

## TEMA 3. ENCABEZADO Y ETIQUETAS SEMÁNTICAS.

1. CABECERA DEL DOCUMENTO.....	2
1.1. ELEMENTO HEAD.....	2
1.2. CODIFICACIÓN DEL CONTENIDO.....	2
1.3. LINKS.....	2
CÓMO ENLAZAR EL FAVICON.....	3
2. METADATOS.....	3
3. DOCTYPE.....	4
4. ETIQUETAS SEMÁNTICAS.....	4
4.1. ESTRUCTURA DE UN DOCUMENTO HTML5.....	5
4.2. USO DE LAS NUEVAS ETIQUETAS SEMÁNTICAS.....	6
<HEADER>.....	6
<NAV>.....	6
<MAIN>.....	7
<FOOTER>.....	7
<ARTICLE>.....	8
<SECTION>.....	8
<ASIDE>.....	10
<FIGURE>.....	10
5. ESTRUCTURA DE UN DOCUMENTO HTML CON ETIQUETAS SEMÁNTICAS.....	11

## 1. CABECERA DEL DOCUMENTO

### 1.1. ELEMENTO HEAD

El elemento head contiene los metadatos que aportan información extra sobre la página, como su título, descripción, autor, etc. Además, puede incluir referencias externas a contenidos necesarios para que el documento se muestre y comporte de manera correcta:

```
<head>
  <meta charset=UTF-8" >
  <title>Mi Primera página</title>
  <link rel="stylesheet" href="style.css">
  <link rel="icon" href="img/favicon.ico" >
</head>
```

### 1.2. CODIFICACIÓN DEL CONTENIDO

Cuando se piensa en "texto", probablemente nos venga a la cabeza una definición de "caracteres y símbolos que veo en la pantalla de mi ordenador". Pero realmente se tratan de bits y bytes. Cada cadena de caracteres que se muestra en la pantalla se almacena con una codificación de caracteres en particular. Hay cientos de codificaciones de caracteres diferentes, algunos optimizado para ciertos idiomas como el ruso, el chino o inglés, y otros que se pueden utilizar para múltiples idiomas. En términos generales, la codificación de caracteres proporciona una correspondencia entre lo que se muestra en la pantalla y lo que un equipo realmente almacena en la memoria y en el disco.

En el caso de HTML5, es aún más sencillo definir esta etiqueta *meta*:

```
<meta charset="utf-8">
```

Debemos acostumbrarnos a especificar la codificación de nuestros documentos, aunque no vayamos a utilizar caracteres especiales o nuestro documento no se presente en otros idiomas.

### 1.3. LINKS

Dentro del elemento head, las etiquetas <link> son una manera de acceder o declarar contenido externo al documento actual, que puede cumplir distintos objetivos:

- ✓ Es una hoja de estilo contiene las reglas CSS que su navegador debe aplicar al presente documento.
- ✓ Es un feed que contiene el mismo contenido que esta página, pero en un formato estándar (RSS).

En HTML5 en enlace de los ficheros .css, suprime el tipo de relación, es decir, el tipo de fichero, ya que solo existe un lenguaje para estilos que es CSS:

```
<link rel="stylesheet" href="style.css" />
```

Antes teníamos que indicar el tipo de fichero que estábamos enlazando con el atributo type.

```
<link rel="stylesheet" href="style.css" type="text/css" />
```

Existen un gran número de etiquetas metas que podemos utilizar en nuestro documento, para diferentes funcionalidades, que puedes consultar el siguiente enlace:

<https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/los-meta-tags-mas-importantes-y-su-funcion/>

### Cómo Enlazar el favicon

El *favicon* o icono para favoritos es el pequeño icono que muestran las páginas en varias partes del navegador. Dependiendo del navegador que se utilice, este icono se muestra en la barra de direcciones, en la barra de título del navegador y/o en el menú de favoritos/marcadores.

```
<link rel="shortcut icon" href="imagenes/favicon.ico">
```

Aunque en principio la imagen debería ser de tipo .ICO (formato gráfico de los iconos), algunos navegadores soportan favicons en otros formatos gráficos más habituales (como, por ejemplo .PNG).

## 2. METADATOS

Una de las partes más importantes de la metainformación de la página son los metadatos, que permiten incluir cualquier información relevante sobre la propia página.

La especificación oficial de HTML no define la lista de metadatos que se pueden incluir, por lo que las páginas tienen libertad absoluta para definir los metadatos que consideren adecuados. La etiqueta empleada para la definición de los metadatos es <meta>.

Etiqueta	<meta>
<b>Atributos propios</b>	<ul style="list-style-type: none"><li>• name = "texto" - El nombre de la propiedad que se define</li><li>• content = "texto" - El valor de la propiedad definida (no existe una lista de valores permitidos)</li><li>• http-equiv = "texto" - En ocasiones, reemplaza al atributo "name" y lo emplean los servidores para adaptar sus respuestas al documento</li><li>• scheme = "texto" - Indica el esquema que se debe emplear para interpretar el valor de la propiedad</li></ul>
<b>Descripción</b>	Permite definir el valor de los metadatos que forman la metainformación del documento

Los metadatos habituales utilizan solamente los atributos name y content para definir el nombre y el valor del metadato:

```
<meta name="autor" content="Juan Pérez" />
```

Algunas etiquetas <meta> hacen uso del atributo http-equiv. Este atributo se utiliza para indicar que el valor establecido por este metadato puede ser utilizado por el servidor al entregar la página al navegador del usuario. El siguiente metadato indica al servidor que el contenido de la página es código HTML y su codificación de caracteres es UTF-8:

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
```

El atributo scheme no suele utilizarse, aunque permite proporcionar información de contexto para que el navegador interprete correctamente el valor del metadato. En el siguiente ejemplo, el atributo scheme indica al navegador que el valor del metadato hace referencia al código ISBN:

---

```
<meta scheme="ISBN" name="identificador" content="789-1392349610">
```

Como hemos visto anteriormente no existe una lista oficial con los metadatos que se pueden definir, algunos de ellos se utilizan en tantas páginas que se han convertido prácticamente en un estándar. A continuación, se muestran algunos de los metadatos más utilizados:

Definir el autor del documento:

```
<meta name="author" content="Juan Pérez" />
```

Definir el programa con el que se ha creado el documento:

```
<meta name="generator" content="WordPress 2.8.4" />
```

Definir el conjunto de caracteres del documento:

```
<meta charset=UTF-8" />
```

Definir el copyright del documento:

```
<meta name="copyright" content="librosweb.es" />
```

Definir el comportamiento de los buscadores:

```
<meta name="robots" content="index, follow" />
```

Definir las palabras clave que definen el contenido del documento:

```
<meta name="keywords" content="diseño, css, hojas de estilos, web, html" />
```

Definir una breve descripción del sitio:

```
<meta name="description" content="Artículos sobre diseño web, usabilidad y accesibilidad" />
```

### 3. DOCTYPE

La forma en la que expresamos con HTML 5 el doctype de nuestro documento es:

```
<!DOCTYPE html>
```

Esta forma de expresarlo difiere mucho de como se hacía anteriormente con HTML4 y sus sucesivas versiones, en las que utilizábamos una forma más compleja de expresarlo

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

Cuando nos encontremos con un doctype de este tipo sabremos que se trata de una página bastante antigua y que no se ha actualizado.

El doctype determina qué versión de (X)HTML utiliza el documento, y es utilizada por el navegador para saber cómo de procesar el documento.

### 4. ETIQUETAS SEMÁNTICAS

En 2004, Ian Hickson, el autor de la especificación de HTML5, analizó 1.000.000.000 de páginas web utilizando el motor de Google, intentando identificar la manera en la que la *web real* estaba construida. Este estudio revela que los desarrolladores utilizan *clases* o *ids* comunes para estructurar los documentos. Esto llevó a considerar que quizás fuese una buena idea crear etiquetas concretas para reflejar estas estructuras.

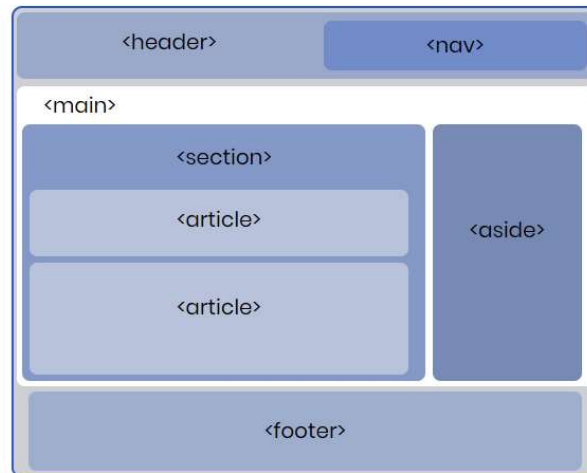
Este tipo de etiquetas que componen la web semántica nos sirven para que cualquier mecanismo automático (un navegador, un motor de búsqueda, un lector de feeds...) que lea

un sitio web sepa con exactitud qué partes de su contenido corresponden a cada una de las partes típicas de un sitio. Observando esas etiquetas semánticas estructurales, cualquier sistema podrá procesar la página y saber cómo está estructurada. Veamos algunas de estas etiquetas que introduce HTML5 en este sentido.

- ✓ `<header></header>`: representa un grupo de artículos introductorios o de navegación. Está destinado a contener por lo general la cabecera de la sección (un elemento h1-h6 o un elemento hgroup), pero no es necesario.
- ✓ `<nav></nav>`: representa una sección de una página que enlaza a otras páginas o a otras partes dentro de la página. No todos los grupos de enlaces en una página necesita estar en un elemento nav, sólo las secciones que constan de bloques de navegación principales son apropiadas para el elemento de navegación.
- ✓ `<main></main>`: representa el contenido principal de la página.
- ✓ `<section></section>`: se utiliza para representar una sección "general" dentro de un documento o aplicación, como un capítulo de un libro. Puede contener subsecciones y si lo acompañamos de h1-h6 podemos estructurar mejor toda la página creando jerarquías del contenido, algo muy favorable para el buen posicionamiento web.
- ✓ `<article></article>`: representa un componente de una página que consiste en una composición autónoma en un documento, página, aplicación, o sitio web con la intención de que pueda ser reutilizado y repetido. Podría utilizarse en los artículos de los foros, una revista o el artículo de periódico, una entrada de un blog, un comentario escrito por un usuario, un widget interactivo o cualquier otro artículo independiente de contenido. Cuando los elementos de `<article>` son anidados, los elementos interiores representan los artículos que en principio son relacionados con el contenido del artículo externo. Por ejemplo, un artículo de un blog que permite comentarios de usuario, dichos comentarios se podrían representar con `<article>`.
- ✓ `<aside></aside>`: representa una sección de la página que abarca un contenido relacionado con el contenido que lo rodea, por lo que se le puede considerar un contenido independiente. Este elemento puede utilizarse para efectos tipográficos, barras laterales, elementos publicitarios, para grupos de elementos de la navegación, u otro contenido que se considere separado del contenido principal de la página.
- ✓ `<footer></footer>`: representa el pie de una sección, con información acerca de la página/sección que poco tiene que ver con el contenido de la página, como el autor, el copyright o el año.

#### 4.1. ESTRUCTURA DE UN DOCUMENTO HTML5

Como hemos visto con las nuevas etiquetas semánticas introducidas en HTML5, aportan un significado concreto al documento que estamos definiendo, y por lo tanto, afectan de manera directa a la forma en la estructuramos el contenido de nuestra página.



## 4.2. USO DE LAS NUEVAS ETIQUETAS SEMÁNTICAS

### `<header>`

Según la especificación, un elemento `<header>` representa lo siguiente:

Un grupo de navegación o contenido introductorio. Un elemento header normalmente contiene una sección de encabezado (un elemento `h1-h6` o un elemento `hgroup`), pero puede contener otro tipo de elementos, como una tabla de contenidos, un formulario de búsqueda o cualquier logo importante.

`<header>`

```
<a href="/"><img src=img/logo.png alt="home"></a>
```

```
<h1>Título de la Página</h1>
```

`</header>`

También es muy común que los sitios web muestren un lema o subtítulo bajo el título principal. Para dar mayor importancia a este subtítulo, y relacionarlo de alguna manera con el título principal de la web, es posible agrupar los dos titulares bajo un elemento `<hgroup>`.

`<header>`

```
<a href="/"><img src=img/logo.png alt="home"></a>
```

```
<hgroup>
```

```
<h1>Título de la Página</h1>
```

```
<h2 class="segundo-encabezado">
```

```
  Esto es un encabezado h2
```

```
</h2>
```

```
</hgroup>
```

`</header>`

### `<nav>`

Según la especificación, un elemento `<nav>` representa lo siguiente:

El elemento `<nav>` representa una sección de una página que enlaza con otras páginas o partes de la misma página: una sección con enlaces de navegación.

El elemento `<nav>` ha sido diseñado para identificar la navegación de un sitio web. La navegación se define como un conjunto de enlaces que hacen referencia a las secciones de

una página o un sitio, pero no todos los enlaces son candidatos de pertenecer a un elemento `<nav>`: una lista de enlaces a patrocinadores o los resultados de una búsqueda, no forman parte de la navegación principal, sino que corresponden con el contenido de la página.

Como ocurre con los elementos `<header>`, `<footer>` y el resto de nuevas etiquetas, no estamos obligados a utilizar un único elemento `<nav>` en toda la página. Es posible que tengamos una navegación principal en la cabecera de la página, una tabla de contenidos o enlaces en el pie de la página, que apuntan a contenidos secundarios. Todos ellos son candidatos para pertenecer a un elemento `<nav>`.

La etiqueta `<nav>` puede ir dentro del header o fuera de él.

```
<nav>
  <ul>
    <li><a href="#">home</a></li>
    <li><a href="#">blog</a></li>
    <li><a href="#">gallery</a></li>
    <li><a href="#">about</a></li>
  </ul>
</nav>
```

### **<main>**

main contiene todo el contenido principal de la página, con las siguientes características:

- ✓ Sólo puede existir un único main en la página.
- ✓ No puede estar dentro de ninguna otra caja.
- ✓ Dentro contendrá aquel contenido que no se repita en cada página (conteniendo `<div>`, `<section>`, `<article>` o incluso `<aside>`), pero nunca cajas como `<header>`, `<nav>` o `<footer>`).

```
<main>
  <section>
    <article> contenido </article>
    <article> contenido </article>
  </section>
</main>
```

### **<footer>**

Según la especificación, un elemento `<footer>` representa lo siguiente:

Representa el pie de una sección. Un pie tradicionalmente contiene información acerca de su sección, como quién escribió el contenido, enlaces relacionados, copyright y similares.

Al igual que ocurre con el elemento `<nav>`, podemos tener tantos elementos `<footer>` como sea necesario. Lo normal es que nuestro sitio web disponga de al menos un pie principal, que contiene los avisos legales (privacidad, condiciones del servicio, copyright...), mapa del sitio web, accesibilidad, contacto y otros muchos enlaces que pueden ir incluidos en un elemento `<nav>`.

## <article>

Según la especificación, un elemento <article> representa lo siguiente:

Este elemento representa un contenido completo, auto-contenido en un documento, página, aplicación o sitio web, que es, en principio, independiente de ser distribuido y reutilizado, por ejemplo en un RSS. Puede ser un post de un foro, un artículo de un periódico o revista, una entrada de un blog, un comentario de un usuario, un widget o cualquier otro elemento independiente.

Cuando los artículos están anidados, los artículos interiores representan contenido que en principio está relacionado con el artículo que los contiene. Por ejemplo, una entrada de un blog puede aceptar comentarios de usuarios, que están incluidos dentro del contenido principal y relacionados con el mismo.

Por lo tanto, la etiqueta <article> se utiliza para encapsular contenido, que tiene significado en sí mismo, y que puede ser distribuido y reutilizado en otros formatos de datos. No nos referimos únicamente a contenidos clásicos de texto, sino que incluso un contenido multimedia con su transcripción, un mapa o email pueden ser totalmente válidos para ser incluidos en una etiqueta <article>.

## <section>

```
<h1>Comments</h1>
```

```
<article id="c1">
```

```
<footer>
```

```
<p>Posted by: <span>George Washington</span></p>
```

```
<p><time datetime="2022-10-10">15 minutes ago</time></p>
```

```
</footer>
```

```
<p>Yeah! Especially when talking about your lobbyist friends!</p>
```

```
</article>
```

```
<article id="c2">
```

```
<footer>
```

```
<p>Posted by: <span itemprop="name">George Hammond</span></p>
```

```
<p><time datetime="2019-10-10">5 minutes ago</time></p>
```

```
</footer>
```

```
<p>Hey, you have the same first name as me.</p>
```

```
</article>
```

```
</section>
```

## <section>

A diferencia del elemento <article>, este elemento es utilizado para dividir el documento (o artículos) en diferentes áreas, o como su propio nombre indica, en secciones. Según la especificación, un elemento <section> representa lo siguiente:

Representa una sección genérica de un documento o aplicación. Una sección, en este contexto, es un grupo temático de contenido, que generalmente incluye una cabecera.



```
<article>
  <h1>Rules for Munchkins</h1>
  <section>
    <h2>Yellow Brick Road</h2>
    <p>It is vital that Dorothy follows it—so no selling
      bricks as "souvenirs"</p>
  </section>
  <section>
    <h2>Fan Club uniforms</h2>
    <p>All Munchkins are obliged to wear their "I'm a friend
      of Dorothy!" t-shirt when representing the club</p>
  </section>
  <p><strong>Vital caveat about the information above:
    does not apply on the first Thursday of the month.</strong></p>
</article>

<article>
  <h1>Rules for Munchkins</h1>
  <section>
    <h2>Yellow Brick Road</h2>
    <p>It is vital that Dorothy follows it—so no selling
      bricks as "souvenirs"</p>
  </section>
  <section>
    <h2>Fan Club uniforms</h2>
    <p>All Munchkins are obliged to wear their "I'm a friend
      of Dorothy!" t-shirt when representing the club</p>
    <p><strong>Vital caveat about the information above:
      does not apply on the first Thursday of the month.</strong></p>
  </section>
</article>
```

Como podemos observar en los dos ejemplos anteriores, es muy sencillo agrupar contenido que pertenece a una misma sección, permitiendo incluirlo dentro de un contexto semántico.

Otra de las posibilidades que nos ofrece esta etiqueta, es la de dividir nuestro documento en secciones, que incluyen contenido de temáticas diferentes entre sí. Si además queremos separar estos contenidos visualmente en dos columnas, lo lógico sería utilizar las tradicionales etiquetas <div> para agrupar los artículos según su temática, y posteriormente aplicar estilos CSS o JavaScript para presentarlos en forma de *pestañas*.

En este caso, la etiqueta <div> no nos aporta ningún significado semántico, tan sólo estructural. La etiqueta <section> es la encargada de añadir semántica en estos casos:

```
<section>
  <h1>Articles about llamas</h1>
  <article>
    <h2>The daily llama: Buddhism and South American camelids</h2>
    <p>blah blah</p>
  </article>
  <article>
    <h2>Shh! Do not alarm a llama</h2>
    <p>blah blah</p>
  </article>
</section>
<section>
  <h1>Articles about root vegetables</h1>
  <article>
    <h2>Carrots: the orange miracle</h2>
    <p>blah blah</p>
  </article>
  <article>
    <h2>Eat more Swedes (the vegetables, not the people)</h2>
    <p>blah blah</p>
  </article>
</section>
```

### EJERCICIO: SEPARAR EN DOS COLUMNAS

#### **<aside>**

Según la especificación, un elemento <aside> representa lo siguiente:

Una sección de una página que consiste en contenido tangencialmente relacionado con el contenido alrededor del elemento, y puede considerarse separado de este contenido. Estas secciones son normalmente representadas como elementos laterales en medios impresos. Este elemento puede utilizarse para contener citas, anuncios, grupos de elementos de navegación y cualquier otro contenido separado del contenido principal de la página.

Dentro de un artículo, por ejemplo, puede ser utilizado para mostrar contenido relacionado como citas u otros artículos relacionados.

#### **<figure>**

Es un nuevo tipo de caja que nos permite contener imágenes, con la posibilidad de añadir un pie de imagen vinculada a ésta. De esta forma el navegador y cualquier robot que acceda a nuestra página conocerá la naturaleza del contenido que contiene este tipo de bloque y su información asociada.

<figure> no tiene las mismas funcionalidades ni características que el resto de cajas, tiene como objetivo contener imágenes que opcionalmente pueden tener vinculadas un pie de imagen (información más detallada en el tema figure y figcaption).

```
<figure>
  
  <figcaption>
    Este texto es el <b>figcaption</b> de esta imagen
  </figcaption>
</figure>
```

## 5. ESTRUCTURA DE UN DOCUMENTO HTML CON ETIQUETAS SEMÁNTICAS

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8"/>
    <meta name="description" content="Resumen del contenido de la página">
    <title>Título de la página</title>
    <link rel="stylesheet" href="css/estilo.css" type="text/css"/>
    <style type="text/css">
      código CSS
      código CSS
    </style>
    <script type="text/javascript">
      código javascript
      código javascript
    </script>
    <script type="text/javascript" src="js/codigo.js"></script>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>

  <body>
    <header>cabecera</header>
    <nav>
      enlace1
      enlace2
    </nav>
    <main>
      <section>
        <article> contenido </article>
        <article> contenido </article>
      </section>
    </main>
    <footer> © pie de la página </footer>
  </body>
</html>
```