

Los Formularios

Los formularios son una de las herramientas de que disponemos a la hora de hacer nuestras páginas web interactivas, en el sentido de que nos permiten recopilar información de la persona que ve la página, procesarla y responder a ella, pudiendo de esta forma 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 página. 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 disponemos de gran cantidad de ellos en páginas especializadas de Internet. También es posible procesar esa información mediante páginas ASP (Active Server Pages) o JSP (Java Server Pages), que son páginas que contienen dentro de su código HTML trozos de código escrito en Visual Basic Script o Java Script 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.

Una vez procesada de esta forma 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 está asociado a un tipo concreto de datos o una acción predeterminada: botones de envío y borrado de datos, listas de selección, cajas de entrada de texto, etc.

El resultado de cualquier formulario es una lista de variables y valores asignados a las mismas, que tienen todos ellos un atributo en común: el nombre de su variable.

Etiquetas <FORM>...</FORM>

Estas etiquetas son las que delimitan el comienzo y el final de un formulario. Si queremos mostrar en una página un elemento aislado de un formulario, aunque sea con otra finalidad diferente a la del envío de los datos que contenga al servidor (por ejemplo para tratar esa información mediante Javascript), es necesario delimitar ese elemento de formulario mediante las etiquetas <FORM> y </FORM>. Si no lo hacemos así, algunos navegadores lo mostraran y otros no.

La pareja de etiquetas se introduce en el código HTML dentro del BODY de la página, en el lugar en que queramos que nos aparezcan los elementos del formulario.

Sus principales atributos o parámetros son:

* **action = "ruta_programa"**. Indica el programa del servidor que va a "tratar" las variables que se envíen con el formulario o el envío de los datos mediante correo electrónico mediante el valor "mailto: direccion_de_correo", en cuyo caso deberemos añadir el parámetro ENCTYPE="text/plain" para que lo que recibamos resulte legible. También puede ser una URL usando el método GET (por ejemplo, un botón para ir a otra página).

En el caso de que definamos aquí un programa para el tratamiento de datos deberemos especificar su ruta relativa respecto al directorio de carpetas del servidor en dónde tengamos situado nuestra aplicación web o bien una URL completa, si el programa está en otra dirección de Internet.

* **method = " POST / GET "**. Indica el método según el que se van a transferir las variables del formulario. **POST** envía los datos, normalmente al programa CGI definido en **action** o por correo si en **action** hemos utilizado **mailto**. El método **GET** añade los argumentos del formulario al URL recogido en action (utilizando como separador de las variables la "?"). El método de uso más normal es **POST**, y en el caso de que estemos mandando el formulario a nuestra dirección de correo electrónico es obligado usarlo.

Con **GET** los datos son encadenados al URL especificado en action , utilizando el tipo de codificación especificado en el atributo enctype. Este método se utiliza cuando los datos no modifican la base de datos, por ejemplo, al realizar una búsqueda, y los caracteres a enviar tienen que pertenecer obligatoriamente al conjunto ASCII.

Con el método **POST** se realiza una transacción mediante el protocolo HTTP, utilizando la codificación enctype. Se utiliza para aplicaciones que modifican la base de datos de destino.

* **enctype = " tipo "**. Indica el tipo de documento en formato **MIME**. El más usado para que lleguen el contenido a través de un mail con el método post sería "text/plain". Especifica el tipo de encriptación que se va a realizar con los datos que se van a enviar. Este atributo solamente se aplica si method es POST. Indica como se codificará el formulario si utiliza el método POST. Por defecto es "application/x-www-form-urlencoded". Otro posibles valor es "multipart/form-data". Más adelante veremos en que caso se utiliza uno u otro tipo.

* **accept = " tipo MIME "**, que indica al servidor qué tipos de archivo MIME se van a aceptar en el envío. Normalmente no se usa, pero si nos es necesario el administrador de nuestro servidor web nos puede indicar cuales son aceptados.

* **accept-charset** = "**charset** ", que especifica la lista de caracteres permitidos que deben ser interpretados correctamente por el servidor receptor. Por ejemplo, si el servidor admite como caracteres permitidos código HTML, en valor de charset será "text/html".

* **target**= " **nombre** ".. Este atributo funciona igual que el homónimo de las etiquetas <A> y <FRAME>, e indica en qué ventana de una página con frames se debe mostrar el resultado del proceso de datos mediante el programa CGI.

Ejemplo.-

```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">  
...  
...elementos del formulario  
...  
</form>
```

Etiqueta <INPUT>

La etiqueta <INPUT> va a definir la mayoría de los diferentes elementos que va a contener el formulario.

Sus atributos y valores son:

* **TYPE** = " **tipo** ", donde tipo puede ser uno cualquiera de los elementos que veremos a continuación:

- **text**, que sirve para introducir una caja de texto simple, y admite los parámetros:
 - **name**= " **nombre** ", que asigna de forma unívoca un nombre identificador a la variable que se introduzca en la caja de texto.
 - **maxlength**= " **n** ", que fija el número máximo de caracteres que se pueden introducir en la caja de texto.
 - **size**= " **n** ", que establece el tamaño de la caja de texto en pantalla.
 - **value**= " **texto** ", que establece el valor por defecto del texto que aparecerá inicialmente en la caja de texto.
 - **disabled**, que desactiva la caja de texto, por lo que el usuario no podrá escribir nada en ella.
 - **accept** = " **lista de content-type** ", Indica el tipo de contenido que aceptará el servidor.

Sus posibles valores son:

1. text/html
2. application/msexcel
3. application/msword
4. application/pdf
5. image/jpg

6. image/gif otros.

- **readonly**, que establece que el texto no puede ser modificado por el usuario.
- **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- **alt= " texto "**, que asigna una pequeña descripción al elemento.

Ejemplo.-

```
<form action="" method="post" enctype="text/plain" name="miform">
  introduce tu nombre:
  <input type="text" maxlength="10" size="10" name="nombre">
</form>
```

con el que obtenemos:

introduce tu nombre:

- **radio**, que define un conjunto de elementos de formulario de tipo circular, en los que el usuario debe optar por uno solo de ellos, que se marca con el ratón o tabulador. Admite los parámetros:
 - **name=" nombre"**, que asigna un nombre identificador único a la variable definida por el elemento. Este identificador debe ser el mismo para todos los elementos radio de un grupo.
 - **value = " valor "**, que define un valor posible de la variable para cada uno de los radio botones.
 - **checked**, que marca por defecto uno de los radio botones del grupo.
 - **disabled**, que desactiva el radio botón, por lo que el usuario no podrá marcarlo.
 - **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
  marca tu equipo favorito:
  <input type="Radio" name="equipo" value="madrid" checked>Real Madrid
  <input type="Radio" name="equipo" value="atletico">Atlético de Madrid
  <input type="Radio" name="equipo" value="barcelona">Barcelona
</form>
```

con el que obtenemos:

marca tu equipo favorito: ☒ Real Madrid ☐ Atlético de Madrid ☐ Barcelona

- **checkbox**, que define una o más casillas de verificación, pudiendo marcar el usuario las que desee del conjunto total. Si pinchamos una casilla con el ratón o la activamos con el tabulador y le damos a la barra espaciadora la casilla se

marca; si volvemos a hacerlo, la casilla se desmarca. Sus parámetros complementarios son:

- o **name=" nombre "**, que asigna un nombre identificador único a la variable definida por el elemento. Este identificador debe ser el mismo para todos los elementos conjunto de casillas.
- o **value = " valor "**, que define un valor posible de la variable para cada uno de casillas de verificación.
- o **checked**, que marca por defecto una o más de las casillas del grupo.
- o **disabled**, que desactiva la casilla de verificación, por lo que el usuario no podrá marcarla.
- o **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
marca tu música favorita:
<input type="checkbox" name="musica" value="rock" checked>Rock
<input type="checkbox" name="musica" value="pop" checked>Pop
<input type="checkbox" name="musica" value="heavy">Heavy
<input type="checkbox" name="musica" value="tecno">Tecno
</form>
```

con el que obtenemos:

marca tu música favorita: ☒ Rock ☒ Pop ☐ Heavy ☐ Tecno

- **button**, que define un botón estándar. Este botón puede ser usado para diferentes acciones, pero normalmente se le da utilidad mediante JavaScript, mediante el evento "OnClick". Sus parámetros son:
 - o **name=" nombre "**, que asigna un nombre al botón, que nos puede servir para acciones con lenguaje de script.
 - o **value=" valor "**, que define el texto que va a figurar en el botón.
 - o **disabled**, que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, pues permanece inactivo.
 - o **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
<input type="button" name="boton" value="pulsame">
</form>
```

con el que obtenemos:

- **image**, que introduce un botón definido por una imagen, en vez del formato estándar de botones, con lo que podemos así personalizar el botón. Inicialmente su función era contener una localización de las

coordenadas que pinchara el usuario, para que luego el programa CGI realizara una acción u otra dependiendo de estas. Pero actualmente se usa más para personalizar un botón de envío de datos; es decir, su funcionalidad es análoga a la de submit, pero nos permite personalizar este elemento. Además añadirá como información en el formulario las coordenadas x e y donde el usuario lo pulsó. Admite los parámetros:

- **name= " nombre "**, que asigna un nombre al botón para identificarlo de forma única y poder así acceder luego a sus propiedades.
- **src= " ruta imagen "**, que establece la ruta dónde localizar el fichero de imagen. Esta ruta puede ser relativa al directorio web en el servidor o una URL absoluta que define unívocamente la localización de la imagen.
- **width= " w "**, que fija la anchura del botón de imagen.
- **height= " h "**, que fija la altura del botón de imagen.
- **align= " left / middle / right / bottom / baseline..."**, que define la posición del texto que rodea el botón respecto a este.
- **hspace= " x "**, que fija el espacio horizontal que habrá entre el botón y el texto que lo circunda.
- **vspace= " x "**, que fija el espacio vertical que habrá entre el botón y el texto que lo circunda.
- **alt = " texto "**, que asocia un texto explicativo al botón de imagen. Al situar el cursor encima del botón se mostrará este texto en forma de tip de Windows. Siempre es conveniente poner este atributo, para que en caso de navegadores de sólo texto, de que la imagen no se encuentre en la ruta por cualquier motivo o de que la página tarde mucho en cargarse por su peso, el usuario puede ver qué hace el botón de imagen y pueda en consecuencia utilizarlo para tal fin.
- **disabled**, que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, pues permanece inactivo.
- **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

- **usemap= " #nombre de mapa "**, que nos permite el uso de un mapa de imágenes en el cliente como un dispositivo más de entrada de datos.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
  <input type="image" name="boton" src="/images/contenidos/lecciones/1.jpg" width="50"
height="20" hspace="10" align="middle">
</form>
```

con el que obtenemos:



- **password**, que define una caja de texto para contener una clave o password, por lo que el texto que introduzca el usuario aparecerá como asteriscos, por motivos de seguridad. Sus parámetros opcionales son los mismos que los del tipo text.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
  <input type="password" size="15" maxlength="10">
</form>
```

con el que obtenemos:

introduce la clave de acceso:

- **hidden**, que define un campo invisible, por lo que no se ve en pantalla. Aunque parece así definido que no tiene utilidad, sus usos son varios e importantes, y los veremos más tarde. Sus atributos son:
 - **name=" nombre "**, que asigna un nombre identificador único al campo oculto.
 - **value=" valor "**, que va a ser el valor fijo que se le va a pasar al programa del servidor, ya que el usuario no puede modificarlo. En realidad este valor no tiene porqué ser fijo, ya que lo vamos a poder modificar mediante código de script, lo que nos va a permitir ir pasando una serie de variables ocultas de una página a otra.

Ejemplo.-

```
<form action="mailto:yo@miservidor.com" method="post" enctype="text/plain" name="miform">
  <input type="hidden" name="contraseña" value="123ABC">
</form>
```

- **file = " ruta archivo "**, que define un archivo que puede ser enviado junto con los datos del formulario, y en donde la ruta puede ser relativa al directorio de carpetas del servidor o una ruta URL absoluta de Internet. Debemos asegurarnos de que el tipo de archivo enviado esté dentro de los permitidos por la etiqueta ACCEPT.
- **submit**, que incorpora al formulario un botón de envío de datos. Cuando el usuario pulsa este botón los datos que ha introducido en los diferentes campos del formulario son enviados al programa del servidor o a la dirección de correo indicada en action. Sus atributos son:
 - **value=" valor "**, que define el texto que va a aparecer en el botón de envío.
 - **disabled**, que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, pues permanece inactivo.
 - **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- **reset**, que define un botón que al ser pulsado por el usuario borra todos los datos que hubiera introducido en cualquiera de los campos del formulario. Sus atributos son los mismos que los de SUBMIT.

Etiquetas <SELECT>...</SELECT>

Mediante esta etiqueta podemos definir una entrada de datos en forma de lista desplegable, que presenta varias opciones de elección, de tal forma que el usuario puede elegir una o varias de ellas, si así se especifica.

Sus atributos y valores son:

- **name=" nombre"**, que asigna un nombre identificador al campo, de tal forma que al enviar los datos del formulario la opción elegida será asociada a este nombre.
- **size= " n "**, con n= nº entero, que define el número de opciones visibles. Si n=1 el campo de selección se presenta como una lista desplegable, mientras que si se indica otro valor se presenta como una caja de lista con barra de desplazamiento.
- **multiple**, que permite elegir varias de las opciones a la vez. Si no se especifica este atributo solamente se podrá escoger una de las opciones. Para ello hay que

mantener pulsada la tecla CONTROL mientras se seleccionan y/o deseleccionan las diversas opciones.

- o **disabled**, que desactiva la lista, de tal forma que no se produce ninguna acción cuando se pulsa una opción, pues permanece inactiva.
- o **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Para introducir cada una de las diferentes opciones de la lista se utiliza la etiqueta **<OPTION>**, que admite como parámetros:

- o **value**, que fija el valor que será asociado al parámetro name de **<SELECT>** cuando se envíe el formulario. Este valor debe ser único para cada opción.
- o **selected**, que establece la opción por defecto. Si no se especifica este parámetro en ninguna opción se tomará la primera de ellas por defecto.

La anchura de la lista desplegable vendrá determinado por el número de caracteres de la opción que más tenga.

Ejemplos.-

```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">
selecciona una de los siguientes valores:
<select>
<option value="uno">uno
<option value="dos">dos
<option value="tres">tres
</select>
</form>
```

con el que obtenemos:

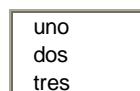
selecciona un de los siguientes valores:



```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">
selecciona una de los siguientes valores:
<select size="3">
<option value="uno">uno
<option value="dos">dos
<option value="tres">tres
</select>
</form>
```

con el que obtenemos:

selecciona un de los siguientes valores:



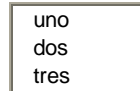
```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">
selecciona una de los siguientes valores:
<select size="3" multiple>
```

```

<option value="uno">uno
<option value="dos">dos
<option value="tres">tres
</select>
</form>

```

con el que obtenemos:



selecciona un de los siguientes valores:

Etiquetas <TEXTAREA>...</TEXTAREA>

Esta pareja de etiquetas inserta una caja de texto de múltiples líneas, que normalmente se utiliza para introducir comentarios o datos largos en un formulario. Si no se introduce ningún texto entre ambas etiquetas la caja de texto aparecerá vacía, para que el usuario lo introduzca, pero si introducimos algún texto entre ambas este aparecerá inicialmente en la caja. Si el texto que contiene supera el espacio disponible en la caja aparecerá una barra de desplazamiento vertical.

Sus atributos y valores son:

- o **name=" nombre"**, que asigna un nombre identificador al campo, que será asociado en el envío del formulario al texto introducido en la caja de texto.
- o **cols=" x "**, que define el número de columnas visibles de la caja de texto.
- o **rows=" y "**, que define el número de filas visibles de la caja de texto.
- o **wrap= " valor "**, que justifica automáticamente el texto en el interior de la caja. Este atributo es de uso complicado.

Según las recomendaciones si su valor es **VIRTUAL** se enviará todo el texto seguido, en una línea, mientras que si vale **PHYSICAL** el texto se enviará separado en líneas físicas, pero así como se define no lo admiten los navegadores. Usemos cual usemos de los dos el resultado es el mismo: justificación por líneas físicas, y así y todo se producen saltos de línea indeseados.

Si construimos la página web con un programa de ayuda tipo HomeSite nos encontramos con tres posibles valores de wrap: **SOFT** y **HARD**, que hacen lo mismo, alineando el contenido en líneas físicas, como en los casos anteriores y es el valor por defecto, y **OFF**, que coloca todo el texto de la caja en una sola línea, apareciendo entonces una barra de desplazamiento horizontal para poder visualizar todo el contenido de la caja.

- o **disabled**, que desactiva la caja de texto. Su utilidad es escasa.
- o **readonly**, que impide que el contenido de la caja sea modificado por el usuario.
- o **tabindex = " n "**, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Ejemplos.-

```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">  
  introduce un comentario:  
  <textarea name="comentario" cols="20" rows="10">  
  </textarea>  
</form>
```

con el que obtenemos:

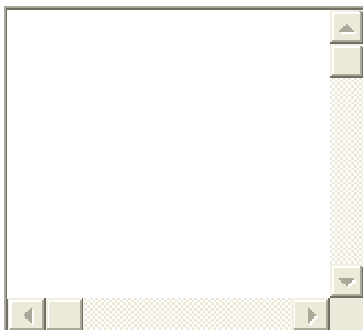
introduce un comentario:

A screenshot of a web browser window. The address bar shows a URL starting with 'http://'. The main content area displays the text 'introduce un comentario:' followed by a large, empty text area. The text area has a light gray border and a small vertical scrollbar on the right side. The browser's status bar at the bottom shows the page title and address.

Ejemplos.-

```
<form action="cgi-bin/control.exe" method="post" enctype="text/plain" name="miform">  
  <textarea name="comentario" cols="20" rows="10" readonly>  
    ahora no puedes escribir nada.  
  </textarea>  
</form>
```

con el que obtenemos:

A screenshot of a web browser window. The address bar shows a URL starting with 'http://'. The main content area displays the text 'introduce un comentario:' followed by a large, empty text area. The text area has a light gray border and a small vertical scrollbar on the right side. The browser's status bar at the bottom shows the page title and address.