

Etiquetas HTML

Etiquetas de estructura de documento.

Etiquetas de estructura básica HTML

- Las primera etiquetas que vamos a ver son aquellas que forman la estructura base de cualquier documento HTML, y son las siguientes:
 - `<!DOCTYPE>`
 - `<html>`
 - `<head>`
 - `<title>`
 - `<base>`
 - `<meta>`
 - `<link>`
 - `<script>`
 - `<style>`
 - `<body>`

<!DOCTYPE>

- Esta etiqueta sirve para decirle al navegador qué tipo de documento HTML o XML se utilizó para escribir el documento.
- Se sitúa al principio de todos los documentos HTML
- En las versiones anteriores a HTML5 había diversos tipos de DOCTYPE
 - **HTML 4.01 Strict:**
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">`
 - **HTML 4.01 Transitional:**
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">`
 - **HTML 4.01 Frameset:**
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">`

<!DOCTYPE>

- **XHTML 1.0 Strict:**

- `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`

- **XHTML 1.0 Transitional:**

- `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`

- **XHTML 1.0 Frameset:**

- `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">`

- **XHTML 1.1:**

- `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">`

<!DOCTYPE>

- Afortunadamente en la versión HTML5, el DOCTYPE es muchísimo más simple y sencillo:
 - `<!DOCTYPE html>`
- No necesitamos poner nada más.

<html></html>

- Esta etiqueta le dice al navegador que es un documento HTML
- Representa la raíz de un documento HTML
- Se sitúa justo debajo del <!DOCTYPE> y es el contenedor del resto de etiquetas HTML salvo el propio <!DOCTYPE>
- Es muy recomendable que en esta etiqueta se especifique un atributo **lang** en el que se define el idioma principal del documento. De este modo ayudamos a los lectores de pantalla y a los programas de síntesis de habla a saber cual es la pronunciación adecuada, a las herramientas de traducción qué reglas deben utilizar, etc.
- Ejemplo:
 - <html lang="es"></html>
- Aquí se puede consultar un listado completo de los códigos de lenguaje que se pueden utilizar con el atributo **lang**:
 - [Códigos de representación de lenguajes ISO 639-1](#)

<head></head>

- Contenedor de los metadatos. Es el primer elemento situado dentro del par de etiquetas `<html></html>` y antes del par de etiquetas `<body></body>` y solo puede existir uno.
- Los metadatos definen el título del documento, conjunto de caracteres, enlaces a hojas de estilo, scripts y URL base
- Etiquetas permitidas dentro del contenedor `<head>`
 - `<title></title>` :En esta etiqueta va el título del documento. Sólo puede haber una.
 - `<base>` :Esta etiqueta nos permite definir la URL base del documento. Sólo puede haber una.
 - `<link>` :Define un enlace entre el documento y una fuente externa. Se usa para cargar hojas de estilo.
 - `<script></script>` :Aquí cargamos archivos JavaScript que se usarán en el documento*.
 - `<style></style>` :Aquí podemos definir estilos CSS en el propio documento. NO recomendable.
 - `<meta>` :Aquí definimos metadatos acerca del documento HTML
- Aquí podéis probar una web interesante sobre metadatos:
 - <https://www.metatags.org/>

<head></head>

- He aquí un ejemplo de uso de la etiqueta <head></head>

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">
  <meta name="description" content="Página de ejemplo HTML">
  <meta name="author" content="Juan Nadie">
  <meta http-equiv="refresh" content="30">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <base href="https://www.paginadeejemplo.com/" target="_blank">
  <link rel="stylesheet" type="text/css" href="app.css">
  <script type="text/javascript" async src="ejemplo.js"></script>
  <style></style>
  <title>Documento de ejemplo</title>
</head>
<body>
  <p>Documento de ejemplo</p>
</body>
</html>
```


<body></body>

- Es el segundo elemento situado dentro del par de etiquetas `<html></html>`, tras la etiqueta `<head></head>` y solo puede haber uno.
- Representa el contenido principal del documento.
- Con la llegada de HTML5 todos los atributos de layout se eliminaron.

<a>: El elemento ancla

- Es un hipervínculo hacia otras páginas web, archivos, localizaciones dentro de la misma página, direcciones de email o cualquier otra URL.
 - **Atributos más comunes:**
 - **download:** Este atributo le pide al navegador que descargue la URL en lugar de navegar hacia ella, pidiéndole al usuario que la guarde en algún lugar.
 - **href:** Dirección del hipervínculo. Puede ser:
 - Una URL absoluta hacia otra página web.
 - Una URL relativa que apunta a un archivo dentro de la web.
 - Un enlace a otro elemento con una id específica dentro de la página.
 - Otros protocolos (https://, ftp://, mailto:, file:, tel:)
 - Un script
 - **hreflang:** Indica el idioma del hipervínculo enlazado. Es sólo de aviso, no contiene funcionalidad

<a>: El elemento ancla

- **ping**: Especifica una lista de URLs a las cuales notificar si el usuario sigue el hipervínculo. Desactivado en algunos navegadores por motivos de seguridad (Firefox).
- **target**: Especifica dónde visualizar la URL enlazada
 - **_self**: Carga la URL dentro de la misma ventana en la que estamos. Valor por defecto.
 - **_blank**: Carga la URL en un nuevo contexto. Suele ser una nueva pestaña o ventana.
 - **_parent**: Carga la URL en el contexto padre del navegador. Si no existe el padre, se comporta igual que si fuese **_self**
 - **_top**: Carga la URL al nivel más alto del contexto de navegación. Si no existe el padre, se comporta igual que si fuese **_self**
- **type**: Especifica el tipo de medio en formato MIME para la URL enlazada. Es sólo de aviso, no contiene funcionalidad.

<button></button>: El Elemento Botón

- Representa un botón pulsable, que puede utilizarse en formularios o en cualquier otra parte del documento en la que haga falta una funcionalidad simple y estándar de botón.
- Los elementos <button></button> usualmente se representan con el aspecto por defecto del sistema operativo de la plataforma donde se muestran, pero puede modificarse con **CSS**
- Dentro de un botón se permiten otros elementos en línea salvo que estos sean a su vez interactivos (como puede ser un <a>)

[Especificación del elemento <button></button> en el W3C](#)

: El Elemento Imagen

- Sirve para embeber una imagen dentro del documento.
 - **Atributos más comunes:**
 - **alt**: Texto que reemplaza a la imagen en caso de que esta no esté disponible. También sirve como descripción para los lectores de pantalla.
 - **src**: URL con la dirección de la imagen a embeber.
 - **width**: Establece un ancho de la imagen en píxeles.
 - **height**: Establece un alto de la imagen en píxeles.

[Especificación del elemento en el W3C](#)

<video></video>: El Elemento Vídeo

- Este elemento sirve para embeber un reproductor de medios audiovisuales que nos permite reproducir un vídeo dentro del documento.



- **Atributos más comunes:**

- **autoplay:** Atributo booleano. Si se especifica, reproduce el vídeo tan pronto como le es posible sin esperar a cargar todo el vídeo. No es recomendable usarlo por problemas de accesibilidad y experiencia de usuario.
- **controls:** Atributo booleano. Si se especifica, el navegador ofrecerá controles para que el usuario pueda controlar la reproducción del vídeo (play, pausa, búsqueda y volumen).
- **height:** Aquí podemos establecer la altura del área de visualización en píxeles (sin porcentajes).
- **loop:** Atributo booleano. Si se especifica repite el vídeo en bucle.
- **muted:** Atributo booleano. Si se especifica el vídeo aparecerá inicialmente silenciado.
- **src:** Aquí definimos la URL desde la que se cargará el vídeo.
- **width:** Aquí podemos establecer la anchura del área de visualización en píxeles (sin porcentajes).

[Especificación del elemento <video></video> en el W3C](#)

<article></article>: El Elemento Artículo



- Este elemento representa una composición auto-contenida en un documento, página, aplicación o site. Está pensado para ser distribuido o reutilizado de forma independiente.
- Un artículo puede posicionarse en cualquier documento sin perder su significado e identidad propias, independiente al contexto en el que se encuadre el documento.
- Algunos ejemplos son posts en un foro, un artículo de una revista o periódico o una entrada en un blog.
- Un documento puede tener numerosos artículos, y estos a su vez tener secciones o artículos dentro.
- Deben identificarse incluyendo un heading (<h1>-<h6>) dentro del mismo.

<aside></aside>: El Elemento Acotación



- Representa una porción de un documento cuyo contenido está indirectamente relacionado con el contenido principal del documento, a modo de acotación, aclaración, o de proporcionar información adicional al contenido principal.
- Un <aside></aside> depende del contexto del documento en el que se sitúa, de modo que carece de significado propio si se extrae del mismo.
- Se suelen presentar en forma de barras laterales o cajas de texto o contenido anexas.

[Especificación del elemento <aside></aside> en el W3C](#)

`<div></div>`: El Elemento División de contenido

- Es un contenedor genérico y neutro de bloque, sin ninguna carga semántica.
- Pueden anidarse, así como contener a su vez dentro de si otros elementos de bloque y/o línea.
- Sirve para seccionar o dividir los contenidos con propósitos estéticos o necesidades de estructura.
- Debe utilizarse cuando no sea apropiado el uso de otros elementos de agrupación semántica.

[Especificación del elemento `<div></div>` en el W3C](#)

`<footer></footer>`: El Elemento Pie



- Representa un pie para su elemento de seccionado ancestro más cercano o el elemento raíz de seccionado.
- Usualmente contiene información sobre el autor de la sección, datos de copyright o enlaces a documentos relacionados.

[Especificación del elemento `<footer></footer>` en el W3C](#)

<h1>-</h6>: El Elemento Encabezado

- Representan seis niveles de encabezado de sección. El elemento `<h1></h1>` es el nivel más alto de sección, mientras que el elemento `<h6></h6>` es el de menor nivel.
- Se utilizan para otorgarle un orden jerárquico al contenido del documento, en cada una de sus secciones.
- Debe respetarse la jerarquía de los encabezados; después de un `<h1></h1>` no podemos poner un `<h4></h4>` porque rompemos el orden jerárquico.
- Sólo debería existir un único `<h1></h1>` en todo el documento, que actúa como encabezado principal.
- **JAMÁS DEBEN UTILIZARSE** con propósitos estéticos y visuales para cambiar el tamaño del texto, puesto que son elementos con gran carga semántica. Para eso debe utilizarse CSS.

<header></header>: El Elemento Cabecera



- Representa un contenido introductorio usualmente un grupo de ayudas de navegación o introductorias. También puede contener encabezados, un logotipo, un formulario de búsqueda, el nombre del autor, y otros elementos.

[Especificación del elemento <header></header> en el W3C](#)

<main></main>: El Elemento Principal

- Representa el contenido dominante del <body> de un documento. El área de contenido principal consiste en contenido que está directamente relacionado con el tema central de un documento o la funcionalidad central de una aplicación.
- Debe ser único, no debe haber más de un elemento <main></main> en el documento.
- El contenido que se repite en un conjunto de documentos o secciones de documentos, como barras laterales, enlaces de navegación, información de copyright, logotipos de sitios y formularios de búsqueda, no debe incluirse a menos que el formulario de búsqueda sea la función principal de la página.
- El elemento <main></main> no contribuye al esquema del documento; es decir, a diferencia de elementos como <body></body>, los encabezados como <h2></h2>, y similares, <main></main> no afecta al concepto del **DOM** sobre la estructura de la página. Es estrictamente informativo.



<nav></nav>: El Elemento Navegación



- Representa una sección de una página cuyo propósito es proporcionar enlaces de navegación, ya sea dentro del propio documento actual o hacia otros documentos. Algunos ejemplos comunes de secciones de navegación son los menús, tablas de contenido e índices.
- Los enlaces de navegación deben estar insertados en una lista ordenada o desordenada; `` O ``
- No es necesario que todos los enlaces estén contenidos en un elemento `<nav>`. El elemento `<nav>` está destinado únicamente a los principales bloques de enlaces de navegación; el elemento `<footer>` suele tener una lista de enlaces que no necesitan estar en un elemento `<nav>`.
- Un documento puede tener varios elementos de navegación. Para distinguirllos es conveniente utilizar `aria-label`.

<p></p>: El Elemento Párrafo

- El elemento HTML <p> representa un párrafo. Los párrafos suelen representarse en los medios visuales como bloques de texto separados de los bloques adyacentes por líneas en blanco y/o sangría de primera línea, pero los párrafos HTML pueden ser cualquier agrupación estructural de contenido relacionado, como imágenes o campos de formulario.
- **NO** se deben utilizar párrafos en blanco para separar un párrafo con contenido de otro. Para ello lo correcto es utilizar **CSS**
- **NUNCA** se debe anidar un elemento <p></p> dentro de otro, porque es semánticamente absurdo.

<section></section>: El Elemento Sección



- Representa una sección independiente -que no tiene un elemento semántico más específico para representarla- contenida en un documento HTML. Es una agrupación temática de contenido. El elemento `<section>` debe llevar un encabezado con un título.
- Si el contenido de una sección tiene significado por si mismo si lo separamos del documento principal, entonces es más adecuado utilizar un elemento `<article>`
- No hay que utilizar el elemento `<section>` como contenedor genérico o para dividir contenido con propósitos estéticos; para eso ya tenemos el elemento `<div>`