

Formularios HTML

Índice

1. [¿Qué es un formulario?](#)
2. [Elementos del formulario](#)

1

¿Qué es un
Formulario?

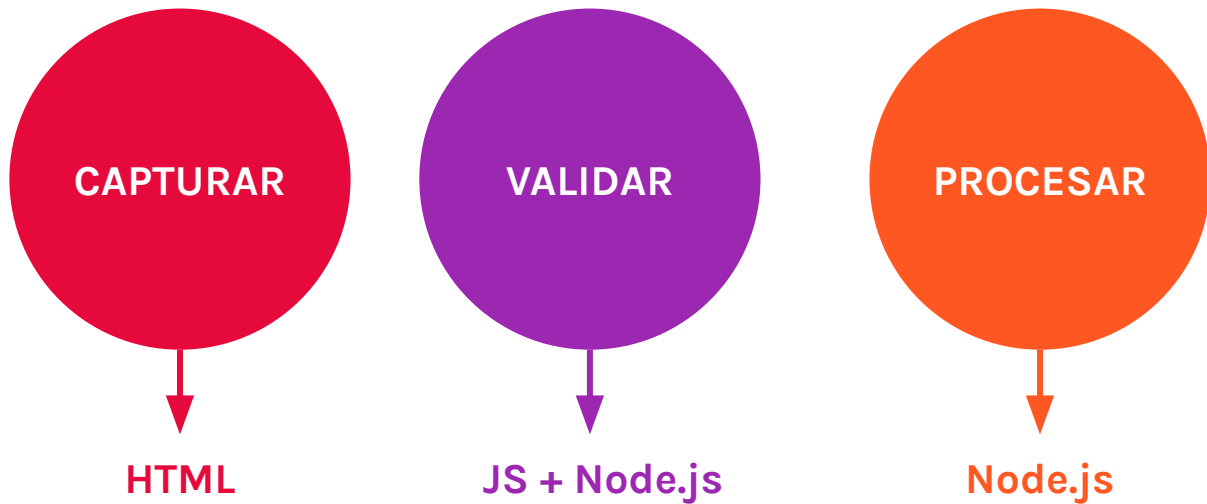


Los formularios son elementos de HTML que nos permiten pedirle datos al usuario que luego podremos validar y procesar.



Cómo Funcionan

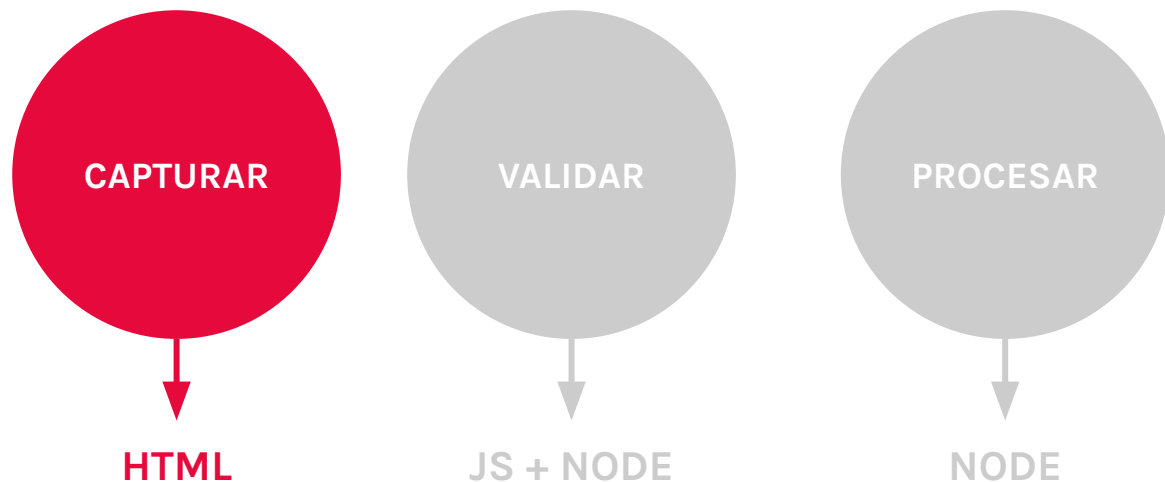
Para lograr que un formulario funcione correctamente, hacen falta las siguientes **3 instancias**:



Cómo Funcionan

Primero nos enfocaremos en la parte concerniente al **HTML**, es decir, que el formulario se vea correctamente y que capture la información.

Más adelante veremos cómo validar y procesar la información.



2 | Elementos del formulario

La etiqueta **<form>**

Con esta etiqueta indicamos que vamos a crear un formulario en nuestro documento HTML.

Todos los elementos que queramos incluir en el formulario van a ir dentro de las etiquetas: `<form>` `</form>`.

```
html1  <form action="/registro" method="POST">
        <!-- Aquí van los campos -->
        </form>
```


Los atributos de **<form>**

Esta etiqueta posee dos atributos: `action` y `method`.

Cada uno cumple un objetivo particular:

- `action` define la ruta en donde se va a procesar la información capturada.
- `method` define cómo se enviará la información. En la versión actual de HTML, hay solo dos valores posibles: `GET` y `POST`.

```
html1  <form action="/registro" method="POST">
        <!-- Aquí van los campos -->
        </form>
```

El elemento `<input>`

Es un elemento multifunción. Cambiando el valor del atributo `type`, podemos obtener distintos tipos de campos.

El atributo `name` identifica el campo y lo diferencia de los demás campos.

Es **importante** saber que **los elementos que no tengan `name`, no se envían**.

html

```
<form action="/registro" method="POST">
  <input type="text" name="usuario">
  <input type="password" name="clave">
  <input type="email" name="email">
  <input type="tel" name="telefono">
  <input type="number" name="edad">
</form>
```

El elemento `<label>`

Sirve para asociar un **texto descriptivo** a un campo determinado.

for **contenido**

Asocia el label con un campo por su propiedad `id`.

Contiene el texto que vamos a mostrar. Si el usuario hace click en el texto, el cursor irá al campo asociado.

html

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

id del campo

Para que el label quede asociado, el campo a asociar debe poseer un `id` con el mismo valor que se usó en el `for` del label.

El elemento **<label>**

Por lo general, cada campo o grupo de campos tendrá su `label` asociado.

Cuando el usuario haga click en el label, se activará automáticamente el campo que corresponda.

html

```
<form action="/registro" method="POST">
  <label for="nombre">Nombre:</label>
  <input type="text" name="nombre" id="nombre">

  <label for="email">Email:</label>
  <input type="email" name="email" id="email">

  <label for="telefono">Edad:</label>
  <input type="tel" name="edad" id="telefono">
</form>
```

El elemento `<textarea>`

Los anteriores campos, especialmente los de texto, están pensados para poca información, una sola línea de texto generalmente.

El `textarea` nos permite texto de gran tamaño y con múltiples líneas.

A diferencia del input, el `textarea` tiene etiqueta de apertura y cierre.

html

```
<form action="/contact" method="POST">
  <label for="comentario">Deje su Comentario:</label>
  <textarea name="comentario" id="comentario">Contenido</textarea>
</form>
```

El elemento `<select>`

Este elemento nos permite agregar un componente que muestra **opciones**. Como el resto de los campos, debe llevar el parámetro `name` para enviarse.

html

```
<form action="/colors" method="GET">
  <label>Seleccione un color:</label>
  <select name="color">
    <option value="#ff0000">Rojo</option>
    <option value="#00ff00">Verde</option>
    <option value="#0000ff">Azul</option>
    <option value="#770077">Morado</option>
  </select>
</form>
```

El elemento `<option>`

Las opciones de un `select` son representadas por elementos `option`.
Los elementos `option` tienen etiqueta de apertura y cierre.

value **selected**

Es el valor que será enviado por el formulario. **Es opcional**, se encarga de preseleccionar la opción.

html

```
<option value="#ff0000" selected>Rojo</option>  
<option value="#00ff00">Verde</option>
```

Contenido de la opción

Es lo que verá el usuario al desplegar el select.

El elemento **<button>**

Nos permite generar un **botón**. Con la propiedad **type** definimos el tipo.

- Un botón de tipo **reset** reinicia el formulario a su estado inicial.
- Un botón de tipo **submit** se encarga de enviar el formulario.
- Un botón de tipo **button** no realizará ninguna acción por defecto. Por lo general, programaremos una con ayuda de JavaScript.

html

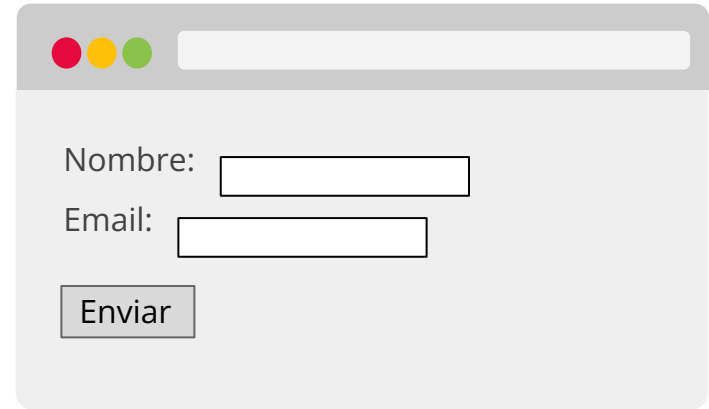
```
<form action="" method="">
  <label for="usuario">Ingrese el usuario:</label>
  <input type="text" name="usuario">

  <button type="reset">Cancelar</button>
  <button type="submit">Enviar</button>
</form>
```


Ejemplo

html

```
<form action="/login" method="POST">
  <p>
    <label>Nombre:</label>
    <input type="text" name="usuario">
  </p>
  <p>
    <label>Email:</label>
    <input type="email" name="email">
  </p>
  <p>
    <button type="submit">Enviar</button>
  </p>
</form>
```



A visual representation of the HTML form rendered in a browser window. The window has a standard macOS-style title bar with red, yellow, and green window control buttons. The form content is displayed on a light gray background. It features two labels, 'Nombre:' and 'Email:', each followed by a corresponding text input field. Below these fields is a button labeled 'Enviar'.

DigitalHouse>
Coding School