

# Formularios en HTML

Tema 2\_6

# Índice

Formularios

Envíos GET y POST

La etiqueta <form>

La etiqueta <input>

La etiqueta <label>

Áreas de texto <textarea>

Listas de selección <select>

Botones <button>

Agrupación de elementos <fieldset>

La etiqueta <output>

La etiqueta <progress>

# Formularios

Los formularios en HTML tienen el propósito de recolectar información proporcionada por los usuarios de un sitio web

Los datos del formulario normalmente se envían al servidor que se encargará de recibir y procesar la información: se almacena en base de datos, se envía un correo electrónico...

También se pueden usar en el lado del cliente para provocar de alguna manera una actualización inmediata de la interfaz (se añade otro elemento a una lista, se muestra u oculta una función de interfaz de usuario...)

# Formularios

- Un formulario web está compuesto por uno o más controles, además de algunos elementos adicionales que ayudan a estructurar el formulario general
- Los controles pueden ser campos de texto, cajas desplegadas, botones, casillas de verificación, botones de opción...

The diagram illustrates a web form with various input types and their corresponding HTML input types. The form fields and their labels are as follows:

- Nombre**: Input type text
- Apellidos**: Input type text
- F. Nacimiento**: Input type text
- Calle y Número**: Input type text
- Código postal**: Input type text
- Provincia**: Input type text
- Teléfono**: Input type text
- Sexo**: Input type radio (Hombre, Mujer)
- Estado Civil**: select lista/menú (Soltero)
- Idiomas**: Input type checkbox (Español, Francés, Inglés, Alemán)
- Comentarios Personales**: Input type textarea
- Pulse aqui**: Input type submit (enviar los datos), Input type reset (borrar los datos)

Labels on the right side of the form indicate the input type for each field:

- Input type text
- Input type radio
- select lista/menú
- Input type checkbox
- Input type textarea
- Input type submit envío de datos
- Input type reset restaurar datos

# Envío de datos (GET y POST)

- En las páginas web, los clientes (navegadores) no solo obtienen elementos HTML del servidor, sino que también envían información:
  - El texto de búsqueda que el usuario ha escrito en el motor de búsqueda
  - El contenido de los formularios
  - El filtro de selección en tiendas online
- Para enviar información al servidor, el protocolo HTTP provee diferentes métodos de petición.
  - Los dos más importantes son GET y POST que, aunque entregan los mismos resultados, tienen algunas diferencias

# Envío de datos GET

- GET: los datos que se envían al servidor se escriben en la misma dirección URL.
  - [www.ejemplo.com/registrar.php?nombre=pedro&apellido=perez&edad=55&](http://www.ejemplo.com/registrar.php?nombre=pedro&apellido=perez&edad=55&)
- Ventajas de GET
  - Los parámetros URL se pueden guardar junto a la dirección URL como marcador. Se puede volver a acceder a la página a través del historial del navegador.
- Desventajas de GET
  - Nula protección de los datos: los parámetros URL que se envían son visibles en la barra de direcciones del navegador y son accesibles sin clave en el historial de navegación, en el caché y en el log de los servidores.
  - Capacidad es limitada: dependiendo del servidor y del navegador, no es posible introducir más de 2000 caracteres.
  - Los parámetros URL solo pueden contener caracteres ASCII (letras, números, signos, etc.) y no datos binarios como archivos de audio o imágenes.

# Envío de datos POST

- POST: introduce los parámetros en el cuerpo de la solicitud HTTP para el servidor y no quedan visibles para el usuario.
- Ventajas de POST
  - Los datos no se muestran en el caché ni tampoco en el historial de navegación.
  - Se puede enviar todo tipos de información, como fotos o vídeos.
  - La capacidad del método POST es ilimitada.
- Desventajas de POST
  - Cuando una página web que contiene un formulario se actualiza los datos del formulario deben transferirse de nuevo: riesgo de duplicación del envío
  - Los datos transferidos con POST no pueden guardarse junto al URL como marcador.

# Contenedor del formulario:<form>

La etiqueta **<form>** se utiliza para crear un formulario en una página web. Los formularios están compuestos de campos de entrada, como texto, números, fechas, archivos, entre otros.

## Atributos:

- **id**: especifica el identificador del formulario
- **action**: especifica el script o el destino que manejará los datos del formulario una vez que el usuario los ha enviado
- **method** especifica cómo se envían los datos al destino especificado en el atributo *action* (GET y POST)
- **autocomplete**: permite o desactiva la autocompleción automática de campos de entrada (on, off)
- **enctype**: especifica el tipo de codificación de los datos del formulario, importante para enviar ficheros
- **novalidate**: especifica que el formulario no debe ser validado al enviarlo
- **target**: define dónde se debe abrir el resultado de enviar el formulario.



# Controles de entrada de datos: <input>

La etiqueta **<input>** es fundamental en los formularios ya que permite a los desarrolladores recopilar información del usuario a través de diversos tipos de campos de entrada.

Se utiliza dentro de un formulario <form> para crear campos de entrada interactivos que permiten a los usuarios ingresar información.

Esta etiqueta es altamente versátil y puede adaptarse a una amplia variedad de tipos de datos y formatos, como texto, números, fechas, contraseñas, casillas de verificación y botones de opción, entre otros.

Su sintaxis básica es:

```
<input type="tipo_de_input" value="valor" name="nombre / id" ...>
```

# Controles de entrada de datos: <input>

Existen varios atributos que pueden aplicarse a la etiqueta **<input>** para modificar su comportamiento, apariencia y características.

## Atributos:

- **id**: identificador del campo
- **name**: identifica un campo de entrada en particular. Es útil en el servidor, ya que actúa como clave para los valores ingresados.
- **value**: especifica el texto predeterminado de un campo de entrada
- **placeholder**: muestra un texto de sugerencia en el campo de entrada cuando está vacío, y desaparece cuando el usuario comienza a escribir
- **required**: indica si el campo es obligatorio y debe completarse antes de rellenar el formulario
- **readonly**: hace que un campo de entrada sea de solo lectura, los usuarios no pueden modificar su valor, pero se envía con el formulario.
- **disabled**: hace que un campo de entrada sea de solo lectura, los usuarios no pueden modificar su valor, y no se envía con el formulario.
- **type**: especifica el tipo de control

# Tipos de elementos <input>

La etiqueta **<input>** admite varios tipos de campos de entrada que se definen mediante el atributo *type*.

## Texto

Para crear un campo de entrada de texto simple, se utiliza el tipo *text*. Los usuarios pueden ingresar cualquier cadena de texto en este campo.

```
<input type="text" name="nombre" placeholder="Ingrese su nombre">
```

## Contraseña

El tipo *password* crea un campo de entrada donde los caracteres ingresados se ocultan.

```
<input type="password" name="contrasena" placeholder="Ingrese su contraseña">
```

# Tipos de elementos <input>

## Checkbox

Las casillas de verificación permiten a los usuarios seleccionar múltiples opciones de una lista. El tipo es “checkbox”.

```
<input type="checkbox" name="intereses" value="deporte"> Deporte  
<input type="checkbox" name="intereses" value="arte" checked> Arte
```

## Radio

Los botones de opción, tipo *radio*, permiten seleccionar una única opción de un conjunto.

```
<input type="radio" name="genero" value="masculino"> Masculino  
<input type="radio" name="genero" value="femenino"> Femenino
```

Se puede usar el atributo ***checked*** para marcar previamente alguna opción

# Tipos de elementos <input>

## Number

El tipo *number* crea un campo de entrada para ingresar valores numéricos. Puedes especificar el rango de números permitidos utilizando los atributos min y max.

```
<input type="number" name="edad" min="1" max="120" placeholder="Edad">
```

## Email

El tipo *email* crea un campo de entrada específico para direcciones de correo electrónico. Este tipo de campo proporciona una validación básica de correo electrónico de forma automática.

```
<input type="email" name="correo" placeholder="Correo electrónico">
```

## Hidden

El tipo *hidden* crea un campo invisible para el usuario, pero que se manda al servidor

```
<input type="hidden" name="oculto" value="formulario_peticion">
```

# Tipos de elementos <input>

## File

El tipo *file* permite a los usuarios cargar archivos. Los usuarios pueden seleccionar un archivo de su dispositivo utilizando el cuadro de diálogo de carga de archivos.

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

## Submit

El tipo *submit* crea un botón que envía el formulario y sus datos al servidor.

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

## Reset

El tipo *reset* un botón que resetea el formulario.

```
<input type="reset" name="reseteo" value="Limpiar formulario">
```

# Tipos de elementos <input>

## Button

El tipo *button* crea un botón genérico que puede usarse para diversas acciones en la página, como abrir una ventana emergente, enviar datos mediante JavaScript, etc.

Se puede agregar funcionalidad utilizando JavaScript y escuchando el evento click en el botón.

```
<input type="button" value="Haga clic aquí" onclick="miFuncion()">
```

## Range

El tipo *range* crea un control deslizante que permite a los usuarios seleccionar un valor dentro de un rango específico. El rango se establece utilizando los atributos min, max y step.

```
<input type="range" name="volumen" min="0" max="100" step="1">
```

## Tel

El tipo *tel* crea un campo de entrada para números de teléfono. Este tipo de campo no proporciona validación, pero ofrece una mejor experiencia de usuario en dispositivos móviles al mostrar el teclado numérico adecuado.

```
<input type="tel" name="telefono" placeholder="Número de teléfono">
```

# Tipos de elementos <input>

## Color

El tipo *color* crea un botón de selección de un color a través de una paleta de colores.

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

## Image

El tipo *image* crea un control como el *submit*, pero usando una imagen.

```
<input type="image" name="image" src="imagenes/ldownload.png">
```

## Url

El tipo *url* crea un campo de entrada para URL, que proporciona validación.

```
<input type="url" name="url" placeholder="Pon un URL">
```



# Tipos de elementos <input>

Hay varios campos que permiten introducir **fechas** y **horas**

## Date

El tipo *date* crea un campo de entrada para seleccionar fechas utilizando un selector de fecha nativo del navegador.

```
<input type="date" name="fecha_nacimiento">
```

## Datetime-local

El tipo *datetime-local* crea un campo de entrada para seleccionar fecha y hora

```
<input type="datetime-local" name="fecha_hora_hoy">
```

## Month

El tipo *month* crea un campo de entrada para seleccionar un mes y de un año

```
<input type="month" name="mes_anio">
```

## Week

El tipo *week* crea un campo de entrada para seleccionar una semana de un año

```
<input type="week" name="semana_anio">
```

# Etiquetas para controles:<label>

La etiqueta **<label>** se utiliza para asociar una descripción con un elemento de formulario específico. Por ejemplo, si tiene una entrada de texto en su formulario, puede usar una etiqueta de etiqueta para describir el propósito de ese campo de entrada.

Siempre debes asociar un elemento **<label>** con cada campo de entrada utilizando el atributo ***for*** en la etiqueta **<label>** y el atributo ***id*** en el elemento **<input>**.

- Esto mejora la accesibilidad al proporcionar información contextual sobre el campo de entrada y permitir que los usuarios hagan clic en la etiqueta para enfocar el campo de entrada asociado.

Atributos:

- **for**: especifica el identificador del campo del formulario relacionado con la etiqueta label
- **form**: Define el id del formulario al cual pertenece el elemento de formulario al cual se está asociando la etiqueta

# Áreas de texto:<textarea>

La etiqueta **<textarea>** se utiliza para crear un área de texto multilínea en un formulario web.

## Atributos:

- **autofocus:** Especifica que el área de texto debe tener el foco automáticamente cuando la página carga.
- **cols:** Especifica el número de columnas que se deben mostrar en el área de texto.
- **disabled:** Especifica que el área de texto está deshabilitada y el usuario no puede interactuar con ella.
- **form:** Especifica el formulario al que pertenece el área de texto.
- **maxlength:** Especifica el número máximo de caracteres que se pueden ingresar en el área de texto.
- **name:** Especifica el nombre del área de texto.
- **placeholder:** Especifica un texto de ejemplo
- **readonly:** Especifica que el área de texto es de solo lectura y el usuario no puede ingresar texto.
- **required:** Especifica que el usuario debe ingresar un valor en el área de texto antes de enviar el formulario.
- **rows:** Especifica el número de filas que se deben mostrar en el área de texto.
- **wrap:** Especifica cómo se deben manejar los saltos de línea en el área de texto. (hard, soft).

# Áreas de texto :<textarea>

En este ejemplo se utilizan los atributos **rows**, **cols**, **maxlength** y **required** para especificar el tamaño y la validación del área de texto. El atributo **placeholder** es usado para proporcionar una indicación al usuario sobre qué debe ingresar en el área de texto.

```
<form>
  <label for="feedback">Deja tu comentario:</label>
  <textarea id="feedback" name="feedback" rows="5" cols="30" maxlength="250" required placeholder="Ingresa
tu comentario aquí">
</textarea> <br>
  <input type="submit" value="Enviar">
</form>
```

# Listas de selección:<select>

La etiqueta **<select>** se utiliza para crear un menú desplegable. Este elemento permite al usuario elegir una opción de un conjunto de opciones predefinidas.

La etiqueta **<select>** se utiliza junto con la etiqueta **<option>** para especificar las opciones disponibles en el menú desplegable..

## Atributos:

- **autofocus:** Especifica que el área de texto debe tener el foco automáticamente cuando la página carga.
- **disabled:** Especifica que el área de texto está deshabilitada y el usuario no puede interactuar con ella.
- **form:** Especifica el formulario al que pertenece el área de texto.
- **multiple:** permite seleccionar varias opciones al mismo tiempo
- **name:** Especifica el nombre del área de texto.
- **required:** Especifica que el usuario debe ingresar un valor en el área de texto antes de enviar el formulario.
- **size:** Especifica el número de opciones que se deben mostrar en el control de selección.

# Listas de selección:<select>

Ejemplo básico de una lista desplegable creada con el control **select**

```
<label for="color">Selecciona tu color favorito:</label>
<select id="color" name="color">
  <option value="rojo">Rojo</option>
  <option value="verde">Verde</option>
  <option value="azul">Azul</option>
</select>
```

# Elementos de listas de selección:<option>

La etiqueta **<option>** representa una opción en una lista desplegable, ya sea en una etiqueta **<select>** o **<optgroup>**. Cada opción tiene un valor asociado que se envía al servidor cuando se selecciona la opción en el formulario.

Atributos:

- **value**: Especifica el valor que se enviará al servidor.
- **label**: Especifica una etiqueta para la opción
- **selected**: especifica que la opción estará seleccionada de manera predeterminada cuando se cargue la página. Solo se puede seleccionar una opción como predeterminada
- **disabled**: deshabilita la opción, impidiendo que el usuario seleccione la opción

```
<select id="select-language" name="language">
  <option value="en" label="Inglés">English</option>
  <option value="es" label="Español" selected>Español</option>
  <option value="fr" label="Francés">Français</option>
</select>
```

# Grupos en listas de selección:<optgroup>

La etiqueta **<optgroup>** se usa para agrupar opciones relacionadas dentro de una lista desplegable (elemento **<select>**).

Hay un atributo obligatorio *label* que especifica el título o títulos del grupo. Dentro de la etiqueta **<optgroup>** se colocan las opciones relacionadas mediante la etiqueta **<option>**.

Atributos:

- **label**: Especifica el título del grupo
- **disabled**: deshabilita todas las opciones del grupo, impidiendo que el usuario seleccione ninguna opción

```
<select>
  <optgroup label="Cítricos">
    <option value="limon">Limón</option>
    <option value="naranja">Naranja</option>
  </optgroup>
  <optgroup label="No cítricos" disabled>
    <option value="manzana">Manzana</option>
    <option value="pera">Pera</option>
  </optgroup>
</select>
```



# Listas de selección:<datalist>

La etiqueta **<datalist>** se utiliza para crear una lista de opciones predefinidas que se pueden utilizar en un campo de entrada de texto. La lista de opciones se muestra como un menú desplegable cuando se hace clic en el campo de entrada o se comienza a escribir en él.

Al ser un campo de texto se pueden enviar opciones que no estén en la lista

La etiqueta <datalist> se compone de varias etiquetas <option>, cada una de las cuales especifica una opción disponible en la lista. El valor de cada etiqueta <option> es el valor que se insertará en el campo de entrada si se selecciona esa opción.

Atributos:

- **id**: Este atributo es necesario para asociar una lista de opciones con un campo de entrada mediante el atributo list en el campo de entrada.

```
<input type="text" id="colors" name="colors" list="colors-list">
<datalist id="colors-list" " title="List of available colors">
  <option value="Red">
  <option value="Green">
  <option value="Blue">
  <option value="Yellow">
  <option value="Orange">
</datalist>
```

# Botones:<button>

La etiqueta **<button>** se utiliza para crear un botón interactivo en una página web.

La etiqueta `<button>` permite mucha más flexibilidad a la hora de crear nuestros botones que la etiqueta `<input type="button">`, ya que permite utilizar HTML dentro del botón.

Atributos:

- **type**: tipo de botón. Puede ser *submit*, *reset* o *button*. El valor predeterminado es *submit*.
- **autofocus**: Especifica que el botón debe tener el foco automáticamente cuando la página carga.
- **disabled**: Especifica que el botón está deshabilitado y el usuario no puede interactuar con el.
- **form**: Especifica el formulario al que pertenece el botón.
- **name**: Especifica el nombre del botón

```
<button type="reset" >Botón de <span style="color:red">limpiar el formulario</span></button>  
<button >Enviar</button>  
<button type="button" >Botón normalito</button>
```

# Agrupación de elementos:<fieldset> <legend>

La etiqueta **<fieldset>** se utiliza para agrupar elementos relacionados en un formulario web.

El elemento <fieldset> se utiliza junto con la etiqueta **<legend>** para proporcionar un título para el grupo de elementos

Atributos:

- **disabled**: Especifica que el conjunto de datos está deshabilitado y el usuario no puede interactuar con él.
- **form**: Especifica el formulario al que pertenece el conjunto de campos.
- **name**: Especifica el nombre del conjunto de controles

```
<fieldset name="personal_info" disabled>
  <legend>Información personal:</legend>
  <label for="name">Nombre:</label>
  <input type="text" id="name" name="name" required><br><br>
  <label for="email">Email:</label>
  <input type="email" id="email" name="email"><br><br>
</fieldset>
```

# Salida de datos:<output>

La etiqueta **<output>** se utiliza para representar el resultado de un cálculo o una operación realizada por un script en una página web.

Es un elemento de formulario que permite mostrar información generada dinámicamente, como resultados de una operación matemática o una consulta a una base de datos.

Atributos:

- **for**: Especifica los controles asociados con la salida. Es una lista de ID de controles separados por espacios.
- **form**: Especifica el formulario al que pertenece el campo.
- **name**: Especifica el nombre del control
- **value**: Especifica el valor inicial del elemento de salida.

```
<form>
  <output id="result"></output>
  <script>
    var result = document.getElementById("result");
    result.innerHTML = new Date().toString();
  </script>
</form>
```

# Avance de procesos: <progress>

La etiqueta **<progress>** se utiliza para mostrar el progreso de una tarea, ya sea en forma de barra de progreso o de un porcentaje. La barra de progreso se puede mostrar de forma horizontal o vertical dependiendo de cómo se estructure el código.

Atributos:

- **max**: Especifica el valor máximo del progreso, y debe ser mayor que *value*.
- **form**: Especifica el formulario al que pertenece el campo.
- **name**: Especifica el nombre del control
- **value**: Especifica el valor actual de la barra de progreso.

```
<progress id="fileUpload" value="35" max="100" form="form1"></progress>  
<br>  
<label for="fileUpload">Upload Progress: 35%</label>
```