

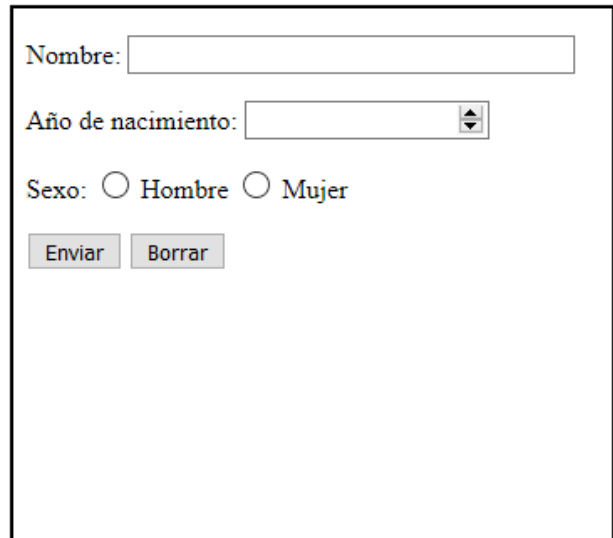
Formularios

El formulario: **<form>**

<form>

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.

```
<form action="ejemplo.php" method="get">
  <p>Nombre: <input type="text"
name="nombre" size="40"></p>
  <p>Año de nacimiento: <input type="number"
name="nacido" min="1900"></p>
  <p>Sexo:
    <input type="radio" name="hm" value="h">
Hombre
    <input type="radio" name="hm" value="m">
Mujer
  </p>
  <p>
    <input type="submit" value="Enviar">
    <input type="reset" value="Borrar">
  </p>
</form>
```



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, el formulario se comporta 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.

Las 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

Algunos atributos se pueden utilizar en varios tipos de controles.

Falta comentar algunos atributos como autocomplete, height, list, pattern, title, width.

El atributo type

El atributo type de la etiqueta <input> indica el tipo de control de que se trata.

En HTML 4.01, los tipos de <input> eran: botón para enviar [submit](#), botón para reiniciar [reset](#), texto [text](#), contraseña [password](#), archivo [file](#), imagen [image](#) y oculto [hidden](#).

En HTML 5 se añadieron los tipos de <input>: botón [button](#), número [number](#), búsqueda [search](#), teléfono [tel](#), dirección web [url](#), dirección de correo [email](#), fecha [date](#), hora [time](#), color [color](#) y rango [range](#).

En HTML 5.1 se añadieron los tipos de <input>: mes [month](#), semana [week](#) y fecha y hora local [datetime-local](#).

El atributo name

El atributo [name](#) identifica al control.

Todos los controles de un formulario deben tener el atributo [name](#) porque, en general, sólo se envían los controles que lo tienen (aunque hay excepciones, como los controles de tipo imagen o los botones, que no los necesitan).

En general, los atributos [name](#) no se deben repetir, porque no se recibiría toda la información al enviarla. La única excepción es el [botón radio](#), como se explica en el apartado correspondiente.

El atributo value

El atributo [value](#) permite establecer el valor inicial de un control, aunque cada control lo utiliza de una forma ligeramente distinta y algunos controles no lo admiten (área de texto, archivo o imagen).

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) 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, los navegadores no lo admiten por motivos de seguridad.

El atributo **required**

El atributo **required** permite indicar qué controles es obligatorio rellenar para enviar el formulario.

```
<p>Nombre: <input type="text" name="nombre"
required></p>

<p>Sexo:
  <input type="radio" name="hm" value="h" required>
Hombre
  <input type="radio" name="hm" value="m" required>
Mujer
</p>
```

Nombre:

Sexo: ☐ Hombre ☐ Mujer

Si alguna de las opciones de un botón radio tiene establecido el atributo **required**, el control es obligatorio.

```
<p>Sexo:
  <input type="radio" name="hm" value="h" required>
Hombre
  <input type="radio" name="hm" value="m"> Mujer
</p>

<p>Edad:
  <input type="radio" name="edad" value="menor">
Menor
  <input type="radio" name="edad" value="adulto"
required> Adulto
</p>
```

Sexo: ☐ Hombre ☐ Mujer

Edad: ☐ Menor ☐ Adulto

El atributo **placeholder**

El atributo **placeholder** permite mostrar en los controles de texto un texto que desaparece al escribir en el control.

```
<p>Apellidos:
  <input type="text" name="apellidos" size="40"
placeholder="Escriba sus dos apellidos">
</p>
```

Apellidos:

El atributo **size**

El atributo **size** permite establecer la longitud de los controles de texto (text, password, search, etc.). Por omisión, las cajas suelen tener 20 caracteres de longitud.

```
<p>Nombre: <input type="text" name="nombre"></p>
<p>Apellidos: <input type="text" name="apellidos"
size="40"></p>
```

Nombre:

Apellidos:

Los atributos **maxlength** y **minlength**

Los atributos **maxlength** y **minlength** permiten establecer la longitud máxima y mínima, respectivamente, que puede escribir el usuario en un control de texto (**text**, **password**, **search**, etc.).

```
<p>Usuario: <input type="text" name="usuario"
maxlength="6"></p>
<p>Contraseña: <input type="password"
name="password" minlength="6"></p>
```

Usuario:

Contraseña:

El atributo **autofocus**

El atributo **autofocus** permite indicar el control que debe tener el foco al cargarse el formulario.

Este atributo facilita la accesibilidad del formulario pues permite que el usuario pueda empezar a rellenar el formulario sin necesidad de hacer clic en el elemento. Como el navegador desplaza automáticamente la página hasta la posición del elemento, se recomienda que el elemento con atributo **autofocus** se encuentren al principio de la página, para que el desplazamiento no confunda al usuario.

El atributo **disabled**

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

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

```
<input type="text" name="texto" disabled>
```

El atributo **readonly**

El atributo **readonly** hace 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>
```

¡A que no me cambias!

El atributo **tabindex**

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

Texto 1	Texto 2
Texto 3	Texto 4

```
<input type="text" name="texto1" value="Texto 1"
tabindex="5">
<input type="text" name="texto2" value="Texto 2"
tabindex="7">
<input type="text" name="texto3" value="Texto 3"
tabindex="6">
<input type="text" name="texto4" value="Texto 4"
tabindex="8">
```

Texto 1	Texto 2
Texto 3	Texto 4

El atributo **accesskey**

El atributo **accesskey** establece la combinación de teclas que coloca el foco en un elemento de un formulario.

- Firefox utiliza la combinación Alt+Shift+tecla
- Google Chrome utilizan normalmente la combinación Alt+tecla, pero cuando esa combinación ya está asociada a alguna acción usa la combinación Alt+Shift+tecla
- Edge combina ambos comportamientos: utiliza la combinación Alt+Shift+tecla, pero también permite la combinación Alt+tecla cuando no está asociada a ninguna acción

En el ejemplo siguiente Firefox utiliza la combinación Alt+Shift+tecla para situar el cursor en las cajas de texto, Google Chrome utiliza la combinación Alt+tecla y Edge utiliza ambas:

```
<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">
```

Acceso con a	Acceso con i
Acceso con o	Acceso con u

Chrome utiliza la combinación Alt+tecla, que tiene el inconveniente que entra en conflicto con combinaciones de teclas ya definidas por el navegador o el sistema operativo. Firefox utiliza la combinación Alt+Shift+tecla, más incómoda de utilizar, pero que no interfiere con las combinaciones ya definidas por el navegador o el sistema operativo.

En el ejemplo siguiente, Chrome y Edge no pueden acceder a la caja de texto con la combinación Alt+d, ya que esta combinación traslada el foco a la barra de dirección, pero pueden hacerlo con la combinación Alt+Shift+d:

<pre><input type="text" name="texto1" value="Acceso con d" accesskey="d"></pre>	<input type="text" value="Acceso con d"/>
---	---

En el ejemplo siguiente, Chrome no puede acceder a la caja de texto con la combinación Alt+e, ya que esta combinación abre el menú de herramientas, pero puede hacerlo con la combinación Alt+Shift+e:

<pre><input type="text" name="texto2" value="Acceso con e" accesskey="e"></pre>	<input type="text" value="Acceso con e"/>
---	---

Controles: **<input>** y otros

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

La mayoría de controles se obtienen mediante la etiqueta vacía `<input>`.

En HTML 4.01, los controles disponibles eran: botón para enviar `<input type="submit">`, botón para reiniciar `<input type="reset">`, botón `<button>`, texto `<input type="text">`, contraseña `<input type="password">`, archivo `<input type="file">`, imagen `<input type="image">`, oculto `<input type="hidden">`, área de texto `<textarea>` y menú `<select>`.

En HTML 5 se añadieron los controles: botón general `<input type="button">`, número `<input type="number">`, búsqueda `<input type="search">`, teléfono `<input type="tel">`, dirección web `<input type="url">`, dirección de correo `<input type="email">`, fecha `<input type="date">`, hora `<input type="time">`, color `<input type="color">`, rango `<input type="range">`, opciones `<option>` (para `<datalist>`), lista de valores `<datalist>`, salida `<output>`, barra de progreso `<progress>` y medidor `<meter>`.

En HTML 5.1 se añadieron los controles: mes `<input type="month">`, semana `<input type="week">` y fecha y hora local `<input type="datetime-local">`.

Falta comentar algunos controles HTML 5 como `<output>`, `<progress>` y `<meter>`.

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

Los botones se crean mediante la etiqueta `<input>` o mediante la etiqueta `<button>`. La diferencia entre ellos es que `<input>` sólo puede contener texto, mientras que `<button>` permite incluir elementos html como imágenes.


El contenido del botón `<input>` se define mediante el atributo `value`, por lo que sólo puede contener texto.

```
<input type="button" value="Este botón no hace nada">
```

Este botón no hace nada

El contenido del botón `<button>` se escribe dentro del elemento, por lo que puede incluir texto e imágenes.

```
<button type="button">
  
  Este botón no hace nada
</button>
```

 Este botón no hace nada

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 quisiéramos saber en cuál de ellos ha hecho clic el usuario, pero esto no es muy habitual).

Botones Submit mediante `<input>` o `<button>`

El botón Submit es el que permite al usuario remitir los datos al servidor. Se crea mediante una etiqueta `<input>` o `<button>` cuyo atributo `type` tiene el valor `submit`.


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

Escriba algo:

Enviar

```
<button type="submit">
  
  Enviar
</button>
```


Escriba algo:

 Enviar

Si un botón `<button>` no lleva el atributo `type`, se comporta como un botón de tipo

```
<button>
  
  Enviar
</button>
```

Escriba algo:

 Enviar

Botones Reset mediante `<input>` o `<button>`

El botón Reset restablece los valores iniciales del formulario. Se crea mediante una etiqueta `<input>` o `<button>` cuyo atributo `type` tiene el valor `reset`.

```
<input type="reset" value="Borrar">
```

Caja de texto:

☐ Casilla de verificación


☐ Radio 1 ☐ Radio 2 ☐ Radio 3

```
<button type="reset">
  
  Borrar
</button>
```

Caja de texto:

☐ Casilla de verificación

☐ Radio 1 ☐ Radio 2 ☐ Radio 3

 Borrar

Si una caja de texto u otro control similar tiene establecido el atributo `value`, el botón Reset restablece ese valor si se ha modificado, pero no deja vacío el control.

```
<p>Caja de texto: <input type="text" name="texto"
value="Bla bla bla"></p>
<p><input type="reset" value="Borrar"></p>
```

Caja de texto:

Botón general mediante `<input>` o `<button>`

Un botón de tipo general se suele asociar a un script para hacer algo. Se crea mediante una etiqueta `<input>` o `<button>` cuyo atributo `type` tiene el valor `button`.



```
<button type="button" onclick="mensaje('Ha pulsado el botón')">Botón</button>
```

```
<script>
  function mensaje(texto) {
    alert(texto);
  }
</script>
```

Botón

```
<button type="button" onclick="mensaje('Ha pulsado el botón')">
  
  Botón
</button>
```

```
<script>
  function mensaje(texto) {
    alert(texto);
  }
</script>
```

 Botón

Caja de texto: `<input type="text">`

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

```
<input type="text" name="texto">
```

Escriba algo:

Enviar

El atributo `value` (optativo) permite incluir un texto en la caja de texto para que el usuario lo modifique. HTML 5 introdujo el atributo `placeholder`, mucho más cómodo para el usuario, ya que no tiene que borrar el texto de ayuda.

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

Escriba algo

```
<input type="text" name="texto"
placeholder="Escriba algo">
```

Escriba algo

Caja de texto de contraseña: `<input type="password">`

Las cajas de texto de una sola línea específicas para **contraseñas** se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `password`.

```
<input type="password" name="contrasena">
```

Contraseña:

Enviar

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.

Número: `<input type="number">`

Las cajas de texto de una sola línea específicas para **números** se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `number`.

En principio, los valores admitidos por el control son números enteros, aunque el atributo `step` permite aceptar números decimales. Los navegadores alertan al usuario cuando no ha escrito un valor correcto, como mínimo al enviar el formulario.

En el caso de los números decimales:

- Chrome admite la coma (,) y el punto (.) como separadores de la parte decimal.
- Firefox sólo admite la coma (,) como separador de la parte decimal.
- Edge sólo admite el punto (.) como separador de la parte decimal.

Escriba un número:
`<input type="number" name="numero">`

Escriba un número:

Enviar

El atributo `min` establece un valor mínimo en la respuesta:

Escriba un número mayor o igual que 20:
`<input type="number" name="numero" min="20">`

Escriba un número mayor o igual que 20:

Enviar

El atributo `max` establece un valor máximo en la respuesta:

Escriba un número menor o igual que 100:
`<input type="number" name="numero" max="100">`

Escriba un número menor o igual que 100:

Enviar

El atributo `step` establece la cantidad, entera o decimal, en la que aumenta o disminuye el valor:

Escriba un múltiplo de 10:
<input type="number" name="numero" step="10">

Escriba un múltiplo de 10:
Enviar

Escriba un valor de 0,5 en 0,5:
<input type="number" name="numero" step="0.5">

Escriba un valor de 0,5 en 0,5:
Enviar

Para poder escribir cualquier número, el atributo **step** debe tener el valor **any**:

Escriba un número cualquiera:
<input type="number" name="numero" step="any">

Escriba un número cualquiera:
Enviar

Búsqueda: <input type="search">

Las cajas de texto de una sola línea específicas para **búsquedas** se crean mediante una etiqueta <input> cuyo atributo **type** tiene el valor [search](#).



Búsqueda: <input type="search" name="busqueda" size="40">

Búsqueda:
Enviar

La recomendación HTML 5 aclara que la diferencia entre [text](#) y [search](#) es únicamente de estilo y que el cuadro de búsqueda debería tener el estilo de los cuadros de búsqueda del sistema utilizado.

Teléfono: <input type="tel">

Las cajas de texto de una sola línea específicas para **números de teléfonos** se crean mediante una etiqueta <input> cuyo atributo **type** tiene el valor **tel**.

Teléfono: <input type="tel" name="telefono" size="40">

Teléfono:
Enviar

La recomendación HTML 5 aclara que los formatos de números de teléfono siguen formatos tan variados que este control no obliga a utilizar ninguna sintaxis en particular.

URL: <input type="url">

Las cajas de texto de una sola línea específicas para **direcciones web (URL)** se crean mediante una etiqueta `<input>` cuyo atributo **type** tiene el valor `url`.



```
URL: <input type="url" name="direccion"
size="40">
```

URL:

Enviar

Actualmente (octubre de 2018), los navegadores sólo comprueban que el texto escrito empiece por `http:` o `https:`, pero no obligan a que las URL sean sintácticamente válidas.

Dirección de correo electrónico: `<input type="email">`

Las cajas de texto de una sola línea específicas para **direcciones de correo electrónico** se crean mediante una etiqueta `<input>` cuyo atributo **type** tiene el valor `email`.

```
Correo: <input type="email" name="correo"
size="40">
```

Correo:

Enviar

Actualmente (octubre de 2018), los navegadores sólo comprueban que el texto escrito contenga una arroba `@` y no contengan caracteres no permitidos, pero no obliga a que las direcciones sean válidas.

Fecha: `<input type="date">`

Las cajas de texto de una sola línea específicas para **fechas (días, meses, años)** se crean mediante una etiqueta `<input>` cuyo atributo **type** tiene el valor `date`. El dato se envía con el formato AAAA-MM-DD donde AAAA es el número de año, MM el número de mes y DD el número de día.

```
Fecha: <input type="date" name="fecha" size="20">
```

Fecha:

Enviar

Mes: `<input type="month">`

Las cajas de texto de una sola línea específicas para **meses (y años)** se crean mediante una etiqueta `<input>` cuyo atributo **type** tiene el valor `month`. El dato se envía con el formato AAAA-MM donde AAAA es el número de año y MM el número de mes.



```
Mes: <input type="month" name="mes" size="20">
```

Mes:

Enviar

Semana: `<input type="week">`



```
Semana: <input type="week" name="semana"
size="20">
```

Semana:

Enviar

Las cajas de texto de una sola línea específicas para **semanas (y años)** se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `week`. El dato se envía con el formato AAAA-WSS donde AAAA es el número de año y SS el número de semana.



```
Semana: <input type="week" name="semana"
size="20">
```

Semana:

Enviar

Hora: `<input type="time">`

Las cajas de texto de una sola línea específicas para **tiempos (horas, minutos)** se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `time`. El dato se envía con el formato HH-MM donde HH son las horas y MM los minutos.

```
Hora: <input type="time" name="hora" size="20">
```

Hora:

Enviar

Fecha y hora: `<input type="datetime-local">`

Las cajas de texto de una sola línea específicas para **fechas (días, meses, años, horas y minutos)** se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `datetime-local`. El dato se envía con el formato AAAA-MM-DDTHH-MM donde AAAA es el número de año, MM el número de mes, DD el número de día, HH son las horas y MM los minutos.



```
Fecha y hora: <input type="datetime-local"
name="fecha" size="20">
```

Fecha y hora local:

Enviar

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

Las casillas de verificación se crean mediante una etiqueta `<input>` cuyo atributo `type` tiene el valor `checkbox`.

```
<input type="checkbox" name="casilla">Casilla 1
```

☐ Casilla 1

Si el atributo **checked** tiene el valor **checked**, la casilla aparece marcada.

```
<input type="checkbox" name="casilla" checked>Casilla 1
```

☒ Casilla 1

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**.

```
<input type="checkbox" name="casilla">Casilla 1
```

☐ Casilla de verificación

Enviar

```
<input type="checkbox" name="casilla" value="hola">Casilla 1
```

☐ Casilla de verificación

Enviar

Botón radio: **<input type="radio">**

Los botones radio se crean mediante una etiqueta **<input>** cuyo atributo **type** tiene el valor **radio**.

```
<input type="radio" name="radio" value="1">Opción 1
```

☐ Opción 1

Los 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.

```
<input type="radio" name="radio" value="1">Opción 1  
<input type="radio" name="radio" value="2">Opción 2
```

☐ Opción 1
☐ Opción 2

Los dos ejemplos anteriores, aunque estén separados, forman el mismo botón radio ya que su atributo **name** tiene el mismo valor (en este caso "radio"). Se puede comprobar pulsando en uno de los ejemplos y observando cómo se desmarca el otro ejemplo. Para que fueran independientes, bastaría con que sus atributos **name** fueran distintos, como en el ejemplo siguiente:

```
<input type="radio" name="boton1"> Hombre
<input type="radio" name="boton1"> Mujer
```

☐ Hombre ☐ Mujer

```
<input type="radio" name="boton2" value="hombre">
Hombre
<input type="radio" name="boton2" value="mujer">
Mujer
```

☐ Hombre ☒ Mujer

Si uno de los botones tiene el atributo **checked** con el valor **checked**, el botón aparece marcado.

```
<input type="radio" name="radio3" value="1">Opción
1<br>
<input type="radio" name="radio3" value="2"
checked>Opción 2
```

☐ Opción 1

☒ Opción 2

Los botones radio sólo se envían si se han marcado. El atributo **value** contiene el valor que envía el formulario si el botón radio está marcado. Si el atributo **value** no está establecido, el formulario envía el valor **on**, así que para poder saber cuál ha sido la opción elegida por el usuario es necesario establecer con valores distintos los atributos **value** de todos los elementos de un botón radio.

```
<input type="radio" name="boton1"> Hombre
<input type="radio" name="boton1"> Mujer
```

☐ Hombre ☒ Mujer

```
<input type="radio" name="boton2" value="hombre">
Hombre
<input type="radio" name="boton2" value="mujer">
Mujer
```

☐ Hombre ☒ Mujer

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


El control de selector de color se crea mediante una 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">
```

Elija un color:

El atributo **value** permite especificar un color inicial.

```
<input type="color" name="color" value="#ff0000">
```


Elija un color: 

Enviar

Rango: `<input type="range">`

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

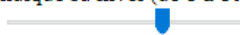
Indique su nivel:
`<input type="range" name="rango">`

Indique su nivel: 

Enviar

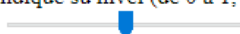
Los atributos `min`, `max` y `step` permiten elegir respectivamente el valor mínimo, el máximo y el incremento.

Indique su nivel (de 1 a 10, de 3 en 3):
`<input type="range" name="rango"
min="1" max="10" step="3">`

Indique su nivel (de 1 a 10, de 3 en 3): 

Enviar

Indique su nivel (de 0 a 1, de 0,1 en 0,1):
`<input type="range" name="rango"
min="0" max="1" step="0.1">`

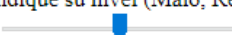
Indique su nivel (de 0 a 1, de 0,1 en 0,1): 

Enviar

El atributo `list` debería añadir etiquetas al control.



Indique su nivel (Malo, Regular, Bueno):
`<input type="range" name="rango"
min="1" max="3" list="lista-rango">
<datalist id="lista-rango">
 <option value="1" label="Malo"></option>
 <option value="2" label="Regular"></option>
 <option value="3" label="Bueno"></option>
</datalist>`

Indique su nivel (Malo, Regular, Bueno): 

Enviar

Selector de archivo: `<input type="file">`

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

```
<input type="file" name="archivo">
```

Examinar... No se ha seleccionado ningún archivo.

Tradicionalmente, los navegadores no han admitido el atributo **value** en el selector de archivo por motivos de seguridad, ya que podría utilizarse para "robar" ficheros al usuario sin su autorización. La recomendación HTML 5 indica que el atributo **value** se puede utilizar para seleccionar selectores de archivos distintos al general, pero los navegadores no parecen utilizar este atributo.



```
<input type="file" name="archivo" value="image/*">
```

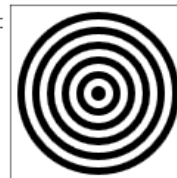
Examinar... No se ha seleccionado ningún archivo.

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="diana" src="diana.svg">
```

Haga clic en algún punto de la imagen:



Si se define el atributo **value**, el formulario debe enviar tanto las coordenadas como el nombre del control con el valor del atributo **value**.



```
<input type="image" name="diana" value="click me" src="diana.svg">
```

Haga clic en algún punto de la imagen:



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

El control oculto se crea mediante una 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).

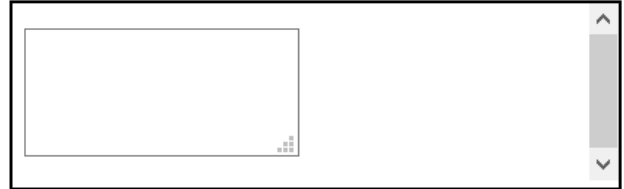
```
<input type="hidden" name="nombre" value="pepito">
```

Enviar

Área de texto: `<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 iniciales de la caja, aunque los navegadores permiten modificarlo arrastrando la esquina inferior derecha.

```
<textarea name="texto" rows="4" cols="20">
</textarea>
```

A visual representation of a text area created with the provided HTML code. It shows a rectangular box with a light gray border and a vertical scrollbar on the right side. The box is empty, representing the initial state of the text area.

```
<textarea name="texto" rows="3" cols="30">
</textarea>
```

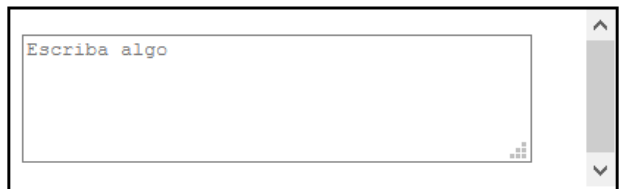
A visual representation of a text area created with the provided HTML code. It shows a rectangular box with a light gray border and a vertical scrollbar on the right side. The box is empty, representing the initial state of the text area.

Si se escribe texto en el interior de la etiqueta, este se muestra en la caja de texto. HTML 5 introdujo el atributo `placeholder` que puede ser más cómodo para el usuario si tiene que borrar todo el texto incluido.

```
<textarea name="texto" rows="4" cols="40">Escriba
algo</textarea>
```

A visual representation of a text area created with the provided HTML code. It shows a rectangular box with a light gray border and a vertical scrollbar on the right side. The box contains the text "Escriba algo" in a light gray font, which is the placeholder text.

```
<textarea name="texto" rows="4" cols="40"
placeholder="Escriba algo"></textarea>
```

A visual representation of a text area created with the provided HTML code. It shows a rectangular box with a light gray border and a vertical scrollbar on the right side. The box contains the text "Escriba algo" in a light gray font, which is the placeholder text.

Menú: `<select>`

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

```
<select name="menu">
  <option>Uno</option>
  <option>Dos</option>
  <option>Tres</option>
</select>
```

A visual representation of a select menu created with the provided HTML code. It shows a label "Elija una opción:" followed by a dropdown menu with the text "Uno" and a downward arrow. Below the dropdown menu is a button labeled "Enviar".

El valor que se envía es el texto que aparece en el menú, salvo si el elemento `<option>` contiene el atributo `value`.

```
<select name="menu">
  <option value="1">Uno</option>
  <option>Dos</option>
  <option value="3">Tres</option>
</select>
```

Elija una opción:

El atributo **selected** indica la opción por omisión.

```
<select name="menu">
  <option value="1">Uno</option>
  <option value="2">Dos</option>
  <option value="3" selected>Tres</option>
</select>
```

Elija una opción:

Si ningún elemento posee el atributo **selected**, los navegadores muestran la primera opción del menú.

```
<select name="menu">
  <option value="1">Uno</option>
  <option value="2">Dos</option>
  <option value="3">Tres</option>
</select>
```

Elija una opción:

El problema de mostrar una de las opciones de forma predeterminada es que al recibir el formulario no se puede saber si el usuario ha elegido realmente esa opción o simplemente no ha modificado el control. Por ello, se aconseja incluir una opción en blanco al principio de los menús

```
<select name="menu">
  <option value="0">...</option>
  <option value="1">Uno</option>
  <option value="2">Dos</option>
  <option value="3">Tres</option>
</select>
```

Elija una opción:

El atributo **size** permite definir la altura del control.

```
<select name="menu" size="3">
  <option value="1">Uno</option>
  <option value="2">Dos</option>
  <option value="3">Tres</option>
  <option value="4">Cuatro</option>
  <option value="5">Cinco</option>
</select>
```

Elija una opción:

El atributo **multiple** permite elegir varias opciones simultáneamente (con ayuda de la tecla Control o Mayúsculas)

```
<select name="menu" size="5" multiple>
  <option value="1">Uno</option>
  <option value="2">Dos</option>
  <option value="3">Tres</option>
  <option value="4">Cuatro</option>
  <option value="5">Cinco</option>
</select>
```

Elija una opción:

- Uno
- Dos
- Tres
- Cuatro
- Cinco

Enviar

Se pueden agrupar opciones utilizando la etiqueta **<optgroup>**.

```
<select name="menu">
  <option 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>
```

Elija una opción: ...

Enviar

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>Datos personales</legend>
  <p>Nombre: <input type="text" name="nombre"
size="30"></p>
  <p>Edad: <input type="number" name="edad"></p>
</fieldset>

<fieldset>
  <legend>Cultura general</legend>
  <p>Capital de Noruega:
    <input type="radio" name="p1" value="1">
Copenhague
    <input type="radio" name="p1" value="2">
Helsinki
    <input type="radio" name="p1" value="3"> Oslo
  </p>
  <p>Fecha de la Revolución francesa:
    <input type="radio" name="p2" value="1"> 1492
    <input type="radio" name="p2" value="2"> 1789
    <input type="radio" name="p2" value="2"> 1917
  </p>
</fieldset>
```

Datos personales

Nombre:

Edad:

Cultura general

Capital de Noruega: ☐ Copenhague ☐ Helsinki ☐ Oslo

Fecha de la Revolución francesa: ☐ 1492 ☐ 1789 ☐ 1917

Enviar Borrar

Sólo puede haber una etiqueta **<legend>**.

Nota:

- [The state of fieldset interoperability](#), 19/09/2018, WHATWG blog.

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.

Relación implícita

Se dice que la relación es implícita cuando el control se encuentra en el interior de la etiqueta.

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, como se muestra en los ejemplos siguientes:

<code><input type="checkbox" name="casilla">Casilla 1</code>	<input type="checkbox"/> Casilla 1
<code><label><input type="checkbox" name="casilla">Casilla 1</label></code>	<input type="checkbox"/> Casilla 1

En el caso de las cajas de texto **<input type="text">**, la etiqueta **<label>** permite que el cursor se sitúe en la caja de texto haciendo clic en el texto, como se muestra en los ejemplos siguientes:

Nombre: <code><input type="text" name="casilla"></code>	Nombre: <input type="text"/>
<code><label>Nombre: <input type="text" name="casilla"></label></code>	Nombre: <input type="text"/>

Relación explícita: el atributo **for**

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, como se muestra en los ejemplos siguientes:

```
<input type="checkbox" name="casilla">Casilla 1
```

☐ Casilla 1

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

☒ Casilla 1

En el caso de las cajas de texto `<input type="text">`, la etiqueta `<label>` permite que el cursor se sitúe en la caja de texto haciendo clic en el texto, como se muestra en los ejemplos siguientes:

```
Nombre: <input type="text" name="casilla">
```

Nombre:

```
<label for="text1">Nombre:</label>
<input type="text" name="casilla" id="text1">
```

Nombre:

La forma explícita se necesita cuando el control y el texto asociado se encuentran en elementos distintos (por ejemplo en una tabla) y la etiqueta `<label>` no puede incluir ambos, como muestra el ejemplo siguiente:

```
<table>
  <tbody>
    <tr>
      <th>Sexo</th>
      <th></th>
    </tr>
    <tr>
      <td><input type="radio" name="hm" value="h"
id="h"></td>
      <td><label for="h">Hombre</label></td>
    </tr>
    <tr>
      <td><input type="radio" name="hm" value="m"
id="m"></td>
      <td><label for="m">Mujer</label></td>
    </tr>
  </tbody>
</table>
```

Sexo	
<input type="radio"/>	Hombre
<input type="radio"/>	Mujer