

# Lenguajes de marcas y sistemas de gestión de la información

## TEMA 2: Lenguajes de marcas en entornos Web: HTML5.

### HTML

#### **Introducción:**

El padre del HTML es Tim Berners Lee, que desarrolló su idea mientras trabajaba en el CERN (Centro Europeo para la investigación Nuclear), su objetivo inicial era facilitar el acceso a todos los datos del Centro independientemente del ordenador en el que se encontrara esa información (en el propio centro o en los ordenadores de las diferentes instituciones que colaboraban con el CERN).

El producto de su idea sería una plataforma de tipo hipertexto y un protocolo de comunicaciones que se denominó HTTP (Hyper Text Transfer Protocol).

Hablamos del concepto de Hipertexto (Conocido también como link) con el que se pueden conectar dos elementos entre si y el SGML (Standard General Markup Language) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse.

La primera versión de HTML sólo mostraba texto con formato, contenía etiquetas para títulos, párrafos, listas y viñetas, entre otras características.

A partir de entonces se formó el grupo Internet Engineering Task Force (IETF), encargado de estandarizar este lenguaje para que fuera compatible y mostrado en cualquier navegador. Bastaron cuatro años para que se lograra la meta y en 1995 publica el lenguaje estándar HTML 2.0:

- HTML 2.0 fue la primera versión oficial de HTML el IETF publicó el estándar en septiembre de 1995.
- HTML 3.2 se publicó el 14 de enero de 1997 por el W3C. Incorpora los applets de Java y texto alrededor de las imágenes.
- HTML 4.0 se publicó el 24 de abril de 1998. Entre las novedades que presenta se encuentran las hojas de estilos CSS y la posibilidad de incluir pequeños programas en las páginas web.
- HTML 4.01 se publicó el 24 de diciembre de 1999. Es una actualización de la versión anterior.
- En ese momento el W3C detuvo la actividad de estandarización de HTML hasta marzo de 2007, momento en que se retoma debido a la fuerza de las empresas que forman el grupo **WHATWG(Web Hypertext Application Technology Working Group)**, auspiciada por Apple, Google y Mozilla entre otras empresas a las que se han ido añadiendo muchas más creador de HTML5.
- HTML 5, se publicó el 28 de octubre de 2014. Es la versión actual e incorpora etiquetas nuevas para el marcado de la multimedia, sobre todo el video.

Actualmente la última versión es <https://www.w3.org/TR/html52/>

## Estructura HTML

**Prólogo:** Todo documento HTML ha de tener una declaración del tipo de documento para que el navegador sepa:

- El tipo de documento que se va a iniciar.

- La versión de HTML utilizada para la codificación del mismo para poder interpretarlo correctamente.

### **Tipos de documentos HTML:**

<https://www.w3.org/QA/2002/04/valid-dtd-list.html>

- HTML 4.0 Strict. DTD utilizada por defecto con HTML 4.0. En estos documentos no se permite el uso de los elementos declarados deprecated (obsoletos) en otras versiones o Recomendaciones HTML. La declaración del tipo de documento correspondiente es:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 //EN"  
"http://www.w3.org/TR/REC-html40/strict.dtd">
```

- HTML 4.0 Transitional. Permite el uso de todos los elementos que permite el HTML 4.0 Strict, además de los elementos deprecated. La declaración del tipo de documento correspondiente es:


```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN"  
"http://www.w3.org/TR/REC-html40/loose.dtd">
```

- HTML 4.0 Frameset. Es una variante de HTML 4.0 Transitional para documentos que usan frames. En estos documentos el elemento body hay que reemplazarlo por un elemento frameset. La declaración del tipo de documento correspondiente es:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN"  
"http://www.w3.org/TR/REC-html40/frameset.dtd">
```

- HTML 5. La declaración del tipo de documento correspondiente es:

```
<!DOCTYPE HTML>
```



**Ejemplar:** Un documento HTML está delimitado por las etiquetas <html> y </html>. El ejemplar que alberga se divide en dos partes:

- **Head (cabecera):** delimitada por las etiquetas <head> y </head>.
  - Contiene la información sobre el título de la página, el autor, palabras clave, etc.
  - Es obligatorio definir el título del documento, para ello se usan las etiquetas <title> </title>.
  - Esta información no se presentará en la ventana del navegador, salvo el título que aparecerá en la barra de título de la parte superior.
- **Body (Cuerpo):** contiene la información que se va a presentar en la pantalla. Está limitado por las etiquetas <body> y </body>, salvo en los documentos de tipo HTML 4.0 Frameset donde éstas se sustituyen por <frameset> y </frameset>.

## Elementos de la estructura básica del documento

Para crear una página web usando HTML 5 la estructura básica es la siguiente:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8" >
    <title></title>
  </head>
  <body>

  </body>
</html>
```

- **Línea DOCTYPE.** La etiqueta DOCTYPE indica el tipo de HTML que estamos utilizando (concretamente HTML 5). Se trata de una línea que en la actualidad siguen ignorando casi todos los navegadores, pero que es muy recomendable (y muy respetuoso con la norma) escribir. En ella se indica la versión HTML que sigue el documento.
- **Etiqueta html.** Todo documento HTML está encerrado dentro de la apertura y el cierre de esa etiqueta. Marca el principio y fin del mismo (marca el elemento raíz de un documento HTML). Se le debe añadir el atributo lang para indicar el idioma en el que está escrito el documento.

- **Cabecera, etiqueta head.** Encierra las etiquetas de cabecera, las cuales proporcionan información que el navegador debe tener en cuenta para el contenido del documento aparezca de forma correcta y también contiene elementos que proporcionan información a los buscadores (como Google). Los elementos habituales que contiene son:
- **Etiquetas meta.** Que sirven para indicar aspectos de funcionamiento de la página web como, cabeceras del protocolo http, indicaciones sobre cada cuánto caduca la página, palabras clave para buscar la página en los navegadores, codificación de caracteres de la página. En este caso sólo se usa la etiqueta meta obligatoria que indica que estamos codificando el texto usando el formato utf-8 de Unicode.
- **título, etiqueta title.** El título de la página, que los navegadores suelen colocar en la barra de título de la ventana en la que se muestra la página.
- **Declaración de archivos externos y scripts.** En el ejemplo no hay declaraciones de este tipo.
- **Cuerpo.** Encerrado en la etiqueta body, encierra el contenido en sí de la página web. Lo que el navegador muestra por pantalla es lo que se coloca en este apartado.

## HTML. Etiquetas y Atributos

Un documento HTML está formado por etiquetas y atributos.

Al igual que en XML las etiquetas pueden ser de apertura, <etiqueta>, o de cierre, </etiqueta>.

*A diferencia de XML, la cantidad de etiquetas en HTML es limitada.*

Aunque HTML define una gran cantidad de etiquetas, estas no son suficientes para crear páginas complejas ya que la definición completa de ciertos elementos, como las imágenes y los enlaces, requiere información adicional. Como no es posible crear una etiqueta por cada elemento diferente, se añade la información adicional a las etiquetas mediante los atributos dando lugar a los elementos.

Para cada uno de los atributos hay definido un conjunto de valores que se le puede asignar, si el valor de un atributo no es válido, el navegador lo ignora.

La forma de utilizar atributos es:

<elemento nombreDeAtributo1="valor1" nombreDeAtributo2="valor1" ... >

A los atributos se les asignan valores que deben ir entre comillas. El símbolo de igualdad (=) es obligatorio. En el cierre solo se cierra la etiqueta, sin indicar los atributos.

Cada una de las etiquetas HTML define los atributos que puede utilizar, aunque algunos de ellos son comunes a muchas etiquetas.

## Atributos comunes o básicos

Todos los elementos tienen una serie de atributos comunes a todos ellos. Eso significa que podemos utilizarlos independientemente de cuál sea el elemento.

El atributo común más conocido es id, que permite identificar a un elemento con un identificador único, pero hay muchos más. Los principales son:

Atributo	Significado
lang	Indica el lenguaje en el que está escrito el texto del elemento
style	Permite indicar código CSS directamente. Ese código solo afectará a ese elemento.
class	Permite indicar una clase CSS para el elemento. No pueden empezar por números y sólo puede contener letras, números, guiones medios y/o guiones bajos.
id	Permite identificar al elemento. Un identificador es un nombre único que se le da al elemento y que no se puede repetir en otro elemento dentro de la misma página web. El identificador solo puede contener letras (del alfabeto inglés) y números (no se pueden usar espacios en blanco). Además debe empezar por una letra. El atributo id es uno de los fundamentales.
title	Permite poner un título al elemento. Es una especie de descripción corta que es muy útil en aquellos elementos que requieren de una explicación semántica como una abreviatura, una sigla, una imagen, etc. Normalmente, los navegadores reaccionan a este atributo mostrando un cartelito con el texto indicado en title, cuando el cursor del ratón se aproxima a un elemento que use este atributo

## Atributos para la internacionalización

Los utilizan las páginas que muestran sus contenidos en varios idiomas o aquellas que quieren indicar

de forma explícita el idioma de sus contenidos.

Atributo	Significado
<b>dir</b>	Indica la dirección del texto por lo que sólo puede tomar dos valores:  <b>ltr (left to right)</b> de izquierda a derecha. Es el valor por defecto.  <b>rtl (right to left)</b> de derecha a izquierda.
<b>lang = "codigo"</b>	Especifica el idioma del elemento mediante un código predefinido. Los posibles valores de este atributo se encuentran en el documento RFC 1766, algunos de los valores posibles son:  en(inglés de Gran Bretaña), en-US (inglés americano), es (Español)
<b>xml:lang = "codigo"</b>	Especifica el idioma del elemento mediante un código definido según la recomendación RFC 1766.Procede de XML, significa lo mismo que lang, pero es entendible por software capaz de analizar XML. Solo se usa en XHTML (no se usa en HTML 5)

En las páginas XHTML, el atributo xml:lang tiene más prioridad que lang y es obligatorio incluirlo siempre que se incluye el atributo lang.

Como se ha indicado anteriormente, el atributo lang permite indicar dos letras que significan el idioma en el que está escrito el texto del elemento. Lo normal es indicarle en el elemento html (por ejemplo <html lang="es">), pero hay veces que una página contiene elementos cuyo contenido está escrito en un idioma diferente al indicado como general en el apartado <html>.

Como el atributo lang, es común a todos los elementos, se puede indicar el mismo cuando sea necesario. Ejemplo:

<p>

Como dirían en Estados Unidos:

<span lang="en">Welcome!</span>

</p>

¿Por qué tomarse tantas molestias? Para que, por ejemplo, los buscadores de Internet distingan el texto en el idioma apropiado y adapten mejor nuestras búsquedas. Otras herramientas (como correctores ortográficos, traductores en línea, etc.) funcionan mucho mejor marcando adecuadamente el idioma.

### Atributos de eventos y atributos para los elementos que pueden obtener el foco

Solo se utilizan en las páginas web dinámicas creadas con JavaScript.

Atributo	Significado
<b>onclick</b>	Permiten asignar código Javascript a alguna de esas acciones (un clic de ratón, un doble clic, pulsar tecla, dejar de pulsar tecla, mantener pulsada la tecla, entrar con el ratón en el elemento, mover el ratón por encima, sacar el ratón del elemento, pulsar tecla de ratón, liberar tecla de ratón,...)
<b>ondblclick</b>	
<b>onkeyup</b>	
<b>onkeydown</b>	
<b>onkeypress</b>	
<b>onmouseover</b>	
<b>onmouseout</b>	
<b>onmousedown</b>	
<b>onmouseup</b>	



<b>onmousemove</b>	
<b>draggable</b>	Atributo incluido en HTML 5. Especifica (con valor <b>true</b> ) que el párrafo tiene posibilidad de ser arrastrado con el ratón o los dedos (en dispositivos táctiles). El control del arrastre de todos modos debe de ser controlado por JavaScript.


## Otros atributos

Atributo	Significado
<b>hidden</b>	No soportado por Internet Explorer. Cuando aparece este atributo (sin valor) hace que el elemento que lo contiene se le marque como no relevante y quede oculto.
<b>contenteditable</b>	Si vale <b>true</b> , el usuario puede modificar el contenido del elemento. Es de HTML 5
<b>spellcheck</b>	Si vale <b>true</b> el párrafo queda marcado como revisable por herramientas de corrección de texto. Es de HTML 5

## Codificación de caracteres en HTML

### Uso de la etiqueta meta charset

Los documentos de texto se pueden codificar de diferentes formas.



Hasta hace unos años, para escribir en español se aconsejaba usar la codificación ISO-8859-1 (todavía hay millones de páginas que usan dicha codificación), pero ahora lo recomendable es usar Unicode, concretamente en su forma UTF-8.

Para indicar que nuestra página utiliza Unicode UTF-8 basta con indicar en la cabecera, la etiqueta:

**<meta charset="UTF-8">**

En páginas escritas en versiones anteriores a HTML 5 se usaba esta versión, más compleja:

**<meta http-equiv="content-type" content="text/html; charset=UTF-8">**

### **Uso de entidades**

Las entidades son códigos especiales que comienzan con el símbolo & tras el cual se indica el nombre de la entidad (un nombre nemotécnico) y final se delimita con un punto y coma (;).

Por ejemplo

&ntilde; es la entidad que representa a la ñe.

Es lo mismo escribir ñ que escribir &ntilde;;

Evidentemente es muy pesado escribir todos los símbolos nacionales usando entidades (aunque hay software que es capaz de hacer la traducción automáticamente a medida que escribimos el texto), por ello y porque ahora ya está muy implantado Unicode para codificar el texto, ya no se usan las entidades con ese propósito.

### **Funcionamiento de los espacios**

Solo se considera el primer espacio en blanco, el resto se eliminan.

## Etiqueta pre

Si deseamos que los espacios que colocamos en el código sí sean tenidos en cuenta, podemos utilizar la etiqueta `pre`.

Realmente, con carácter general, no se aconseja el uso de **pre** porque ayuda a adquirir malos hábitos y dificulta el aprendizaje de las formas avanzadas de maquetar páginas web.

## Uso de `&nbsp;`

Otra forma de obligar a tener en cuenta espacios en blanco es el código `&nbsp;`. Dicho código es una entidad. Si, por ejemplo, escribimos cuatro veces seguidas `&nbsp;` estaremos dejando cuatro espacios en blanco que el navegador sí tendrá en cuenta.

## Comentarios

Sirve para insertar una o varias líneas con comentarios que no serán interpretados por el navegador. Importante para la documentación.

```
<!--    mi comentario  -->
```

## Contenido y contenedores

En HTML los elementos que muestran algo en la página, se colocan dentro de la sección **body**.

Además hay que tener en cuenta que hay elementos pensados para contener grandes cantidades de información (como **div**) y elementos que contienen poco texto (como **strong**). Es correcto que aparezca un elemento **strong** dentro de un elemento **p**, pero no al revés:

```
<strong><p>Hola</p></strong> -->¡¡Incorrecto!!
```



`<p><strong>Hola</strong></p> -->Correcto`

La mayoría de elementos en HTML son contenedores de otros textos y elementos. Hay que conocer que tipos de elementos tenemos. Los tipos están relacionados con la propiedad **CSS display**, que se encarga de decir cómo tiene que mostrarse un elemento. Por defecto los elementos funcionan de esta forma:

### **Elementos de línea. (Elementos de tipo inline).**

Son elementos que se muestran seguidos en la misma línea. Los elementos de este tipo son los que marcan texto simple. Todas las etiquetas de marcado de texto tienen este display: **strong, em, span, mark, abbr, etc.** No comienzan en línea nueva. Estos elementos solo pueden contener datos y otros elementos de línea.

Ejemplo : `<strong>Hola</strong> <em>Hola</em>`

### **Elementos de bloque. Elementos de tipo block.**

Son contenedores más grandes. Los elementos que **marcan párrafos enteros** (como p, h1, h2 o blockquote) y **los contenedores de secciones** (como div, section, article o figure) tienen este display.

Su característica es que, por defecto, tienen forma rectangular y abarcan toda la anchura de su contenedor. Si indicamos para ellos un color de fondo, este se muestra en forma de rectángulo (en los **inline**, el color de fondo abarca exactamente al texto, como si lo hubiéramos pintado con un rotulador). Crean estructuras más grandes que pueden contener a los elementos de línea y a otros elementos de bloque. Además siempre comienzan en una línea nueva.

### **Elementos de tipo inline-block.**

Son rectangulares como los block, pero no abarcan todo su contenedor. Se adaptan a su contenido. Ejemplos son los elementos button o image:

Ejemplo : `<strong>Hola</strong> <button>Hola</button>`

El resultado muestra un botón (elemento rectangular) seguido del texto de tipo strong. Si el botón tuviera un display de tipo block, saltaría a la línea siguiente.

En todo caso, **el display de un elemento se puede modificar desde lenguaje CSS**, pero no conviene hacer combinaciones antinaturales ya que se pierde el sentido semántico del código.

## Elementos HTML

### Elementos de la estructura básica

Elemento	Descripción
<b>html</b>	Documento HTML.
<b>head</b>	Cabecera del documento.
<b>body</b>	Cuerpo del documento. Permite definir formatos que se aplican a los elementos de la página de manera global, como son el color del fondo del texto, los márgenes, el color de los enlaces, ...

## Elementos de la sección cabecera

Elemento	Descripción
<b>Title</b>	Título del documento.
<b>Script</b>	Script incrustado. Su contenido ha de ir situado entre las marcas de comentarios ya que no ha de ser interpretado.
<b>Style</b>	Estilo aplicado al documento utilizando CSS. Su contenido ha de ir situado entre las marcas de comentarios ya que no ha de ser interpretado.

Elemento	Descripción
<b>Base</b>	URI base del documento
<b>Isindex</b>	Prompt de entrada de datos.
<b>Link</b>	Enlaces a documentos externos de librerías
<b>Meta</b>	Información que agiliza la búsqueda del documento en buscadores.

## Elementos que dan formato al texto de un párrafo

Elemento	Descripción
<b>p</b>	Delimita los párrafos

<b>h1 – h2 – h3 – h4 –h5 –h6</b>	<p>Encabezado de nivel i, donde i es un número entero entre 1 y 6, ambos inclusive. El tamaño de la letra del encabezado es mayor cuanto menor sea el valor de i: H1 es la mayor y H6 es la menor.</p> <p>No deben de usarse estas etiquetas para formatear texto. Sólo estarán bien usadas para designar títulos de párrafos.</p>
<b>b</b>	Indica que el texto que está en ese elemento se le pondrá en negrita.
<b>i</b>	Indica que el texto que está en ese elemento se le pondrá en itálica ó cursiva.
<b>u</b>	Indica que el texto que está en ese elemento se le pondrá subrayado.
<b>sup</b>	Indica que el texto que está en ese elemento es un supraíndice.
<b>sub</b>	Indica que el texto que está en ese elemento es un subíndice.
<b>strong</b>	<p>Indica que el texto que está en ese elemento es importante.</p> <p>Habitualmente los navegadores resaltan el texto poniéndolo en negrita aunque podría haber algún navegador que resaltase el texto poniéndolo en cursiva y en naranja.</p>
<b>em</b>	Indica que el texto que está en ese elemento está enfatizado.
<b>mark</b>	Indica que el texto que está en ese elemento está resaltado.
<b>br</b>	Salto de línea. No lleva etiqueta de fin

## Elementos de listas

Las listas permiten crear párrafos agrupados y alineados mediante símbolos como viñetas o números para facilitar la lectura y organización de las ideas del documento.

Elemento	Descripción
<b>ul</b>	Delimita los elementos que forman una lista desordenada
<b>ol</b>	Delimita los elementos que forman una lista ordenada
<b>li</b>	Indica cada uno de los elementos de una lista
<b>dl</b>	Delimita los elementos que forman una lista de definición
<b>dt</b>	Cada uno de los términos que se definen de una lista de definición.
<b>dd</b>	Cada una de las definiciones de una lista de definición.

### Lista con viñetas

Las listas con viñetas se deben englobar dentro de un elemento `ul` (acrónimo de `unordered list`, lista no ordenada), después cada párrafo de la lista estará dentro de elementos de tipo `li` (de `list item`, elemento de lista).

Ejemplo:

```
<p>Lista de la compra</p>
<ul>
  <li>Agua</li>
  <li>Vino</li>
  <li>Cerveza</li>
</ul>
```



## Numéricas

Las listas numéricas aparecen dentro del elemento ol (de ordered list, lista ordenada), después cada párrafo de la lista estará dentro de elementos de tipo li, al igual que las anteriores. La diferencia ahora es que cada párrafo con li, aparece con un número y no con una viñeta.

Ejemplo:

```
<p>Lista de la compra</p>
<ol>
  <li>Agua</li>
  <li>Vino</li>
  <li>Cerveza</li>
</ol>
```

## Listas anidadas

Es posible meter una lista dentro de otra, por ejemplo:

```
<p>Lista de la compra</p>
<ul>
  <li>
    No alcohólicas
    <ul>
      <li>Agua</li>
    </ul>
  </li>
  <li>
    Alcohólicas
    <ul>
      <li>Vino</li>
      <li>Cerveza</li>
    </ul>
  </li>
</ul>
```

## Listas de términos

Permite crear una lista de definiciones de términos. En ellas se indica el término a definir y su significado. Ejemplo:

```
<dl>
  <dt>Windows</dt>
  <dd>
    Sistema operativo de <strong>Microsoft</strong> disponible para PC
    disponible en versiones de 32 y 64 bits y para servidores,
    ordenadores e incluso tabletas y móviles.<br>
    La última versión es la 8 y la 2012 para servidores.
  </dd>
  <dt>Linux</dt>
  <dd>
    Sistema operativo de código abierto disponible
    en numerosas distribuciones gratuitas y de pago.
    Es la base del sistema <strong>Android</strong>.
  </dd>
  <dt>Mac OS</dt>
  <dd>
    Sistema operativo de los ordenadores de la empresa <strong>Apple</strong>
    <br>
    La última versión es la <strong>Mountain Lion</strong>
  </dd>
</dl>
```


## Elementos de tabla

Hoy en día se recomienda no usar las tablas de esta forma ya que las mejoras que proporciona CSS en el uso de capas, permiten olvidar a las tablas como recurso de maquetación.

El uso actual de las tablas es más semántico: indicar una disposición en celdas las cuales se marcan también de forma semántica (celdas de cabecera, celdas normales, filas, columnas, pie de tabla, título de tabla, etc.).

## Creación de tablas simples

En HTML cada tabla está asociado a un elemento **table**, dentro de este elemento se indican las filas mediante el elemento **tr** y dentro de cada fila se indican las celdas mediante elementos **td**.



En HTML 5 las etiquetas **table** y **tr** no tienen atributos, aunque hay atributos, como por ejemplo el atributo **border**, que reconocen los navegadores por estar presentes en versiones anteriores de HTML.

### Elementos de marcado semántico en las tablas

celdas de cabecera <th> /<th>

títulos - A las tablas se les puede poner un título con ayuda de la etiqueta **caption**

### Agrupación de filas

Hay tres elementos HTML que sirven para diferenciar las tres partes principales de una tabla, son:

- **thead**. Sirve para indicar las filas que forman la cabecera de la tabla
- **tfoot**. Indica el pie de la tabla
- **tbody**. Indica el cuerpo de la tabla

De esa forma se podrá más adelante dar formato diferencial a cada parte.

### Agrupación de columnas

Permite realizar agrupaciones de columnas de una tabla con la finalidad de dar un formato homogéneo a cada grupo de columnas indicado. El único atributo permitido en HTML 5 es **span** que sirve para indicar el número de columnas que agrupamos.

Otra posibilidad, más recomendable, es usar dentro de **colgroup** elementos **col** los cuales dan un mayor sentido y son más versátiles.

Esta etiqueta debe de ser la primera de una tabla excepto si se usa título en la tabla (etiqueta **caption**) en cuyo caso será la segunda.

## Elemento col

Se usa dentro de la anterior para dar propiedades a cada grupo de columnas realizado. Permite el uso del atributo col para indicar cuántas columnas se incluyen en el grupo.

## combinar celdas

Es posible unir celdas y de esta forma conseguir tablas de formas caprichosas que permiten una maquetación más poderosa.

Las etiquetas de columna (**td** y **th**) son las que poseen los atributos que permiten esta operación.

En concreto son los atributos:

Atributo	significado
<b>colspan</b>	Combina la celda actual con el número de celdas a la derecha que se indique. Por ejemplo <i>colspan="3"</i> une esta celda con las dos que tiene a su derecha, formando una combinación de tres celdas en horizontal.
<b>rowspan</b> <b>n</b>	Combina la celda actual con el número de celdas hacia abajo que se indique. Por ejemplo <i>rowspan="3"</i> une esta celda con las dos que tiene hacia abajo, formando una combinación de tres celdas en vertical.

Elemento	Descripción
<b>table</b>	Delimita el contenido de una tabla.
<b>tr</b>	Delimita cada una de las líneas de la tabla.
<b>td</b>	Delimita el contenido de cada celda de la tabla.

<b>colgroup</b>	Permite agrupar columnas.
<b>tbody</b>	Permite agrupar líneas de la tabla.
<b>thead</b>	Define la línea cabecera de la tabla.
<b>th</b>	Delimita cada una de las celdas de la cabecera
<b>tfoot</b>	Define la fila pie de la tabla.

## Elementos de formularios

Elemento	Descripción
<b>form</b>	<b>Delimita el contenido del formulario</b>
<b>input</b>	<p><b>Caja de texto para texto corto. Dependiendo del valor que tome el atributo type de este elemento podemos estar ante un texto sin más, un campo de texto donde al escribir no se visualice el contenido si no que escriba asteriscos, un botón de radio que el usuario podrá elegir, una opción que el usuario podrá activar, botones, ...</b></p> <p><a href="https://www.w3schools.com/html/html_form_input_types.asp">https://www.w3schools.com/html/html_form_input_types.asp</a></p> <p><a href="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input">https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input</a></p>

<b>textarea</b>	<b>Caja de texto para texto largo.</b>
<b>select</b>	<b>Crea un menú desplegable que permite elegir de una lista de opciones que contiene el elemento.</b>
<b>option</b>	<b>Delimita cada una de las opciones de un menú desplegable que le contiene.</b>
<b>button</b>	<b>Permite definir un botón. Su principal ventaja frente a los botones hechos con input es que este elemento permite introducir en el botón cualquier otro elemento de HTML, como por ejemplo imágenes.</b>
<b>fieldset</b>	<b>Permite agrupar elementos de un un formulario.</b>
<b>legend</b>	<b>Permite poner un título al fieldset.</b>
<b>label</b>	<b>Etiqueta de un campo del formulario.</b>

## Enlaces

### Enlaces globales

Los enlaces permiten colocar un texto (u otro elemento, como una imagen o un botón) resaltado de forma especial, de modo que cuando se le hace clic, el navegador web nos llevará al destino de la URL (si está disponible).

La etiqueta que permite realizar enlaces es la etiqueta `a`. El atributo `href` permite indicar la URL a la que se realiza el salto.

Ejemplo:

A Augusto le sucedió el emperador `<a href="http://es.wikipedia.org/wiki/Tiberio">Tiberio</a>`

La palabra Tiberio estará remarcada de modo que al hacer clic en ella saltaremos a la URL <http://es.wikipedia.org/wiki/Tiberio>.

### Enlaces locales relativos

La mayor parte de enlaces de un sitio web en Internet no saltan a otros sitios, sino que son saltos a recursos del mismo servicio web. Es decir, se salta muy a menudo a recursos del mismo servidor web.

En ese caso no se indica una URL global tal cual se explicó anteriormente, sino que se indica un salto local. Es decir, un salto a un recurso en nuestro propio servidor web que toma como punto de partida el directorio en el que se encuentra la página web que realiza el salto.

Por ejemplo:

Imaginemos que estamos creando una página web relacionada con la URL <http://www.miservidor.com/noticias/n1.html>

y queremos en esa página hacer un enlace a la URL indicada mediante

<http://www.miservidor.com/noticias/n2.html>



Podríamos hacerlo con el código:

```
<a href="http://www.miservidor.com/noticias/n2.html"> Ver segunda noticia</a>
```

El enlace funcionaría perfectamente. Pero hay que tener en cuenta que la página a la que saltamos está en el mismo directorio.

Si algún día movemos todo nuestro sitio web al dominio nuevodominio.com resulta que tendremos que cambiar todo el código para hacer frente a la nueva situación. Por ello lo lógico es usar rutas locales.

El enlace usando una ruta local sería:

```
<a href="n2.html">Ver segunda noticia</a>
```

Hay que observar que no se ha indicado protocolo ni ruta (tampoco se indicaría puerto), simplemente se empieza indicando la ruta.

### **Enlaces locales absolutos**

Hay otra posibilidad de enlace local.

Por ejemplo:

```
<a href="/tiberio.html">Tiberio</a>
```

Ese enlace (/tiberio.html) buscaría la página tiberio.html en el directorio raíz del servidor ya que la ruta comienza con el símbolo /.

Permiten ahorrar mucho texto y son más fáciles de usar que los enlaces relativos, pero en la práctica no se usan mucho, ya que, si movemos de sitio nuestro sitio web, por ejemplo a un directorio interno del servidor, la raíz ya no sería la misma y el enlace no funcionaría.

### **Enlaces a una posición interna dentro del propio documento**

Hay un tipo de enlace que permite posicionarnos en un punto concreto del documento. Este tipo de enlace se denomina interno, ya que salta dentro del propio documento.



Este salto requiere primero marcar la posición a la que deseamos saltar. Antes de HTML 5 se hacía usando el atributo **name** del propio elemento a. Pero con HTML 5 se prohíbe el uso de este atributo con este fin y, en su lugar, se utiliza el atributo **id**.

**id** es un atributo presente en cualquier elemento de la página web. Sirve para identificar a dicho elemento dentro de la página, por ello no podemos repetir valor para ese atributo en ningún otro elemento.

Supongamos que tenemos este código:

```
<h1 id="tituloPrincipal">Curso de HTML</h1>
```

```
<p>
```

Este es un curso que versa sobre cómo escribir páginas web  
mediante...

...

```
</p>
```

El párrafo de tipo **h1** está identificado con el identificador *tituloPrincipal*. Gracias a ello si en otra parte de la página tenemos el código:

```
<p><a href="#tituloPrincipal">Volver al título principal</a></p>
```

Al hacer clic en ese enlace, se usa el identificador como un marcador de posición y así nos colocaremos en dicho párrafo.

### Enlaces a una posición interna dentro de otro documento

También podemos realizar un salto a una posición interna a otro documento. Por ejemplo si este enlace:

```
<a href="manual.html">Abrir manual</a>
```

Nos permite abrir una página web llamada manual.html (que estará en el mismo directorio que la página actual). Siempre que se abre un documento se muestra el inicio del mismo. Sin embargo este otro enlace:

```
<a href="manual.html#comentarios"> Abrir manual por la zona de comentarios</a>
```

Abre el mismo documento, pero se intenta colocar en la zona marcada por el identificador comentarios. Lógicamente para que esto sea posible, debe existir un elemento marcado con ese identificador en el documento.

Es posible indicar una posición interna en una URL absoluta. Por ejemplo este enlace permite abrir la página dedicada a la ciudad de Palencia de la Wikipedia y colocarse en la zona dedicada a la historia:

```
<a href="http://es.wikipedia.org/wiki/Palencia#Historia"> Historia de Palencia</a>
```

## Atributos del elemento a

Atributo	significado
<b>href</b>	Sirve para especificar la URI destino del hipervínculo
<b>accesskey</b>	Asigna una tecla única de acceso al hipervínculo
<b>Shape</b>	Establece la forma de la imagen que contiene el hipervínculo
<b>coords</b>	Establece la posición en pantalla de la imagen
<b>target</b>	Establece el marco de destino( por defecto se toma la ventana actual) valores posibles: _blank, _self, _parent, _top
<b>tabindex</b>	Establece un número de orden
<b>download</b>	permite indicar que al hacer clic en el enlace, no se abrirá en el navegador, sino que se descargará en nuestro directorio predeterminado de descargas.  <a href="documentos/manual.pdf" download> Descargar manual</a>

<b>type</b>	<p>Permite indicar el tipo de contenido (según la normativa oficial de tipos de la IANA, véase:</p> <p><a href="https://www.iana.org/assignments/media-types/media-types.xhtml">https://www.iana.org/assignments/media-types/media-types.xhtml</a></p> <p><code>&lt;a href="obtenerimagen.php?id=98" type="image/png"&gt;Abrir imagen nº 98&lt;/a&gt;</code></p> <p><b>media</b> permite indicar el medio idóneo para mostrar el contenido del enlace.</p> <p>Ejemplo:</p> <p><code>&lt;a href="documentos/manualImpr.pdf" media="print and(resolution:300dpi)"&gt;Descargar manual&lt;/a&gt;</code></p>
-------------	--

## Elementos estructurales

### Elemento div

Se trata de un elemento clásico (presente en las primeras versiones de HTML) que permite agrupar contenido. Inicialmente se usaba para usar su atributo align para que todo un bloque de párrafos tuviera la misma alineación.

Desde hace años se ha convertido en el elemento utilizado, combinado con CSS, para crear capas. Las capas son elementos contenidos en otro elemento, el cual podemos colocar libremente en cualquier parte de la página. Dentro de div se pueden colocar todo tipo de etiquetas (tablas, párrafos, imágenes,...)

Por defecto, los navegadores no dan formato al contenido de un elemento div, con lo que visualmente no se diferencia del resto de elementos.

## Elemento span

Es muy similar al anterior, pero como los navegadores no agregan nada (ni siquiera espacios) a esta etiqueta cuando se muestra por pantalla, en lugar de para definir capas (aunque se podría) se usa para marcar contenido dentro de un párrafo, a fin de que a ese contenido se le pueda dar un formato especial mediante CSS. Es decir, con `div` marcamos capas (contenedores grandes) y con `span` marcamos bloques pequeños de texto.

## Elementos estructurales en HTML5

La idea en los nuevos estándares de HTML (especialmente en HTML5) es que los elementos HTML aporten valor semántico al contenido. Es decir, se trata de que el código HTML indique qué tipo de contenido tenemos en las páginas. Eso vale para casi todos los elementos ya conocidos como **p** (párrafo), **abbr** (abreviatura), **strong** (remarcado fuerte), etc.

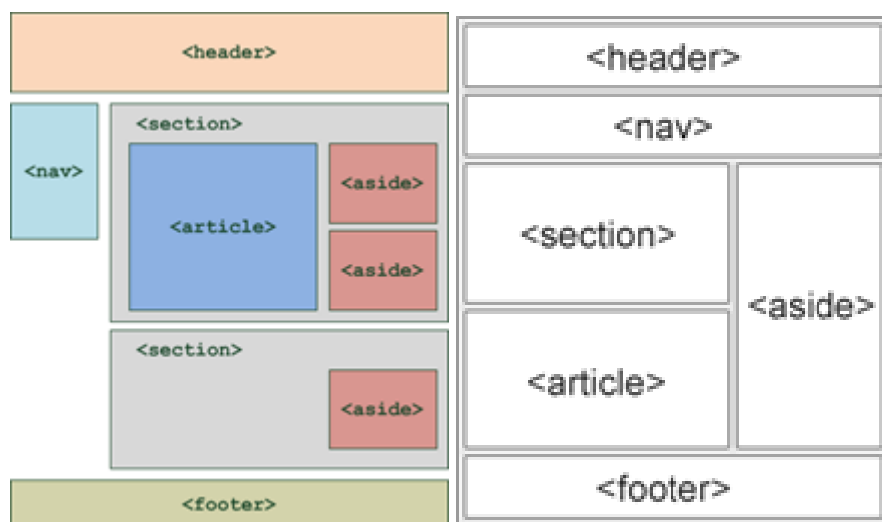
La idea es marcar la semántica con HTML y el formato con las hojas de estilo. A este respecto hay una serie de elementos incluidos en HTML 5 que no dan ningún formato al texto, pero permiten darle un significado. Posteriormente, podemos hacer que ese contenido tenga un formato diferente mediante CSS.

Como el elemento clásico para crear capas es `div`, nos encontramos con el problema de que este elemento no indica un valor semántico, sino funcional. La idea actual es que estos elementos sustituyan el sobreuso que se hace de la etiqueta `div`.

No es fácil esta idea, puesto que existen numerosas plantillas y patrones de trabajo profesionales (como por ejemplo **bootstrap**) exigen el uso de `div`.

Elemento	Descripción
<b>article</b>	<p>Secciones.</p> <p>El contenido dentro de &lt;article&gt; tiene un valor semántico alto, y debe aportar información relevante de nuestro documento. Se usa para representar un contenido específico en nuestra web.</p> <p>Puede representar un artículo, un comentario, entrada de blog, etc.</p> <p>Suele ir con &lt;h1&gt;...&lt;h6&gt; y se puede añadir &lt;footer&gt;, con el autor, fecha publicación, etc.</p>
<b>section</b>	<p>Secciones.</p> <p>Se usa para representar un bloque con valor semántico.</p> <p>Es menos importante que &lt;article&gt; pero más importante que &lt;div&gt;</p>
<b>footer</b>	<p>Texto – sección.</p> <p>Se usa para definir el pie de un texto o de una sección.</p>
<b>header</b>	<p>Texto – sección.</p> <p>Se usa para definir el encabezado de un texto o de una sección.</p>
<b>figure</b>	<p>Figuras.</p> <p>Una imagen, una ilustración, un diagrama, un fragmento de código, o un esquema al que se hace referencia en el texto principal, pero que se puede mover a otra página o a un apéndice sin que afecte al flujo principal.</p>
<b>figcaption</b>	<p>Figuras.</p> <p>Se usa para enmarcar un titular dentro de &lt;figure&gt;</p>

<b>hgroup</b>	<p>Permite agrupar varios títulos ( elementos de h1 a h6) dentro de esta etiqueta para darles estilo común.</p> <p>Se utiliza dentro de <b>article</b> para marcar la zona de títulos del artículo.</p>
<b>aside</b>	<p>Permite marcar texto dentro de un artículo para que no se tenga en cuenta como parte del texto del artículo, sino como un texto aparte que permite realizar aclaraciones del artículo, referencias, resúmenes remarcados y sobre todo cuadros de texto de estilo periodístico para destacar partes del artículo.</p>
<b>main</b>	<p>Permite indicar una sección (que podrá incluir artículos, cabeceras, títulos, párrafos, secciones de imagen,..) de forma que se entenderá que esta es la sección principal del documento.</p>



## Elementos para indicar secciones de tipo mostrar/ocultar

Se trata de secciones que están dentro de un elemento llamado **details**. Dentro de ese elemento se puede poner una sección de tipo **summary** la cual muestra un texto de resumen de la sección. Al hacer clic en el apartado **summary**, se muestra todo el contenido de la sección.

Es decir en definitiva consigue crear en un documento HTML un apartado de tipo resumen/detalle.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>detalle y resumen</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h3>detalle y resumen</h3>
    <details>
      <summary>Cliqueame para mostrar o esconder</summary>
      <p>Aquí va el contenido que se muestra o esconde dependiendo de nuestro
      clic. Recuerda, aquí no se está usando absolutamente nada de
      Javascript!</p>
    </details>
  </body>
</html>
```

## Elementos de cuadro de diálogo

Se suelen llamar cuadros de diálogo a las ventanas que los sistemas muestran a los usuarios para darles mensajes, o para permitirles configurar una determinada respuesta u operación.

En HTML 5 se ha incorporado una nueva etiqueta llamada **dialog**, precisamente con esta finalidad.

Normalmente requiera que para que tenga funcionalidad utilizar javascript.

## Objetos multimedia e imágenes

Elemento	Descripción
----------	-------------



<b>iframe</b>	<p>Inserta un marco en línea dentro de un documento, es equivalente a insertar una página dentro de otra con el tamaño establecido.</p> <p>Se suele utilizar par publicidad o sitios de colaboración.</p>
<b>object</b>	<p>Inserta un objeto en el documento. El tipo de objeto viene determinado por atributos y puede ser una imagen, un video, una animación flash, otro objeto HTML</p>
<b>param</b>	<p>Sirve para inicializar variables de objetos, que serán utilizadas en tiempo de ejecución</p>
<b>img</b>	<p>Sirve para insertar la imagen determinada por el atributo src. Se trata de un elemento en línea. Sin etiqueta de cierre.</p>
<b>map</b>	<p>Sirve para crear un mapa de imágenes en el lado del cliente.</p>
<b>area</b>	<p>Sirve para especificar una región geométrica del mapa y el vínculo asociado a esa región.</p>
<b>audio</b>	<p>Contenido de tipo sonido</p>
<b>video</b>	<p>Para definir un video o película.</p>
<b>embed</b>	<p><b>Multimedia.</b></p> <p>Se utiliza para insertar elementos multimedia de audio y video.</p>
<b>canvas</b>	<p><b>Gráficos.</b></p> <p>Permite generar gráficos dinámicamente a través de scripting (por ejemplo, con JavaScript).</p> <p>Es útil para dibujar gráficos, hacer composiciones de fotos, procesar video en tiempo real.</p>
<b>source</b>	<p><b>Multimedia</b></p> <p>Para introducir varias fuentes (&lt;audio&gt; y &lt;video&gt;).</p>

## Otros elementos de HTML

Elemento	Descripción
<b>cite</b>	<b>Texto. Citas.</b> Inserta una cita o referencia. Se suele usar para hacer referencia a los autores de alguna cita.
<b>output</b>	<b>Aplicaciones externas.</b> Mostrar el resultado de un cálculo.
<b>progress</b>	<b>Aplicaciones externas.</b> Se usa para representar el progreso de una tarea, suele ir con JavaScript.
<b>time</b>	<b>Definir fecha y hora</b>