



HTML

DISEÑO WEB

Carlos Rojas Sánchez

Licenciatura en Informática

Universidad del Mar

1. Estructura básica de un documento HTML
2. Etiquetas meta
3. Etiquetas head

Estructura básica de un documento HTML

HyperText Markup Language (Lenguaje de marcas de hipertexto), un lenguaje de etiquetas que permite incluir o hacer referencia a todo tipo de información. [1]

Las etiquetas son la base del lenguaje HTML. Existen muchas etiquetas y cada una se utiliza para contener información y darle un cierto significado a dicha información, dependiendo de la etiqueta que se trate. Las etiquetas HTML tienen la siguiente estructura:

```
<etiqueta>contenido</etiqueta>
```

En HTML no se puede utilizar cualquier palabra como etiqueta (en el ejemplo anterior, es incorrecto utilizar la etiqueta etiqueta). En su lugar, existen una serie de etiquetas concretas, cada una de ellas con su finalidad y características propias, que tendremos que utilizar según requiera la ocasión. Por norma general, las etiquetas deben cerrarse para indicar donde finaliza su contenido.

Año	Evento
1991	Se publica la primera versión de HTML (HTML 1.0).
1993	Se publica la segunda versión de HTML (HTML 2.0).
1995	Se publica la tercera versión de HTML (HTML 3.0), que incluye soporte para hojas de estilo en cascada (CSS) y scripts.
1997	Se publica la cuarta versión de HTML (HTML 4.0), que incluye más elementos y atributos.
1999	Se publica la especificación de XHTML 1.0, una versión más estricta y modular de HTML.
2000	Se publica HTML 4.01, una revisión menor de HTML 4.0.
2004	Se publica la especificación de XHTML 2.0, una versión radicalmente diferente de XHTML que nunca se implementó completamente.
2008	Se publica HTML5, que incluye nuevas funcionalidades como video y audio nativos, canvas y geolocalización.
2014	Se publica la especificación de HTML5.1, una revisión menor de HTML5.
2016	Se publica la especificación de HTML5.2, otra revisión menor de HTML5.
2017	El W3C anuncia que HTML5 es la especificación oficial para la web.

¿Qué es la semántica?

En el documento HTML debe aparecer información correctamente individualizada, de modo que al leer una página HTML comprendamos su significado, y si queremos cambiar la apariencia, lo hagamos en el documento CSS. Esto es lo que comunmente se conoce como separación de la presentación del contenido.

El objetivo de crear documentos HTML semánticos es que, aunque estamos acostumbrados a crear páginas para usuarios (o más concretamente, para navegadores), cada vez tendemos más a una Internet capaz de procesar información de forma autónoma.

ESTRUCTURA DE UNA ETIQUETA HTML



The diagram illustrates the structure of an HTML tag: `<etiqueta atributo = "valor"> contenido </etiqueta>`. The components are color-coded as follows: `<etiqueta` is in an orange box, `atributo` is in a green box, `= "valor"` is in a blue box, `>` is in a yellow box, `contenido` is in a purple box, and `</etiqueta>` is in an orange box. The entire tag is shown within a larger orange container box.

Existen 3 tipos de atributos dependiendo de sus valores:

Conjunto de valores Son aquellos atributos en los que sólo se permiten unos valores concretos. Cualquier otro valor diferente, no será válido.

Valores libres Son los atributos en los que puedes especificar un valor libremente, como una dirección URL o un texto, y no existe una serie de valores específicos para escribir.

Valores booleanos Son los atributos que deben tener un valor verdadero (true) o un valor falso (false). En HTML5 un atributo sin valor (solo el atributo) significa verdadero y si se omite el atributo se considera falso.

Atributos comunes

Atributo	Valor	Descripción
id	<u>nombre</u>	Establece un identificador único a la etiqueta HTML. Sólo el mismo nombre una vez por documento.
class	<u>nombre</u>	Establece una clase (género) a una etiqueta HTML. Puede repetirse por documento.
style	<u>estilos CSS</u>	Aplica propiedades CSS directamente al elemento HTML en cuestión.

El atributo `class` puede tener múltiples clases diferentes, no necesariamente una sola. Esto nos da más flexibilidad a la hora de crear clases específicas, donde cada una se ocupa de un detalle diferente.

ESTRUCTURA DE UN DOCUMENTO HTML

```
<!DOCTYPE html>
```

TIPO DE DOCUMENTO

```
<html>
```

DOCUMENTO

```
<head>  
</head>
```

CABECERA DE PÁGINA

```
<body>  
</body>
```

CUERPO DE PÁGINA

```
</html>
```

El !DOCTYPE o tipo de documento es una etiqueta especial que se escribe en la primera línea del documento HTML.

`<!-- Actual -->`

`<!DOCTYPE html>`

`<!-- Antes -->`

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

La etiqueta head de un documento HTML, muchas veces la denominamos cabecera HTML. Sin embargo, esta cabecera no es una parte visual de la página web, sino una parte de nuestro código HTML donde se incluyen ciertas etiquetas de metadatos, es decir, unas etiquetas que establecen ciertos datos que no tienen que verse necesariamente de forma visual. También es el lugar indicado para incluir etiquetas que relacionen el documento actual con otros documentos externos (por ejemplo, para cargar los estilos .css o ficheros javascript .js).

- Título de la página web (aparece en la pestaña del navegador). Esta etiqueta es obligatoria.
- Descripción de la página (aparece en los resultados de Google)
- Miniatura/preview de la página (aparece al poner enlace en redes sociales)
- Icono de la página (aparece en la pestaña del navegador)

La etiqueta `body` es donde van todos los elementos visuales de una página, por lo que es una de las partes más importantes de una página web.

Con el paso del tiempo y la transición desde versiones anteriores a HTML5 (por ejemplo, desde HTML4 o XHTML), hay muchas etiquetas HTML que han sido marcadas como obsoletas y se recomienda dejar de utilizarlas cuanto antes. Es muy posible que algunas incluso funcionen aún en navegadores actuales, pero esto ocurre por retrocompatibilidad y llegará un momento que dejarán de ser soportadas por el navegador.

Etiqueta obsoleta	Descripción	Alternativa
<applet>	Etiqueta para applets Java.	Javascript, DOM y CSS
<acronym>	Indica un acrónimo.	<abbr>
<bgsound>	Especifica un sonido de fondo.	<audio>
<dir>	Indica una lista de archivos o carpetas.	
<frame>	Define un marco específico.	<iframe>
<frameset>	Define un conjunto de marcos.	↑
<noframes>	Indica una alternativa si el navegador no soporta marcos.	↑
<isindex>	Campo para buscar en el documento.	<input>
<listing>, <xmp>	Fragmentos de código fuente.	<pre><code>
<noembed>	Alternativa (fallback) para contenidos.	<object>
<strike>	Muestra un texto tachado.	<s>
<basefont>	Define una tipografía por defecto.	CSS: font-family
<big>	Aumenta el tamaño del texto.	CSS: font-size
<blink>	Muestra el texto de forma parpadeante.	CSS: animation

<code><center></code>	Centra el texto.	CSS: text-align
<code></code>	Cambia la tipografía o sus características.	CSS: font-family
<code><marquee></code>	Muestra el texto moviéndose de un lado a otro.	CSS: animation
<code><multicol></code>	Columnas múltiples.	CSS: columns
<code><nobr></code>	Evita que un texto haga un salto de línea.	CSS: white-space
<code><spacer></code>	Inserta un espacio horizontal.	<code>&nbsp;</code>
<code><tt></code>	Muestra el texto con una fuente monoespaciada.	<code><code></code>

Atributo obsoleto	Pertenece a...	Descripción	Alternativa
charset	<a>, <link>	Codificación del enlace.	
name	<a>, <embed>, , <option>	(Usándolo para establecer un ancla).	id
language	<script>	Indica el lenguaje utilizado.	type
link	<body>	Indica el color de los enlaces.	CSS: :link
alink	<body>	Indica el color de los enlaces.	CSS: :active
vlink	<body>	Indica el color de los enlaces.	CSS: :visited
bgcolor	<body>	Indica el color de fondo de página.	CSS: background-color
align, valign	<table>	Indica la alineación vertical/horizontal.	CSS: alineaciones
hspace, vspace	<table>, <iframe>	Indica espacios en tablas/marcos.	CSS: modelo de cajas
allowtransparency	<iframe>	Permite transparencia.	
frameborder	<iframe>	Indica el borde de los marcos flotantes.	CSS: bordes
scrolling	<iframe>	Permite o no hacer scroll en un marco.	CSS: overflow
align, border	, <table>	Indica que alineación o borde utilizar.	
cellpadding	<table>	Indica espacios entre celdas de tabla.	CSS: tablas
cellspacing	<table>	Indica espacios entre celdas de tabla.	CSS: tablas
nowrap	<td>, <th>	Establece fijo el tamaño de una celda.	CSS: tamaños

Tipo de error	Descripción
Sintaxis	El código está mal escrito y es incorrecto. El navegador podría no mostrar correctamente ciertos detalles.
Accesibilidad	El código no tiene porque estar mal escrito o ser incorrecto, sin embargo, tiene problemas que harán que no se vean bien en determinados dispositivos o que un usuario invidente o similar no pueda utilizarla.
Posicionamiento (SEO)	El código no tiene porque estar mal escrito o ser incorrecto, sin embargo, los buscadores como Google no aceptarán favorablemente la página y podría tener un rendimiento menor de posicionamiento SEO.
Rendimiento	El código no tiene porque estar mal escrito o ser incorrecto, sin embargo, el navegador puede ser lento a la hora de cargarlo o tardar en realizar sus tareas.
Usabilidad	El código no tiene porque estar mal escrito o ser incorrecto, sin embargo, la forma en la que se disponen los elementos o está pensado puede frustrar al usuario final que utiliza la web.


Para asegurarnos de que nuestro código está correctamente escrito, podemos utilizar un Validador HTML, que no es más que un sistema que analiza nuestro código y nos dice el número de errores que tenemos, junto a una breve descripción del mismo para facilitar su corrección.


Este proceso de validación se puede realizar mediante la herramienta oficial HTML Validator de W3C, un validador de código HTML5, donde podemos validar nuestra página de tres formas diferentes:

By URI Indicando la URL de la página que se quiere validar. Obviamente, la página debe estar publicada en Internet.

By File upload Seleccionando el archivo .html a analizar y subiéndolo al validador.

By Direct Input Pegando directamente el código HTML que se quiere analizar en una casilla de texto.

← → ↻  validator.w3.org



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI

Validate by File Upload

Validate by Direct Input


Validate by URI

Validate a document online:

Address:


► [More Options](#)

This validator checks the [markup validity](#) of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as [RSS/Atom feeds](#) or [CSS stylesheets](#), [MobileOK content](#), or [validators and tools](#) available. As an alternative you can also try our [non-DTD-based validator](#).



Interested in understanding what new technologies are coming out of W3C? Follow [@w3cdevs on X](#) to keep track of what the future has in store. [Donate](#) and help us build better tools for a better web.

[Home](#) [About...](#) [News](#) [Docs](#) [Help & FAQ](#) [Feedback](#) [Contribute](#)



COPYRIGHT © 2024 WORLD WIDE WEB CONSORTIUM. [W3C](#) LIABILITY, TRADEMARK AND PERMISSIVE LICENSE RULES APPLY.

Etiquetas meta

Atributos de metadatos

Atributo	Valor	Descripción
<code>title</code>	<code>mensaje</code>	Mensaje mostrado en un tooltip (aviso emergente) al mover el ratón encima.
<code>data-*</code>	<code>texto</code>	Metadatos en la propia etiqueta. Se puede usar cualquier nombre con prefijo <code>data-</code> .
<code>accesskey</code>	<code>atajo</code>	Combinación de teclas que puede pulsar el usuario para activar el elemento.

Atributos de metadatos

Aunque se suele hacer sobre todo en las etiquetas de imágenes `img`, en la mayoría de las etiquetas HTML podemos indicar el atributo `title` para especificar un mensaje de texto que aparezca cuando el usuario detenga el ratón sobre el elemento un instante.

```
<div>
  
  <p>Mantén el ratón unos instantes sobre la imagen anterior.</p>
</div>
```

Atributos de metadatos

En un documento HTML, la mayoría de los metadatos (información adicional) se incluyen en el interior de la etiqueta head del documento HTML. Sin embargo, también se pueden incluir metadatos en una etiqueta HTML a través de un atributo que comienza con el prefijo data-.

```
<div id="social-networks">  
  <a href="https://twitter.com/Manz" data-subs="44000">Twitter</a>  
  <a href="https://youtube.com/@ManzDev" data-subs="10000">Youtube</a>  
  <a href="https://twitch.tv/ManzDev" data-subs="13000">Twitch</a>  
</div>
```

Atributos de metadatos

En HTML es posible añadir el atributo `accesskey` para indicar un atajo de teclado que puede pulsar el usuario para activar ese elemento.

```
<form>
  <input accesskey="N" placeholder="Nombre (ALT+N)" />      <⚡ Campo de datos →
  <input accesskey="A" placeholder="Apellidos (ALT+A)" />    <⚡ Campo de datos →
  <a accesskey="L" href="https://manz.dev/">Enlace (ALT+L)</a> <⚡ Enlace HTML →
  <button accesskey="B">Botón (ALT+B)</button>               <⚡ Botón HTML →
</form>
```

Atributos de idioma

Atributo	Valor	Descripción
lang	<u>idioma</u>	Indica el idioma del contenido de la etiqueta HTML.
translate	yes no	Indica si el contenido de la etiqueta se debería traducir o no.
dir	ltr rtl	Establece la direccionalidad del texto (<u>left to right</u> o <u>right to left</u>).

Atributos de idioma

Mediante el atributo lang podemos indicar el idioma del contenido de la etiqueta. El valor de dicho atributo tendrá que ser el código ISO 639-1 del idioma al que queremos hacer referencia. Aunque en principio podemos hacer esto en cualquier etiqueta HTML, es obligatorio hacerlo en la etiqueta html que es la etiqueta que abarca todo el documento.

```
<!DOCTYPE html>  
<html lang="es">  
  ...  
</html>
```

En las etiquetas HTML se puede indicar el atributo `translate`, el cuál acepta los valores `yes` y `no`. Por defecto, aunque este atributo no sea añadido en una etiqueta, el valor por defecto que tiene es `yes`. Por lo tanto, todas las etiquetas están marcadas como traducibles.

```
<p>  
  Hace algunos días fui a ver la nueva película de <span translate="no">StarWars</span>.  
</p>
```

Existe un atributo `dir` que permite al desarrollador indicar la direccionalidad del texto en el documento. Esto es ideal para idiomas en los que se escribe o lee de derecha a izquierda, en lugar de izquierda a derecha. El valor por defecto de este atributo es `ltr` (left to right, de izquierda a derecha), pero podemos modificarlo y establecer el valor `rtl` (right to left, de derecha a izquierda).

```
<p>Esto es un ejemplo de texto con direccionalidad de izquierda a derecha.</p>  
<p dir="rtl">Esto es un ejemplo de texto con direccionalidad de derecha a izquierda.</p>
```

Etiquetas head

Etiqueta head

La etiqueta head simplemente es una etiqueta contenedora que puede englobar muchas otras etiquetas, generalmente para definir metadatos o relaciones. Es importante tener en cuenta que, en general, las etiquetas que se colocan en la parte del head son etiquetas que necesitamos que se lean y carguen antes de que el navegador comience a dibujar la página.

Etiqueta HTML	¿Dónde se pueden usar?		Descripción
<title>	✓ <head>	✗ <body>	Indica el título de la página (pestaña o SEO en Google).
<base>	✓ <head>	✗ <body>	URL base de los enlaces (para gestionar rutas relativas).
<link>	✓ <head>	✗ <body>	Establece una relación del documento actual con otro externo.
<meta>	✓ <head>	✗ <body>	Establece un metadato específico en el documento actual.
<style>	✓ <head>	✓ <body>	Estilos CSS que afectan únicamente al documento actual.
<script>	✓ <head>	✓ <body>	Indica un script a cargar o ejecutar en la página actual.

Etiqueta meta

La etiqueta meta con el atributo charset es una etiqueta que se utiliza para indicar al navegador la codificación a utilizar. En nuestro caso, usaremos UTF-8, que es la estándar hoy en día.

```
<head>  
  <title>Título del documento</title>  
  <meta charset="UTF-8">  
</head>
```

La etiqueta title sirve para indicar un título para nuestra página o documento HTML, y es obligatoria, por lo que cualquier documento HTML debería tener una etiqueta title (y sólo una, no deben existir varios).

- Los títulos no deben estar vacíos, ni con texto incompleto.
- El título debe ser descriptivo, breve y claro y reflejar el contenido de la página.
- Se suele decir que lo recomendable es entre 50-60 caracteres.



J. Román.

Lenguaje HTML5.

<https://lenguajehtml.com/>, 2008.

[Online; accessed 5-October-2024].