

Programación Web Desde Cero

Formularios en HTML, CSS, JS

¡Hola! 🖐️

Hoy veremos una herramienta muy utilizada en las páginas webs. Los formularios.

Permite a los usuarios ingresar y enviar datos al servidor web para su procesamiento. Los formularios se crean utilizando elementos HTML específicos y se componen de varios componentes clave. Aquí te explico los conceptos básicos para crear formularios en HTML.

¡Comencemos! 🚀

Formularios

Un **formulario** es un elemento que se utiliza para recopilar información ingresada por el usuario como nombres, correos electrónicos, contraseñas, comentarios, entre otros. Son esenciales en las páginas web para permitir interacciones entre el usuario y el servidor, como iniciar sesión, registrarse, enviar comentarios y realizar pedidos en línea.



Formularios

A visual representation of an HTML form. It consists of a light gray rounded rectangle containing the following elements: a 'Nombre' label above a text input field, an 'Apellido' label above a text input field, an 'Email' label above a text input field, a 'Mensaje' label above a larger text area, and a blue button with the text 'Enviar' at the bottom left.

Un formulario típico de HTML se crea utilizando la etiqueta **<form>** y contiene varios elementos de entrada como **<input>**, **<textarea>**, **<select>** y **<button>** que facilitan la recopilación de datos de los usuarios.

Veamos un ejemplo de una estructura HTML básica para un formulario:

```

<form>
  <!-- Campo para nombre -->
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre">

  <!-- Campo para correo electrónico -->
  <label for="email">Correo electrónico:</label>
  <input type="email" id="email" name="email">

  <!-- Campo para mensaje -->
  <label for="mensaje">Mensaje:</label>
  <textarea id="mensaje" name="mensaje"></textarea>

  <!-- Campo para selección de país -->
  <label for="pais">País:</label>
  <select id="pais" name="pais">
    <option value="argentina">Argentina</option>
    <option value="chile">Chile</option>
    <option value="mexico">México</option>
    <option value="peru">Perú</option>
  </select>

  <!-- Botón de envío -->
  <button type="submit">Enviar</button>
</form>

```

- **<form>**: Envuelve todos los campos del formulario. Se utiliza para agrupar los elementos relacionados en un solo bloque.
- **<label>**: Se utiliza para etiquetar los campos del formulario. El atributo "for" se utiliza para vincular la etiqueta con el campo.
- **<input>**: Se utiliza para crear campos de entrada de texto. En este caso, se utilizó el tipo "text" para el campo de nombre y el tipo "email" para el campo de correo electrónico. El atributo "id" se utiliza para identificar el campo y el atributo "name" se utiliza para definir el nombre del campo en el servidor.
- **<textarea>**: Se utiliza para crear áreas de texto más grandes. En este caso, se utiliza para el campo de mensaje.
- **<select>**: Se utiliza para crear listas desplegables. En este caso, se utiliza para el campo de selección de país. Cada opción dentro del elemento <select> se crea con el elemento **<option>**, que contiene un valor y el texto a mostrar en la lista.

- **<button>**: Se utiliza para crear botones en el formulario. En este caso, se utiliza para el botón de envío del formulario. El atributo "type" se establece en "submit" para enviar el formulario.

¡Genial! Ahora, pasemos a ver cómo manejar y validar los formularios con JavaScript...

Manejo y validación de formularios

El **manejo y validación de formularios** es un aspecto clave para garantizar que los **datos ingresados por los usuarios sean correctos** y estén en el formato esperado antes de ser procesados o almacenados en un servidor.

El formulario muestra cuatro campos de entrada:

- Usuario:** Contiene el texto "john123". Tiene un borde rojo y un icono de "X" rojo, indicando un error. Debajo del campo, se muestra el mensaje de error: "El usuario tiene que ser de 4 a 16 dígitos y solo puede contener números, letras y guion bajo."
- Nombre:** Contiene el texto "Carlos Arturo". Tiene un borde gris y un icono de "check" verde, indicando que es válido.
- Contraseña:** Contiene 10 puntos para ocultar el texto. Tiene un borde azul y un icono de "check" verde, indicando que es válido.
- Repetir Contraseña:** Contiene 10 puntos para ocultar el texto. Tiene un borde rojo y un icono de "X" rojo, indicando un error. Debajo del campo, se muestra el mensaje de error: "Ambas contraseñas deben ser iguales."

💡 *Esta práctica mejora la calidad de los datos y la experiencia del usuario, al tiempo que minimiza posibles problemas o errores en el procesamiento de la información.*

El **manejo de formularios** en JavaScript implica la interacción con elementos de formulario HTML como campos de entrada, botones, casillas de verificación y listas desplegables.

Algunas tareas comunes en el manejo de formularios incluyen:

- Acceder a los valores de los campos de entrada.

- Cambiar dinámicamente el contenido o atributos de un elemento de formulario.
- Enviar un formulario o realizar una acción cuando se hace clic en un botón.
- Utilizar eventos de formulario, como submit, change, focus y blur, para detectar y responder a las interacciones del usuario.
- Validación de formularios

La **validación de formularios** es el proceso de verificar que los datos ingresados por el usuario cumplen con ciertos criterios o reglas. Estos pueden incluir:

- Comprobar que un campo es obligatorio y no está vacío.
- Validar que el valor ingresado sea del tipo de datos esperado como un número, una dirección de correo electrónico o una fecha.
- Verificar que el valor esté dentro de un rango específico como una cantidad mínima y máxima, o una longitud mínima y máxima.
- Comprobar que el valor cumpla con un patrón específico como una expresión regular para una contraseña segura o un formato de número de teléfono.

Resumen

¡Felicidades!

Esperamos que esta información te haya sido de utilidad y que te lleves para repasar:

- Cómo armar formularios en Javascript.
- Cuáles son los elementos necesarios para armar un formulario.
- La utilidad de la validación en Javascript.