

# Guía de Desarrollo Web: HTML, CSS y JavaScript

Fecha: 11 de octubre de 2025

## Parte 1: HTML — Las 50 etiquetas más usadas

El HTML (HyperText Markup Language) es el esqueleto de cualquier página web. A continuación, se presentan las 50 etiquetas más comunes y esenciales para estructurar el contenido.

Etiqueta	Función	Ejemplo de Uso
<html>	Raíz del documento HTML.	<html>...</html>
<head>	Contiene metadatos y enlaces a scripts o estilos.	<head><title>Mi Página</title></head>
<title>	Define el título que aparece en la pestaña del navegador.	<title>Título de la Página</title>
<body>	Contenedor principal para todo el contenido visible.	<body><h1>Hola, mundo</h1></body>
<h1> a <h6>	Encabezados de diferentes niveles de importancia.	<h1>Encabezado Principal</h1>
<p>	Define un párrafo de texto.	<p>Este es un párrafo.</p>
<a>	Crea un hipervínculo (enlace).	<a href="https://google.com">Ir a Google</a>
<img>	Inserta una imagen en la página.	
<div>	Contenedor genérico para agrupar otros elementos.	<div>Contenido agrupado</div>

<code>&lt;span&gt;</code>	Contenedor en línea para estilizar una parte del texto.	<code>p&gt;Texto con &lt;span style="color:blue;"&gt;estilo&lt;/span&gt;.&lt;/p&gt;</code>
<code>&lt;ul&gt;</code>	Define una lista desordenada (con viñetas).	<code>&lt;ul&gt;&lt;li&gt;Elemento 1&lt;/li&gt;&lt;/ul&gt;</code>
<code>&lt;ol&gt;</code>	Define una lista ordenada (numerada).	<code>&lt;ol&gt;&lt;li&gt;Primer paso&lt;/li&gt;&lt;/ol&gt;</code>
<code>&lt;li&gt;</code>	Representa un elemento dentro de una lista ( <code>&lt;ul&gt;</code> u <code>&lt;ol&gt;</code> ).	<code>&lt;li&gt;Elemento de lista&lt;/li&gt;</code>
<code>&lt;header&gt;</code>	Define la cabecera de una página o sección.	<code>&lt;header&gt;&lt;h1&gt;Logo y Navegación&lt;/h1&gt;&lt;/header&gt;</code>
<code>&lt;footer&gt;</code>	Define el pie de página de una página o sección.	<code>&lt;footer&gt;&lt;p&gt;Copyright 2025&lt;/p&gt;&lt;/footer&gt;</code>
<code>&lt;nav&gt;</code>	Contiene los enlaces de navegación principales.	<code>&lt;nav&gt;&lt;a href="/inicio"&gt;Inicio&lt;/a&gt;&lt;/nav&gt;</code>
<code>&lt;main&gt;</code>	Representa el contenido principal y único del documento.	<code>&lt;main&gt;&lt;article&gt;...&lt;/article&gt;&lt;/main&gt;</code>
<code>&lt;section&gt;</code>	Agrupar contenido temáticamente relacionado.	<code>&lt;section&gt;&lt;h2&gt;Sobre Nosotros&lt;/h2&gt;&lt;/section&gt;</code>
<code>&lt;article&gt;</code>	Contenido independiente y autocontenido (ej. un post).	<code>&lt;article&gt;&lt;h3&gt;Título del Blog&lt;/h3&gt;&lt;/article&gt;</code>
<code>&lt;aside&gt;</code>	Contenido secundario o lateral (ej. una barra lateral).	<code>&lt;aside&gt;&lt;p&gt;Publicidad&lt;/p&gt;&lt;/aside&gt;</code>
<code>&lt;form&gt;</code>	Crea un formulario para la entrada de datos del usuario.	<code>&lt;form action="/enviar"&gt;&lt;input type="submit"&gt;&lt;/form&gt;</code>

<code>&lt;input&gt;</code>	Campo de entrada de datos (texto, email, contraseña, etc.).	<code>&lt;input type="text" name="usuario"&gt;</code>
<code>&lt;textarea&gt;</code>	Campo de entrada de texto de múltiples líneas.	<code>&lt;textarea name="comentario"&gt;&lt;/textarea&gt;</code>
<code>&lt;button&gt;</code>	Define un botón interactivo.	<code>&lt;button&gt;Haz Clic&lt;/button&gt;</code>
<code>&lt;label&gt;</code>	Asocia un texto descriptivo a un campo <code>&lt;input&gt;</code> .	<code>&lt;label for="email"&gt;Email:&lt;/label&gt;</code>
<code>&lt;table&gt;</code>	Crea una tabla para mostrar datos.	<code>&lt;table&gt;...&lt;/table&gt;</code>
<code>&lt;tr&gt;</code>	Define una fila dentro de una tabla.	<code>&lt;tr&gt;&lt;td&gt;Dato 1&lt;/td&gt;&lt;/tr&gt;</code>
<code>&lt;td&gt;</code>	Define una celda de datos en una tabla.	<code>&lt;td&gt;Contenido de la celda&lt;/td&gt;</code>
<code>&lt;th&gt;</code>	Define una celda de encabezado en una tabla.	<code>&lt;th&gt;Nombre&lt;/th&gt;</code>
<code>&lt;thead&gt;</code>	Agrupar el contenido del encabezado en una tabla.	<code>&lt;thead&gt;&lt;tr&gt;&lt;th&gt;...&lt;/th&gt;&lt;/tr&gt;&lt;/thead&gt;</code>
<code>&lt;tbody&gt;</code>	Agrupar el contenido principal del cuerpo en una tabla.	<code>&lt;tbody&gt;&lt;tr&gt;&lt;td&gt;...&lt;/td&gt;&lt;/tr&gt;&lt;/tbody&gt;</code>
<code>&lt;br&gt;</code>	Inserta un salto de línea.	Línea 1 Línea 2
<code>&lt;hr&gt;</code>	Crea una línea horizontal para separar contenido.	<code>&lt;p&gt;Texto.&lt;/p&gt;&lt;hr&gt;&lt;p&gt;Más texto.&lt;/p&gt;</code>
<code>&lt;strong&gt;</code>	Indica que el texto tiene una gran importancia (negrita).	<code>&lt;strong&gt;Importante:&lt;/strong&gt; no olvidar.</code>
<code>&lt;em&gt;</code>	Enfatiza un texto (cursiva).	Debes hacerlo <code>&lt;em&gt;ahora&lt;/em&gt;</code> .

<code>&lt;b&gt;</code>	Pone el texto en negrita sin connotación semántica.	<code>&lt;b&gt;Texto en negrita&lt;/b&gt;</code>
<code>&lt;i&gt;</code>	Pone el texto en cursiva sin connotación semántica.	<code>&lt;i&gt;Texto en cursiva&lt;/i&gt;</code>
<code>&lt;script&gt;</code>	Incrusta o hace referencia a un script de JavaScript.	<code>&lt;script src="app.js"&gt;&lt;/script&gt;</code>
<code>&lt;link&gt;</code>	Vincula recursos externos, como hojas de estilo CSS.	<code>&lt;link rel="stylesheet" href="style.css"&gt;</code>
<code>&lt;meta&gt;</code>	Proporciona metadatos sobre el documento HTML.	<code>&lt;meta charset="UTF-8"&gt;</code>
<code>&lt;blockquote&gt;</code>	Define una sección que es una cita de otra fuente.	<code>&lt;blockquote&gt;Cita célebre.&lt;/blockquote&gt;</code>
<code>&lt;code&gt;</code>	Muestra un fragmento de código de programación.	<code>&lt;p&gt;Usa la etiqueta &lt;code&gt;&amp;lt;p&amp;gt;&lt;/code&gt;.&lt;/p&gt;</code>
<code>&lt;pre&gt;</code>	Muestra texto preformateado, conservando espacios y saltos.	<code>&lt;pre&gt;Código con formato.&lt;/pre&gt;</code>
<code>&lt;iframe&gt;</code>	Incrusta otro documento HTML en la página actual (ej. un video).	<code>&lt;iframe src="https://youtube.com/..."&gt;&lt;/iframe&gt;</code>
<code>&lt;video&gt;</code>	Incrusta contenido de video.	<code>&lt;video src="video.mp4" controls&gt;&lt;/video&gt;</code>
<code>&lt;audio&gt;</code>	Incrusta contenido de audio.	<code>&lt;audio src="sonido.mp3" controls&gt;&lt;/audio&gt;</code>
<code>&lt;source&gt;</code>	Especifica múltiples fuentes de medios para <code>&lt;video&gt;</code> o <code>&lt;audio&gt;</code> .	<code>&lt;video&gt;&lt;source src="vid.webm" type="video/webm"&gt;&lt;/video&gt;</code>

<code>&lt;figure&gt;</code>	Contenedor para contenido ilustrativo como imágenes o diagramas.	<code>&lt;figure&gt;&lt;img src="foto.jpg"&gt;&lt;figcaption&gt;...&lt;/figcaption&gt;&lt;/figure&gt;</code>
<code>&lt;figcaption&gt;</code>	Añade una leyenda o descripción a un <code>&lt;figure&gt;</code> .	<code>&lt;figcaption&gt;Foto de la ciudad.&lt;/figcaption&gt;</code>
<code>&lt;details&gt;</code>	Crea un widget interactivo que el usuario puede abrir y cerrar.	<code>&lt;details&gt;&lt;summary&gt;Más info&lt;/summary&gt;&lt;p&gt;...&lt;/p&gt;&lt;/details&gt;</code>

Exportar a Hojas de cálculo

## Parte 2: CSS — 20 formas de añadir estilo a tu HTML

CSS (Cascading Style Sheets) es el lenguaje que usamos para dar estilo y diseño a los elementos HTML. Aquí tienes 20 propiedades clave.

Propiedad	Qué hace	Ejemplo de Uso	Se suele aplicar a...
<code>color</code>	Cambia el color del texto.	<code>p { color: #333333; }</code>	Elementos de texto como <code>&lt;p&gt;</code> , <code>&lt;h1&gt;</code> , <code>&lt;span&gt;</code> .
<code>background-color</code>	Establece el color de fondo de un elemento.	<code>div { background-color: lightblue; }</code>	Casi cualquier elemento visible: <code>&lt;div&gt;</code> , <code>&lt;body&gt;</code> , <code>&lt;section&gt;</code> .
<code>font-size</code>	Define el tamaño de la fuente del texto.	<code>h1 { font-size: 24px; }</code>	Elementos de texto.
<code>font-family</code>	Especifica la familia tipográfica del texto.	<code>body { font-family: 'Arial', sans-serif; }</code>	Elementos de texto o contenedores para herencia.
<code>font-weight</code>	Define el grosor del texto (normal, bold).	<code>strong { font-weight: bold; }</code>	Elementos de texto.

<code>text-align</code>	Alinea el texto horizontalmente (left, center, right).	<code>h2 { text-align: center; }</code>	Elementos de bloque que contienen texto.
<code>margin</code>	Añade un espacio exterior (margen) alrededor de un elemento.	<code>div { margin: 20px; }</code>	Elementos de bloque como <code>&lt;div&gt;</code> , <code>&lt;p&gt;</code> , <code>&lt;section&gt;</code> .
<code>padding</code>	Añade un espacio interior (relleno) dentro de un elemento.	<code>button { padding: 10px 15px; }</code>	Casi cualquier elemento visible.
<code>border</code>	Define un borde alrededor de un elemento.	<code>img { border: 1px solid black; }</code>	Casi cualquier elemento visible.
<code>width</code>	Establece el ancho de un elemento.	<code>.container { width: 80%; }</code>	Elementos de bloque.
<code>height</code>	Establece la altura de un elemento.	<code>.hero { height: 500px; }</code>	Elementos de bloque.
<code>display</code>	Define cómo se muestra un elemento (block, inline, flex, grid).	<code>nav ul { display: flex; }</code>	Todos los elementos. Es fundamental para el layout.
<code>position</code>	Especifica el método de posicionamiento (static, relative, absolute).	<code>.modal { position: absolute; }</code>	Elementos que necesitan un posicionamiento preciso.
<code>flex</code>	Propiedad abreviada para controlar la flexibilidad de un item flex.	<code>.item { flex: 1; }</code>	Hijos directos de un contenedor con <code>display: flex</code> .
<code>grid-template-columns</code>	Define las columnas en un contenedor grid.	<code>.grid { grid-template-columns: 1fr 1fr 1fr; }</code>	Un contenedor con <code>display: grid</code> .

<code>justify-content</code>	Alinea los items en el eje principal de un contenedor flex o grid.	<code>.flex-container { justify-content: center; }</code>	Contenedores flex o grid.
<code>align-items</code>	Alinea los items en el eje secundario de un contenedor flex o grid.	<code>.flex-container { align-items: center; }</code>	Contenedores flex o grid.
<code>box-shadow</code>	Añade una sombra a la caja de un elemento.	<code>.card { box-shadow: 2px 2px 5px rgba(0,0,0,0.2); }</code>	Casi cualquier elemento visible.
<code>border-radius</code>	Redondea las esquinas de un elemento.	<code>button { border-radius: 5px; }</code>	Elementos con bordes visibles.
<code>opacity</code>	Define el nivel de transparencia de un elemento.	<code>.overlay { opacity: 0.5; }</code>	Todos los elementos.

## JavaScript — ¿Qué es el DOM y para qué se usa?

### ¿Qué representa el DOM en una página web?

El **DOM (Document Object Model)** o Modelo de Objetos del Documento es una **representación** de la estructura de una página web. Cuando un navegador carga un documento HTML, crea un "mapa" o un "árbol" de objetos en memoria. Cada etiqueta HTML, como `<body>`, `<h1>`, o `<p>`, se convierte en un "nodo" en este árbol.

Imagina que tu página HTML es un árbol genealógico:

- El nodo `<html>` es el ancestro de todos.
- `<body>` y `<head>` son sus hijos directos.
- `<body>` puede tener hijos como `<h1>`, `<p>` y `<div>`, y así sucesivamente.

El DOM no es el código HTML en sí, sino una **interfaz viva y dinámica** de ese código. Esto permite que lenguajes de programación como JavaScript puedan interactuar con la página, leer su contenido y modificar su estructura, estilo y contenido en tiempo real.

## ¿Cómo se puede manipular el DOM con JavaScript?

JavaScript puede "hablar" con el DOM para cambiar casi cualquier cosa en la página web después de que se haya cargado. El proceso generalmente sigue estos pasos:

1. **Seleccionar un elemento:** Primero, JS necesita identificar el elemento (o nodo) que quiere modificar. Para ello, utiliza métodos como `document.getElementById()`, `document.querySelector()` o `document.getElementsByClassName()`.
2. **Modificar el elemento:** Una vez seleccionado, se puede cambiar su contenido, sus atributos o su estilo.
  - a. **Contenido:** Usando propiedades como `innerHTML` o `textContent`.
  - b. **Estilo:** Accediendo a la propiedad `style` (ej. `elemento.style.color`).
  - c. **Atributos:** Usando métodos como `setAttribute()` (ej. para cambiar el `src` de una imagen).
3. **Crear y añadir nuevos elementos:** JS también puede crear nuevos nodos (ej. `document.createElement('p')`) y añadirlos al DOM en la posición deseada.

## 2 ejemplos prácticos de manipulación del DOM

Aquí hay dos ejemplos sencillos de cómo JavaScript puede cambiar una página web dinámicamente.

HTML

```
<h1 id="tituloPrincipal">¡Hola, Mundo!</h1>
<button id="miBoton">Cambiar Saludo</button>
```

JavaScript:

```
// 1. Seleccionar los elementos
const titulo = document.getElementById('tituloPrincipal');
const boton = document.getElementById('miBoton');

// 2. Añadir un "escuchador" de eventos al botón
boton.addEventListener('click', () => {
  // 3. Cambiar el contenido del título cuando se hace clic
  titulo.textContent = '¡Bienvenido a JavaScript!';
});
```