



# Diseño Web Responsive HTML5 y CSS3

Unidad 4: Maquetación con Flexbox



## Indice

### Unidad 4: Maquetación con Model Box

#### Elementos semánticos HTML5





## Objetivos

### Que el alumno logre:

- Logre aplicar los atributos de estilos CSS a la estructura del sitio a través de la utilización de cajas.





## ¿Qué aporta HTML 5?

HTML 5 ha sido definido pensando en las personas que construyen páginas web y desarrollan aplicaciones para ellas, por lo que su orientación es ante todo práctica, con el objetivo de resolver los muchos problemas de versiones anteriores.

HTML es ahora una herramienta más útil y cercana a las personas que van a utilizarla. Además de simplificar el desarrollo, lo ha dotado de herramientas más potentes, reduciendo complejidades innecesarias. Ha eliminado componentes superfluos y que aportaban poco, y ha incorporado mecanismos ampliamente utilizados por la comunidad de desarrollo del lenguaje.

La potencial extensibilidad de HTML 5 trata de evitar que el lenguaje vaya quedando atrás cada vez más lejos de la continua evolución de la web.

HTML 5 va a suponer un marco estable para el desarrollo de páginas web, respondiendo a necesidades reales, y aportando medios para acceder a funciones potentes y sofisticadas, especialmente en lo que se refiere a los contenidos multimedia.

## HTML5: la estructura básica del documento

Comenzando desde las primeras líneas de código, vamos a repasar los elementos que ya conocemos y cuales se han modificado en esta nueva versión de HTML.

### *doctype*

Para empezar cualquier página nuestra primera línea de código es el **doctype**.

Históricamente este elemento nunca ha tenido un buen acompañamiento, ya que había que indicarle algunos parámetros y versiones que normalmente nunca se ajustaban a la realidad. Ahora, con el HTML 5 esto queda reducido a una única opción muy simple:

```
<!DOCTYPE HTML>
```

Es un DOCTYPE mucho más simplificado que XHTML (cuyas reglas siguen siendo usadas) y te permite usar todas las habilidades de HTML5 sin que nada de lo que ya tienes programado deje de funcionar.

Este elemento no hay que cerrarlo ni nada, simplemente será la primera línea de cada una de nuestras páginas.



## html

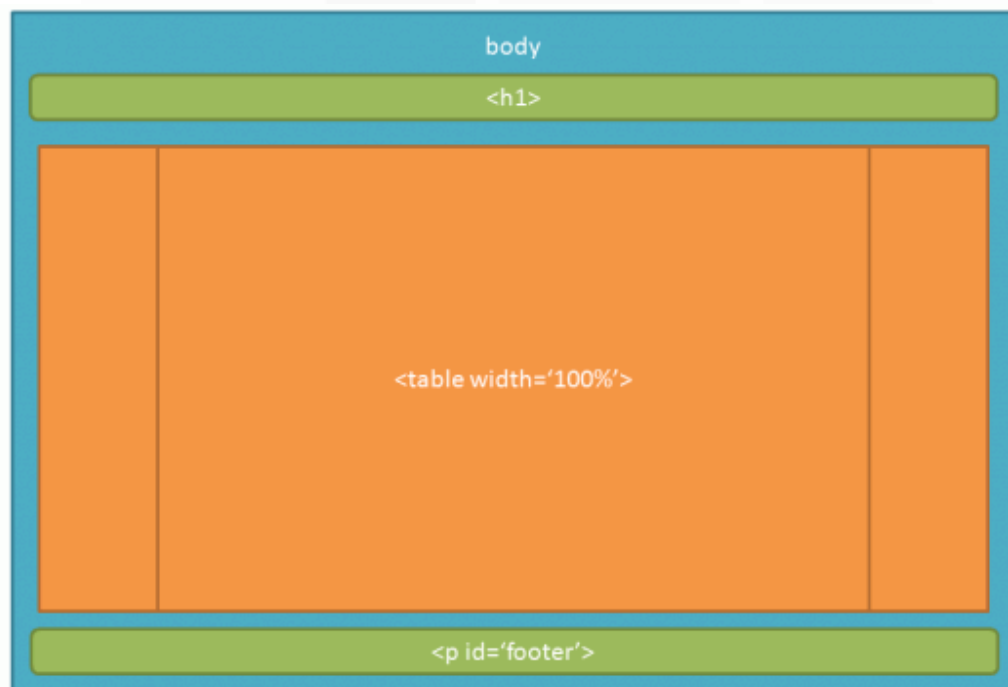
Tanto la etiqueta `<html> ... </html>` como la etiqueta `<head> ... </head>` se mantienen igual que en las versiones anteriores (en las próximas unidades veremos algunos atributos que podemos agregar)

Dentro de la etiqueta `<body> </body>`, la estructura de maquetación fue variando de acuerdo a cada versión de HTML.

Hace unos 10 años, cuando se quería hacer una página, te podías plantear el uso de los *frames* que eran la forma más sencilla de no tener que repetir y repetir infinidad de veces el mismo código. Con la entrada de los lenguajes de programación esto comenzó a variar y se usaban *tablas* para dar formato al sitio... La aparición de los CSS hizo que los *div* se convirtieran en la mejor opción para formatear los sitios, y ahora llega el HTML 5 que incluye unas nuevas etiquetas que solucionan todos estos procesos genéricos.

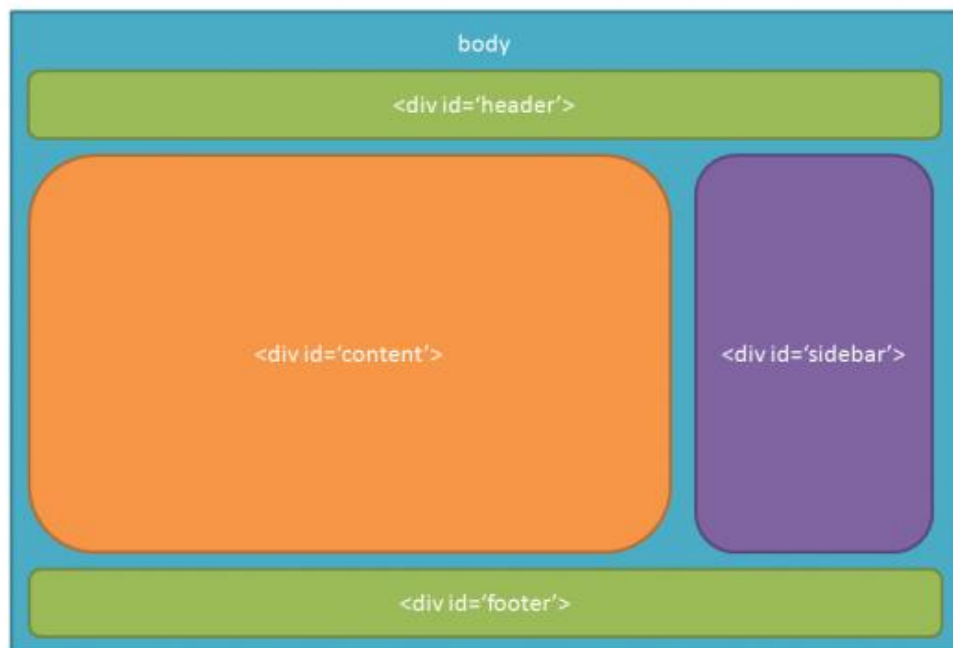
Volviendo sobre las versiones anteriores de HTML podemos definir la estructura del sitio de la siguiente forma:

Estructura básica en **HTML3**:

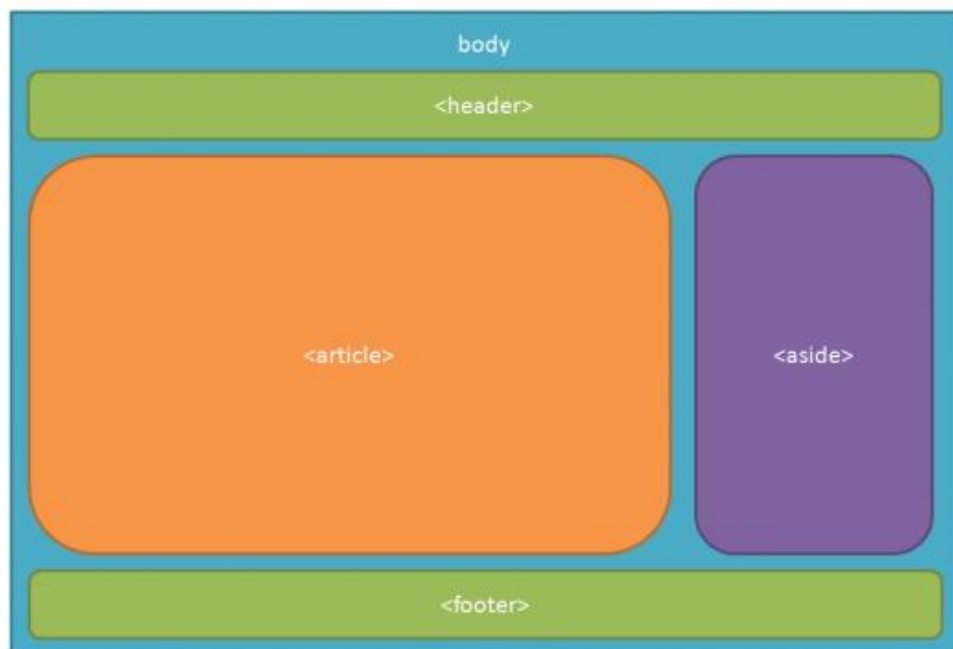




Estructura básica en **HTML4** (o en **XHTML1**):



Estructura básica en **HTML5**:





## *header*

En principio se presenta como un contenedor que agrupa elementos introductorios o un conjunto de información referente a lo que a continuación se pondrá.

Podemos darle dos grandes usos de este elemento:

- **Cabecera de la página web**, donde se incluirían el logo, el menú de navegación, etc.
- **Encabezado de algunos bloques de información**. Por ejemplo, si tuviésemos un contenido que es la ficha de un libro, en este *header* incorporaríamos como bloque el título del libro y una pequeña ficha técnica, como encabezado del resto de información.

<header> está diseñada para reemplazar la necesidad de crear divs sin significado semántico.

## *footer*

El elemento **footer** va a ser muy parecido al *header* anterior.

Básicamente se ha incluido como agrupador de elementos al final de la estructura del sitio.

Por norma general los sitios web tienen un pie de página en el que se indican datos legales y de contacto, y esta podría ser la principal función, aunque no está pensado para que sea la única.

De la misma forma que el header, hay dos funciones principales para el elemento:

- Pie de página del sitio, donde se encuentran los datos legales, etc.
- Pie de bloque de contenido, donde incorporar información como la fecha, enlaces relacionados, nube de tags y similares.

## *article*

El nuevo elemento **article** está destinado a revolucionar los contenedores y el SEO.



Este elemento viene a ser el “contenedor” de lo que es importante en la página. Puede haber varios y es el agrupador de lo que vendría a ser “lo indexable” por un buscador, el contenido principal de la página.

Para hacernos una idea básica, una página tendría este formato:

```
<body>
  <header>...</header>
<section>
  <article>...</article>
  <article>...</article>
  <article>...</article>
</section>
  <aside>...</aside>
  <footer>...</footer>
</body>
```

Esto daría a entender que hay 3 bloques de información principales en el sitio, que son los contenidos que el usuario ha de contemplar, además de una cabecera, un menú lateral y un pie de página.

### *aside*

Otro de los nuevos elementos es el **aside**. Viene a ser “la barra lateral” que creábamos con divs.

Cualquier contenido que no esté relacionado con el objetivo primario de la página va en un aside.

Es el contenido que, página a página dentro del sitio web se repite junto a la cabecera y el pie, y a su vez, los buscadores le darán menos peso a este bloque que al contenido dentro de **article**.

### *section*

El nuevo elemento **section** viene a ser el `<div id="nombre">` que llevamos usando hasta ahora.

Define un área de contenido única dentro del sitio.

```
<section>
  <header>...</header>
  <article class="noticia">...</article>
  <article class="relacionados">...</article>
  <article class="multimedia">...</article>
</section>
```





## *nav*

Este elemento es *el que agrupa los enlaces*. Es un *section* especial para los enlaces, tanto absolutos como relativos.

Dentro de este elemento es donde incluiremos nuestra botonera.

Se puede utilizar cualquier etiqueta para organizar los enlaces, lo recomendado es utilizar listas no ordenadas (<ul>).

## *h1 – h6*

Estos elementos son los títulos de las diferentes *sections* y cada *h* tiene un peso (ranking específico en base al número que tiene, del 1 (el más alto) al 6 (el más bajo)).



## Atributos nuevos globales

Empezaremos a desarrollar algunos de los nuevos atributos incorporados por HTML 5 para las etiquetas nuevas y existentes:

### *dir*

El atributo **dir** básicamente indica la dirección del texto, o, mejor dicho, de escritura, para aquellos idiomas en los que se escribe de derecha a izquierda y no de izquierda a derecha como lo hacemos nosotros.

Tiene tan sólo dos opciones que corresponden a lo que acabo de comentar: *ltr* (left to right) o *rtl* (right to left).

```
<div dir="rtl">Este contenido está escrito de derecha a izquierda...</div>
```

### *draggable*

Otra de las nuevas incorporaciones es el **draggable**.

Permite incorporar o eliminar elementos de una página simplemente arrastrándolos.

Un ejemplo sería una imagen que, al arrastrarla fuera de la ventana y colocarla en un editor de imágenes, se podía abrir. Otro ejemplo el de un enlace que se arrastraba sobre una pestaña del navegador y ejecutaba esa dirección.

Los parámetros que permite son *true*, *false* o *auto*, que es el que está por defecto, dejando a cada usuario / navegador decidir los elementos.

### *hidden*

Hasta ahora cuando se requería que un bloque de contenido no se viera había que utilizar los CSS para ocultarlo. Ahora eso ya no será necesario, ya que si un elemento incorpora este atributo simplemente no se mostrará por pantalla.

De todas formas, este elemento no debe usarse para ocultar bloques o pestañas, sino que se ha de utilizar para elementos que no son todavía o no van a ser relevantes.



`<div hidden>Este contenido está oculto por no ser relevante para el usuario...</div>`

## *lang*

El atributo **lang** marca el idioma del contenido de ese fragmento de texto.

Hay que tener en cuenta que, aunque un documento se le aplique un idioma, por normal general, puede haber fragmentos de texto de contenidos que se encuentran en otros idiomas, por lo que hay que indicarlos.

`<div lang="es">este bloque se encuentra escrito en español...</div>`

`<div lang="en">...and this is in english...</div>`

## *title*

**Title** básicamente hace lo que su nombre indica: *titular*. Y es que muchos elementos, cuando se pone el ratón encima permiten mostrar más información de lo que eso significa. Un ejemplo muy habitual sería el las abreviaturas:

`<p>Internet usa mucho el protocolo <abbr title="Hypertext Transport Protocol">HTTP</abbr> y el <abbr title="File Transport Protocol">FTP</abbr>.</p>`



## Resumen

### En esta Unidad...

Trabajamos con el Modelo de cajas, aplicado al a maquetación.

### En la próxima Unidad...

En la próxima unidad vamos a comenzar a trabajar con las nuevas incorporaciones de CSS3.