<html>

<body>

<h1>rango</h1>

<form action="/registrardatos.php">

rango

<input type="range" min="2" max="5" name="rango">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada buscar**

<input type=”search” name=”buscar” >

<!DOCTYPE html>

<html>

<body>

<h1>buscar</h1>

<form action="/registrardatos.php">

Buscar en google

<input type="search" name="busco">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada url**

<input type=”url” name=”buscar”>

<!DOCTYPE html>

<html>

<body>

<h1>url</h1>

<form action="/registrardatos.php">

Buscar en google

<input type="url" name="url">

<input type="submit">

</form>

</body>

</html>

**Tipo entrada semana**.

<input type=”week” name=”semana”>

<!DOCTYPE html>

<html>

<body>

<h1>url</h1>

<form action="/registrardatos.php">

Buscar en google

<input type="week" name="semana">

<input type="submit">

</form>

</body>

</html>