

Objetivos

Las páginas web que va a diseñar deben estar estructuradas con contenedores. Se trata de contenedores que permiten construir la arquitectura de las páginas web. Estos contenedores van a contener tipos de contenido variados: una barra de navegación, una columna lateral, encabezados, pies de página...

Con HTML 4, los desarrolladores utilizaban principalmente el elemento `<div>` para diseñar estos contenedores. Los desarrolladores hablan de "cajas" `<div>`. Una página web que comprenda una buena cantidad de información estructurada que deba mostrar va a contener una cantidad importante de cajas `<div>`, sin que su contenedor sea distinto. La única distinción clara se realizará mediante selectores CSS que indicarán su utilidad. Nos encontramos con una gran cantidad de contenidos no semánticos.

Con la aparición de HTML5, el W3C nos propone toda una serie de contenedores estructurales semánticos.

El elemento <div>

¡No por utilizar HTML5 debe abandonarse el uso de cajas <div>! ¡Sigue siendo posible usarlas y siguen teniendo su utilidad! Las cajas <div> se usan a menudo para disponer de contenidos "neutros", sin necesidad de tener un contenido semántico preciso.

El elemento

El elemento se utiliza a menudo para realizar una división dentro de un párrafo de texto, por ejemplo. Resulta muy práctico para poner en un formato particular un texto que esté formateado de otra manera.

Veamos un ejemplo concreto: deseamos poner sobre fondo gris y con un borde cierta parte de un texto dentro de un párrafo.

He aquí el selector CSS:

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

He aquí su aplicación en el código HTML:

```
<p>Donec ullamcorper nulla non metus auctor  
fringilla. Morbi leo risus, <span class="fondo-gris">porta  
ac consectetur ac vestibulum</span> at eros. Donec sed odio...</p>
```

He aquí su representación:

Donec ullamcorper nulla non metus auctor fringilla. Morbi leo risus, porta ac consectetur ac vestibulum at eros. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur blandit tempus porttitor.

El archivo de este ejemplo se llama C05-span.html.

El elemento <header>

El nuevo elemento <header> permite insertar una zona de representación para los encabezados. Estos encabezados pueden utilizarse en distintos lugares:

- a nivel de la página, se trata del clásico encabezado, situado a menudo en la parte superior de la pantalla, con un logotipo, un eslogan, una barra de navegación principal...
- a nivel de los contenidos, esto permite tener una zona de introducción para el contenido que viene a continuación, como el encabezado de un artículo, por ejemplo.

Este contenedor puede contener cualquier tipo de elemento: títulos, párrafos, enlaces...

El elemento <footer>

El nuevo elemento <footer> permite insertar una zona de representación para los pies de página. Encontramos la misma semántica que para los encabezados. Estos pies de página pueden definirse para la página o para cualquier otra zona de representación dentro de ella, o para un artículo. Observe que el uso de un <footer> no implica necesariamente el uso de un <header>.

El elemento <aside>

El elemento <aside> permite mostrar un contenido vinculado a un contenido principal al cual está asociado. Esto se corresponde a menudo con las clásicas barras laterales (sidebar), a zonas de widgets, a complementos de artículos o cualquier otro contenido de redacción.

Habitualmente se tiene el contenido principal que se muestra en la parte central de la página, y con frecuencia en la parte derecha la representación del contenido asociado con el elemento <aside>.

El elemento <nav>

El elemento <nav>, como su nombre permite adivinar, está dedicado a la representación de una navegación con enlaces de hipertexto. Pero, preste atención, no se sienta obligado a tener una única zona de navegación por página. Puede crear tantos elementos <nav> como navegaciones existan en sus páginas, siempre que cada una de ellas esté bien identificada. El elemento <nav> está pensado principalmente para la navegación principal del sitio, para la creación de enlaces entre las páginas del sitio.

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

El elemento <main>

El elemento <main> permite indicar el contenido principal de la página. Este contenido debe ser único y no estar repetido en la página. Además, W3C indica con precisión su contexto de uso: no debe utilizarse en el interior (elemento descendiente) de los elementos <article>, <aside>, <footer>, <header> o <nav>.

El elemento `<section>`

El elemento `<section>` permite agrupar elementos que comparten una misma temática. Esto permite agrupar en un mismo elemento un contenido estructurado, con su encabezado y su pie de página, de modo que se pueden distinguir varias partes, varias secciones en el seno de una misma página, con otros elementos de estructura anidados.

El elemento <article>

El elemento <article> permite insertar un contenido autónomo, pues puede reutilizarse en otros lugares dentro del sitio, sin que su comprensión se vea afectada. El uso más habitual se corresponde con el nombre del elemento: creación de artículos de blog y de actualidad.

Dos ejemplos de estructura semántica de página

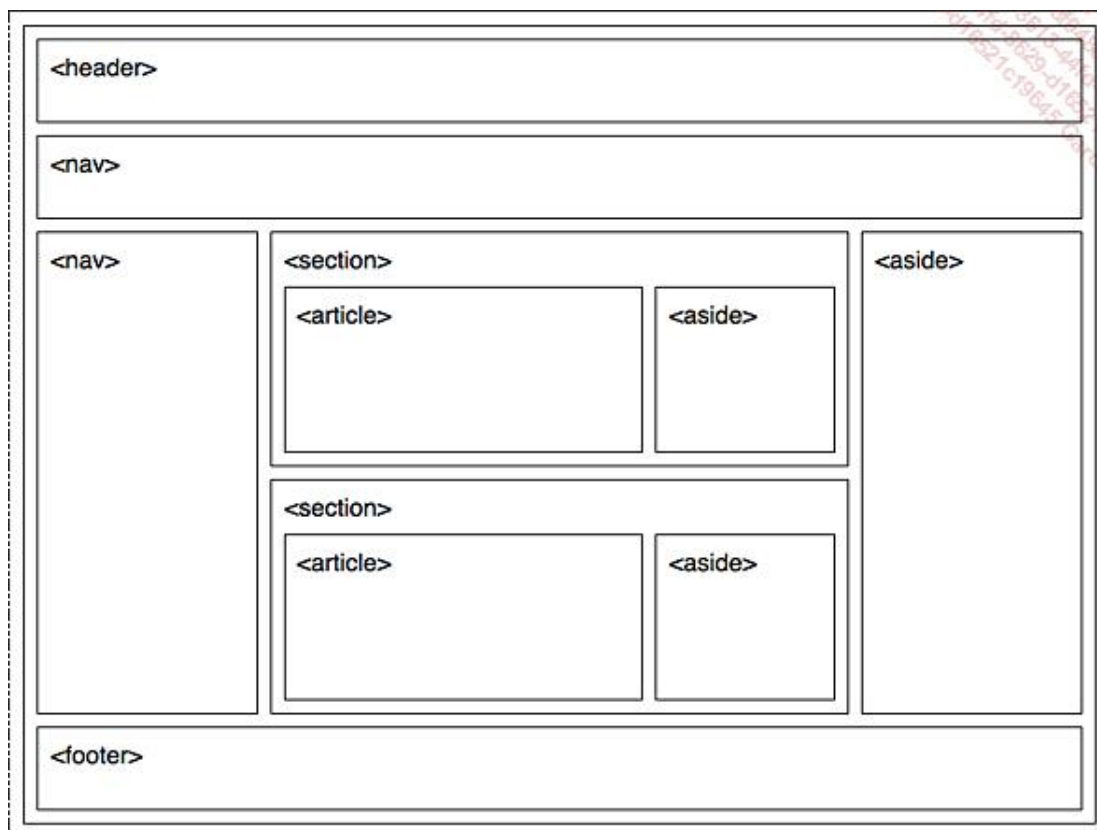
He aquí una estructura semántica muy sencilla con un primer ejemplo.



Tenemos:

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

He aquí una segunda estructura más elaborada.

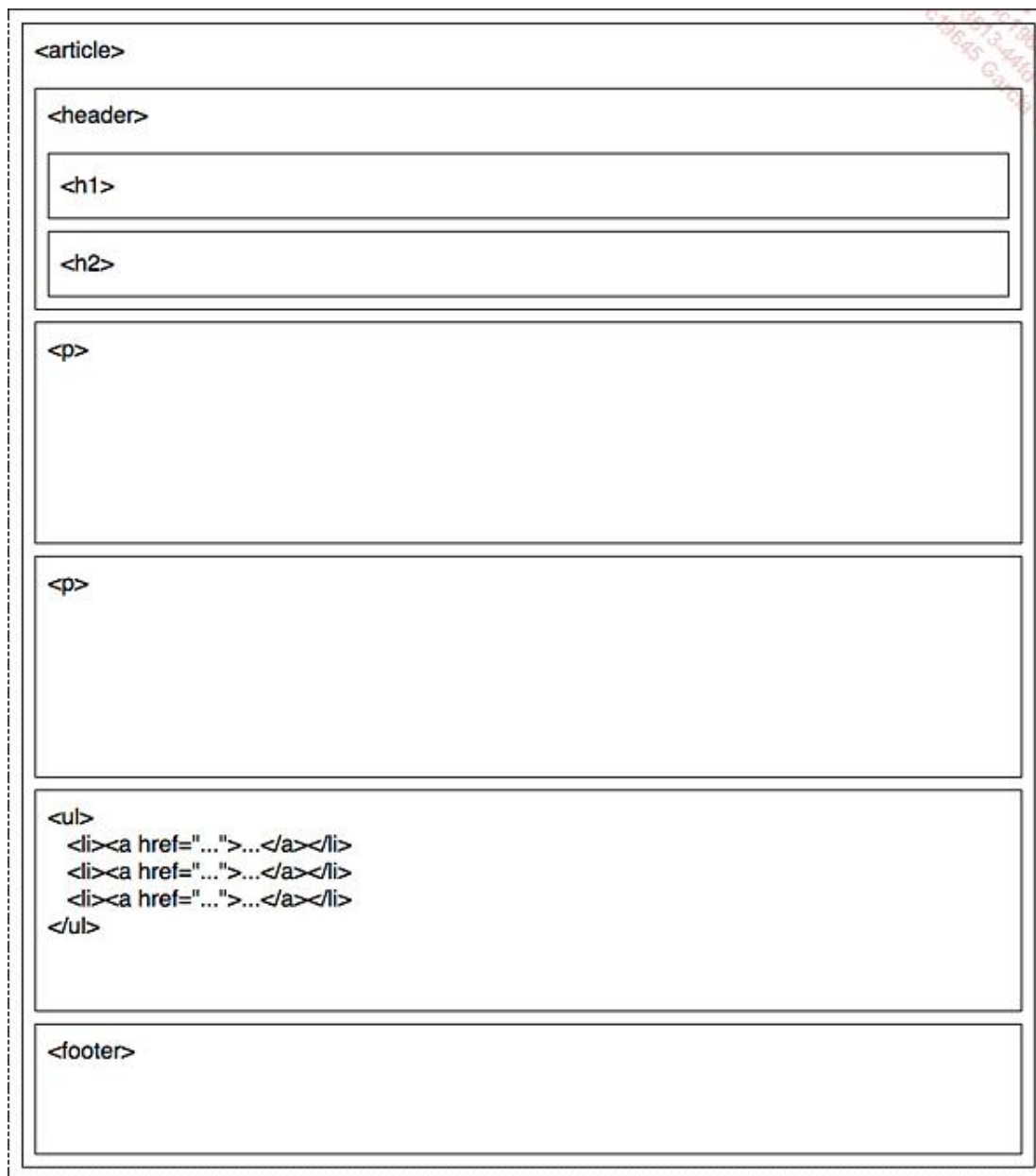


Tenemos:

- Un encabezado `<header>` en la parte superior.
- Debajo, una barra de navegación `<nav>` para la navegación general en el sitio, para navegar de página en página.
- En la parte izquierda, una segunda sección `<nav>` se utiliza para la navegación secundaria, para los enlaces vinculados directamente con el contenido de la página que se está mostrando.
- En la parte derecha, un elemento `<aside>` que muestra información vinculada con el contenido de la página, como enlaces promocionales, contenidos relacionados...
- El contenido de la página se muestra en dos elementos `<section>` que permiten de este modo diferenciar ambos contenidos. Cada `<section>` contiene un elemento `<article>` para el contenido de texto y un elemento `<aside>` para mostrar elementos de información complementaria vinculados con el artículo (iconografía, enlaces...).
- Por último, la página se termina con un pie de página `<footer>` para mostrar la información legal, las condiciones de venta, un enlace de contacto, un plano de acceso...

Un ejemplo de estructura semántica de un artículo

He aquí un ejemplo de un artículo, con contenido de texto, que utiliza estos elementos de estructura semántica.



Tenemos:

- Un elemento `<article>` como contenedor general.
- Nuestros artículos contienen encabezados, a modo de introducción, de modo que utilizaremos el elemento `<header>`. El elemento `<header>` contiene el título `<h1>` del artículo y su subtítulo `<h2>`.
- El contenido de texto del artículo se ubica en los elementos `<p>`. El artículo contiene enlaces hacia artículos complementarios, que se enumeran en una lista ``.
- El artículo termina con un pie de página `<footer>`, o mejor dicho un pie de artículo, con la fecha de publicación, la categoría del artículo y el nombre del autor, por ejemplo.

Objetivos

Cuando incorpore texto en sus páginas web, debe situarlo en contenedores. Incluso aunque sea perfectamente válido no situar el texto en ningún contenedor, conviene evitar estar práctica. En efecto, si lo hace así, no podrá acceder al texto, de modo que no podrá asignarle un formato.

Tiene a su disposición todos los contenedores necesarios: para los títulos, para el texto común, para crear listas...

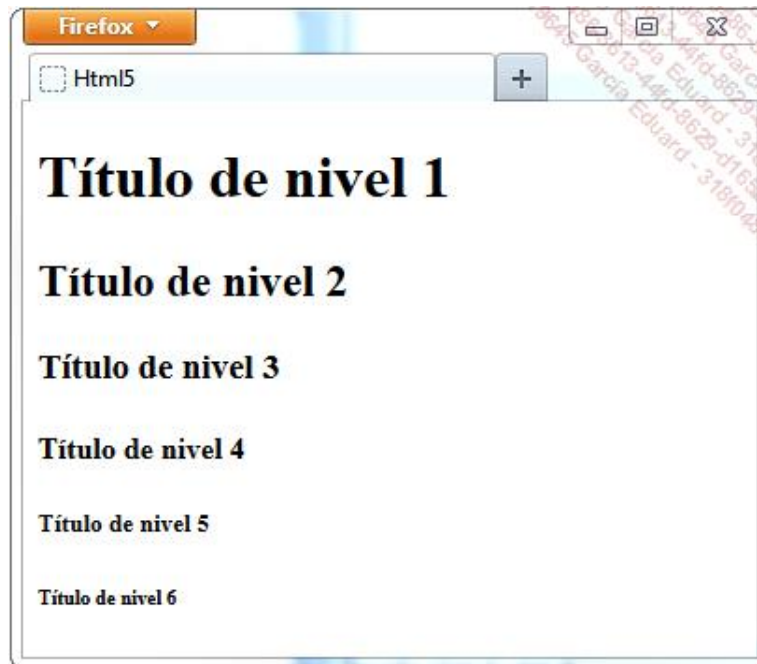
Títulos

¿Qué puede resultar más útil que prever distintos tamaños de título para estructurar el contenido? Basta con pensar en los títulos que identifican una gran sección o una tabla de contenidos.

El lenguaje HTML5 proporciona seis tamaños de letra para los títulos. Cuando se sabe que título en inglés se dice *heading*, la creación de la etiqueta de título se hace evidente: `<h>` por *heading* seguido de un número 1 (para el más grande), 2, 3, 4, 5, 6 (para el más pequeño) y el símbolo `>` de cierre. En adelante, usaremos la notación `<hx>` . . . `</hx>` donde x es un nivel de 1 a 6.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
  <h1>Título de nivel 1</h1>
  <h2>Título de nivel 2</h2>
  <h3>Título de nivel 3</h3>
  <h4>Título de nivel 4</h4>
  <h5>Título de nivel 5</h5>
  <h6>Título de nivel 6</h6>
</body>
</html>
```



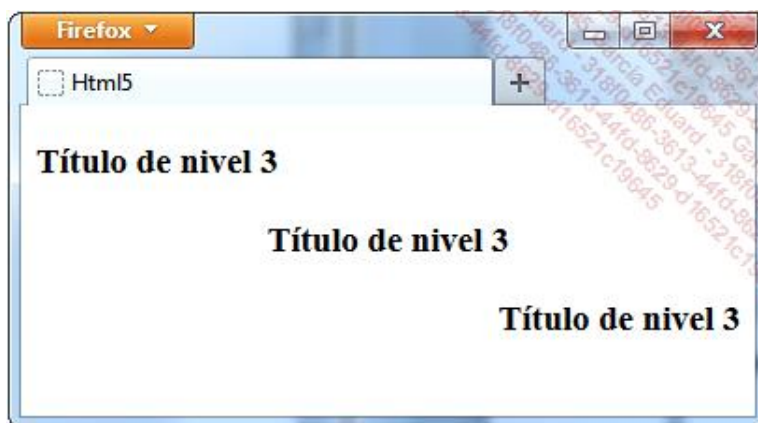
Comentarios

A pesar de la simplicidad de esta etiqueta, se imponen algunas observaciones:

- Esta etiqueta existe desde los comienzos de HTML y es, por tanto, perfectamente compatible con todos los navegadores.
- Por defecto, con la etiqueta `<hx> . . . </hx>`, el texto se escribe en negrita.
- También por defecto, como para todos los elementos de bloque, también llamados elementos de división, la etiqueta incluye una línea vacía entre el contenido de la etiqueta y el resto del documento.
- Para obtener caracteres más grandes o más pequeños que los proporcionados por defecto, o bien un tipo de letra diferente al proporcionado por defecto por el navegador, es preciso usar una hoja de estilo CSS.
- Las etiquetas del título al principio de la página se tienen muy en cuenta en los motores de búsqueda como Google, e influyen de forma importante en la clasificación y recomendación de los sitios web, en particular de aquellos que contienen las palabras clave adecuadas.
- El W3C insiste para que, en HTML5, las etiquetas de título `<hx>` encuentren su objetivo de calidad a la hora de estructurar las páginas web. El uso de las etiquetas de título debería reservarse a su función original. Es importante no desviarlas de su objetivo principal para, por ejemplo, escribir un texto en negrita o con un tamaño de letra determinado. Las hojas de estilo CSS están ahí para realizar estas tareas de presentación.
- El atributo `align`, disponible en HTML 4.0, no se ha mantenido en HTML5. La alineación del párrafo se realiza mediante una propiedad de estilo CSS.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
  <style>
    #centrado { text-align: center;}
    #derecha { text-align: right;}
  </style>
</head>
<body>
  <h3>Título de nivel 3</h3>
  <h3 id="centrado">Título de nivel 3</h3>
  <h3 id="derecha">Título de nivel 3</h3>
</body>
</html>
```



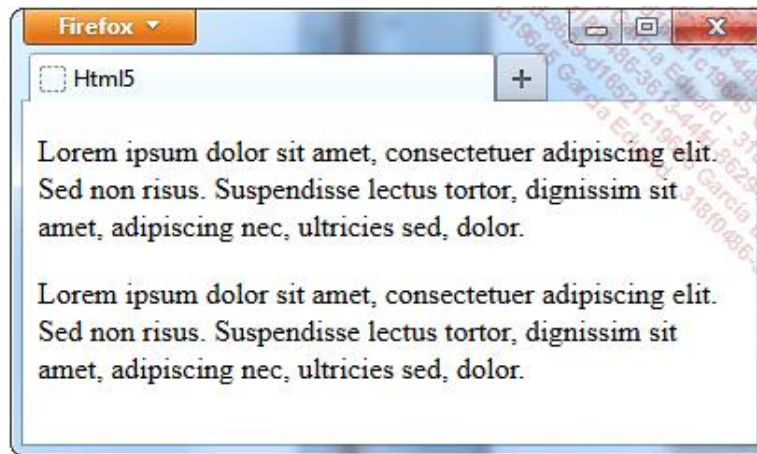
Párrafos

Un contenido de tipo texto gana en legibilidad y comprensión cuando está dividido en distintos párrafos.

Para definir un párrafo, se utiliza la etiqueta `<p> . . . </p>`.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed
non risus. Suspendisse lectus tortor, dignissim sit amet,
adipiscing nec, ultricies sed, dolor.</p>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed
non risus. Suspendisse lectus tortor, dignissim sit amet,
adipiscing nec, ultricies sed, dolor.</p>
</body>
</html>
```



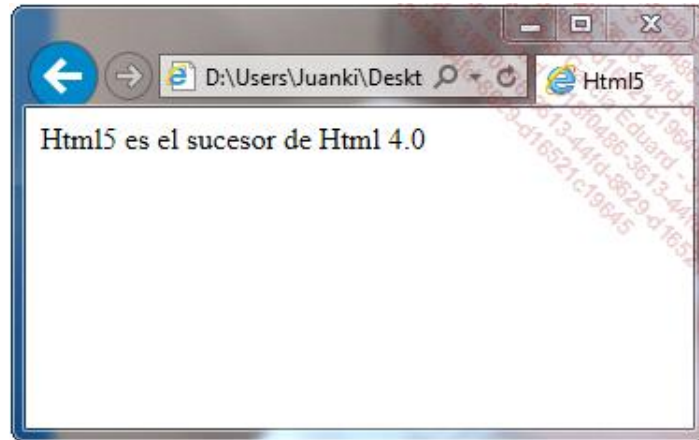
Comentarios

- Esta etiqueta es perfectamente compatible con los navegadores antiguos y modernos.
- Por defecto, cada párrafo está precedido y seguido del espacio correspondiente a un salto de línea para marcar la separación con el contenido anterior y posterior. Este espacio puede modificarse mediante las hojas de estilo.
- El atributo `align` de HTML 4.0 ha desaparecido en la especificación HTML5. La alineación (izquierda, centrado, derecha y justificado) debe realizarse mediante una propiedad de estilo CSS.

En HTML, los retornos de carro o saltos de línea que aparecen en el código se ignoran y se sustituyen por un espacio.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p>Html5 es el sucesor de
Html 4.0 </p>
</body>
</html>
```

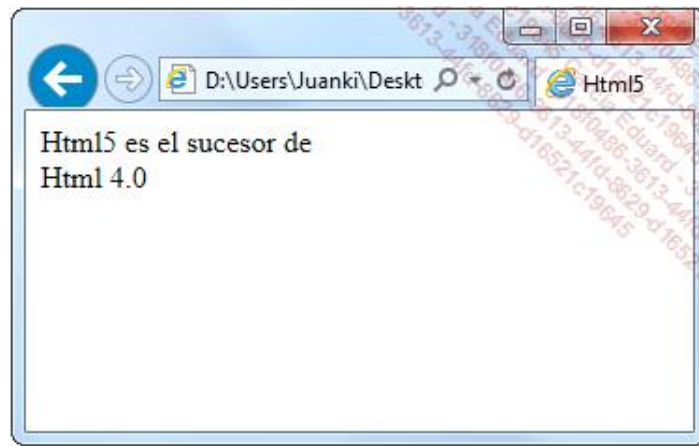


Vemos que el texto HTML ocupa una sola línea.

Para forzar el salto de línea, es preciso usar la etiqueta
.

Nuestro ejemplo queda así:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p>Html5 es el sucesor de<br>
Html 4.0 </p>
</body>
</html>
```



Comentarios

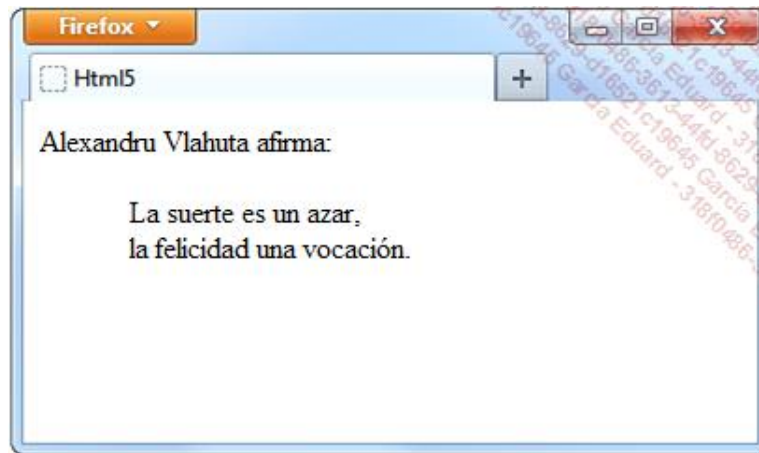
- La etiqueta `
` es una etiqueta llamada única, pues no tiene etiqueta de cierre.
- La notación `
`, usada en XHTML 1.0, también está aceptada en HTML5.

Citas

Las citas se escriben por delante del contenido de tipo texto mediante las etiquetas `<blockquote>` ... `</blockquote>`.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p>Alexandru Vlahuta afirma:</p>
<blockquote>La suerte es un azar,<br>
la felicidad una vocación.</blockquote>
</body>
</html>
```

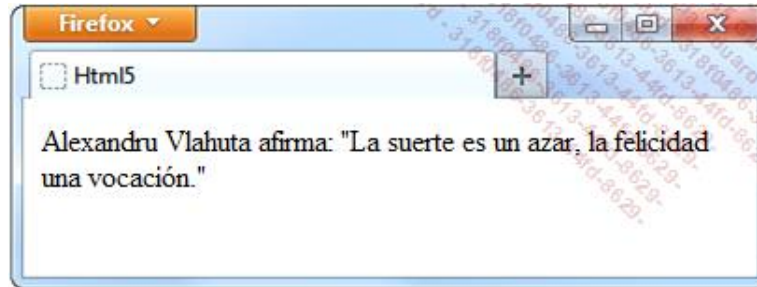


Comentarios

- La cita se visualiza con un salto de línea antes y otro después, igual que con los párrafos.
- Se muestra, a su vez, con un ligero desplazamiento respecto al margen izquierdo. En ocasiones se usa la etiqueta `<blockquote>` para obtener este desplazamiento. Dé preferencia a la propiedad de estilo `margin-left` para conseguir este efecto.
- Existe también la etiqueta `<q>` ... `</q>` para las citas. Esta no incluye espacio antes ni después, ni tampoco desplazamiento a la derecha. Se visualiza entre comillas.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
```

```
<p>Alexandru Vlahuta afirma:  
<q>La suerte es un azar,  
la felicidad una vocación.</q>  
</p>  
</p>  
</body>  
</html>
```



Listas

Las listas, muy conocidas por los usuarios de procesadores de texto, son una forma muy eficaz de estructurar un contenido de tipo texto.

1. Listas ordenadas

La etiqueta ` ... ` escribe una lista ordenada.

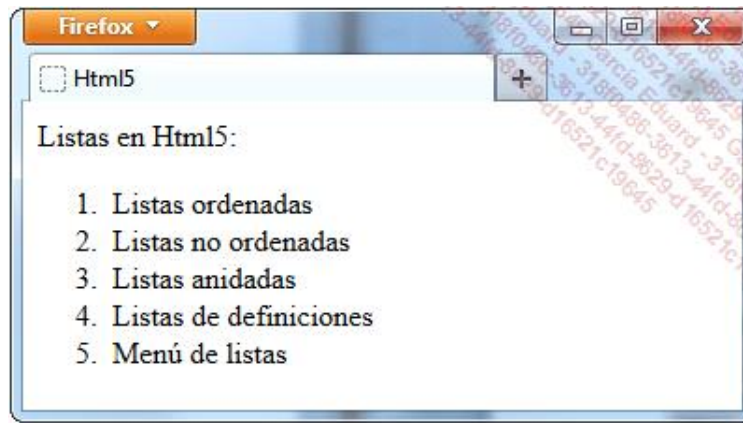
Se definen a continuación, en el interior de estas etiquetas, los elementos de la lista mediante las etiquetas ` ... `.

La estructura general de una lista ordenada o numerada es:

```
<ol>
  <li>Capítulo 1</li>
  <li>Capítulo 2</li>
  <li>Capítulo 3</li>
  <li>Capítulo 4</li>
</ol>
```

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
Listas en Html5:
<ol>
  <li>Listas ordenadas</li>
  <li>Listas no ordenadas</li>
  <li>Listas anidadas</li>
  <li>Listas de definiciones</li>
  <li>Menú de listas</li>
</ol>
</body>
</html>
```



Comentarios

- La etiqueta `` es un clásico de HTML. Se reconoce perfectamente en todos los navegadores.
- El atributo `type`, que permitía reemplazar las cifras por letras (mayúsculas o minúsculas) o números romanos, ya no forma parte de HTML5. Veremos en la parte dedicada a las hojas de estilo CSS cómo modificar la numeración.
- El atributo `start` permite empezar la numeración en una cifra distinta al 1. Este atributo, para los amantes de la precisión, que había sido desvalorizado (*deprecated*) en HTML 4.0, se ha retomado en HTML5.
- El nuevo atributo HTML5 `reversed` permite realizar una numeración descendente (5, 4, 3, 2, 1) en vez de la numeración ascendente habitual (1, 2, 3, 4, 5). En la actualidad, las últimas versiones de Firefox, Chrome y Opera han implementado este atributo.

2. Listas no ordenadas

Las listas no ordenadas, también llamadas listas no numeradas, se escriben mediante la etiqueta ` ... `.

Se definen a continuación, en el interior de esta etiqueta, los elementos de la lista mediante las etiquetas ` ... `.

La estructura general de una lista no ordenada es:

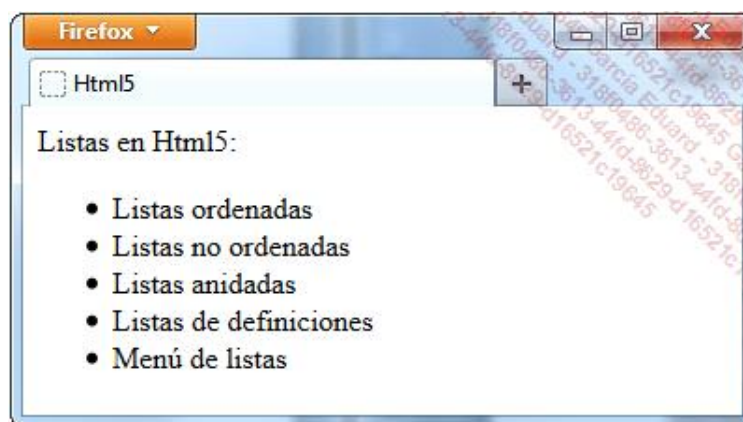
```
<ul>
  <li>Capítulo 1</li>
  <li>Capítulo 2</li>
  <li>Capítulo 3</li>
  <li>Capítulo 4</li>
</ul>
```

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
```

Listas en Html5:

```
<ul>
  <li>Listas ordenadas</li>
  <li>Listas no ordenadas</li>
  <li>Listas anidadas</li>
  <li>Listas de definiciones</li>
  <li>Menú de listas</li>
</ul>
</body>
</html>
```



Comentarios

- La etiqueta `` es un clásico de HTML. Se reconoce perfectamente en todos los navegadores.
- El atributo `type`, que permitía cambiar de algún modo el tipo de marca (redonda llena, redonda vacía, cuadrada), ya no existe en HTML5. Veremos en la parte dedicada a las hojas de estilo CSS cómo modificar la forma de las marcas o cómo reemplazarlas por imágenes.
- La forma de las marcas (disco lleno o disco vacío) la determina el navegador.

3. Listas anidadas

Nada impide elaborar listas ordenadas y listas no ordenadas anidadas. La codificación es simplemente un poco más compleja.

Ejemplo

Una lista numerada en la que se anida una lista no ordenada.

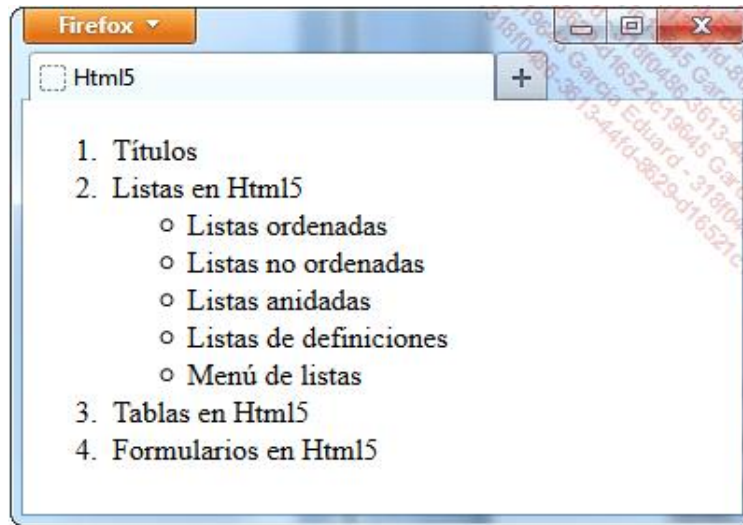
```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<ol>
```



```

<li>Títulos</li>
<li>Listas en Html5
  <ul>
    <li>Listas ordenadas</li>
    <li>Listas no ordenadas</li>
    <li>Listas anidadas</li>
    <li>Listas de definiciones</li>
    <li>Menú de listas</li>
  </ul></li>
<li>Tablas en Html5</li>
<li>Formularios en Html5</li>
</ol>
</body>
</html>

```



4. Listas de definiciones

HTML proporciona un último tipo de lista, particularmente útil para presentar, por ejemplo, glosarios. Esta lista, llamada de definición, presenta una lista de términos, cada uno seguido de su descripción.

La lista se construye en primer lugar mediante la declaración de una lista de definición (*definition list*), es decir, `<dl> ... </dl>`.

Entre estas etiquetas, se declara lo que se llama el término mediante la etiqueta `<dt> ... </dt>`, seguida de su descripción en la etiqueta `<dd> ... </dd>`.

La estructura general de una lista de definición es la siguiente:

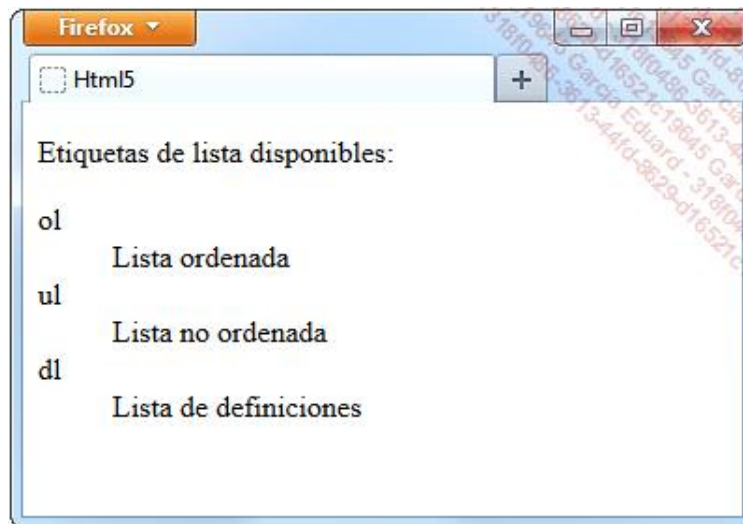
```

<dl>
<dt>Término 1</dt>
  <dd>Definición 1</dd>
<dt>Término 2</dt>
  <dd>Definición 2</dd>
<dt>Término 3</dt>
  <dd>Definición 3</dd>
</dl>

```

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p>Etiquetas de lista disponibles:</p>
<dl>
  <dt>ol</dt>
  <dd>Lista ordenada</dd>
  <dt>ul</dt>
  <dd>Lista no ordenada</dd>
  <dt>dl</dt>
  <dd>Lista de definiciones</dd>
</dl>
</body>
</html>
```

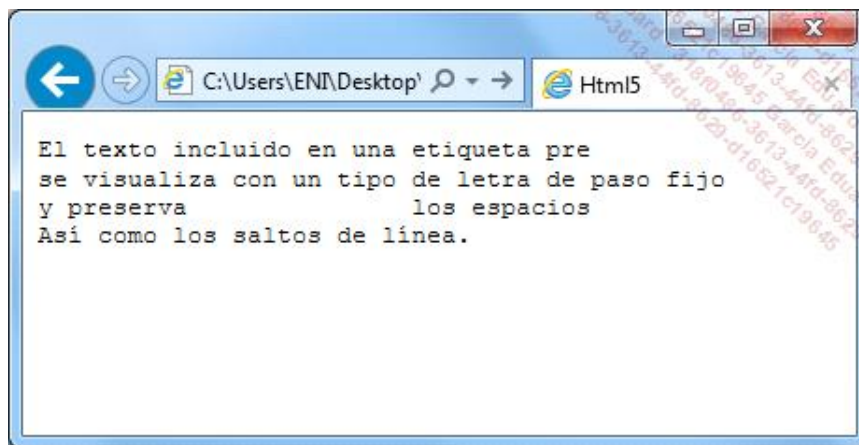


Texto preformateado

La etiqueta `<pre> ... </pre>` permite visualizar el texto tal y como se ha codificado en el editor de texto. Los espacios, tabulaciones y retornos de carro se respetan tal cual en la pantalla. Si incluye texto, se visualizará con un tipo de letra de paso fijo.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<pre>
El texto incluido en una etiqueta pre
se visualiza con un tipo de letra de paso fijo
y preserva          los espacios
Así como los saltos de línea.
</pre>
</body>
</html>
```

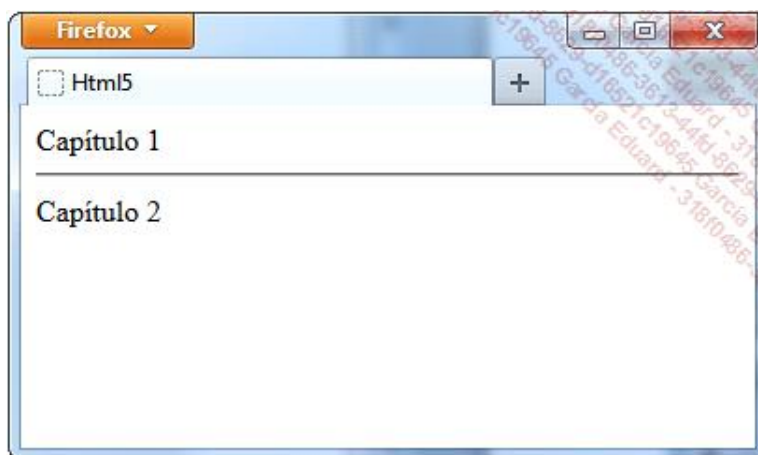


Separadores horizontales

El trazo horizontal puede ser muy útil para definir un cambio en el contenido. La etiqueta `<hr>` realiza esta función.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
  <p>Capítulo 1</p>
  <hr>
  <p>Capítulo 2</p>
</body>
</html>
```



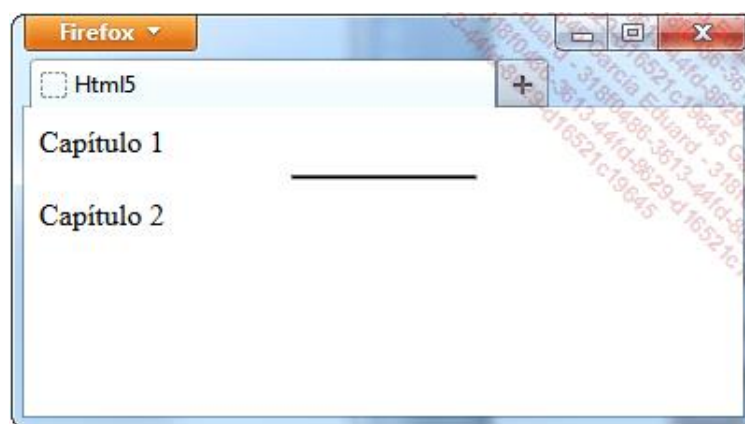
Comentarios

- No existen problemas de compatibilidad con esta etiqueta, que es bien antigua.
- La etiqueta `<hr>` es una etiqueta única que no tiene etiqueta de cierre. HTML5 admite también la notación `<hr />`, derivada de XHTML 1.0.
- Por defecto, esta línea horizontal ocupa toda la longitud de la ventana del navegador.
- Los diseñadores prefieren a menudo reemplazar este trazo, muy básico en su aspecto estético, por una imagen gráficamente más elaborada.
- Todos los atributos `align`, `noshade`, `size`, `width`, desvalorizados (*deprecated*) en HTML 4.0, ya no existen en HTML5. Es preciso usar hojas de estilo para determinar la alineación, el tamaño y la longitud.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
```

```
<title>Html5</title>
<meta charset="UTF-8">
<style>
hr { text-align: center;
    height: 3px;
    background-color: black;
    width: 100px;}
</style>
</head>
<body>
    <p>Capítulo 1</p>
    <hr>
    <p>Capítulo 2</p>
</body>
</html>
```



Dos ejemplos de formato de los contenedores con CSS

Terminaremos este capítulo viendo dos ejemplos rápidos de formato CSS para los dos contenedores `<p>` y `<blockquote>`.

En este primer ejemplo, vamos simplemente a aplicar un fondo gris a un párrafo.

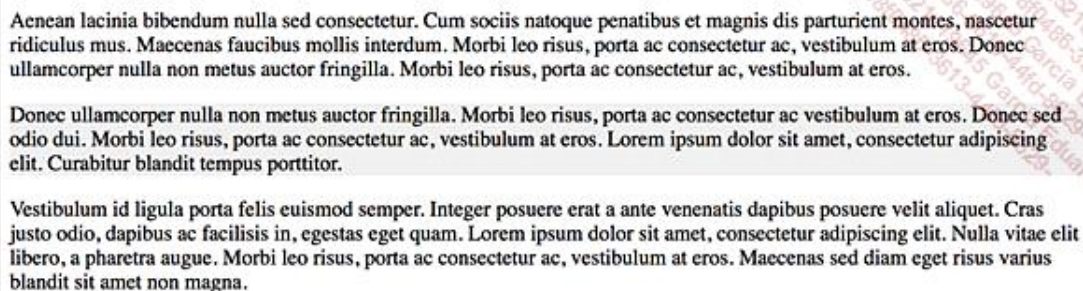
He aquí el código utilizado:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Párrafo con fondo gris</title>
  <meta charset="UTF-8" />
  <link href="#" rel="stylesheet" />
  <style>
    .fondo-gris {
      background-color: #eee;
    }
  </style>
</head>
<body>
<p>Aenean lacinia...</p>
<p class="fondo-gris">Donec ullamcorper...</p>
<p>Vestibulum id ligula porta...</p>
</body>
</html>
```

El archivo de ejemplo se llama C06-01-p.html.

Utilizaremos la propiedad CSS `background-color` para aplicar un fondo coloreado. El color es un gris claro: `#eee`. Aplicaremos a continuación este selector al párrafo deseado: `<p class="fondo-gris">Donec...</p>`.

He aquí su representación:



Aenean lacinia bibendum nulla sed consectetur. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Maecenas faucibus mollis interdum. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Donec ullamcorper nulla non metus auctor fringilla. Morbi leo risus, porta ac consectetur ac, vestibulum at eros.

Donec ullamcorper nulla non metus auctor fringilla. Morbi leo risus, porta ac consectetur ac vestibulum at eros. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur blandit tempus porttitor.

Vestibulum id ligula porta felis euismod semper. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla vitae elit libero, a pharetra augue. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Maecenas sed diam eget risus varius blandit sit amet non magna.

Para el segundo ejemplo, vamos a aplicar un borde a la izquierda de un elemento `<blockquote>`.

He aquí el código utilizado:

```
<!DOCTYPE html>
<html lang="es">
```

```

<head>
  <title>Cita con un borde izquierdo</title>
  <meta charset="UTF-8" />
  <link href="#" rel="stylesheet" />
  <style>
    .borde-recto {
      border-left: 5px solid #333;
      padding-left: 10px;
    }
  </style>
</head>
<body>
<p>Aenean lacinia...</p>
<blockquote class="borde-recto">Donec ullamcorper...</blockquote>
<p>Vestibulum id ligula...</p>
</body>
</html>

```

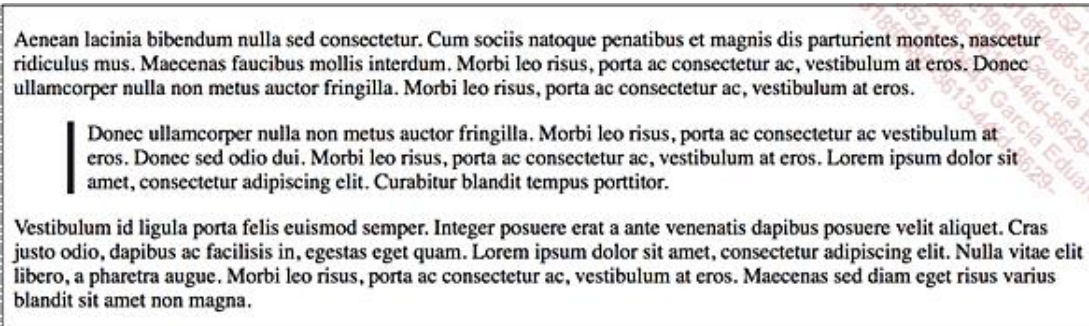
El archivo de ejemplo se llama C06-02-blockquote.html.

Utilizaremos la propiedad `border-left` para crear un borde sobre el lado izquierdo. Este borde mide 5 píxeles de ancho, tiene un trazo continuo y es de color gris.

Utilizaremos a continuación la propiedad `padding-left` para espaciar 10 píxeles el borde del texto.

A continuación, aplicaremos esta regla CSS al bloque de cita.

He aquí la representación obtenida:



Objetivos

En el capítulo anterior, hemos visto los principales contenedores semánticos de texto. En este capítulo, vamos a estudiar el formato semántica del texto. Una vez situados sus textos en contenedores, de tipo párrafo <p> principalmente, podrá aplicar elementos HTML de formato que son también semánticos. Es decir, el formato dará sentido al contenido correspondiente, no es un simple formato gráfico.

Caracteres especiales

Existe toda una maraña de caracteres especiales que es preciso codificar, en ocasiones, de forma especial.

Pensemos en un primer momento en los caracteres que se usan en la codificación HTML5. Por ejemplo, el signo menor que (<) que marca la apertura de una etiqueta. A la hora de leer este código el navegador puede encontrarse con alguna dificultad. El validador de W3C le advierte de esta ambigüedad a la hora de verificar su página.

Un poco como para las palabras reservadas en un lenguaje de programación, existe una lista de elementos que es preciso evitar en la codificación del texto. Estos elementos son, por ejemplo:

- El signo < que indica la apertura de una etiqueta.
- El signo > que marca el cierre de una etiqueta.
- El signo " usado en los atributos de la etiqueta.
- El signo & que marca el inicio de una referencia de carácter.

Los distintos signos deben codificarse usando las entidades siguientes:

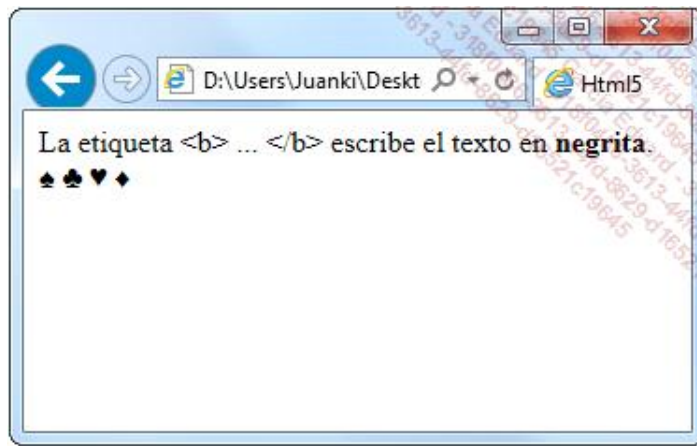
<	<
>	>
"	"
&	&

Preste atención al punto y coma final, pues es obligatorio.

Por otro lado, existe toda una serie de caracteres que no se encuentran en el teclado normal, como el símbolo © de copyright o el símbolo ® de marca registrada. Hay también otros caracteres específicos de las matemáticas, como el signo ∫ para las integrales. Aquellos lectores interesados podrán encontrar toda una larga lista de caracteres especiales usados en HTML en la dirección <http://www.ascii.cl/es/codigos-html.htm>

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
La etiqueta <b> ... </b> escribe el texto en
<b>negrita</b>.<br>
&spades; &clubs; &hearts; &diamonds;
</body>
</html>
```



¿Qué ocurre con los demás caracteres especiales que se usaban en HTML? ¿Qué ha sido en HTML5 de `´` (para la é), de ``` (è), de `ˆ` (ê), de ` ` (espacio indivisible), etc.?

En principio, si el carácter existe en el juego de caracteres (*charset*) declarado en el documento HTML5, no es preciso escribir los caracteres especiales mediante entidades HTML.

Así, en las codificaciones ISO-8859-1, ISO-8859-15 y en UTF-8, la é existe y no es necesario recurrir a la entidad `´`. Esto mejora sin duda la legibilidad del código.

Por el contrario, con la codificación ISO-8859-1, el signo € no existe, puesto que es demasiado antigua para soportar el carácter en su juego de caracteres. En este caso hay que pasar obligatoriamente por el uso de su entidad HTML `€`. Remarquemos que el símbolo € existe en ISO-8859-15 y UTF-8.

Para saber más acerca de este tema, consulte las indicaciones de W3C en la dirección <https://www.w3.org/TR/1999/REC-html401-19991224/charset.html>.

Muchos navegadores realizan una detección automática de la codificación, pero esto puede dar pie a ciertas circunstancias indeseadas. Evite las sorpresas. Es recomendable especificar bien el *charset*.

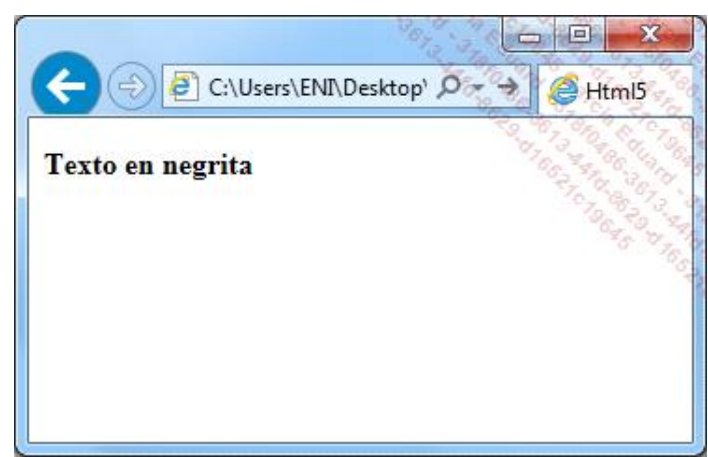
Negrita

El W3C anuncia la separación estricta entre contenido y presentación. HTML se encarga solamente de la estructura del documento, y las hojas de estilo CSS se encargan de todo el aspecto relacionado con el formato visual. Puede parecer, no obstante, paradójico volver a encontrarse etiquetas de presentación. Esté tranquilo: se trata únicamente de algunos formatos básicos del texto, como la negrita, la cursiva, los exponentes y subíndices, etc. Un estudio detallado de las hojas de estilo CSS le permitirá realizar mayores fantasías en la presentación.

Para escribir un texto en negrita (*bold* en inglés), basta con incluirlo entre las etiquetas ` . . . `.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p><b>Texto en negrita</b></p>
</body>
</html>
```



De forma similar, la etiqueta ` . . . ` permite igualmente escribir el texto en negrita en los navegadores visuales.

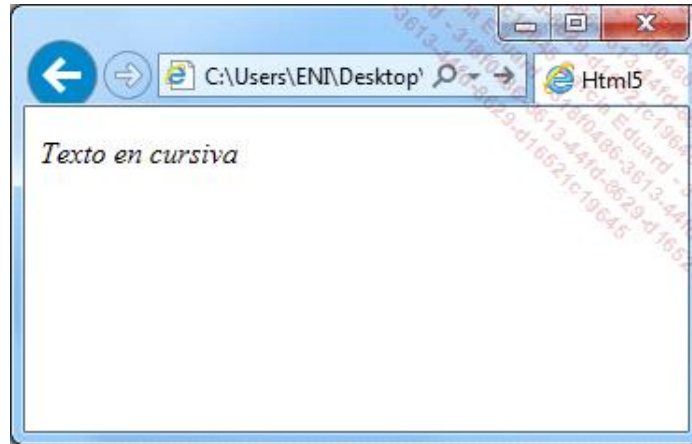
La diferencia entre las etiquetas `` y `` es, no obstante, sensible en HTML. La etiqueta `` provoca un efecto puramente tipográfico y visual. La etiqueta `` tiene como objetivo principal reforzar el texto. Si bien los navegadores visuales han adoptado la misma representación que para la negrita, la interpretación es muy distinta en los programas de síntesis vocal que leen las páginas web a personas con deficiencias visuales. La etiqueta `` refuerza el contenido mediante una entonación de voz distinta o un nivel sonoro más elevado. El sentido que se da al texto con la etiqueta `` es por tanto muy diferente.

Cursiva

Para escribir un texto en cursiva (*italic* en inglés), hay que incluirlo entre las etiquetas `<i>` ... `</i>`.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p><i>Texto en cursiva</i></p>
</body>
</html>
```



Comentarios

- Es posible escribir un texto en negrita y en cursiva enlazando las etiquetas `` e `<i>`. No obstante, sí importa el orden en que se enlazan estas etiquetas. Esto significa que el orden de cierre de las etiquetas ha de ser inverso a su orden de apertura de las mismas. De este modo `<i>texto</i>` está enlazado correctamente. Por el contrario, con `<i>texto</i>` las etiquetas están mezcladas y no se ha respetado la forma de enlazarlas.
- Existe también la etiqueta `` (énfasis). Esta etiqueta también escribe el texto en cursiva en los navegadores. La diferencia con la etiqueta `<i>` está en el sentido que se da a ambas etiquetas. La etiqueta `<i>` es puramente tipográfica y visual. La etiqueta `` tiene como objetivo poner de relieve el texto que la contiene. Este significado visual puede interpretarse de otro modo en los navegadores no visuales.

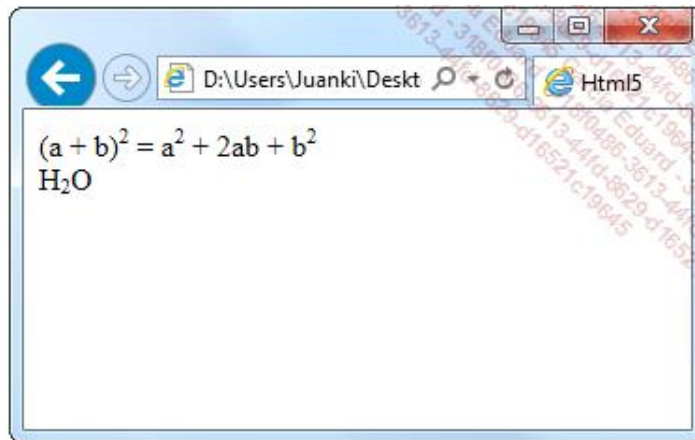
Exponentes y subíndices

En ocasiones es necesario escribir algunos caracteres como exponentes o subíndices, por ejemplo en fórmulas matemáticas o químicas.

La etiqueta `^{...}` escribe el contenido en exponente y la etiqueta `_{...}` en subíndice.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p> (a + b)<sup>2</sup> = a<sup>2</sup> + 2ab + b<sup>2</sup><br>
H<sub>2</sub>O</p>
</body>
</html>
```



Subrayado

El elemento `<u>` permite resaltar una sección del texto sin importancia semántica o poner de relieve un texto mal ortografiado, con una representación subrayada.

El hecho de que la representación estándar se realice con un subrayado puede llevar a confusión con los enlaces de hipertexto, que también están subrayados por defecto. Por ello, el W3C sugiere realizar un uso prudente.

He aquí el código HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Subrayado</title>
  <meta charset="UTF-8" />
</head>
<body>
<p>Donec ullamcorper nulla non metus auctor fringilla.
<u>Donec sed odio dui</u>. Morbi leo risus, porta ac consectetur ac,
vestibulum at eros.</p>
</body>
</html>
```

El archivo correspondiente a este ejemplo se llama C07-01-underline.html.

He aquí la representación:

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

Tachado

En los textos que evolucionan rápidamente, quizá existan secciones que sean incorrectas. Puede, simplemente, suprimir el código, aunque también puede querer mostrarlas a pesar de todo, para informar a los lectores. En este caso, utilice el elemento `<s>`, que indica esta eliminación. La representación por defecto es un texto tachado.

He aquí el código HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Tachado</title>
  <meta charset="UTF-8" />
</head>
<body>
<p>Donec ullamcorper nulla non metus auctor fringilla.
<s>Donec sed odio dui</s>. Morbi leo risus, porta ac consectetur ac,
vestibulum at eros.</p>
</body>
</html>
```

El archivo correspondiente a este ejemplo se llama C07-02-strike.html.

He aquí la representación:

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

Reducir el tamaño de letra

Inicialmente, en HTML 4, el elemento `<small>` permitía escribir con un tamaño de letra más pequeño que el contexto en curso. En HTML5, este elemento se ha redefinido de una manera más semántica, el uso está pensado para las cláusulas de los contratos, las menciones específicas y legales.

La representación es idéntica: el tamaño de letra es más pequeño que el contexto en curso.

Títulos y las citas cortas

Si debe citar una obra (libro, canción, película, ópera...), utilice el elemento `<cite>`. La representación por defecto es en cursiva.

Antes hemos visto el contenedor `<blockquote>` para incluir citas largas en un único elemento HTML. Si tiene que realizar una cita corta en un elemento HTML, `<p>` por ejemplo, utilice el elemento `<q>`. La representación por defecto presenta el texto entre comillas.

He aquí el código HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Títulos y citas cortas</title>
  <meta charset="UTF-8" />
</head>
<body>
<p>En <cite>Recuerdos de egotismo</cite>, Stendhal dice
<q>Todo se puede conocer, salvo a uno mismo</q>.</p>
</body>
</html>
```

El archivo correspondiente a este ejemplo se llama C07-03-q-cite.html.

He aquí la representación obtenida:

En *Recuerdos de egotismo*, Stendhal dice “Todo se puede conocer, salvo a uno mismo”.

Inserciones y supresiones

Las páginas web no son, la mayor parte del tiempo, inmutables, sino que evolucionan, con pequeños elementos que se agregan y se eliminan tras su primera publicación.

Si desea indicar que un texto se ha agregado desde la publicación inicial, utilice el elemento `<ins>`. El texto correspondiente se mostrará subrayado.

Si desea indicar que un texto se ha eliminado tras la publicación inicial, utilice el elemento ``. El texto correspondiente se mostrará tachado.

Dirección del texto

Algunos sistemas de escritura, tales como los alfabetos árabe y hebreo, se escriben de derecha a izquierda, al contrario que el sentido de escritura convencional, de izquierda a derecha, de las lenguas que usan el alfabeto latino (como por ejemplo el castellano). HTML, lenguaje universal, debe tener en cuenta de algún modo esta especificación.

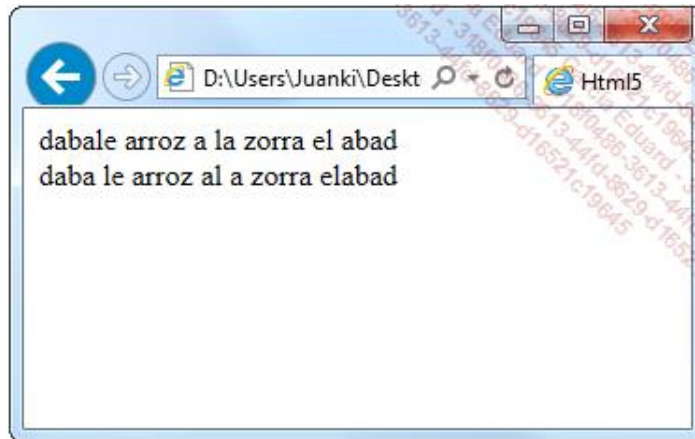
La etiqueta `<bdo> . . . </bdo>` indica el sentido de visualización del texto (de izquierda a derecha o de derecha a izquierda).

Los atributos son:

- `dir="ltr"` (*left to right*) para el sentido de lectura de izquierda a derecha (por defecto).
- `dir="rtl"` (*right to left*) para el sentido de derecha a izquierda.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
<p><bdo dir="ltr">dabale arroz a la zorra el abad</bdo><br>
<bdo dir="rtl">dabale arroz a la zorra el abad</bdo></p>
</body>
</html>
```



Otros formatos semánticos

HTML5 le proporciona otros elementos para aplicar un formato semántico. Estos elementos deben utilizarse dentro de un contenedor de texto, como un párrafo `<p>`, por ejemplo.

- `<dfn>`: permite destacar un término definido en el contexto en curso. No hay un formato por defecto, cada navegador es libre. Sin embargo, la mayoría de las veces se aplica la cursiva.
- `<abbr>`: permite destacar una abreviatura. Puede utilizar el atributo global `title` para indicar la etiqueta completa de la abreviatura.
- `<code>`: permite destacar un código informático. El formato se realiza generalmente con un tipo de letra de paso fijo de tipo Courier.
- `<var>`: permite destacar una variable matemática o informática. El formato se realiza en cursiva.
- `<samp>`: permite destacar un ejemplo o una muestra (*sample*, en inglés). El formato se realiza generalmente con un tipo de letra de paso fijo de tipo Courier.
- `<kbd>`: permite destacar un texto que debe introducirse por teclado, una tecla o una combinación de teclas. El formato se realiza generalmente con un tipo de letra de paso fijo de tipo Courier.
- `<time>`: permite destacar información de fecha y de hora.
- `<mark>`: permite destacar una sección de texto importante, con un formato subrayado con fondo amarillo.
- `<meter>`: permite insertar un indicador que muestra un valor numérico definido sobre una escala específica.
- `<progress>`: permite insertar una barra de progreso, que muestra un porcentaje de avance.
- `<summary>/<details>`: permite mostrar los detalles (con el elemento `<details>`), ocultos por defecto, de un elemento que se muestra en `<summary>`.

He aquí un ejemplo de código HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Detalles de un elemento</title>
  <meta charset="UTF-8" />
</head>
<body>
<details>
<summary>Lista de formatos de imagen</summary>
<ul>
  <li>GIF</li>
  <li>JPEG</li>
  <li>PNG</li>
</ul>
</details>
</body>
</html>
```

El archivo correspondiente a este ejemplo se llama C07-04-summary-details.html.

He aquí la representación inicial:

► Lista de formatos de imagen

He aquí la representación una vez que el usuario ha hecho clic en el triángulo para mostrar:

▼ Lista de formatos de imagen

- GIF
- JPEG
- PNG