



FORMULARIOS EN HTML: INTERACCIÓN CON EL USUARIO

CONCEPTO BÁSICO

- Definición:
 - Un formulario en HTML es una herramienta que permite a los usuarios ingresar datos que se envían a un servidor para su procesamiento.
 - Se define con la etiqueta <form>.
- Uso Principal:
 - Registro de usuarios.
 - Búsquedas.
 - Envío de comentarios.
 - Recopilación de datos.

ESTRUCTURA BÁSICA DE UN FORMULARIO

```
<form action="procesar.php"
method="post">
  <label
for="nombre">Nombre:</label>
  <input type="text" id="nombre"
name="nombre">
  <button
type="submit">Enviar</button>
</form>
```

- <form>: Define el formulario.
 - action: URL del archivo que procesará los datos.
 - method: Método de envío (GET o POST).
- <label>: Asocia un texto descriptivo a un campo de entrada.
- <input>: Crea un campo de entrada.
 - type: Especifica el tipo de dato (texto, contraseña, correo, etc.).
- <button>: Envía el formulario.

CAMPOS DE ENTRADA

1. Texto:

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

- Campo para ingresar texto.

2. Contraseña:

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

- Oculta el texto ingresado.

3. Correo Electrónico:

```
<input type="email" name="correo">
```

- Valida que el formato sea de un correo electrónico.

4. Número:

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

- Permite solo números.

CAMPOS DE SELECCIÓN

1. Casilla de Verificación:

```
<input type="checkbox"
name="aceptar" value="sí"> Acepto
los términos
```

- Permite selecciones múltiples.

2. Botón de Opción:

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

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

- Solo se puede seleccionar una opción.

3. Lista Desplegable:

```
<select name="pais">
```

```
  <option
value="es">España</option>
```

```
  <option
value="mx">México</option>
```

```
</select>
```

- Presenta una lista de opciones.

BOTONES

- Enviar:

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

- Restablecer:

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

ATRIBUTOS DE VALIDACIÓN

- **required:**

- Hace que un campo sea obligatorio.

```
<input type="text" name="nombre" required>
```

- **pattern:**

- Valida un campo con una expresión regular.

```
<input type="text" name="telefono" pattern="[0-9]{10}">
```

- **min y max:**

- Especifica rangos para números.

```
<input type="number" name="edad" min="18" max="99">
```


EJEMPLO PRÁCTICO

```
<form action="procesar.php" method="post">
  <h3>Registro</h3>
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre" required>
  <label for="correo">Correo:</label>
  <input type="email" id="correo" name="correo" required>
  <label for="clave">Contraseña:</label>
  <input type="password" id="clave" name="clave" required>
  <label>Género:</label>
  <input type="radio" id="masculino" name="genero" value="masculino">
  <label for="masculino">Masculino</label>
  <input type="radio" id="femenino" name="genero" value="femenino">
  <label for="femenino">Femenino</label>
  <label for="pais">País:</label>
  <select id="pais" name="pais">
    <option value="es">España</option>
    <option value="mx">México</option>
  </select>
  <input type="checkbox" id="terminos" name="terminos" value="sí" required>
  <label for="terminos">Acepto los términos y condiciones</label>
  <button type="submit">Registrar</button>
</form>
```


RESUMEN FINAL

- Formularios en HTML:
 - Permiten recopilar datos de los usuarios.
 - Se estructuran con `<form>` y campos como `<input>`, `<textarea>`, y `<select>`.
- Validación:
 - Mejora la calidad de los datos enviados al servidor.
- Interactividad:
 - Integración con CSS y JavaScript para personalizar el diseño y comportamiento.

IMPORTANCIA DE LA ACCESIBILIDAD

- **¿Por qué es importante?**
 - Permite que personas con discapacidades utilicen los formularios.
 - Mejora la experiencia del usuario para todos.
 - Es una práctica recomendada por los estándares de la web (WCAG - Web Content Accessibility Guidelines).

REGLAS BÁSICAS DE ACCESIBILIDAD

1. Usar etiquetas <label> correctamente:

- Asocia cada campo con una etiqueta descriptiva usando el atributo for.

```
<label for="nombre">Nombre:</label>
```

```
<input type="text" id="nombre" name="nombre" required>
```

2. Atributos de accesibilidad ARIA:

- Proporciona información adicional a los lectores de pantalla.

```
<input type="text" id="nombre" name="nombre" aria-label="Ingresa su nombre completo">
```

1. Describir los errores:

- Usa mensajes claros y asociarlos con los campos mediante aria-describedby.

```
<input type="text" id="nombre" name="nombre" aria-describedby="error-nombre">
```

```
<span id="error-nombre">El nombre es obligatorio.</span>
```

2. Evitar el uso de placeholder como único texto descriptivo:

- Los placeholder desaparecen al escribir y pueden confundir al usuario. Mejor combina con <label>.

```
<label for="correo">Correo:</label>
```

```
<input type="email" id="correo" name="correo" placeholder="ejemplo@correo.com" required>
```

SEGURIDAD EN FORMULARIOS. PRINCIPALES AMENAZAS

- **Ataques XSS (Cross-Site Scripting):**
 - El atacante introduce código malicioso en campos de entrada.
 - Puede usarse para robar información, realizar phishing o manipular el sitio.
- **Inyección SQL:**
 - El atacante manipula campos de entrada para ejecutar comandos en la base de datos.
 - Puede comprometer datos sensibles.
- **CSRF (Cross-Site Request Forgery):**
 - El atacante engaña al usuario para que realice acciones no deseadas en su nombre.
- **Spam y Bots:**
 - Bots automáticos envían formularios masivamente, saturando el servidor o recolectando información.

REGLAS DE VALIDACIÓN SEGURA

1. Validación del lado del cliente y del servidor:

- **Lado del Cliente (JavaScript):**

- Proporciona retroalimentación inmediata al usuario.
- Ejemplo:

```
<input type="text" name="nombre" pattern="[A-Za-z]+" required>
```

- **Lado del Servidor:**

- Es indispensable para garantizar la seguridad, ya que los atacantes pueden deshabilitar la validación del cliente.

REGLAS DE VALIDACIÓN SEGURA

2. Sanitización de Datos:

- Elimina o escapa caracteres peligrosos del input antes de procesarlo.
- Ejemplo en PHP:

```
$nombre = htmlspecialchars($_POST['nombre'], ENT_QUOTES, 'UTF-8');
```

REGLAS DE VALIDACIÓN SEGURA

2. Filtrar y validar todos los campos:

- Usa funciones específicas para validar datos como correos, números y URLs.
- Ejemplo en PHP:

```
if (filter_var($correo, FILTER_VALIDATE_EMAIL)) {  
    // Correo válido  
} else {  
    // Error  
}
```


REGLAS DE VALIDACIÓN SEGURA

2. Límite de longitud de datos:

- Define la longitud máxima para cada campo para evitar desbordamientos de buffer o inyecciones.

```
<input type="text" name="usuario" maxlength="50" required>
```

PROTECCIÓN CONTRA BOTS

1. CAPTCHA o reCAPTCHA:

- Añade un campo para verificar que el usuario es humano.

```
<div class="g-recaptcha" data-sitekey="tu-clave-site"></div>
```

2. Campos ocultos (Honeypot):

- Añade un campo visible solo para bots, que si se rellena invalida el formulario.

```
<input type="text" name="robot" style="display:none;">
```

3. Limitar Intentos de Envío:

- Implementa un límite de solicitudes por usuario o IP.

BUENAS PRÁCTICAS GENERALES

1. Usa HTTPS para encriptar la transmisión de datos.
2. No almacenes contraseñas en texto plano; usa algoritmos de hashing seguros (como bcrypt).
3. Implementa tokens CSRF para evitar solicitudes no autorizadas.
4. Registra intentos fallidos y revisa patrones sospechosos.

FORMULARIO SEGURO Y ACCESIBLE

```
<form action="procesar.php" method="post">
  <h3>Registro</h3>

  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre"
name="nombre" required maxlength="50" aria-
label="Ingrese su nombre completo">

  <label for="correo">Correo:</label>
  <input type="email" id="correo"
name="correo" required aria-label="Ingrese su
correo electrónico">
```

```
<label for="clave">Contraseña:</label>
  <input type="password" id="clave" name="clave"
required minlength="8" aria-describedby="info-clave">
  <small id="info-clave">La contraseña debe tener al
menos 8 caracteres.</small>

  <div>
    <label for="captcha">Escriba el resultado: 3 + 2 =
?</label>
    <input type="text" id="captcha" name="captcha"
required pattern="5" aria-label="Ingrese el resultado
de la suma">
  </div>

  <button type="submit">Registrar</button>
</form>
```