

Layouts

El layout es como se va a visualizar o distribuir el contenido de una web.

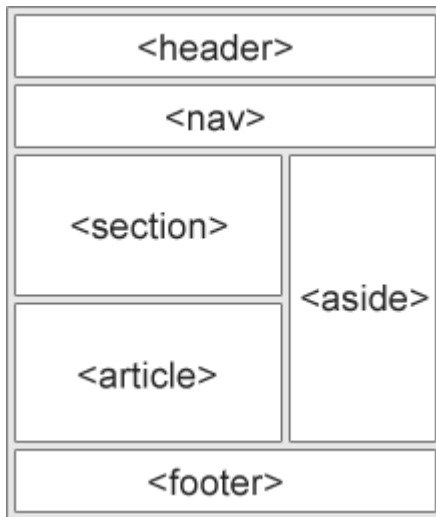
Elementos HTML para Layouts

HTML nos ofrece varias etiquetas semánticas que definen diferentes partes de una web:

- [<header>](#)
 - Se utiliza como una **introducción** del elemento que la contiene. Si colocamos un HEADER en el BODY, a primer nivel, será una introducción o presentación de toda la página. Si colocamos un HEADER dentro de un SECTION, contendrá una introducción a esa sección. También se pueden colocar dentro de un ARTICLE.
 - El HEADER suele llevar el título, la descripción corta y el logo de la página. El HEADER por si mismo no separa contenidos, no se debe utilizar aislado, sino dentro de otra etiqueta semántica, o a primer nivel en el BODY.
 - El HEADER no es equivalente a la parte superior de la página, aunque frecuentemente está en esa posición. La etiqueta HGROUP que se solía utilizar junto con el HEADER ha quedado obsoleta.
 - Es bastante frecuente utilizar las etiquetas H1, H2, ... para escribir los títulos dentro del HEADER.
- [<nav>](#)
 - Esta etiqueta está pensada para contener la **barra de navegación** o los enlaces a las páginas del sitio web.
 - Normalmente va en la parte superior de la página o en un lateral. Suele estar formada por una lista de enlaces con las etiquetas UL y LI. Aunque hay multitud de diseños diferentes de barras de navegación y menús.
- [<section>](#)
 - Una SECTION agrupa contenido con un tema común, por ejemplo, las diferentes partes de un periódico: política, sociedad, deportes, ...
 - Un sitio de cocina podría estar organizado en secciones. Una SECTION podría ser recetas de carne, otra recetas de pescado y otra de verduras. Dentro de cada SECTION tendríamos un ARTICLE por cada receta.

- Cada SECTION puede estar encabezada por una etiqueta H1, H2,... dentro de una SECTION suelen haber etiquetas ARTICLE, y también etiquetas DIV y P.
 - Una SECTION no debe ser usada con el único fin de dar formato, para eso se utiliza la etiqueta DIV.
- [<article>](#)
 - Representa una unidad de contenido, es decir, contenido que responde a un tema concreto. El ejemplo más claro es un artículo dentro de una revista, por ejemplo, en una página de cocina, un ARTICLE sería cómo elaborar un determinado plato. En un foro, un ARTICLE sería un post, o entrada en el foro.
 - Se pueden anidar un ARTICLE dentro de otro ARTICLE. Dentro de un ARTICLE el contenido se suele estructurar con las etiquetas DIV y P. También está permitido poner dentro de un ARTICLE una SECTION, siempre que el artículo sea largo y tenga diferentes secciones, pero lo más natural es que el ARTICLE vaya dentro de una SECTION. Más adelante trataremos cómo crear estructuras de páginas complejas, con tres niveles.
- [<aside>](#)
 - Esta etiqueta está diseñada para contenido tangencial, es decir, menos importante que el cuerpo de la página, y que suele ser de un tema relacionado indirectamente (o no relacionado) con el tema principal. Se suele colocar en los laterales de la página. Pero no todo lo que va en los laterales es ASIDE. Por ejemplo, si la página tiene publicidad en el lateral derecho, puede colocarse en una etiqueta ASIDE, también una pequeña reseña sobre el autor podría ir en esta etiqueta.
- [<footer>](#)
 - Normalmente va al final de la página y contiene información del tipo: autor, copyright, contacto, mapa del sitio, etc. Se pueden poner HEADER y FOOTER dentro de SECTION y ARTICLE, aunque no es lo más normal. No hay nada prohibido, pero debe tener cierta lógica o sentido semántico.
- [<details>](#) Define detalles adicionales
 - [<summary>](#) Define un encabezado para el <details>
 - [<figcaption>](#) Define una leyenda para una <figure> elemento
 - [<figure>](#) Especifica el contenido en sí misma, como ilustraciones, diagramas, fotos, listados de código, etc.

- `<main>` Especifica el contenido principal de un documento
- `<mark>` Define texto marcado / resaltado
- `<time>` Define una fecha / hora



Métodos de Layouts

Hay cinco maneras de crear un sistema de layout, cada una tiene sus ventajas y desventajas:

- Tablas HTML (no recomendado)
- CSS float
- CSS flexbox
- CSS framework
- CSS grid

Tablas HTML

El elemento `<table>` no se creó para utilizarlo para crear layouts. Así que no deberíamos de usar este método.

CSS Frameworks

Es la forma más rápida de crear un layout, usar algún frameworks como lo es [Bootstrap](#).

CSS Floats

Es una forma muy común de maquetar toda la web usando la propiedad de CSS float. Es fácil de aprender, pero en algunos casos puede que no sea todo lo flexible que necesitamos.

CSS Flexbox

Flexbox CSS es un nuevo sistema de layout desde CSS3. Se va ajustando a las diferentes resoluciones de pantalla. Pero no funciona con IE10.

CSS Grid Layout

CSS Grid Layout nos ofrece un layout basado en una rejilla con columnas y filas. Es fácil posicionar los elementos. Pero no funciona en IE y en Edge 15 o inferiores.

Referencias:

w3bai.com