**FORMULARIOS EN HTML**

Los formularios son una característica del estándar HTML que permite a los autores recolectar información provista por los visitantes. Estos formularios pueden resultar útiles para reunir información personal, de contacto, preferencias, opiniones, o de cualquier otro tipo que el autor necesite. En este tutorial exploraremos todas las herramientas disponibles para construir formularios en HTML.

Un formulario puede insertarse en un documento HTML a través del elemento HTML form que actuará como contenedor para todos los elementos de entrada. Toda la información recolectada por un formulario puede ser enviada a un agente procesador (un archivo conteniendo un script hecho especialmente para procesar esta información) que usualmente va especificado en el atributos "action". Lo que el agente procesador haga con la información y cómo la maneje es un tema que no será tratado en este sitio ya que no pertenece al estándar HTML. Para manejar la información de un formulario debes usar un script del lado servidor.

También puedes especificar cómo la información será enviada en el valor del atributo "method": "post" (los datos del formulario son adjuntados al cuerpo del mismo) ó "get" (los datos del formulario son adjuntados a la URL). Se supone que el agente procesador sabe cuál es el método de envío del formulario.

De este modo, un formulario simple puede tener la siguiente declaración:

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

...Controles...

</form>

1. **ELEMENTOS DE ENTRADA**

La mayoría de los controles de entrada son visuales y pueden interactuar con el usuario. Su uso depende del tipo de control y también del tipo de información que pueden recolectar. Los elementos de entrada de un formulario pueden ser definidos mediante el uso de cuatro elementos: el elemento HTML input, el elemento HTML button, el elemento HTML select y el elemento HTML textarea. En este tutorial dividiremos los controles por su funcionalidad.

Como regla general para todos los controles, el atributo "name" identificará la información para el agente procesados, y su valor dependerá de la naturaleza del control (algunas veces, como en las casillas de verificación o botones radio, será el contenido del atributo "value").

Nota: como las descripciones y atributos de cada control son levemente tratados en este tutorial, visita la referencia de los elementos para más información al respecto.

* 1. **ENTRADA TEXTUAL**

Existen tres tipos de entrada textual que pueden recolectar información como nombres, comentarios, opiniones, etc.

1. **Entrada de línea**

Este control recolecta información textual en una sola línea, lo que significa que el usuario no podrá utilizar la tecla "enter" para ir a la siguiente línea (en la mayoría de los formularios, la tecla "enter" presionada en uno de estos campos, envía el formulario que lo contiene).

Este control es insertado en documentos HTML usando el tag HTML input con el valor "text" en su atributo "type". El valor inicial, mostrado cuando la página se carga, puede ser definido usando el atributo "value".

**Código**

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

Ingresa un texto: <input name="ingresatexto" type="text" value="Por favor, ingresa aquí" />

</form>



1. **ENTRADAS DE PASSWORD**

Este control actúa exactamente como el de entrada de línea con la excepción de que "esconde" los caracteres ingresados mostrándolos como puntos o asteriscos para evitar que los usuarios ven su contenido.

Es definido utilizando el valor "pass" en el atributo "type", y su valor inicial puede ser definido usando el atributo "value". Es comúnmente usado para el ingreso de contraseñas.

Código

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

Ingresa tu contraseña: <input name="contrasena" type="password" value="123456" />

</form>



1. **ENTRADA DE MULTI-LÍNEA**

Este control permite a los usuarios ingresar texto en una o más líneas. se inserta utilizando el tag HTML textarea y puede ser usado para recolectar reportes, comentarios, cartas, etc. En este tag, el contenido será el texto inicial.

Código

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

Ingresa tus comentarios:<br /><textarea name="comentarios" rows="2" cols="30">...Tus comentarios aquí...</textarea>

</form>



Nota que los tributos "rows" y "cols" establecen las dimensiones de la caja de texto y son requeridos por algunos DTDs. El valor pasado al agente procesador será el texto ingresado, es decir, el contenido de la caja de texto.

* 1. **OPCIONES**

Los autores pueden también dejar que sus visitantes elijan opciones preestablecidas de una lista. Esto puede lograrse usando uno de los tres controles siguientes, que pueden construir diferentes tipos de listas de opciones.

1. **CASILLAS DE VERIFICACIÓN**

Una casilla de verificación es una opción simple que puede tomar uno de dos valores: "marcado" ó "no marcado" ("checked" ó "unchecked"). Las casillas de verificación pueden ser agrupadas visualmente formando listas de opciones, pero cada una de ellas es tratada individualmente.

Este control es insertado mediante el tag HTML input con el valor "checkbox" en su atributo "type". Inicialmente, la casilla aparece sin marcar a menos que lo especifique de otra forma usando el atributo booleano (verdadero o false) "checked". Recuerda, que para lograr un código correcto en XHTML necesitas definir al atributo booleano con sus nombres como valores (por ejemplo, checked="checked").

Código

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

Selecciona tus intereses:<br />

<input name="cbipeliculas" type="checkbox" />Películas<br />

<input name="cbilibros" type="checkbox" checked="checked" />Libros<br />

<input name="cbiinternet" type="checkbox" />Internet

</form>



En este caso, el valor pasado será booleano y representará el estado de la opción (marcado o no marcado). Dependiendo del agente procesador puede ser "on/off", "checked/unchecked", "true/false", etc.

1. **BOTONES RADIO**

Los botones radio trabajan de la misma forma que las casillas de verificación con una pequeña diferencia: los botones radio que comparte el mismo nombre conforman un grupo de opciones donde el usuario no puede seleccionar más de una a la vez. Esto significa que cuando un usuario elige una opción, las demás son automáticamente deseleccionadas.

El valor inicial para el grupo depende del navegador (la mayorñia muestra todos los controles sin marcar). Esto puede ser cambiado usando el atributo booleano "checked".

Código

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

Selecciona tu actividad favorita:<br />

<input name="intereses" type="radio" value="rbipeliculas" />Películas<br />

<input name="intereses" type="radio" value="rbilibros" />Libros<br />

<input name="intereses" type="radio" value="rbiinternet" checked="checked" />Internet

</form>



Para botones radio el valor pasado al agente procesador es el contenido del atributo "value" de la opción seleccionada, lo que significa que una lista con muchos botones radio solo pasará un valor.

1. **LISTAS**

Estas listas pueden ser construídas utilizando tres elementos: el elemento HTML select (contenedor principal), el elemento HTML option (opción simple) y el elemento HTML optgroup (grupo de opciones). El último elemento es el único prescindible para construir este tipo de listas.

Los controles de lista pueden ser usadas para recolectar información al igual que los botones radio (sólo una opción) o como casillas de verificación (múltiples opciones), dependiendo de la presencia del atributo booleano "multiple". Como en los controles anteriores, el valor inicial de los elementos HTML option puede ser establecido a "seleccionado" usando el atributo booleano "selected".

Código

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

Elije solo una opción, como en los botones radio:

<select name="entradalista">

<optgroup label="Entradas textuales">

<option>Entrada de línea</option>

<option selected="selected">Entrada de contraseña</option>

<option>Entrada multi-línea</option>

</optgroup>

<optgroup label="Opciones">

<option>Casillas de verificación</option>

<option>Botones radio</option>

<option>Listas</option>

</optgroup>

</select>

<br />

Elije tantas opciones como quieras, como con casillas de verificación (manteniendo presionada la tecla "Ctrl"):<br />

<select name="entradalista[]" multiple="multiple">

<optgroup label="Entradas textuales">

<option>Entrada de línea</option>

<option>Entrada de contraseña</option>

<option>Entrada multi-línea</option>

</optgroup>

<optgroup label="Opciones">

<option selected="selected">Casillas de verificación</option>

<option>Botones radio</option>

<option>Listas</option>

</optgroup>

</select>

</form>



Para la primera lista sin el atributo "multiple", el valor pasado al agente procesador será la opción seleccionada, pero para el segundo ejemplo el valor pasado será un arreglo (array) con los valores de las opciones seleccionadas. Es por esto que, para listas con el atributo "multiple" presente, debes agregar los corchetes("[]") al final del nombre del tag HTML select. El valor individual pasado en ambos casos es el contenido del atributo "value" si el mismo está presente, y el contenido del elemento si no (de las opciones seleccionadas).

* 1. **BOTONES**

Un botón es un dispositivo primordialmente diseñado para realizar una acción con el formulario que lo contenga. De echo, existen dos tipos básicos de botones: para enviar el formulario y para reestablecerlo (devuelve todos los campos a sus valores iniciales). Existe sólo un tipo más de botón que no posee una acción predefinida (la misma debe ser especificada con un lenguage del lado cliente).

Los botones pueden insertarse a través del tag HTML input (botones para enviar y reestabler y botones de imagen) o el elemento HTML button (botones de contenido).

1. **BOTONES DE ENVÍO**

Este tipo de botones envía automáticamente el formulario en que se encuentra cuando es presionado. Es insertado utilizando el tag HTML input con el valor "submit" en su atributo "type".

Código

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

<input type="submit" value="Enviar este formulario" />

</form>



1. **BOTONES DE REESTABLECIMIENTO**

Este tipo de botones reestablecen los controles en un formulario a sus valores iniciales cuando es presionado. Son insertados con el tag HTML input con el valor "reset" en su atributo "type".

Código

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

<input type="text" name="texto1" value="Valor por defecto" /><br />

<input type="checkbox" name="reglas" checked="checked" /> Acepto las relgas<br />

<input type="reset" value="Reestablecer todos los campos a su valor predeterminado" />

</form>



En el ejemplo anterior, puedes probar la funcionalidad del botón cambiando los valores de los elementos y presionando el botón para reestablecer los campos a sus valores iniciales.

1. **BOTONES DE IMAGEN**

Los botones de imagen funcionan exactamente como los botones de envío con la única diferencia que los de imagen son representados visualmente con la imagen especificada en el atributo "src". Otra particularidad es éstos envían además las coordenadas donde ha ocurrido el click cuando el formulario fue enviado, tomadas desde la esquina superior izquierda del botón (por ejemplo, para un botón de imagen llamado "boton1"las coordenadas serán enviadas en la forma "boton1.x" y "boton1.y").

Los botones de imagen se insertan con el tag HTML input, usando el valor "image" en su atributo "type".

Código

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

<input type="image" src="/img/p/tutorials/forms/3/submit.png">

</form>



1. **BOTONES DE CONTENIDO**

Los botones de contenido pueden ser usados como botones de envío o restablecimiento, o bien pueden no tener ninguna acción preestablecida (dependiendo del valor de su atributo "type"). Pero su características es que permite a los autores inserta contenido dentro de los mismos. Esto significa que una porción de código HTML puede ser mostrado dentro del botón (vínculos, párrafos, texto en negrita, imágenes, etc.).

Código

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

<button type="button">

El <b>tag HTML button</b><br />

permite contenido.

</button>

</form>



* 1. **ENTRADA DE ARCHIVOS**

La entrada de archivos puede ser utilizada para subir archivos al servidor. Este control muestra una caja de texto donde el usuario debe especificar la ruta del archivo (que será adjuntado localmente por el navegador) que será enviado al servidor. De este modo los autores pueden pedir a los visitantes que envíen archivos de sus computadoras a través de la página. El control habitualmente muestra un botón para elegir el archivo visualmente.

Nota que para los formularios con subida de archivos debes especificar el valor "multipart/form-data" en el atributo "enctype" del tag HTML form, de otro modo, el archivo no será enviado.

Código

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

Ingresa el archivo: <input name="imagen" type="file" />

</form>



1. **ETIQUETANDO ELEMENTOS**

Las etiquetas de los elementos pueden hacer que tu página luzca mejor y sea más funcional en agentes de usuario visuales (cuando un visitante hace click en la etiqueta, el enfoque pasa automáticamente al control asociado), pero éstas sin duda harán una gran diferencia para personas con discapacidades o con navegadores no visuales. Una etiqueta establece una relación entre un control y una porción de texto (que se supone, establece un "título" para el control).

Las etiquetas pueden insertarse con el elemento HTML label y son asociados a los controles mediante el atributo "for", que debe coincidir con el valor del atributo "id" en el control.

Código

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

<label for="idnombre">Nombre:</label> <input type="text" id="idnombre" name="nombre" /><br />

<label for="idapellido">Apellido:</label> <input type="text" id="idapellido" name="apellido" /><br /><br />

Género:<br /><input type="radio" id="idmasculino" name="genero" /><label for="idmasculino">Masculino</label><br />

<input type="radio" id="idfemenino" name="genero" /><label for="idfemenino">Femenino</label>

</form>



Puedes proba en el ejemplo anterior, cómo los controles toman el enfoque cuando hacer click en sus etiquetas.

1. **AGRUPANDO ELEMENTOS**

Todos los elementos en un formulario pueden también ser agrupados temáticamente con el elemento HTML fieldset. Este elemento contiene a un grupo de controles que están relacionados entre sí por alguna razón (por ejemplo, información personal, laboral, financiera, sobre interests, etc.).

El título de cada grupo de elementos puede establecerse con el elemento HTML legend que debe ser definido justo después del tag de apertura del elemento HTML fieldset, y debe proveer un título descriptivo para el grupo.

Código

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

<fieldset>

<legend>Información personal</legend>

<label for="lnombre">Nombre</label>: <input id="lnombre" type="text" name="nombre" /><br />

<label for="lapellido">Apellido</label>: <input id="lapellido" type="text" name="apellido" /><br />

<label for="ldireccion">Dirección</label>: <input id="ldireccion" type="text" name="direccion" /><br />

<label for="ltelefono">Teléfono</label>: <input id="ltelefono" type="text" name="telefono" />

</fieldset>

<fieldset>

<legend>Información laboral</legend>

<label for="ldirecciontrabajo">Dirección</label>: <input id="ldirecciontrabajo" type="text" name="direcciontrabajo" /><br />

<label for="ltelefonotrabajo">Teléfono</label>: <input id="ltelefonotrabajo" type="text" name="telefonotrabajo" />

</fieldset>

</form>



El uso de grupos de controles es amplio y depende de cada formulario en particular, pero puede ser muy beneficioso para los visitantes cuando deben rellenar formularios muy largos (especialmente en agentes de usuario no visuales).