

Capítulo 2-3

Los contenedores semánticos

1. Utilizar correctamente los contenedores semánticos

Para diseñar sus páginas, debe pensar en términos de «contenedor». Un contenedor, como su nombre indica, incluye un contenido de tipo muy variado.

En estos contenedores, puede situar texto, imágenes, formularios, enlaces, tablas, etc. Pero también puede tener contenedores más «pequeños», como para resaltar una o varias palabras o para definir una celda de una tabla.

Los contenedores también sirven para estructurar sus páginas. De esta manera, puede utilizar contenedores para insertar un encabezado de página, una columna de desplazamiento (un *sidebar* en inglés), un pie de página, una barra de navegación, etc.

Como habrá entendido, todos los contenidos se insertan en contenedores. Cada contenedor, fuera de las capas (`<div>` y ``), están dedicados a recibir un contenido específico. Para esto se califican los contenedores semánticos.

2. El elemento <div>

El elemento `<div>` es uno de los contenedores más antiguos del HTML. Permite crear una capa estructural en la página. En estas capas, podemos situar cualquier contenido, incluso otros contenedores, como otras capas `<div>`, párrafos, listas, etc.

El HTML5 introduce nuevos contenedores semánticos que han reducido el uso del elemento `<div>`. Pero el motivo por el que debe evitar usar cajas `<div>` no es porque utilice HTML5. Siempre se pueden usar y efectivamente tienen su utilidad. Las cajas `<div>` habitualmente se usan para conseguir contenedores «neutros», que no necesariamente tienen un sentido semántico concreto.

3. El elemento

El elemento `` se utiliza habitualmente para aplicar un formato específico a una parte del texto dentro de un párrafo. Es especialmente útil cuando se desea destacar un fragmento sin alterar la estructura del contenido.

A continuación se muestra un ejemplo concreto: queremos poner el fondo gris y un borde fino en una parte de un texto de un párrafo.

A continuación se muestra el selector CSS:

```
.fondo-gris {  
    background-color: #eee;  
    padding: 0 5px;  
    border: 1px solid #333;  
}
```

A continuación se muestra su aplicación en el código HTML:

```
<p>Donec ullamcorper nulla no metus auctor  
fringilla. Morbi leo risus, <span class="fondo-gris">porta  
ac consectetur ac vestibulum</span> at eros. Donec sed odio...</p>
```

A continuación se muestra la visualización obtenida:

```
Donec ullamcorper nulla non metus auctor fringilla. Morbi leo risus, porta ac consectetur ac vestibulum at eros. Donec sed odio...
```

4. El elemento `<header>`

El elemento `<header>` permite insertar una zona de visualización para los encabezados. Estos encabezados se pueden utilizar en varios lugares:

- A nivel de la página: es el clásico encabezado de página, habitualmente ubicado en la parte superior de la pantalla, con un logo, un eslogan, una barra de navegación principal, etc.
- A nivel de los contenidos, lo que permite tener una introducción al contenido que sigue, como el encabezado de un artículo, por ejemplo.

Este contenedor puede contener todo tipo de elementos: títulos, párrafos, enlaces, etc.

Puede anidar los elementos `<header>` y `<footer>` en otro elemento `<header>` si los dos primeros elementos se incluyen en un mismo elemento padre.

A continuación se muestra un ejemplo de anidación perfectamente validado:

```
<article>
  <header>
    <h2>Cras Vestibulum Sem Fermentum</h2>
    <aside>
      <header>
        <h3>Inceptos Magna Vehicula Malesuada</h3>
        <p>Mollis Risus Sollicitudin Inceptos...</p>
      </header>
      <p>Sit Mattis Aenean Commodo...</p>
      <footer>
        <p>Parturient Pharetra Quam</p>
      </footer>
    </aside>
  </header>
  <p>Adipiscing Ultricies Dapibus Mollis...</p>
</article>
```

5. El elemento `<footer>`

El elemento `<footer>` se utiliza para definir el pie de página de un documento o de una sección específica, como un artículo. Su función semántica es similar a la del elemento `<header>`, aunque no es obligatorio usar ambos conjuntamente. Es decir, puede haber un `<footer>` sin `<header>`, y viceversa.

6. El elemento `<aside>`

El elemento `<aside>` permite mostrar un contenido relacionado de manera indirecta con el contenido principal al que se le asocia. Esto se corresponde habitualmente con las clásicas barras de desplazamientos (*sidebar* en inglés), las zonas de componentes de interfaz (*widgets* en inglés), los complementos sobre los artículos o cualquier otro contenido textual.

Por lo general, el contenido principal aparece en el centro de la página, mientras que el contenido asociado al elemento `<aside>` se muestra a la derecha.

7. El elemento `<nav>`

El elemento `<nav>`, como su nombre permite intuir, se dedica a la visualización de una barra de navegación con enlaces de hipertextos. Pero atención, no se sienta obligado a tener una única zona de navegación por página. Puede crear tantos elementos `<nav>` como quiera, con navegaciones diferentes en sus páginas, siempre y cuando cada uno de ellos se identifique correctamente. El elemento `<nav>` quizás se dedica más a la navegación principal del sitio web, a la creación de enlaces entre las páginas del sitio web.

Puede incluir una navegación principal `<nav>` en un encabezado `<header>` y una navegación secundaria `<nav>` en un pie de página `<footer>`, por ejemplo.

8. El elemento <main>

El elemento `<main>` permite indicar el contenido principal de la página. Este contenido debe ser único y no repetirse en la página. Además, la norma indica concretamente su contexto de utilización: no se debe usar en el interior, como elemento incluido, e los elementos `<article>`, `<aside>`, `<footer>`, `<header>` o `<nav>`.

9. El elemento <section>

El elemento `<section>` permite agrupar los elementos que comparten una misma temática. Esto permite agrupar, en un mismo elemento, un contenido estructurado, con su encabezado y su pie de página. La utilización de varios elementos `<section>` distinguirá varias partes, varias secciones dentro de una misma página, con otros elementos de estructura anidados.

10. El elemento <article>

El elemento `<article>` permite insertar un contenido autónomo. Está calificado como autónomo porque se puede reutilizar en cualquier lugar en el sitio web, sin que su comprensión se vea afectada. El uso más habitual retoma el nombre del elemento: creación de artículos de blog y de actualidad.

11. El elemento <search>

El elemento `<search>` permite agrupar todos los elementos relacionados con una búsqueda textual en el sitio web. Puede incluir formularios, títulos, párrafos y eventualmente los resultados. Es importante mantener esta área bien estructurada para facilitar su comprensión y accesibilidad.

12. La visualización de los elementos de estructura

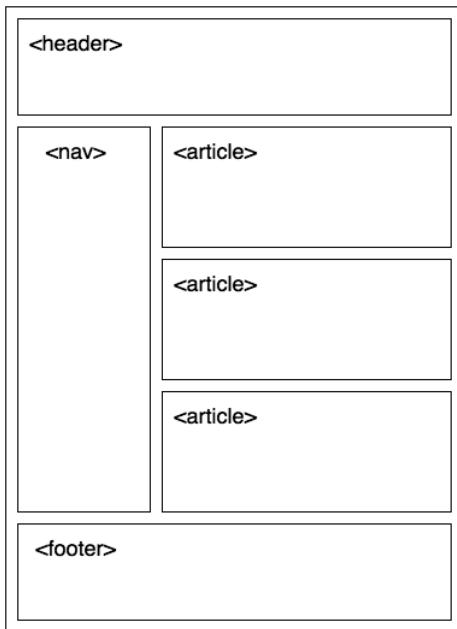
Todos los elementos de estructura de tipo `block` que acabamos de ver, es decir, `<div>`, `<header>`, `<footer>`, `<aside>`, `<nav>`, `<main>`, `<section>` y `<article>`, utilizan la propiedad `display`, pero no tienen un valor específico para los márgenes.

```
div, header, footer, aside, nav, main, section, article {  
    display : block ;  
}
```

13. Dos ejemplos de estructura semántica de página

13.1 Una estructura semántica sencilla

A continuación se muestra un primer ejemplo con una estructura semántica muy sencilla:

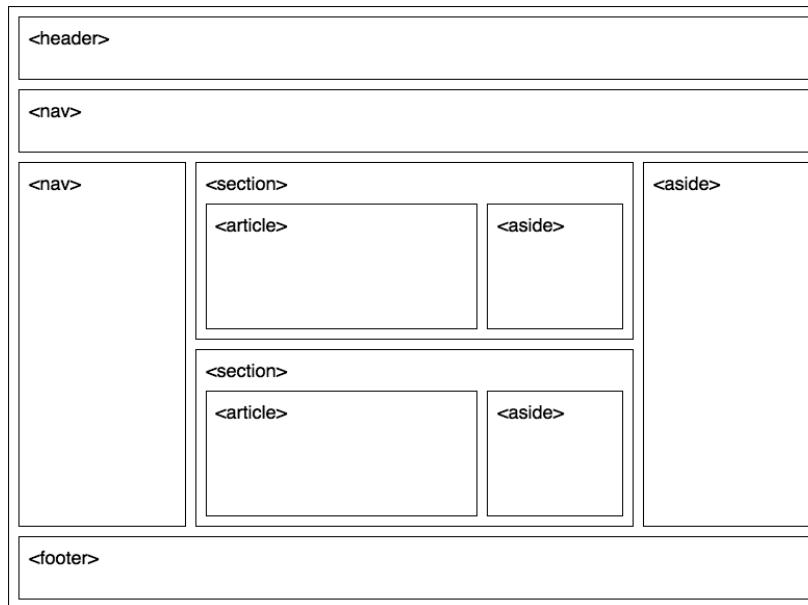


Tenemos:

- Un encabezado `<header>` en la parte superior, con un logo y un eslogan, por ejemplo.
- Una barra de navegación `<nav>` en la parte izquierda de la página.
- Toda la actualidad del sitio web se podrá situar en elementos `<article>`.
- Para terminar, el pie de página `<footer>` podrá contener las menciones legales, los enlaces de contacto, etc.

13.2 Una estructura semántica más elaborada

A continuación se muestra una segunda estructura más elaborada:



Tenemos:

- Un encabezado `<header>` en la parte superior.
- Más abajo, una barra de navegación `<nav>` para la navegación general del sitio web, para navegar entre las páginas.
- A la izquierda, una segunda caja `<nav>` para la navegación secundaria, para los enlaces relacionados directamente con el contenido de la página mostrada.
- A la derecha, un elemento `<aside>` muestra la información relacionada con el contenido de la página, como los enlaces promocionales, los contenidos relacionados, etc.
- El contenido de la página se muestra en dos elementos `<section>`, que permiten de esta manera diferenciar correctamente estos dos contenidos. Cada elemento `<section>` contiene un elemento `<article>` para el contenido textual y un elemento `<aside>` para mostrar los elementos de información adicionales relacionados con el artículo (iconografía, enlaces, etc).
- Finalmente, un pie de página `<footer>` para mostrar la información legal, las condiciones de venta, un enlace de contacto, un plano de acceso, etc.

14. Un ejemplo de estructura semántica de un artículo

A continuación se muestra un ejemplo de un artículo, de un contenido textual que utiliza estos elementos de estructura semántica:



Tenemos:

- Un elemento `<article>` como contenedor general.
- Nuestros artículos contienen encabezados, introducciones; por lo tanto utilizamos el elemento `<header>`. El elemento `<header>` contiene el título `<h1>` del artículo y su subtítulo `<h2>`.
- El contenido textual del artículo se ubica en los elementos `<p>`. El artículo contiene los enlaces a los complementos de artículos, listados en una lista ``.