

# Formularios

El formulario

Un formulario es un conjunto de controles (botones, cajas de texto, casillas de verificación, botones radio, etc) que permiten al usuario introducir datos y enviarlos al servidor web para su procesamiento.

La etiqueta que delimita un formulario es la etiqueta `<form> ... </form>`. Los atributos más importantes de la etiqueta `<form>` son:

- `action`: contiene el nombre del agente que procesará los datos remitidos al servidor (por ejemplo, un script de PHP)
- `method`: define la manera de enviar los datos al servidor. Los valores posibles son:
  - `get`: los valores enviados se añaden a la dirección indicada en el atributo `action`
  - `post`: los valores se envían de forma separada

Si el atributo `method` no está establecido, Internet Explorer y Firefox se comportan como si el valor fuera `get`.

La etiqueta `<form>` es un elemento de bloque. En su interior puede haber cualquier elemento típico de una página web (párrafos, imágenes, divisiones, listas, tablas, etc.), además de las etiquetas que crean los controles.

**Los etiquetas que crean los controles en los formularios son `<input />`, `<button>`, `<select>`, `<optgroup>`, `<option>` y `<textarea>`. Además, se pueden estructurar los controles con las etiquetas `<fieldset>` y `<legend>`. Por último, la etiqueta `<label>` permite mejorar la accesibilidad de los controles.**

El navegador envía únicamente los datos de los controles contenidos en el formulario. En una misma página puede haber varios formularios que envíen datos al mismo o a diferentes agentes.

**Atributos comunes de los controles: `name`, `value`, `disabled`, `readonly`, `tabindex` y `accesskey`**

El atributo `name` identifica al control. El formulario únicamente envía al servidor los datos de los controles que tienen establecido el atributo `name`, por tanto si un control no tiene establecido su atributo `name`, el control simplemente se ignora. Si hubiera varios controles con el mismo atributo `name`, el formulario envía todos los datos.

---

El atributo `value` permite establecer el valor inicial de un control, aunque cada control lo utiliza de una forma ligeramente distinta. El único control sin atributo `value` es el área de texto (`<textarea>`).

En algunos casos (botones) el valor se muestra al usuario y el usuario no puede modificarlo. En otros casos (cajas de texto, contraseña) el valor se muestra en la página y el usuario puede modificarlo. En otros casos (casillas de verificación, botones radio, opciones de menú, oculto, imagen) el valor no se muestra en la página y el usuario no puede modificarlo.

En algunos casos (botones, oculto, botones radio) el atributo value es necesario. En otros casos (casilla de verificación, opciones de menú) es conveniente. En otros casos (cajas de texto, contraseña, imagen) puede omitirse. En el caso del selector de archivo, ni Internet Explorer ni Firefox parecen utilizarlo.

---

El atributo disabled permite deshabilitar el control. Una vez deshabilitado, el control ni siquiera puede coger el foco.

```
<input type="submit" value="Enviar" disabled="disabled" />
```

El atributo readonly permite que el control no sea modificable, aunque el control puede coger el foco.

```
<input type="text" name="texto" value="¡A que no me cambias!"  
readonly="readonly" />
```

El atributo tabindex permite controlar el orden en que el foco pasa de un elemento a otro mediante el tabulador (Tab para avanzar y Shift+Tab para retroceder). Los valores de tabindex pueden ser números naturales (incluido el cero), no necesariamente consecutivos. Si no está presente, los controles se visitan en el orden en que aparecen en el texto. Si está presente, los controles se visitan de menor a mayor.

```
<input type="text" name="texto1" value="Texto 1" tabindex="1" />  
<input type="text" name="texto2" value="Texto 2" tabindex="3" />  
<input type="text" name="texto3" value="Texto 3" tabindex="4" />  
<input type="text" name="texto4" value="Texto 4" tabindex="2" />
```

El atributo accesskey permite definir teclas de acceso (atajos de teclado). El problema con las teclas de acceso es que a veces entran en conflicto con combinaciones de teclas ya definidas por el navegador o el sistema operativo.

Para acceder a las teclas de acceso, cada navegador utiliza una combinación de teclas distinta:

- Firefox utiliza la combinación Alt+Shift+tecla
- Internet Explorer utiliza la combinación Alt+tecla o Alt+Shift+tecla
- Google Chrome utiliza normalmente la combinación Alt+tecla, pero cuando esa combinación ya está asociada a alguna acción usa la combinación Alt+Shift+tecla

En el ejemplo siguiente, Firefox utiliza la combinación Alt+Shift+tecla y Google Chrome e Internet Explorer utilizan la combinación Alt+tecla para acceder a las cajas de texto:

```
<input type="text" name="texto1" value="Acceso con a"  
accesskey="a" />  
<input type="text" name="texto2" value="Acceso con i"  
accesskey="i" />  
<input type="text" name="texto3" value="Acceso con o"  
accesskey="o" />
```

```
<input type="text" name="texto4" value="Acceso con u"
accesskey="u" />
```

En el caso de las teclas de acceso que no son las letras del alfabeto (los acentos no pueden ser teclas de acceso):

- Firefox utiliza la combinación **Alt+Shift+tecla**
- Internet Explorer utiliza la combinación **Alt+tecla**
- Google Chrome utiliza la combinación **Alt+tecla**

Si hay varios elementos con la misma tecla de acceso, o si hay varios elementos con la misma tecla de acceso en mayúsculas y minúsculas:

- Firefox avanza de uno a otro con la combinación **Alt+Shift+tecla**
- Internet Explorer avanza de uno a otro con la combinación **Alt+tecla** y retrocede de uno a otro con **Alt+Shift+tecla**
- Google Chrome no avanza de uno a otro

## Controles

Los controles son los elementos que permiten al usuario introducir información.

### Botones: `<input />` y `<button>`

Los botones se crean mediante la etiqueta `<button>`, aunque los botones más usuales en un formulario (los botones Submit y Reset) se pueden crear también con la etiqueta `<input />`.

Normalmente los botones no suelen llevar atributo name, pero se puede poner si se quiere (tendría sentido si un formulario contuviera dos botones de envío y quisieramos saber en cuál de ellos ha hecho clic el usuario, pero esto no es muy habitual).

### Botones Submit y Reset mediante `<input />`

El botón Submit es el que permite al usuario remitir los datos al servidor. Se crea mediante una etiqueta `<input />` cuyo atributo type tiene el valor submit. El texto que se muestra en el botón se define mediante el atributo value.

El botón Reset restablece los valores iniciales del formulario. Se crea mediante una etiqueta `<input />` cuyo atributo type tiene el valor reset. El texto que se muestra en el botón se define mediante el atributo value.

El atributo value sólo puede contener texto, no imágenes.

### Botones Submit y Reset mediante `<button>`

La etiqueta `<button>` permite crear botones de tipo submit o reset o botones de tipo general que deben asociarse a scripts para hacer algo. Los botones submit o reset se crean mediante el atributo type con el valor submit o reset. El botón de uso general se crea mediante el atributo type con el valor button.

Los botones `<button>` pueden contener texto e imágenes (o estructuras más complejas, pero no mapas de imágenes).

Si un botón no lleva el atributo `type`, se comporta como un botón de tipo `submit`.

**Caja de texto:** `<input type="text" />`, `<input type="password" />` y `<textarea>`

Existen dos tipos de cajas de texto: de una sola línea y de varias líneas

**Caja de texto de una sola línea:** `<input type="text" />` y `<input type="password" />`

Las cajas de texto de una sola línea se crean mediante la etiqueta `<input />` cuyo atributo `type` tiene el valor `text`.

El atributo `value` (optativo) contiene el valor inicial de la caja de texto. El atributo `size` indica el tamaño en caracteres de la caja en la pantalla (por omisión, las cajas suelen tener 20 caracteres de tamaño). El atributo `maxlength` indica el número máximo de caracteres que puede escribir el usuario.

```
<input type="text" name="texto" value="Escribe algo" />
```

```
<input type="text" name="texto" size="10" />
```

```
<input type="text" name="texto" maxlength="5" />
```

Existe una caja de texto de una sola línea especial para escribir contraseñas que se crea mediante la etiqueta `<input />` cuyo atributo `type` tiene el valor `password`. Al escribir en una caja de contraseña, en vez de letras aparecen puntos gruesos. Es importante señalar que estas cajas no proporcionan ninguna seguridad en la transmisión, simplemente ocultan al usuario lo que este escribe.

**Caja de texto de varias líneas:** `<textarea>`

Las cajas de texto de varias líneas se crean mediante la etiqueta `<textarea>`. Los atributos obligatorios `rows` y `cols` establecen el número de filas y columnas de la caja. El atributo `value` (optativo) contiene el valor inicial de la caja de texto de varias líneas.

**Selector de color:** `<input type="color" >`

El control de selector de color se crea mediante la etiqueta `<input />` cuyo atributo `type` tiene el valor `color`. Permite elegir un color mediante el selector de color proporcionado por el sistema operativo.

```
<input type="color" name="color" />
```

**Casilla de verificación:** `<input type="checkbox" />`

Las casillas de verificación se crean mediante la etiqueta `<input />` cuyo atributo `type` tiene el valor

checkbox. Si el atributo checked tiene el valor checked, la casilla aparece marcada. Las casillas de verificación sólo se envían si se han marcado. El atributo value contiene el valor que envía el formulario si la casilla de verificación está marcada. Si el atributo value no está establecido, el formulario envía el valor on.

### **Botón radio: <input type="radio" >**

Los botones radio se crean mediante la etiqueta <input /> cuyo atributo type tiene el valor radio. Lo botones radio que tienen el mismo atributo name forman un grupo, es decir, que si se marca uno de ellos se desmarca automáticamente el resto.

### **Menú: <select>**

Los menús se crean mediante la etiqueta <select>. Cada opción del menú se define mediante la etiqueta <option>.

El atributo **selected** indica la opción por omisión. Si ningún elemento posee el atributo selected, tanto Firefox como Internet Explorer muestran la primera opción del menú. Para evitar malentendidos y forzar al usuario a elegir un valor, se suele incluir una opción en blanco al principio de los menús.

El atributo **size** permite definir la altura del control. El atributo **multiple** permite elegir varias opciones simultáneamente (con ayuda de la tecla **Control** o **Mayúsculas**).

Se pueden agrupar opciones utilizando la etiqueta <optgroup>.

```
<select name="menu">
  <option selected="selected"></option>
  <optgroup label="Grupo1">
    <option value="1">Opción uno</option>
    <option value="2">Opción dos</option>
    <option value="3">Opción tres</option>
  </optgroup>
  <optgroup label="Grupo2">
    <option value="4">Opción cuatro</option>
    <option value="5">Opción cinco</option>
    <option value="6">Opción seis</option>
  </optgroup>
</select>
```

El atributo **value** de cada opción contiene el valor que envía el formulario si la opción está elegida. Si no se define el atributo value, el formulario envía como valor el texto que aparece en el menú.

### **Selector de archivo: <input type="file" />**

El selector de archivo se crea mediante la etiqueta <input /> cuyo atributo type tiene el valor file.

```
<input type="file" name="archivo" value="Selecciona un
archivo">
```

El atributo value no funciona ni con firefox ni con IE.

**Imagen: <input type="image" >**

El control de tipo imagen inserta una imagen que funciona como un botón (aunque los navegadores no le dan relieve como a los botones). Al hacer clic en un punto de la imagen se envía el formulario (como si se hubiera pulsado un botón submit) y se envían las coordenadas del punto en el que se ha hecho clic (junto con los valores de los otros controles del formulario).

```
<input type="image" name="GNU" alt="GNU" src="gnu.jpg" />
```

Si se define el atributo value, el formulario envía también el nombre del control con el valor del atributo.

**Control oculto: <input type="hidden" >**

El control oculto se crea mediante la etiqueta <input /> cuyo atributo type tiene el valor hidden. Lógicamente, los navegadores no muestran estos controles en la pantalla (aunque pueden verse en el código fuente). Se utilizan para almacenar información que de otro modo se perdería (por ejemplo, cuando hay varios formularios encadenados).

**Número: <input type="number" >**

El control de número se crea mediante la etiqueta <input /> cuyo atributo type tiene el valor number.

En principio, los valores admitidos por el control son únicamente números enteros. Los navegadores alertan al usuario cuando no ha escrito un valor correcto, como mínimo al enviar el formulario.

Escriba un número:

```
<input type="number" name="numero" />
```

Escriba un número mayor o igual que 20:

```
<input type="number" name="numero" min="20"/>
```

El atributo max establece un valor máximo en la respuesta. El atributo step establece la cantidad, entera o decimal, en la que aumenta o disminuye el valor. Para poder escribir cualquier número, el atributo step debe tener el valor any. Combinando los atributos min y step se obliga a que el valor escrito pertenezca a una sucesión aritmética.

Escriba un año en la que se haya celebrado un centenario  
de la llegada a América de Cristóbal Colón :

```
<input type="number" name="numero" min="1492" max="2015"  
step="100"/>
```

### **Rango: <input type="range" >**

El control de rango se crea mediante la etiqueta <input /> cuyo atributo type tiene el valor range. El rango de valores predeterminado es de 0 a 100 (valores enteros).

Los atributos min, max y step permiten elegir respectivamente el valor mínimo, el máximo y el incremento. El atributo list, debería añadir etiquetas al control.

Indique su nivel (Malo, Regular, Bueno):

```
<input type="range" name="rango"
  min="1" max="5" step="2" list="lista-rango" />
<datalist id="lista-rango">
  <option value="1" label="Malo"></option>
  <option value="3" label="Regular"></option>
  <option value="5" label="Bueno"></option>
</datalist>
```

### **Grupos de controles: <fieldset>**

La etiqueta <fieldset> permite agrupar un conjunto de controles. Los navegadores muestran una caja alrededor de cada grupo de controles.

La etiqueta <legend> permite añadir una leyenda al <fieldset>. Los navegadores muestran la leyenda sobre el borde que rodea el grupo de controles.

```
<fieldset><legend>Formulario</legend>
  <input type="text" name="texto"/><br/>
  <select name="menu">
    <option>Uno</option>
    <option>Dos</option>
    <option selected="selected">Tres</option>
  </select>
</fieldset>
```

Sólo puede haber una etiqueta <legend>.

### **Accesibilidad: <label>**

La etiqueta <label> permite asociar un control con un texto, de manera que mejore la accesibilidad de los formularios.

La asociación entre el control y la etiqueta <label> puede ser implícita o explícita. Se dice que la relación es implícita cuando el control se encuentra en el interior de la etiqueta. Se dice que la relación es explícita cuando la etiqueta <label> contiene el atributo for, que indica el control afectado (el control tiene entonces que tener establecido el atributo id).

Por ejemplo, en el caso de una casilla de verificación, la etiqueta <label> permite que la casilla se marque o desmarque haciendo clic en el texto

```
<input type="checkbox" name="casilla" id="casilla1" />
<label for="casilla1">Casilla 1</label>
```