

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>
```

Nombre:

Año de nacimiento:

Sexo: ☐ Hombre ☐ Mujer

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.

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

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>  
  
<p>Nacionalidad:  <input  type="text"  name="nacionalidad"  
size="20"></p>
```

Nombre:

Apellidos:

Nacionalidad:

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.

Nota: En estos apuntes (lecciones y ejercicios) no se utiliza este atributo porque como en las páginas se incluyen varios formularios en distintas posiciones de la página, el desplazamiento automático a un elemento con atributo **autofocus** puede confundir al lector.

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>
```

A rectangular button with a light gray background and a thin border. The text 'Enviar' is centered on the button. The button is disabled, as indicated by the 'disabled' attribute in the code above.

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

A rectangular text input field with a light gray background and a thin border. The field is empty. The input is disabled, as indicated by the 'disabled' attribute in the code above.

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>
```

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

Two rectangular text input fields, one above the other, both with a light gray background and a thin border. The first field contains the text '¡A que no me cambias!'. The second field is empty and has a placeholder text '¡A que no me cambias!'. Both inputs are readonly, as indicated by the 'readonly' attribute in the code above.

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" placeholder="Texto 1">  
<input type="text" name="texto2" placeholder="Texto 2">  
<input type="text" name="texto3" placeholder="Texto 3">  
<input type="text" name="texto4" placeholder="Texto 4">
```

| | | |
|---------|---------|---------|
| Texto 1 | Texto 2 | Texto 3 |
| Texto 4 | | |

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

| | | |
|---------|---------|---------|
| Texto 1 | Texto 2 | Texto 3 |
| Texto 4 | | |

El atributo **accesskey**

El atributo **accesskey** establece la tecla que coloca el foco en un elemento de un formulario, pero los navegadores Google Chrome, Firefox y Microsoft Edge no utilizan los mismos atajos de teclado, como se comenta en la [lección de diferencias entre navegadores](#).

- 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:

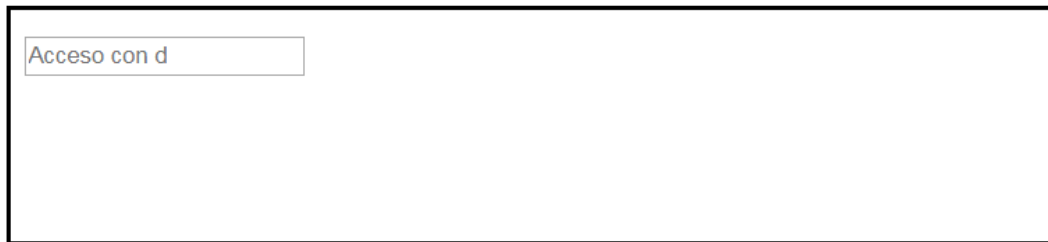
```
<input type="text" name="texto1" placeholder="Acceso con a"
accesskey="a">
<input type="text" name="texto2" placeholder="Acceso con i"
accesskey="i">
<input type="text" name="texto3" placeholder="Acceso con o"
accesskey="o">
<input type="text" name="texto4" placeholder="Acceso con u"
accesskey="u">
```

| | | |
|--------------|--------------|--------------|
| 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**:

```
<input type="text" name="texto1" placeholder="Acceso con d"
accesskey="d">
```

A screenshot of a web browser window. Inside the window, there is a single text input field. The input field has a light gray border and contains the placeholder text "Acceso con d" in a light gray font. The rest of the browser window is empty.

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**:

```
<input type="text" name="texto2" placeholder="Acceso con e"
accesskey="e">
```

A screenshot of a web browser window. Inside the window, there is a single text input field. The input field has a light gray border and contains the placeholder text "Acceso con e" in a light gray font. The rest of the browser window is empty.

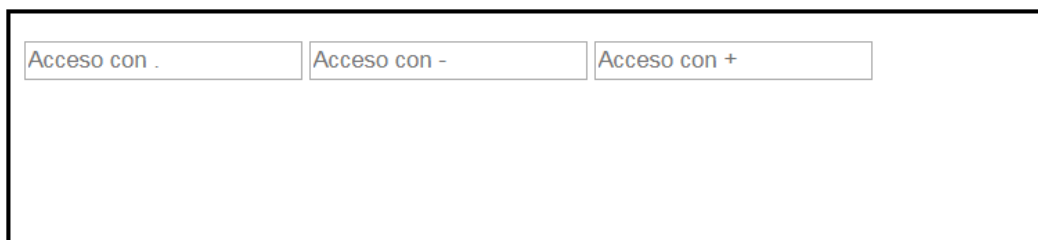
En el caso de las teclas de acceso que no son las letras del alfabeto (los acentos no pueden ser teclas de acceso):

- Firefox utiliza la combinación **Alt+Shift+tecla**
- Chrome y Edge utilizan la combinación **Alt+tecla**

```
<input type="text" name="texto1" placeholder="Acceso con 1"
accesskey="1">
<input type="text" name="texto2" placeholder="Acceso con 2"
accesskey="2">
<input type="text" name="texto3" placeholder="Acceso con 3"
accesskey="3">
<input type="text" name="texto4" placeholder="Acceso con 4"
accesskey="4">
```

A screenshot showing four text input fields arranged in a 2x2 grid. The top row contains two fields: 'Acceso con 1' and 'Acceso con 2'. The bottom row contains two fields: 'Acceso con 3' and 'Acceso con 4'. Each field has a small icon on the left representing the access key.

```
<input type="text" name="texto1" placeholder="Acceso con ,"
accesskey=",">
<input type="text" name="texto2" placeholder="Acceso con ."
accesskey=".">
<input type="text" name="texto3" placeholder="Acceso con +"
accesskey="+">
```

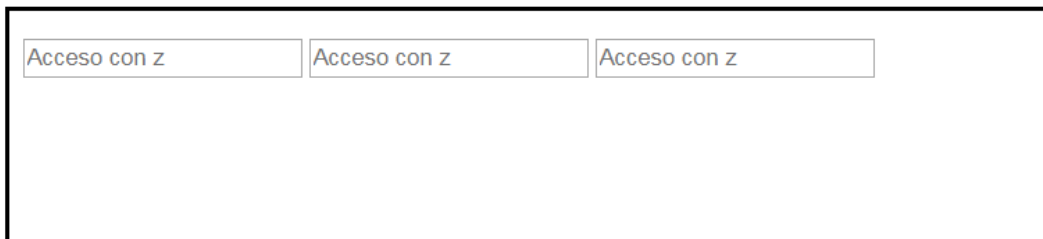
A screenshot showing three text input fields arranged horizontally. The first field is 'Acceso con ,', the second is 'Acceso con .', and the third is 'Acceso con +'. Each field has a small icon on the left representing the access key.

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:

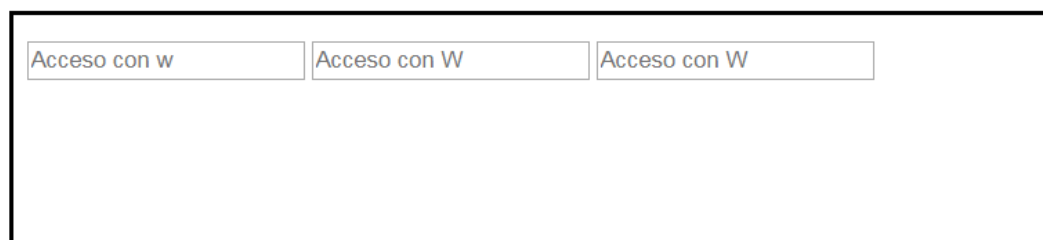
- Google Chrome no avanza de uno a otro
- Firefox avanza de uno a otro con la combinación **Alt+Shift+tecla**

- Edge avanza de uno a otro con la combinación **Alt+tecla** y retrocede de uno a otro con **Alt+Shift+tecla**

```
<input type="text" name="texto1" placeholder="Acceso con z"
accesskey="z">
<input type="text" name="texto2" placeholder="Acceso con z"
accesskey="z">
<input type="text" name="texto3" placeholder="Acceso con z"
accesskey="z">
```

A screenshot showing three text input fields arranged horizontally. Each field contains the placeholder text "Acceso con z". The fields are enclosed in a black rectangular border.

```
<input type="text" name="texto1" placeholder="Acceso con w"
accesskey="w">
<input type="text" name="texto2" placeholder="Acceso con W"
accesskey="W">
<input type="text" name="texto3" placeholder="Acceso con W"
accesskey="W">
```

A screenshot showing three text input fields arranged horizontally. The first field contains the placeholder text "Acceso con w", and the other two fields contain "Acceso con W". The fields are enclosed in a black rectangular border.

Controles: **<input>** y otros

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

La mayoría de controles se obtienen mediante 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 `<datalist>`, `<meter>`, `<output>` y `<progress>`.

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

`<input submit>`

- -

`<input reset>`

- -

`<input button>`

- -

`<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">
```



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>
```



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:

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


Escriba algo:

 Enviar

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

```
<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 tienen establecidos los atributos `value` o `placeholder`, el botón Reset restablece estos valores, pero no deja vacío el control.

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


Caja de texto:

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

`<p>Caja de texto: <input type="text" name="texto" placeholder="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`.

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

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



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

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



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

```
<input text>
```

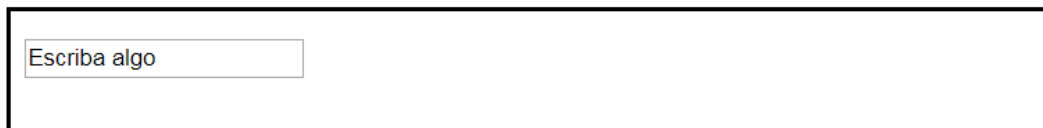
- -

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

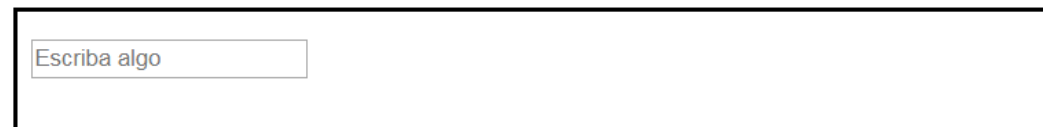
`<input type="text" name="texto">`

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">`

A screenshot of a web form. It features a single text input field with a light gray border. Inside the field, the text "Escriba algo" is displayed in a light gray font, serving as a placeholder. The field is centered within a larger white rectangular container.

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

A screenshot of a web form, similar to the previous one. It shows a text input field with a light gray border. The placeholder text "Escriba algo" is visible in a light gray font. The field is centered within a larger white rectangular container.

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

`<input 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">`

A screenshot of a web form. On the left, the label "Contraseña:" is followed by a password input field with a light gray border. Below the input field is a button with a light gray background and the text "Enviar". The entire form is enclosed in a black rectangular border.

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">`

`<input 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.

Escriba un número:

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

Escriba un número:

El atributo `step` permite que el control admita 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 cualquiera:

`<input type="number" name="numero" step="any">`

Escriba un número cualquiera:

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:

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:

Búsqueda: `<input type="search">`

`<input 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:

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">`

<input 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">`

Un formulario web que contiene el texto "Teléfono:" seguido de un campo de entrada de texto rectangular. Debajo del campo de entrada hay un botón rectangular con el texto "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.

En los teléfonos móviles, al escribir en una caja de texto de tipo `tel` se muestra un teclado numérico.

URL: `<input type="url">`

<input 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">`

Un formulario web que contiene el texto "URL:" seguido de un campo de entrada de texto rectangular. Debajo del campo de entrada hay un botón rectangular con el texto "Enviar".

Actualmente (octubre de 2019), los navegadores sólo comprueban que el texto escrito empiece por `http:` o `https:`, pero no obligan a que las URL sean sintácticamente válidas.

En los teléfonos móviles, al escribir en una caja de texto de tipo `url` se adapta ligeramente el teclado de texto, mostrándose el carácter `/`.

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

`<input 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">`

Un formulario web que contiene un campo de texto etiquetado como 'Correo:' y un botón de envío etiquetado como 'Enviar'.

Actualmente (octubre de 2019), 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.

En los teléfonos móviles, al escribir en una caja de texto de tipo `email` se adapta ligeramente el teclado de texto, mostrándose el carácter `@`.

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

`<input 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">`

Un formulario web que muestra el texto "Fecha:" seguido de un campo de entrada con el formato "dd/mm/aaaa". Debajo del campo hay un botón etiquetado "Enviar".

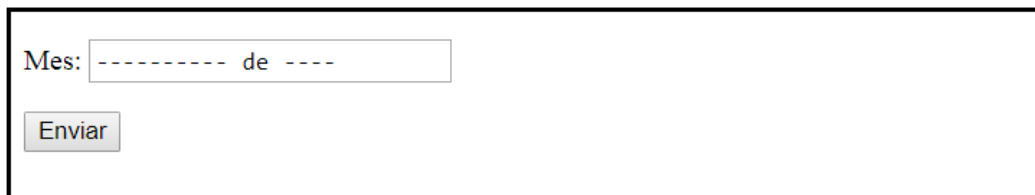
Mes: `<input type="month">`

`<input 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">`

Un formulario web que muestra el texto "Mes:" seguido de un campo de entrada con el formato "----- de ----". Debajo del campo hay un botón etiquetado "Enviar".

Nota: Firefox no muestra un control específico para seleccionar el mes y el año, como se comenta en la [lección de diferencias entre navegadores](#).

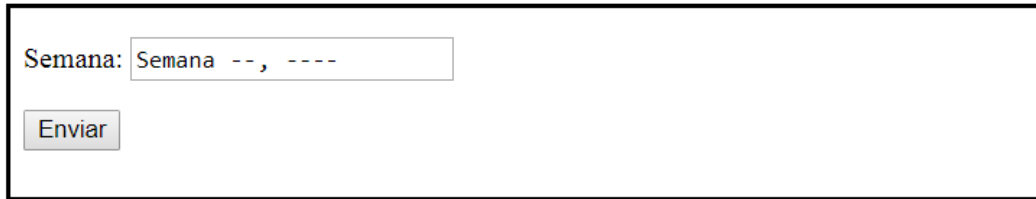
Semana: `<input type="week">`

`<input week>`

- - -

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">`



Nota: Firefox no muestra un control específico para seleccionar la semana y el año, como se comenta en la [lección de diferencias entre navegadores](#).


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

`<input 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">`



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

`<input 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="fechaHora">`

Fecha y hora local:

Nota: Firefox no muestra un control específico para seleccionar la fecha y hora, como se comenta en la [lección de diferencias entre navegadores](#).

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

`<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

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

☐ Casilla de verificación

Botón radio: `<input type="radio">`

`<input radio>`

- -

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

`<input type="radio" name="boton" 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="boton" value="1">Opción 1
`
`<input type="radio" name="boton" value="2">Opción 2
`
`<input type="radio" name="boton" value="3">Opción 3`

☐ Opción 1
☐ Opción 2
☐ Opción 3

Si los atributos **name** no tienen el mismo valor, los botones radio son independientes (al marcar uno no se desmarca el otro):

```
<input type="radio" name="boton1" value="1">Opción 1<br>
<input type="radio" name="boton2" value="2">Opción 2
```

Un formulario web que contiene dos botones de radio. El primero está etiquetado como 'Opción 1' y el segundo como 'Opción 2'. Ambos botones están desmarcados. Debajo de los botones hay un botón rectangular con el texto 'Borrar'.

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="boton"> Hombre<br>
<input type="radio" name="boton"> Mujer
```

Un formulario web que contiene dos botones de radio. El primero está etiquetado como 'Hombre' y el segundo como 'Mujer'. Ambos botones están desmarcados. Debajo de los botones hay un botón rectangular con el texto 'Enviar'.

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

Un formulario web que contiene dos botones de radio. El primero está etiquetado como 'Hombre' y el segundo como 'Mujer'. Ambos botones están desmarcados. Debajo de los botones hay un botón rectangular con el texto 'Enviar'.

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


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



Formulario con dos botones radio: "Opción 1" (desactivado) y "Opción 2" (seleccionado). Debajo hay un botón "Borrar".

Una vez se ha hecho clic en uno de los botones radio, siempre habrá una opción seleccionada. Para desactivar un botón radio, se necesita un botón Reset.

```
<input type="radio" name="boton" value="1">Opción 1<br>
<input type="radio" name="boton" value="2">Opción 2<br>
<input type="reset" value="Borrar">
```



Formulario con dos botones radio: "Opción 1" (desactivado) y "Opción 2" (desactivado). Debajo hay un botón "Borrar".

El inconveniente del botón Reset es que reinicia todos los controles del formulario. Este problema no tiene una solución sencilla. Algunos recomiendan incluir en los botones radio una opción neutra, de manera que el usuario pueda cambiar su respuesta sin tener que reiniciar todo el formulario.


El problema es que el usuario puede interpretar esa opción como una opción más y elegirla a propósito, por lo que se recomienda valorar en cada caso la mejor solución.

Quizás en formularios con pocos controles no merezca la pena incluir esa opción neutra, y hacerlo sólo en formularios con muchos controles en los que el usuario tendría que volver a completar todo el formulario si utiliza el botón Reset

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

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

`<input type="radio" name="boton" value="0" checked>`Sin respuesta
`</p>`



Nombre:

☐ Opción 1
☐ Opción 2
☒ Sin respuesta

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

`<input 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.

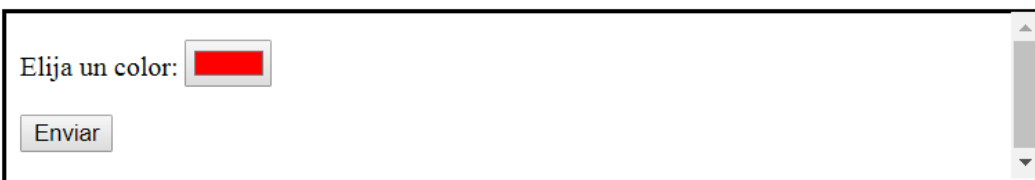
`<input type="color" name="colorElegido">`



Elija un color:

El atributo `value` permite especificar un color inicial. El color se debe indicar en formato hexadecimal (`#RRGGBB`).

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



Elija un color:

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

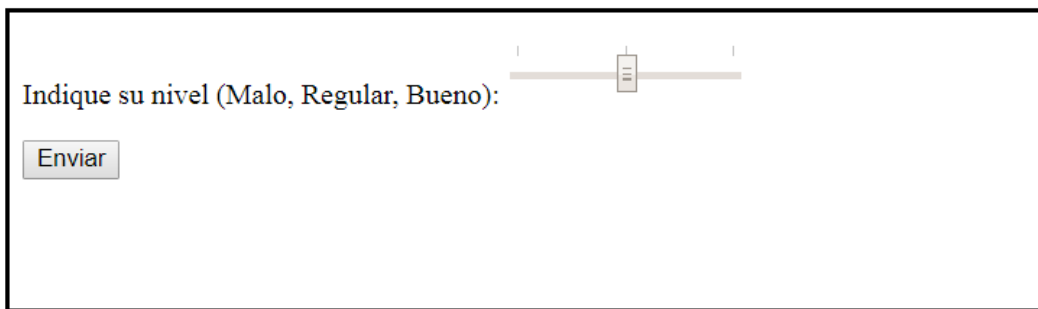
Indique su nivel (de 0 a 1, de 0,1 en 0,1):

Enviar

El atributo **list** debería permitir añadir etiquetas al control, pero actualmente (octubre de 2019) ningún navegador lo hace.

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 value="2" label="Regular">
  <option value="3" label="Bueno">
</datalist>
```

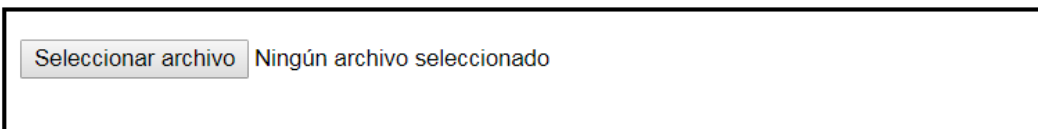


Selector de archivo: `<input type="file">`

```
<input file>
```

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

```
<input type="file" name="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/*">
```

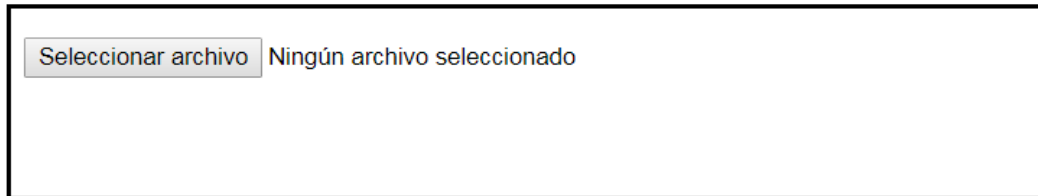


Imagen: **<input type="image">**

```
<input image>
```

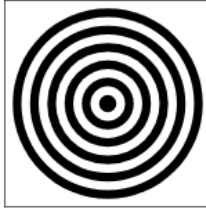
- -

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).

Este control debe incluir siempre el atributo **alt** (con un valor no vacío), para indicar el texto que deben mostrar los navegadores si la imagen no está disponible. Es importante recordar que esta obligatoriedad no existe en el caso de la etiqueta de imagen ****, en la que este atributo está aconsejado, pero no es obligatorio.

```
<input type="image" name="diana" src="diana.svg" alt="Diana">
```

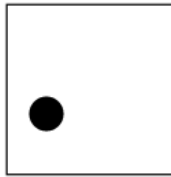
Haga clic en algún punto de la imagen:



El ejemplo siguiente es un pequeño juego en el que se utilizan las coordenadas recibidas para calcular si el usuario ha hecho clic en el punto negro del dibujo.

```
<input type="image" name="punto" src="punto.svg" alt="Punto">
```

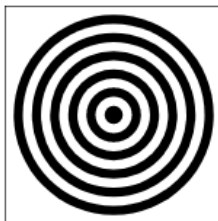
Haga clic en el círculo negro o fuera de él:



Si se define el atributo **value**, Microsoft Edge envía tanto las coordenadas como el nombre del control con el valor del atributo **value**, pero Firefox y Google Chrome no lo hacen, como se comenta en la [lección de diferencias entre navegadores](#).

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

Haga clic en algún punto de la imagen:



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

Escriba su nombre:

Los controles ocultos se pueden utilizar, por ejemplo, para dividir en varias páginas un formulario largo. El problema de dividir en varias páginas es que cada página es independiente de las demás y HTML no permite relacionar una con otra por lo que no se sabría que la información enviada en el segundo formulario completa la del primero (los lenguajes de programación de servidor, como PHP, sí que disponen de mecanismos para relacionar unas páginas con otras, por ejemplo mediante *cookies* o *sesiones*). Una solución es guardar en el segundo formulario la información enviada en el primer formulario en controles ocultos, de manera que cuando el usuario envía el segundo formulario, está enviando a la vez la información del primero y así sucesivamente en los siguientes formularios.

Escriba su nombre: `<input type="text" name="nombre">`

Escriba su edad: `<input type="number" name="edad">
`

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

Escriba su nombre:

Área de texto: <textarea>

<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>




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

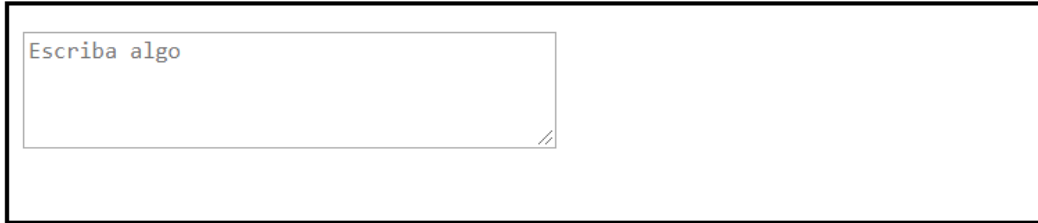


Si se escribe texto en el interior de la etiqueta, este se muestra en la caja de texto. HTML 5 introdujo el atributo **placeholder** que puede ser más cómodo para el usuario si tiene que borrar todo el texto incluido.

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



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

A screenshot of a web browser showing a text area with the placeholder text "Escriba algo". The text area is a simple rectangular box with a light gray border and a small cursor icon at the bottom right.

Menú: **<select>**

```
<select>
```

```
- -
```

```
<option>
```

```
- -
```

```
<optgroup>
```

```
- -
```

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 screenshot of a web browser showing a dropdown menu. The text "Elija una opción:" is followed by a dropdown box containing the word "Uno" and a small downward arrow. Below the dropdown box 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>Uno</option>
```


```
<option value="2">Dos</option>
<option value="three">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>
```



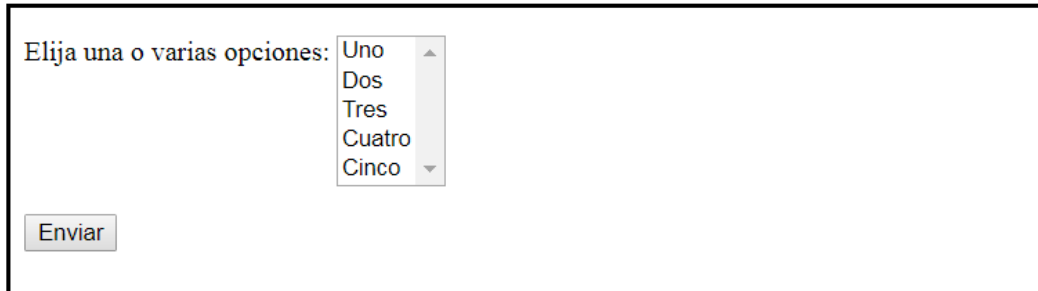
El atributo **size** permite definir la altura del control.

```
<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>
```



El atributo **multiple** permite elegir varias opciones simultáneamente (con ayuda de la tecla **Control** o **Mayúsculas**). En este caso el atributo **name** debe incluir corchetes (**[]**) para que se puedan enviar en su caso varios valores en forma de matriz.


```
<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>
```



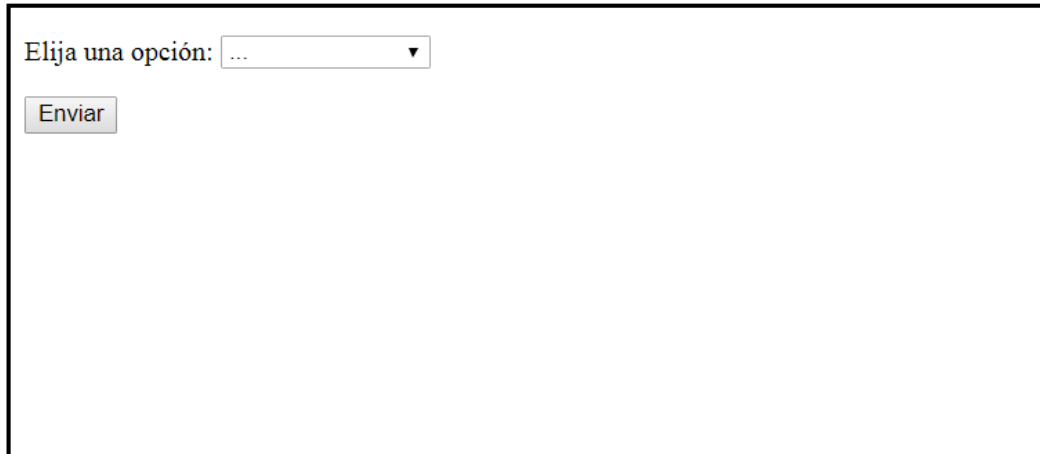
Elija una o varias opciones:

- 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>
```

A screenshot of a web form. At the top, there is a label 'Elija una opción:' followed by a dropdown menu with a downward arrow and three dots. Below this, there is a button labeled 'Enviar'.

Grupos de controles: **<fieldset>**

<fieldset>

- -

<legend>

- -

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="3"> 1917  
</p>  
</fieldset>
```

Datos personales

Nombre:

Edad:

Cultura general

Capital de Noruega: ☐ Copenhague ☐ Helsinki ☐ Oslo

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

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

Nota:

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

Accesibilidad: **<label>**

<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:

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

☐ Casilla 1

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

☐ Casilla 1

En el caso de la 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>Nombre: <input type="text" name="casilla"></label>`

Nombre:

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 la 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 |