

# LLMM\_HTML5

## 1-INTRODUCCIÓN, PÁRRAFOS Y ENLACES

---

### 1.1 ESTRUCTURA BÁSICA

```
<!DOCTYPE html>

<html lang="es">

  <head>

    <meta charset="UTF-8">

    <title>Mi primer documento HTML</title>

  </head>

  <body>

    <p> ¡Hola mundo! </p>

  </body>

</html>
```

---

**<!DOCTYPE html>**: declaración de tipo de documento. Los navegadores la usan para poder representar adecuadamente la página. Debe ser la primera línea de la página.

**<html>**: un “contenedor” general que alberga el conjunto de la página web. Sirve también para declarar el idioma de la página.

**<head>**: Contenedor para etiquetas que dan información sobre el documento. Usadas por el navegador, no aportan contenido visible, salvo “title”, que es el título de la página. En el ejemplo se indica la codificación usada en la página, el juego de caracteres (letras y símbolos).

**<body>**: El contenido real de la página. Lo que es visible para los usuarios. En el ejemplo contiene un párrafo.

## 1.2 TIPOS DE ELEMENTOS HTML

- De bloque (block): definen la estructura del documento, y siempre que puedan ocupan el 100% del ancho que tengan disponible
- En línea (inline): se usan dentro de los elementos en bloque para dar significado y forma al contenido. Ocupan sólo lo imprescindible, no el 100% del ancho disponible.

Por norma, no se pueden usar elementos en bloque dentro de elementos en línea. Si lo hacemos el navegador lo mostrará, pero no es correcto.

Ejemplo, no se debe poner un “p” (block) dentro de “strong” (inline).

---

Ejemplos de elementos en bloque:

- Párrafos y bloques de elementos: <p> y <div>
- Encabezados: <h1> ... <h6>
- Formularios: <form>
- Listas y sus elementos: <ol> <ul> <li>

Ejemplos de elementos en línea:

- Imágenes (<img>)
- Enlaces (<a>)
- Marcadores para énfasis / importancia: <strong>, <em>
- Campos de formulario: <input>, <button>, <select>

## 1.3 ATRIBUTOS HTML

Hay una serie de atributos globales, que se pueden usar en todos los tags HTML. Algunos ejemplos: id, hidden, lang, style, tabindex. Son muchos para enumerarlos todos.

Hay atributos específicos de ciertos elementos. Por ejemplo, el atributo href es exclusivo de ciertos elementos, como enlaces (<a>), enlaces a hojas de estilo (<link>) y otros.

Hay cuatro atributos de especial relevancia:

- Id: permite identificar un elemento de forma única en el documento HTML. Esto implica que no puede (o no debería) haber dos elementos HTML con el mismo ID.
- Class. Sirve para especificar una clase de elemento. Puede haber múltiples elementos con la misma clase.
- Title. Sirve para añadir un texto de ayuda a cualquier elemento
- Style. Sirve para definir estilos CSS directamente en el elemento.

Los atributos id, class y style se usan en CSS para definir estilos, y en JavaScript pueden usarse para añadir comportamiento.

## 1.4 ENLACES

Un enlace es una conexión de una página con otra. La sintaxis de un enlace es:

```
<a href="url">Texto del enlace</a>
```

El atributo “href” indica el destino del enlace, dónde debe “saltar” el navegador.

Puede ser de distinto tipo, como: Otra página, Un punto dentro de la misma página, Una imagen o un vídeo, Una dirección de correo, etc....

Por defecto, los navegadores muestran los enlaces en distintos colores en función de si han sido visitados o no. Podemos abrir el enlace en otras ventanas con el atributo “target”. La sintaxis de un enlace con este atributo es:

```
<a href="url" target="...">Texto del enlace</a>
```

Posibles valores del atributo “target”:

- `_blank`: nueva ventana o pestaña.
- `_parent`: en la ventana padre del documento.
- `_self`: en la ventana del documento. Valor por defecto.
- `_top`: en la ventana general del navegador.

### 1.4.1 Marcadores y enlaces a marcadores

Un marcador es un punto dentro de un documento al que deseamos “saltar” directamente.

Se definen con el tag “a”, pero en lugar de usar el atributo “href” se usa el atributo “name”.

El enlace a un marcador usa el nombre del marcador en el atributo href, precedido de la almohadilla (#).

Aplicación: crear índices internos o índices que navegan directamente a un punto de otra página.

```
<h1>Ejemplo de enlaces internos</h1>
<h2>Índice</h2>
<ul>
  <li><a href="#s1">Sección 1</a></li>
  <li><a href="#s2">Sección 2</a></li>
  <li><a href="#s3">Sección 3</a></li>
  <li><a href="#s4">Sección 4</a></li>
</ul>

<h2><a name="s1">Sección 1</a></h2>
<p>Contenido de la sección 1</p>
<p>Lorem ipsum dolor sit amet consectetur adipi
<h2><a name="s2">Sección 2</a></h2>
<p>Contenido de la sección 2</p>
<p>Lorem ipsum dolor sit amet consectetur adipi
<h2><a name="s3">Sección 3</h2>
<p>Contenido de la sección 3</p>
<p>Lorem ipsum dolor sit amet consectetur adipi
<h2><a name="s4">Sección 4</h2>
<p>Contenido de la sección 4</p>
<p>Lorem ipsum dolor sit amet consectetur adipi
```

#### 1.4.1.1 Ejemplos de enlaces

Enlace a una página “coches.html”

```
<a href="coches.html">Ir a la página de coches</a>
```

Enlace a marcador interno con nombre “detalles” en la propia página:

```
<a href="#detalles">Detalles de esta sección</a>
```

Enlace a un vídeo con formato mp4, que abre en ventana nueva:

```
<a href="video.mp4" target="_blank">
```

Detalles de esta sección

```
</a>
```

Enlace a otra página, pero a un marcador dentro de esa página:

```
<a href="otrapagina.html#motos">
```

Detalles de esta sección</a>

## 2-DESTACADOS, CITAS, OTROS

---

### 2.1 <STRONG> Y <EM>

El <strong> se considera más “potente”, con más “importancia” que <em>.

- strong y em si queremos destacar o enfatizar contenido.
- CSS si simplemente queremos cambiar el aspecto gráfico del contenido.

### 2.2 <MARK>

Permite remarcar un texto. Se usa cuando se quiere marcar algo a lo que se hace referencia.

Ejemplo: resaltar en un texto el resultado de una búsqueda.

**Has buscado "color"**

`<h2>Has buscado "color"</h2>`

El coche tenía color rojo

`<p>El coche tenía <mark>color</mark> rojo</p>`

Vamos a pintar de color blanco las paredes.

`<p>Vamos a pintar de <mark>color</mark> blanco las paredes</p>`

### 2.3 <SMALL>

Identifica textos legales, derechos de autor, descarga de responsabilidad, etc. Se puede usar también para definir comentarios al margen.

Ejemplos:

El propietario del sitio no se hace responsable del uso del mismo.

`<p><small>El propietario del sitio no se hace responsable del uso del mismo</small></p>`

Precio por noche: 135 € desayuno no incluido

`<p>Precio por noche: 135€; <small>desayuno no incluido</small></p>`

### 2.4 <SUB> Y <SUP>

Identifica subíndices y superíndices tal y como se usan en fórmulas matemáticas y químicas.

No debe usarse para posicionar texto por estética.

Ejemplo:

`<p>`

La fórmula del agua es H<sub>2</sub>O,  
y la del área del círculo es  $A = \pi R^2$

`</p>`

La fórmula del agua es H<sub>2</sub>O, y la del área del círculo es  $\pi R^2$

### 2.4 <DEL>

Indica una porción de texto que se ha eliminado del documento. Puede incluir atributos específicos:

- cite: indica la URL donde consultar el motivo del borrado.
- datetime: indica la fecha en la que se eliminó el contenido

Ejemplo:

`<p>Esto es <del datetime="2010-10-25">texto eliminado</del></p>`

Esto es ~~texto eliminado~~.

## 2.5 <DEL>

Abreviatura de un texto mayor. El atributo title se usa para especificar el texto completo, y se muestra cuando se posiciona el cursor sobre la abreviatura. Sustituye al obsoleto <acronym>.

Ejemplo:

```
<p>El <abbr title="Hyper Text Markup Language">HTML</abbr>  
es el lenguaje de marcado de la World Wide Web</p>
```

El HTML es el lenguaje de marcado de la World Wide Web.

## 2.6 CITAS- <BLOCKQUOTE>

Cita larga que se representa como elemento de tipo bloque.

En versiones anteriores de HTML, el contenido de <blockquote> debía escribirse dentro de párrafos <p>. HTML5 elimina este requisito, pero se recomienda hacerlo.

Ejemplo:

```
<blockquote>
```

```
  cite="Don Quijote de la Mancha, Miguel de cervantes">
```

```
    <p>Primera cita: En un lugar de la mancha, de cuyo  
      nombre no quiero acordarme...</p></blockquote>
```

```
<blockquote><p>Segunda cita: En un lugar de la mancha,  
de cuyo nombre no quiero acordarme... <cite>Don Quijote de  
La Mancha</cite></p></blockquote>
```

Primera cita: En un lugar de La Mancha, de cuyo nombre no quiero acordarme...

Segunda cita: En un lugar de La Mancha, de cuyo nombre no quiero acordarme...

*Don Quijote de La Mancha*

## 2.6 CITAS- <Q>

Cita corta, en línea dentro de un <p> u otro elemento de bloque. Genera automáticamente comillas de apertura y cierre. Se debe usar para fuentes externas, no para destacar una frase.

El origen puede indicarse con cite.

```
<p>Como Tim Berners-Lee, el padre de la web, escribió en  
  <cite>Tejiendo la web</cite>: <q>Yo llegué en el  
  momento justo, interesado, cuando el hipertexto e  
  Internet habían visto ya la luz</q></p>
```

Como Tim Berners-Lee, el padre de la web, escribió en *Tejiendo la Web*:  
«Yo llegué en el momento justo, interesado, cuando el hipertexto e  
Internet habían visto ya la luz»

## 2.7 DIRECCIÓN DE CONTACTO -<ADDRESS>

Representa la dirección (física o digital) de contacto para una persona, grupo de personas u organización.

El uso ideal es para mostrar la información de contacto relacionada con la página, como el autor del artículo / documento.

```
<address>
```

```
  IES Clara del Rey  
  C/ del padre Claret, 8  
  28002 Madrid, España
```

```
</address>
```

*IES Clara del Rey  
C/ del Padre Claret, 8  
28002 Madrid, España*

## 3-IMÁGENES

---

### 3.1 IMÁGENES – <IMG>

Permite insertar imágenes en el documento. Es una etiqueta “vacía”. No tiene tag de cierre </img>. En HTML5 se puede escribir como <img> o como <img/>. Esta última forma de escribirla es una herencia de XHTML (XML). Tiene dos atributos obligatorios: alt y src

-Atributo alt:

- Descripción de la imagen.
- Su principal uso es dar un contexto para las personas ciegas que usan lectores de pantallas.
- Si la imagen es sólo decorativa (no tiene funcionalidad ni aporta contenido) el atributo alt debe ser vacío (alt="")

-Atributo src:

- Es la URL (dirección) de la imagen que queremos mostrar.
- Puede ser de varios tipos:
  - Dirección relativa a la página.
  - Dirección relativa al sitio web. Empieza con “/”.
  - Absoluta: comienza por http:// o https://
  - Embebida con codificación Base64.
- Se considera buena práctica establecer el tamaño de las imágenes en CSS para evitar distorsiones en la carga de la página, sobre todo en dispositivos móviles.

-Ejemplos de atributo src con dirección relativa:

- `src="girasoles.png"` – En la misma carpeta que la página html.
- `src="img/girasoles.png"` – En una carpeta “img” que está dentro de la carpeta de la página html.
- `src="../img/girasoles.png"` – En una carpeta “img” que está en el sitio web a la misma “altura” que la carpeta de la página html. Por así decirlo la carpeta img es “hermana” de la carpeta de la página.

-Ejemplo de atributo src con dirección relativa al sitio web:

- `src="/imagenes/girasoles.png"` – En la carpeta imágenes en la raíz del sitio web.

-Ejemplos atributo src con dirección absolutas:

- `src="https://i.blogs.es/718a10/img_2085/1200_800.jpeg"` – Imagen de un teléfono iPhone en i.blogs.es.
- `src="https://media.game.es/COVERV2/3D_L/220/220388.png"` – Imagen de la carátula de un videojuego en Game.es

-Ejemplo de atributo src con imagen embebida en Base64.

- `src="data:image/gif;base64, código de base 64"`

```
<!-- Imagen que está en la misma carpeta que nuestra página -->


<!-- Imagen que está en una subcarpeta "imagenes/fotos" -->


<!-- Imagen una carpeta "img/icons" en raíz del servidor -->


<!-- Imagen en un servidor externo -->

```

### 3.2 IMÁGENES – <PICTURE>

Permite establecer varias versiones de imágenes para distintos anchos de pantalla, adaptándose así a móviles, tablets y monitores de sobremesa o portátiles.

Por el máximo de la imagen:

```
<picture>
  <source media="(max-width: 858px)" srcset="imagenes/EvangelionPadre.png">

  <source media="(max-width: 1280px)" srcset="imagenes/SinAcabar(858x480).jpeg">

  <source media="(max-width: 1700px)" srcset="imagenes/One-Piece-11-1280x720.jpg">

  
</picture>
```

Por el mínimo de la imagen:

```
<picture>
  <source media="(min-width: 1700px)" srcset="imagenes/claymore(1920x1080).jpg">

  <source media="(min-width: 1280px)" srcset="imagenes/One-Piece-11-1280x720.jpg">

  <source media="(min-width: 858px)" srcset="imagenes/SinAcabar(858x480).jpeg">

  
</picture>
```

## 4-LISTAS

---

### 4.1 TIPOS DE LISTAS

Las listas HTML se usan para representar listados de información de forma semántica. Hay tres tipos de listas:

- Listas sin ordenar: los elementos están relacionados, pero no hay un orden específico.
- Listas ordenadas: Los elementos deben seguir un orden establecido.
- Listas de definición o de descripción: cada elemento de la lista se acompaña de una descripción de este.

Se pueden anidar listas, incluyendo una lista dentro de un elemento de otra. Y se pueden incluir imágenes en los elementos de la lista.

#### 4.1.1 LISTAS SIN ORDEN-<UL>

<pre>&lt;ul&gt;   &lt;li&gt;Avellanas&lt;/li&gt;   &lt;li&gt;Almendras&lt;/li&gt;   &lt;li&gt;Anacardos&lt;/li&gt;   &lt;li&gt;Pipas de calabaza&lt;/li&gt;   &lt;li&gt;Pipas de girasol&lt;/li&gt; &lt;/ul&gt;</pre>	<ul style="list-style-type: none"><li>• Avellanas</li><li>• Almendras</li><li>• Anacardos</li><li>• Pipas de calabaza</li><li>• Pipas de girasol</li></ul>
---	--

“Cada elemento (item) de la lista se delimita con <li> </li>. Dentro de <ul> ... </ul> sólo puede haber elementos <li>. “

#### 4.1.2 LISTAS CON ORDEN-<OL>

<pre>&lt;ol&gt;   &lt;li&gt;Almendras&lt;/li&gt;   &lt;li&gt;Anacardos&lt;/li&gt;   &lt;li&gt;Avellanas&lt;/li&gt;   &lt;li&gt;Pipas de calabaza&lt;/li&gt;   &lt;li&gt;Pipas de girasol&lt;/li&gt; &lt;/ol&gt;</pre>	<ol style="list-style-type: none"><li>1. Almendras</li><li>2. Anacardos</li><li>3. Avellanas</li><li>4. Pipas de calabaza</li><li>5. Pipas de girasol</li></ol>
---	---

“Igual que <ul>”. Puede comenzar con un número diferente de 1 (atributo “start”), y puede invertirse el orden (de mayor a menor) (atributo “reversed”).

#### 4.1.3 LISTAS DE DEFINICIÓN O DE DESCRIPCIÓN — <DL>

Cada uno de los elementos de la lista ya no son un único elemento <li>, sino dos elementos:

- <dt>: El término a describir (dt => term).
- <dd>: La descripción del término (dd => description).

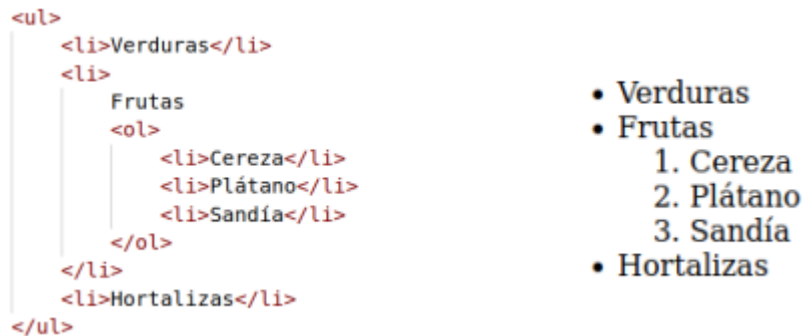
<pre>&lt;dl&gt;   &lt;dt&gt;HTML&lt;/dt&gt;   &lt;dd&gt;Lenguaje usado en la Web &lt;/dd&gt;   &lt;dt&gt;XML&lt;/dt&gt;   &lt;dd&gt;Lenguaje extensible para       definir otros&lt;/dd&gt;   &lt;dt&gt;SVG&lt;/dt&gt;   &lt;dd&gt;Lenguaje para creación de gráficos       vectoriales. Basado en XML.&lt;/dd&gt; &lt;/dl&gt;</pre>	<table border="0"><tr><td>HTML</td><td>Lenguaje usado en la Web</td></tr><tr><td>XML</td><td>Lenguaje extensible para definir otros</td></tr><tr><td>SVG</td><td>Lenguaje para creación de gráficos vectoriales. Basado en XML.</td></tr></table>	HTML	Lenguaje usado en la Web	XML	Lenguaje extensible para definir otros	SVG	Lenguaje para creación de gráficos vectoriales. Basado en XML.
HTML	Lenguaje usado en la Web						
XML	Lenguaje extensible para definir otros						
SVG	Lenguaje para creación de gráficos vectoriales. Basado en XML.						



## 4.2-ANIDAMIENTO DE LISTAS

Una lista puede colocarse dentro de un ítem de lista de otra.

Importante: la lista “interna” debe estar dentro de un ítem de lista, no puede estar directamente dentro de la lista “externa”.



## 4.3-CONSIDERACIONES DE ACCESIBILIDAD EN LISTAS

Las listas son los elementos ideales para expresar elementos relacionados entre sí.

Son especialmente recomendables para los grupos de enlaces, como los menús.

Hay que evitar en la medida de lo posible (complejas de navegar):

- Listas de un único elemento.
- Usarlas porque los navegadores aplican sangría (mejor usar CSS).
- Usar párrafos que comiencen con un guion en lugar de listas
- Abusar de las listas anidadas, con múltiples niveles de anidamiento.

## 5-TABLAS

### 5.1-TABLAS Y ESTRUCTURAS BÁSICAS

Se utilizan para distribuir información en filas y columnas. Se deben usar sólo para representar información que debe tener el formato de tabla.

No se deben usar para:

- Maquetación: distribuir o alinear los contenidos en la página.
- Representar datos que deberían usar un formato de lista, como por ejemplo una lista de definición (<dl>)

```
<table>
  <tr>
    <th>Cabecera
      columna 1</th>
    <th>Cabecera
      columna 2</th>
  </tr>
  <tr>
    <td>Dato</td>
    <td>Dato</td>
  </tr>
  <tr>
    <td>Dato</td>
    <td>Dato</td>
  </tr>
</table>
```

La tabla se divide en filas, que a su vez se dividen en celdas.

**<table>** - **</table>**: Inicio y final de la tabla.

**<tr>** - **</tr>**: Inicio y final de fila de tabla.

**<th>** y **<td>**: Elemento para cada una de las celdas de la tabla.

**<th>** se utiliza para las celdas de encabezado, las cabeceras de la tabla.

**<td>** se utiliza para las celdas de datos.

```
<table>
  <tr>
    <th>Cabecera columna 1</th>
    <th>Cabecera columna 2</th>
  </tr>
  <tr>
    <td>Dato</td>
    <td>Dato</td>
  </tr>
  <tr>
    <td>Dato</td>
    <td>Dato</td>
  </tr>
</table>
```

Por defecto las tablas se dibujan sin bordes. Se deben añadir con CSS.

Hay atributos, como "border", que permiten hacerlo, pero como casi todos los atributos desde la aparición de CSS y HTML5, se desaconseja.

Cabecera columna 1	Cabecera columna 2
Dato	Dato
Dato	Dato

```
<table>
  <caption>Título de la tabla</caption>
  <thead>
    <tr>
      <th>Cabecera 1</th>
      <th>Cabecera 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Celda 1 de la fila 1</td>
      <td>Celda 2 de la fila 1</td>
    </tr>
    <tr>
      <td>Celda 1 de la fila 2</td>
      <td>Celda 2 de la fila 2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Celda 1 de la fila del pie</td>
      <td>Celda 2 de la fila del pie</td>
    </tr>
  </tfoot>
</table>
```

La estructura básica puede ampliarse con más elementos, que ayudan a definir mejor la tabla, o a dar estilos más eficientemente.

**<caption>**: título de la tabla.

**<thead>**: cabecera de la tabla.

**<tbody>**: cuerpo de la tabla.

**<tfoot>**: pie de la tabla.

## 5.2-TABLAS – IDENTIFICAR ÁMBITO DE CABECERAS

Aunque la estructura natural de una tabla es con las cabeceras en la parte superior, hay ocasiones en que las cabeceras están en la parte izquierda.

El atributo “scope” del tag <th> permite indicar si la cabecera aplica:

- A la columna: col
- A la fila: row
- A un grupo de columnas: colgroup
- A un grupo de filas: rowgroup

Esto es importante para que la tabla sea más accesible. Identificar las cabeceras de las tablas ayuda a los lectores de pantalla a leer la información correcta.

Ejemplo tabla scope:

```
<table class="default">
  <tr>
    <td></td>
    <th scope="col">Hoy</th>
    <th scope="col">Mañana</th>
    <th scope="col">Viernes</th>
  </tr>
  <tr>
    <th scope="row">Condición</th>
    <td>Soleado</td>
    <td>Mayormente soleado</td>
    <td>Parcialmente nublado</td>
  </tr>
</table>
```

```
<table>
  <tr>
    <th scope="row">Temperatura</th>
    <td>19°C</td>
    <td>17°C</td>
    <td>12°C</td>
  </tr>
  <tr>
    <th scope="row">Vientos</th>
    <td>E 13 km/h</td>
    <td>E 11 km/h</td>
    <td>S 16 km/h</td>
  </tr>
</table>
```

## 5.3- TABLAS COMBINAR CELDAS

Se pueden combinar celdas horizontal o verticalmente para que aparezcan varias como una sola celda.

-Para combinar celdas horizontalmente se usa el atributo “colspan” (expandirse en columnas), en un elemento <td> o <th>.

-Para combinar celdas verticalmente se usa el atributo “rowspan” (expandirse en filas), en un elemento <td> o <th>.

Estos atributos para combinar celdas no se pueden sustituir por CSS. Son atributos que definen estructura del documento, no apariencia.

Ejemplos tabla combinar celdas:

```
<table border="1">
  <tr>
    <th>Nombre</th>
    <th colspan="2">Apellidos</th>
  </tr>
  <tr>
    <td>José Luis</td>
    <td>López</td>
    <td>Álvarez</td>
  </tr>
  <tr>
    <td>Carlos</td>
    <td>De la Cruz</td>
    <td>Morales</td>
  </tr>
</table>
```

Nombre	Apellidos	
José Luis	López	Álvarez
Carlos	De la Cruz	Morales

```
<table border="1">
  <tr>
    <th>Nombre</th>
    <td>José Luis</td>
    <td>Carlos</td>
  </tr>
  <tr>
    <th rowspan="2">Apellidos</th>
    <td>López</td>
    <td>De la Cruz</td>
  </tr>
  <tr>
    <td>Álvarez</td>
    <td>Morales</td>
  </tr>
</table>
```

Nombre	José Luis	Carlos
Apellidos	López	De la Cruz
	Álvarez	Morales

## 6-ESTRUCTURA DE PÁGINA EN HTML 5

---

### 6.1- ESTRUCTURA DE PÁGINA

CABECERA		
MENÚ	CUERPO	+INFO
PIE		

### 6.2- ESTRUCTURA DE PÁGINA – CON TABLAS

CABECERA <TD COLSPAN="3">		
MENÚ <TD>	CUERPO <TD>	+INFO <TD>
PIE <TD COLSPAN="3">		

Hace ya años que la estructura basada en tablas se considera incorrecta. Tiene problemas de accesibilidad muy graves. Es difícil encontrar hoy en día algo de este estilo, pero no imposible. Puede que en código antiguo.

### 6.3- ESTRUCTURA DE PÁGINA – CON <DIV>

CABECERA <DIV CLASS="CABECERA">		
MENÚ <DIV CLASS="MENU">	CUERPO <DIV CLASS="CUERPO">	+INFO <DIV CLASS="INFO">
PIE <DIV CLASS="PIE">		

<div>: elemento de bloque sin significado semántico. Sirve para agrupar elementos. Se debe dar formato con CSS (de ahí el atributo class). Un gran número de páginas siguen usando este enfoque basado en <div>. No es incorrecto, pero no es el recomendado hoy día.

### 6.4- ESTRUCTURA DE PÁGINA – CON HTML 5

CABECERA <HEADER>		
MENÚ <NAV>	CUERPO <MAIN>	+INFO <ASIDE>
PIE <FOOTER>		

Elementos semánticos (aportan significado) / estructurales. Mejora la accesibilidad de las páginas, y el ranking en buscadores. Es el enfoque recomendado, aunque sigue siendo necesario CSS para posicionar los elementos.

### 6.4.1 Elementos semánticos HTML 5

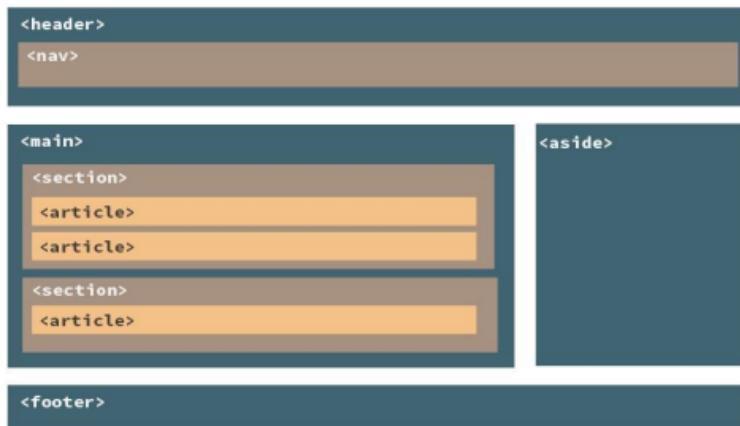
Los utilizados más habitualmente son:

- `<header>`: para cabeceras (no es lo mismo que `h1...h6`).
- `<nav>`: menú de navegación.
- `<main>`: contenido principal de la página o sección.
- `<section>`: para dividir el contenido de la página en secciones.

Normalmente una sección se compone de varios “article”.

- `<article>`: bloques independientes de información. Un article puede ser una noticia, post de blog, servicio, producto.
- `<aside>`: información no relacionada con el resto de la página.
- `<footer>`: pie de página.

Ejemplo de estructura semántica HTML 5:



#### 6.4.1.1 Elementos HTML 5 – `<header>`

Define cabeceras. No confundir cabecera con encabezado (`h1...h6`).

Normalmente hay al menos una cabecera, la principal de la página con logotipo, título, `h1`, etc. Puede haber otras cabeceras dentro de otros elementos, como por ejemplo los artículos (`<article>`). Es habitual que se use en conjunción de un encabezado.

Ojo: sólo puede haber una cabecera dentro de cada elemento. Es decir, una en `<body>`, una en cada uno de los `<article>`, `<aside>`, etc.

No debe colocarse dentro de `<footer>`, o dentro de otro `<header>`.

#### 6.4.1.2 Elementos HTML 5 – `<nav>`

Define una zona con enlaces que sirven para navegar por la web.

No todos los enlaces de la página tienen que ir en nav, sólo cuando haya un grupo de enlaces relacionados. Podemos encontrar `<nav>` como elementos independientes, o dentro de otros elementos como header, footer, section, etc. Es muy habitual, y recomendable para accesibilidad) que dentro contengan una lista desordenada (`ul`) con los enlaces para navegar.

No tiene por qué haber una sola navegación en la página. Por ejemplo:

- Un menú principal con las secciones de la página
- Una navegación para ir a enlaces relacionados con cierto contenido

#### 6.4.1.3 Elementos HTML 5 – <footer>

Define pie de página o de elemento.

Cuando está dentro de <body>, se suele usar para la parte final de la página, que puede contener información como copyright, forma de contacto, enlaces a condiciones legales, etc. Al igual que con <header>, puede haber otros <footer> dentro de otros elementos, como por ejemplo los artículos (<article>)

Ojo: también igual que con <header>, sólo puede haber un <footer> “principal” en la página, o dentro de cada uno de los elementos <article>, <aside>, etc.

Consejo: la información de contacto dentro de <footer>, algo muy habitual, debe incluirse usando el elemento <address>.

#### 6.4.1.4 Elementos HTML 5 – <main>

Identifica el bloque principal de información de una página.

Normalmente están acompañados, como mínimo, de un <header> (antes) y un <footer> (después)

Aunque puede haber varios (el estándar lo admite), la recomendación es que haya un solo elemento <main> en la página, por cuestiones de accesibilidad. En algunas referencias incluso aparece directamente como prohibido que se repita.

El contenido dentro de <main> debe ser exclusivo del documento. No deben incluirse en <main> elementos como menús de navegación generales, pies de página, cabeceras, barras laterales de información, publicidad a nivel de página, etc. Sólo contenido relevante de la página.

#### 6.4.1.5 Elementos HTML 5 – <article>

Define un bloque de información más o menos independiente, como artículos o entradas de un blog.

Independiente significa que, si se leyera de forma independiente, si se extrajera de la página, tendría sentido en sí misma, sin necesidad de otros elementos asociados.

Aunque el tag se traduzca por “artículo”, no necesariamente tienen que ser este tipo de contenido. Puede ser un producto, un servicio, o cualquier otro elemento.

Cuando tenemos varios artículos es posible que se incluyan en una sección de artículos relacionados, y también que un artículo se divida en varias secciones.

#### 6.4.1.6 Elementos HTML 5 – <section>

Permite dividir un elemento en varias secciones, o agrupar elementos relacionados dentro de una sección.

Ejemplo 1: en una página de empresa que tiene múltiples artículos sobre su actividad, agruparlos en tres secciones: servicios, productos y clientes.

Ejemplo 2: dividir un artículo que trata sobre un tema extenso en varias secciones.

Como se ve, podemos encontrar rodeando a otros elementos como , pero también dividiéndolos. No hay una norma que diga si uno puede o no contener al otro.

Hay que decidir, en cada caso, el mejor uso para estructurar la información de forma semántica.

#### 6.4.1.7 Elementos HTML 5 – <aside> y <figure>

##### <aside>

Permite definir un bloque de información que, estando relacionada con la información colindante, es más una información complementaria, no es parte fundamental del contenido. Es muy habitual usarla para las barras laterales que contienen enlaces hacia otro sitio web, o alguna otra clase de información relacionada.

##### <figure>

Permite contener una imagen, para agruparla con su información relacionada: título, descripción, copyright, etc. Para añadir un título a la imagen se usa el elemento <figcaption>

#### 6.4.1.7 Elementos HTML 5 – <details> y <summary>

##### <details>

Sirve para incluir detalles. Este contenido puede mostrarse u ocultarse a demanda, con interacción del usuario. Puede contener cualquier tipo de contenido, pero normalmente se usa con un elemento <summary>

##### <summary>

Define el resumen para <details>. Es lo que el usuario ve por defecto, y al hacer clic en el <summary>, se mostrará todo el contenido de <details>

```
<details>
  <summary>Detalles</summary>
  Contenido que se muestra cuando se
  hace clic en los detalles.
</details>
```

► Detalles

▼ Detalles

Contenido que se muestra cuando se  
hace clic en los detalles.