

## Capítulo 3. Texto

La mayor parte del contenido de las páginas HTML habituales está formado por texto, llegando a ser más del 90% del código de la página. Por este motivo, es muy importante conocer los elementos y etiquetas que define HTML para el manejo del texto.

El lenguaje HTML incorpora al tratamiento del texto muchas de las ideas y normas establecidas en otros entornos de publicación de contenidos. De esta forma, HTML define etiquetas para **estructurar** el contenido en secciones y párrafos y define otras etiquetas para **marcar** elementos importantes dentro del texto.

La tarea inicial del editor de contenidos HTML consiste en estructurar el texto original definiendo sus párrafos, titulares y títulos de sección, como se muestra en la siguiente imagen:

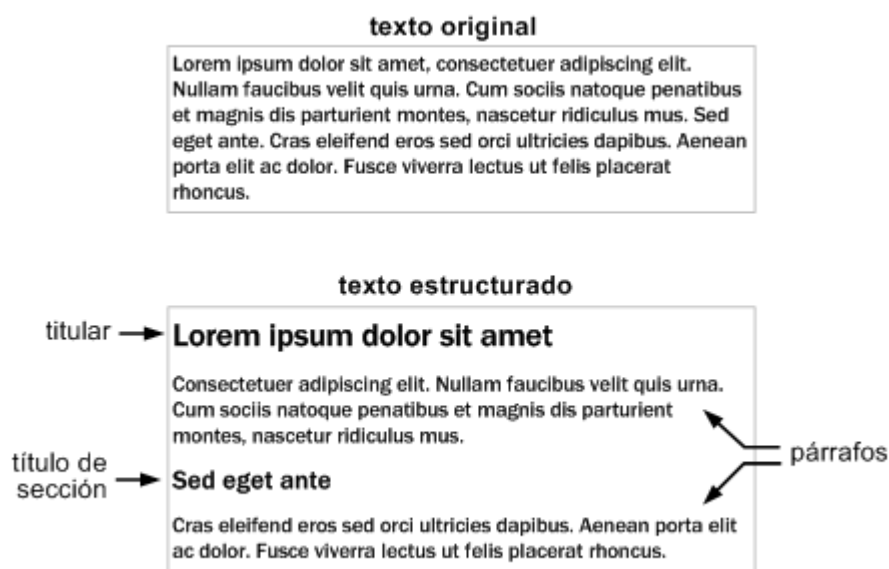


Figura 3.1. Resultado de estructurar un texto sencillo

El proceso de estructurar un texto simple consiste en indicar las diferentes zonas o secciones que componen el texto. De esta forma, los textos estructurados utilizan etiquetas para delimitar cada párrafo y títulos de sección para delimitar cada una de las secciones que forman el texto.

Una vez definida la estructura básica de los contenidos de la página, el siguiente paso consiste en marcar los diferentes elementos dentro del propio texto: definiciones, abreviaturas, textos importantes, textos modificados, citas a otras referencias, etc.

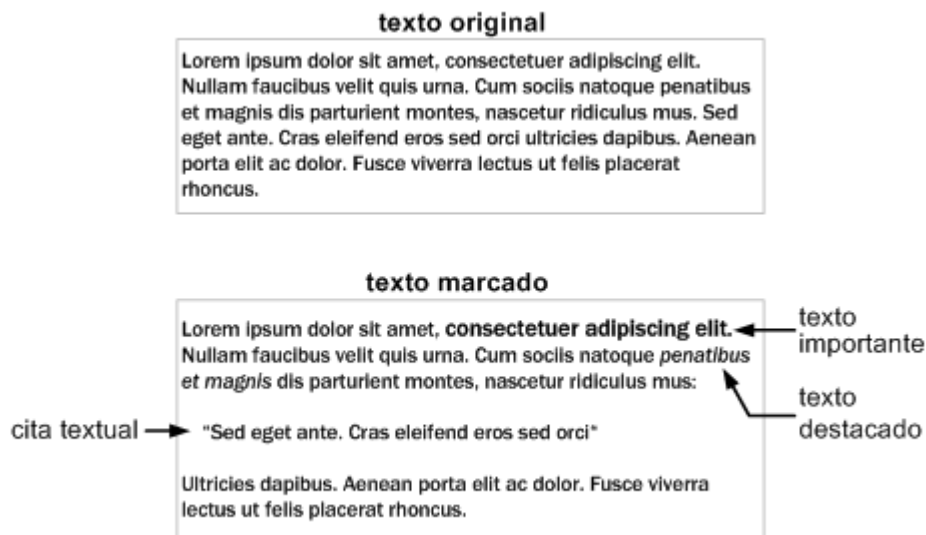


Figura 3.2. Resultado de marcar un texto sencillo

El anterior ejemplo muestra la transformación de un párrafo con un texto simple en un párrafo cuyo texto contiene elementos marcados de forma especial. Así, algunas palabras del texto se muestran en negrita porque se consideran importantes; otras palabras aparecen en cursiva, ya que se han marcado como destacadas e incluso una frase aparece tabulada y entre comillas, indicando que es una cita textual de otro contenido.

En las secciones siguientes se muestran todas las etiquetas que define HTML para estructurar y marcar el texto. Además, se hace una mención especial al tratamiento que hace HTML de los espacios en blanco y las nuevas líneas.

## 3.1. Estructurar

La forma más sencilla de estructurar un texto consiste en separarlo por párrafos. Además, HTML permite incluir títulos que delimitan cada una de las secciones.

### 3.1.1. Párrafos

Una de las etiquetas más utilizadas de HTML es la etiqueta `<p>`, que permite definir los párrafos que forman el texto de una página. Para delimitar el texto de un párrafo, se encierra ese texto con la etiqueta `<p>`, como muestra el siguiente ejemplo:

```
<html>

<head>
<title>Ejemplo de texto estructurado con párrafos</title>
</head>

<body>
<p>Este es el texto que forma el primer párrafo de la página.
Los párrafos pueden ocupar varias líneas y el navegador se
encarga
```

de ajustar su longitud al tamaño de la ventana.</p>

<p>El segundo párrafo de la página también se define encerrando su texto con la etiqueta p. El navegador también se encarga de separar automáticamente cada párrafo.</p>  
</body>

</html>

El ejemplo anterior se visualiza de la siguiente manera en cualquier navegador:

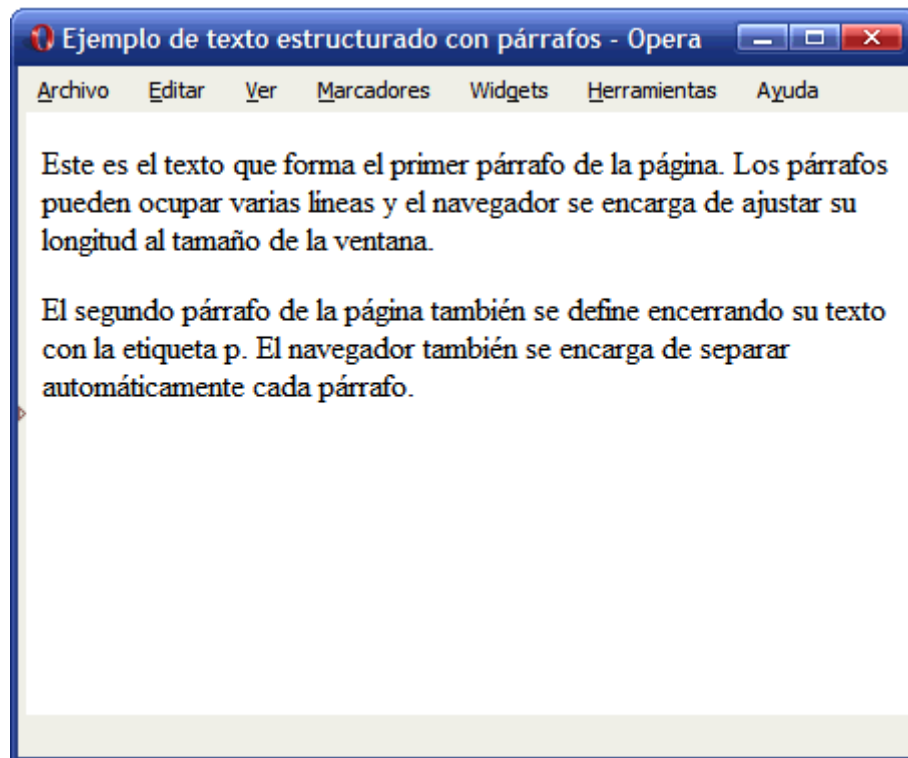


Figura 3.3. Ejemplo de texto HTML estructurado con párrafos

La siguiente tabla recoge la definición formal de la etiqueta <p>:

<p>	Párrafos
Atributos comunes	básicos, i18n y eventos
Atributos específicos	-
Tipo de elemento	Bloque
Descripción	Delimita el contenido de un párrafo de texto

Los párrafos creados con HTML son elementos de bloque, por lo que siempre ocupan toda la anchura de la ventana del navegador. Además, no tienen atributos específicos, pero sí que se les pueden asignar los atributos comunes de HTML básicos, de internacionalización y de eventos.

## 3.1.2. Secciones

Las páginas HTML habituales suelen tener una estructura más compleja que la que se puede crear solamente mediante párrafos. De hecho, es habitual que las páginas se dividan en diferentes secciones jerárquicas.

Los títulos de sección se utilizan para delimitar el comienzo de cada sección de la página. HTML permite crear secciones de hasta seis niveles de importancia. De esta forma, aunque una página puede definir cualquier número de secciones, sólo puede incluir seis niveles jerárquicos.

Las etiquetas que definen los títulos de sección son `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` y `<h6>`. La etiqueta `<h1>` es la de mayor importancia y por tanto se utiliza para definir los titulares de la página. La importancia del resto de etiquetas es descendiente, de forma que la etiqueta `<h6>` es la que se utiliza para delimitar las secciones menos importantes de la página.

A continuación se muestra la definición formal de la etiqueta `<h1>`, siendo idéntica la definición del resto de etiquetas referidas a los títulos de sección:

<b>&lt;h1&gt;</b>	Sección (titular) de nivel 1
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Define los títulos de las secciones de mayor importancia de la página.

Al igual que la etiqueta `<p>`, las etiquetas de título de sección son elementos de bloque y no tienen atributos específicos.

Las etiquetas `<h1>`, ..., `<h6>` definen títulos de sección, no secciones completas. Por este motivo, no es necesario encerrar los contenidos de una sección con su etiqueta correspondiente. Solamente se debe encerrar con las etiquetas `<h1>`, ..., `<h6>` los títulos de cada sección.

El siguiente ejemplo muestra el uso de las etiquetas de título de sección:

```
<html>

<head>
<title>Ejemplo de texto estructurado con secciones</title>
</head>

<body>
<h1>Titular de la página</h1>
```

```
<p>Párrafo de introducción...</p>

<h2>La primera sub-sección</h2>

<p>Párrafo de contenido...</p>

<h2>Otra subsección</h2>

<p>Más párrafos de contenido...</p>
</body>

</html>
```

Los navegadores muestran el ejemplo anterior de la siguiente manera:

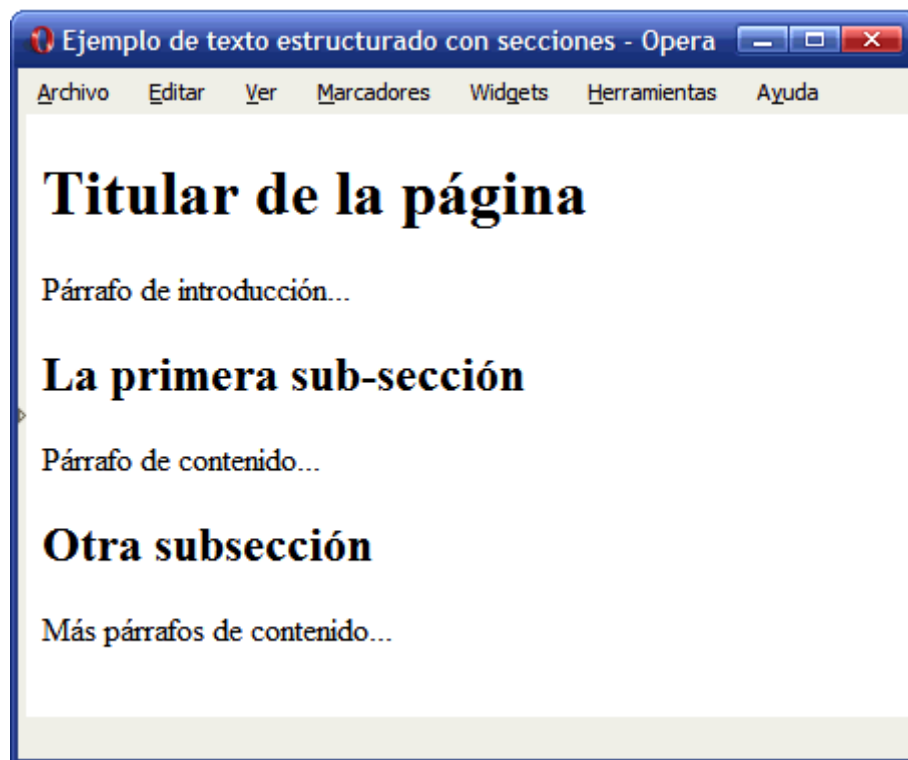


Figura 3.4. Ejemplo de texto HTML estructurado con párrafos y secciones

Los navegadores asignan de forma automáticamente el tamaño del título de cada sección en función de su importancia. Así, los títulos de sección `<h1>` se muestran con el tamaño de letra más grande, ya que son el nivel jerárquico superior, mientras que los títulos de sección `<h6>` se visualizan con un tamaño de letra muy pequeño, adecuado para el nivel jerárquico de menor importancia.

Evidentemente, el aspecto que los navegadores aplican por defecto a los títulos de sección se puede modificar utilizando las hojas de estilos de CSS. La siguiente imagen muestra el tamaño por defecto con el que los navegadores muestran cada titular:

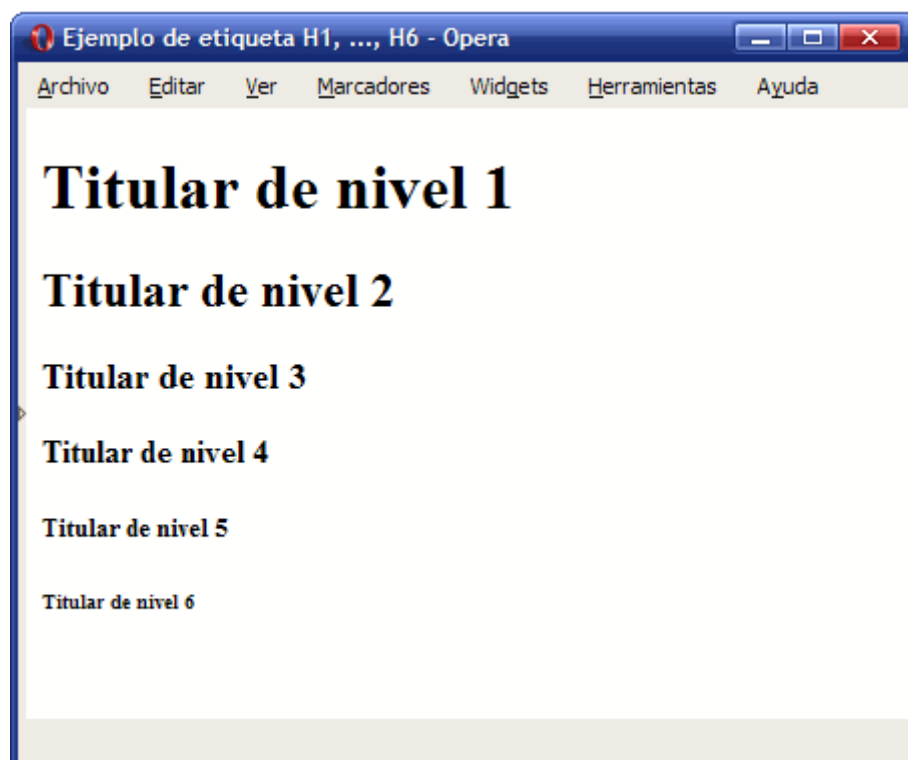


Figura 3.5. Ejemplo de uso de las etiquetas h1, h2, h3, h4, h5 y h6

## 3.2. Marcado básico de texto

Una vez estructurado el texto en párrafos y secciones, el siguiente paso es el marcado de los elementos que componen el texto. Los textos habituales están formados por elementos como palabras en negrita o cursiva, anotaciones y correcciones, citas a otros documentos externos, etc. HTML proporciona varias etiquetas para *marcar* cada uno de los diferentes tipos de texto.

Entre las etiquetas más utilizadas para marcar texto se encuentran `<em>` y `<strong>`. La definición formal de estas dos etiquetas se muestra a continuación:

<b><code>&lt;em&gt;</code></b>	Énfasis
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Realza la importancia del texto que encierra
<b><code>&lt;strong&gt;</code></b>	Énfasis más acentuado
<b>Atributos comunes</b>	básicos, i18n y eventos

<b>&lt;strong&gt;</b>	Énfasis más acentuado
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Realza con la máxima importancia el texto que encierra

La etiqueta `<em>` marca un texto indicando que su importancia es mayor que la del resto del texto. La etiqueta `<strong>` indica que un determinado texto es de la mayor importancia dentro de la página. Ejemplo:

```
<html>
<head>
<title>Ejemplo de etiqueta em y strong</title>
</head>

<body>
<p>El lenguaje HTML permite marcar algunos segmentos de texto
como <em>muy importantes</em> y otros segmentos como
<strong>los
más importantes</strong>.</p>
</body>
</html>
```

Por defecto, los navegadores muestran los elementos `<em>` en cursiva para hacer evidente su importancia y muestran los elementos `<strong>` en negrita, para indicar que son los más importantes:

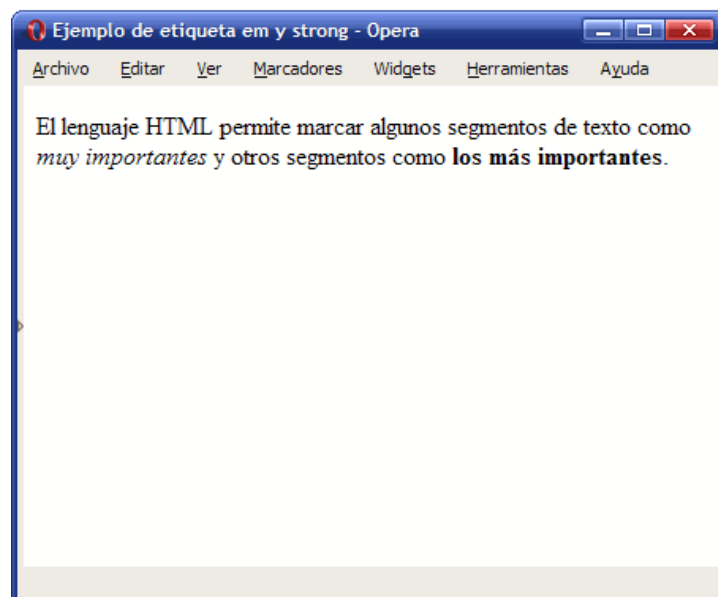


Figura 3.6. Ejemplo de uso de las etiquetas em y strong

## Ejercicio 2

Estructurar y marcar el siguiente texto [extraído de la Wikipedia](#) para que el navegador lo muestre con el aspecto de la siguiente imagen:

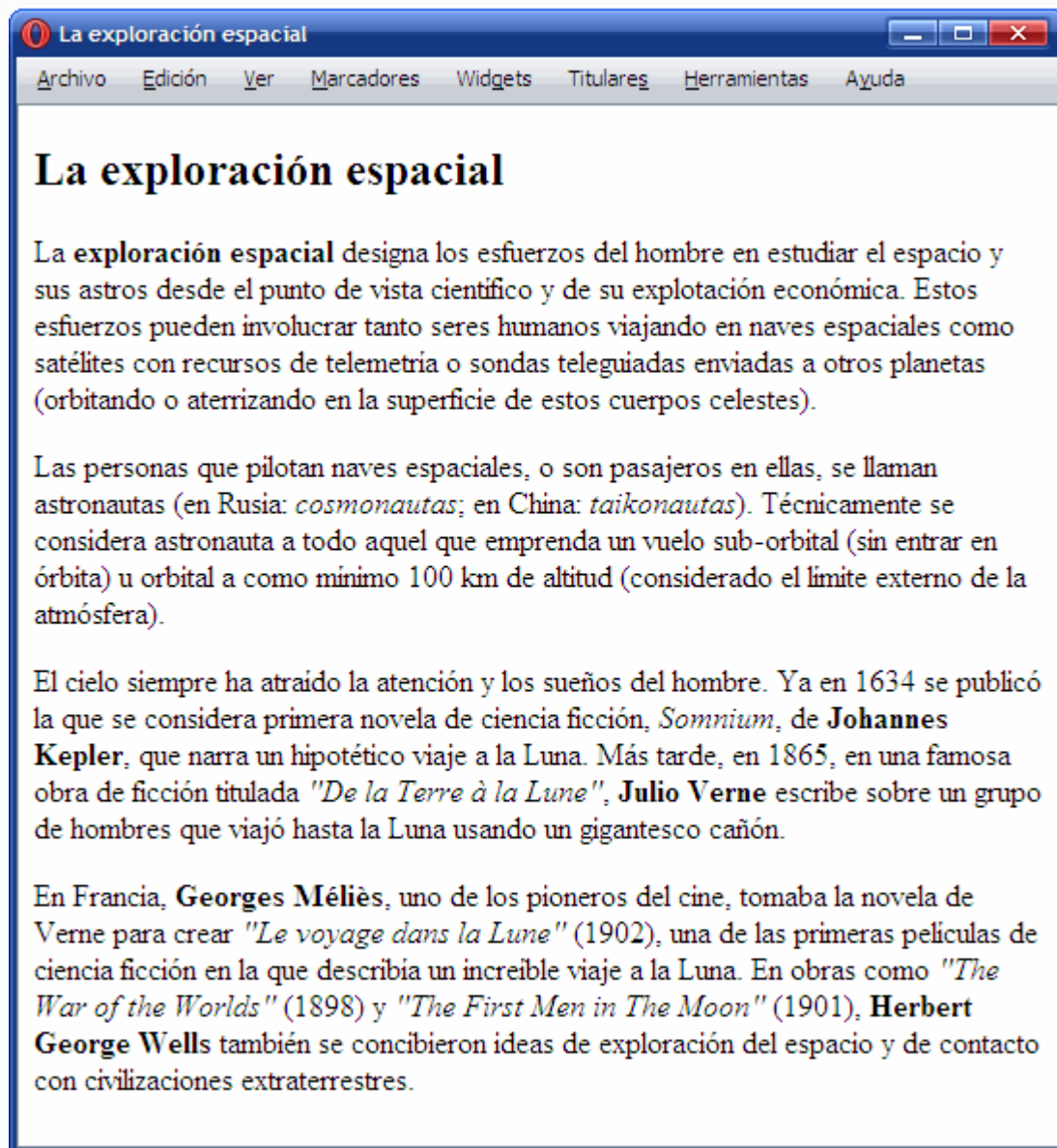


Figura 3.7. Resultado de estructurar y marcar el texto original

HTML también permite marcar de forma adecuada las modificaciones realizadas en el contenido de una página. En otras palabras, HTML permite indicar de forma clara el texto que ha sido eliminado y el texto que ha sido añadido a un determinado texto original. Las etiquetas utilizadas son `<ins>` y `<del>`, cuya definición formal es la siguiente:



<b>&lt;ins&gt;</b>	Insertión
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<p><a href="#">cite = "url"</a> - Indica la URL de la página en la que se puede obtener más información sobre el motivo por el que se realizó la modificación.</p> <p><a href="#">datetime = "fecha"</a> - Especifica la fecha y hora en la que se realizó el cambio</p>
<b>Tipo de elemento</b>	Bloque y en línea
<b>Descripción</b>	Se emplea para marcar una modificación en los contenidos originales consistente en la inserción de un nuevo contenido
<b>&lt;del&gt;</b>	Borrado
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<p><a href="#">cite = "url"</a> - Indica la URL de la página en la que se puede obtener más información sobre el motivo por el que se realizó la modificación.</p> <p><a href="#">datetime = "fecha"</a> - Especifica la fecha y hora en la que se realizó el cambio</p>
<b>Tipo de elemento</b>	Bloque y en línea
<b>Descripción</b>	Se emplea para marcar una modificación en los contenidos originales consistente en el borrado de cierto contenido

Las dos etiquetas cuentan con los mismos atributos específicos, que opcionalmente se pueden añadir para proporcionar más información sobre los cambios realizados. El atributo [cite](#) se emplea para indicar la dirección de un documento externo en el que se puede encontrar más información relacionada con la inserción o el borrado de texto. El atributo [datetime](#) puede utilizarse para indicar la fecha y la hora en la que se realizó cada cambio.

Ejemplo:

```
<html>
<head><title>Ejemplo de etiqueta ins y del</title></head>
<body>

<h3>Ejemplo de etiqueta ins y del</h3>

<p>El HTML, acrónimo inglés de Hyper Text Markup Language
(lenguaje de
<del datetime="20091025"
cite="http://www.librosweb.es/mas_informacion.html">marcado de
hipertexto</del> <ins datetime="20091026"
cite="http://www.librosweb.es/mas_informacion.html">
marcas hipertextuales</ins>) es un lenguaje de marcación
diseñado para estructurar textos y
presentarlos en forma de hipertexto.</p>

</body>
</html>
```

Los navegadores muestran el ejemplo anterior de la siguiente manera:

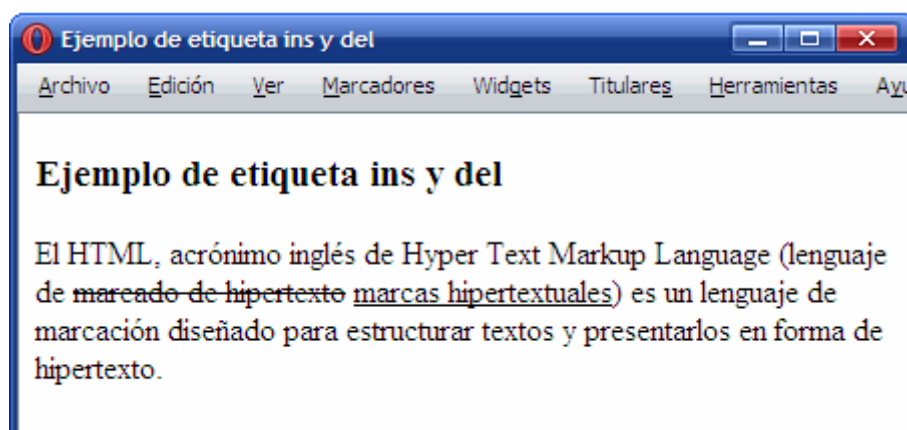


Figura 3.8. Ejemplo de uso de las etiquetas ins y del

Por defecto, el texto eliminado (marcado con la etiqueta `<del>`) se muestra tachado de forma que el usuario pueda identificarlo fácilmente como un texto que formaba parte del texto original y que ya no tiene validez. El texto insertado (marcado con la etiqueta `<ins>`) se muestra subrayado, de forma que el usuario pueda identificarlo como un texto nuevo que no formaba parte del texto original.

Por otra parte, en muchos tipos de páginas (artículos, noticias) es habitual citar literalmente un texto externo. HTML define la etiqueta `<blockquote>` para incluir citas textuales en las páginas web. La definición de la etiqueta HTML con el nombre más largo se muestra a continuación:

<b>&lt;blockquote&gt;</b>	Citas
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<a href="#">cite = "url"</a> - Indica la dirección de la página web original de la que se extrae la cita
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para indicar que el texto que encierra es una cita textual de otro texto externo

Al igual que `<ins>` y `<del>`, la etiqueta `<blockquote>` permite indicar mediante el atributo `cite` la dirección de un documento del que se ha extraído la cita. Ejemplo:

```
<html>
<head><title>Ejemplo de etiqueta blockquote</title></head>

<body>
<p>Según el W3C, el valor del
atributo <em>cite</em> en las etiquetas
<strong>blockquote</strong> tiene el
siguiente significado:</p>

<blockquote
cite="http://www.w3.org/TR/html401/struct/text.html">"El valor
de este atributo
es una dirección URL que indica el documento original de la
cita."</blockquote>
</body>

</html>
```

El aspecto que muestra el ejemplo anterior en cualquier navegador es el siguiente:

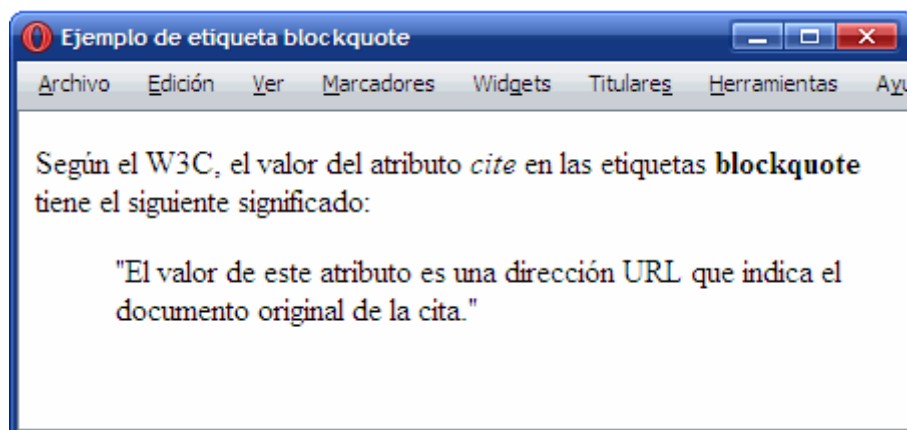


Figura 3.9. Ejemplo de uso de la etiqueta blockquote

Para indicar de forma clara que el texto es una cita externa, los navegadores muestran por defecto el texto del elemento `<blockquote>` con un gran margen en la parte izquierda.

## 3.3. Marcado avanzado de texto

Las páginas y documentos más avanzados suelen incluir otros elementos importantes que se deben marcar de forma adecuada. Por ello, HTML incluye muchas otras etiquetas que permiten marcar más elementos del texto.

La etiqueta `<abbr>` marca las abreviaturas de un texto y la etiqueta `<acronym>` se emplea para marcar las siglas o acrónimos del texto. Su definición es la siguiente:

<b>&lt;abbr&gt;</b>	Abreviaturas
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<code>title = "texto"</code> - Indica el significado completo de la abreviatura
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Se emplea para marcar las abreviaturas del texto y proporcionar el significado de esas abreviaturas
<b>&lt;acronym&gt;</b>	Acrónimos o siglas
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<code>title = "texto"</code> - Indica el significado completo del acrónimo o sigla
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Se emplea para marcar las siglas o acrónimos del texto y proporcionar el significado de esas siglas

En ambos casos, el atributo `title` se puede utilizar para incluir el significado completo de la abreviatura o sigla. Ejemplo:

```

<html>

<head>
<title>Ejemplo de etiqueta acronym</title>
</head>

<body>
<p>El lenguaje <acronym title="HyperText Markup
Language">HTML</acronym> es estandarizado
por el <acronym title="World Wide Web
Consortium">W3C</acronym>.</p>
</body>

</html>

```

La mayoría de navegadores muestran por defecto un borde inferior punteado para todos los elementos `<abbr>` y `<acronym>`. Al posicionar el puntero del ratón sobre la palabra subrayada, el navegador muestra un pequeño recuadro (llamado *tooltip* en inglés) con el valor del atributo `title`:

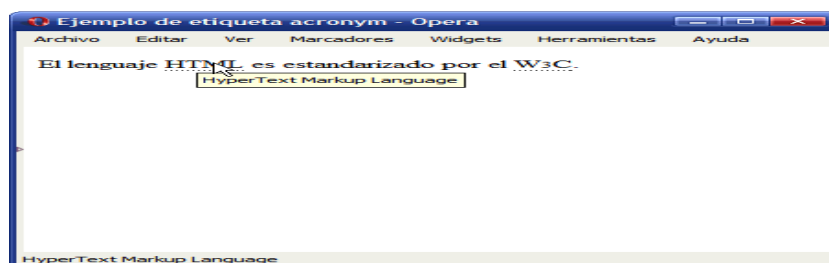


Figura 3.10. Ejemplo de uso de la etiqueta acronym

Por otra parte, en ocasiones resulta útil incluir la definición de una palabra extraña o cuyo uso está restringido a un entorno muy determinado. HTML incluye la etiqueta `<dfn>` para proporcionar al usuario la definición de todas las palabras para las que se considere apropiado. La definición formal de esta etiqueta se muestra a continuación:

<b>&lt;dfn&gt;</b>	Definición
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<code>title = "texto"</code> - Indica el significado completo del término
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Se emplea para marcar las definiciones de ciertos términos y proporcionar el significado de esos términos

El siguiente ejemplo muestra cómo se utiliza la etiqueta `<dfn>` para incluir la definición completa de una palabra cuyo uso no es habitual fuera de los ámbitos médicos y psicológicos:

`<p>Con estos síntomas, podría tratarse de un caso de <dfn title="Imagen o sensación subjetiva, propia de un sentido, determinada por otra sensación que afecta a un sentido diferente">sinestesia</dfn></p>`

Por último, HTML incluye una etiqueta que se puede utilizar para marcar un texto como una citación:

<b><code>&lt;cite&gt;</code></b>	Cita
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Se emplea para marcar una cita o una referencia a otras fuentes

En ocasiones, no está clara la diferencia entre `<cite>` y `<blockquote>`. El elemento `<cite>` marca el autor de la cita (persona, documento, etc.) y `<blockquote>` marca el contenido de la propia cita. En el siguiente ejemplo, `<blockquote>` encierra el contenido de una frase célebre y `<cite>` encierra el nombre de su autor:

Como dijo `<cite>Mahatma Gandhi</cite>`:  
`<blockquote>Vive como si fueras a morir mañana y aprende como si fueras a vivir para siempre.</blockquote>`

### Ejercicio 3

Estructurar y marcar el siguiente texto para que el navegador lo muestre con el aspecto de la siguiente imagen:

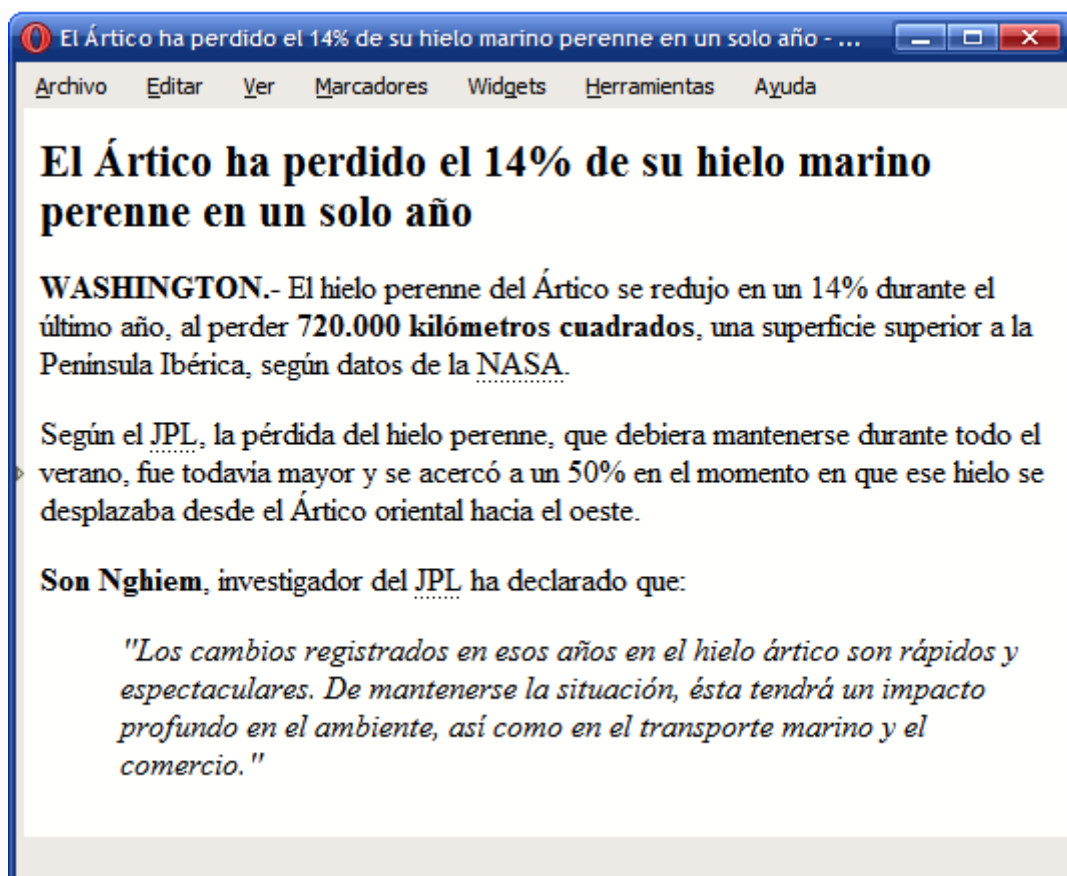


Figura 3.11. Texto HTML correctamente estructurado y marcado

## 3.4. Marcado genérico de texto

El estándar HTML/XHTML incluye numerosas etiquetas para marcar los contenidos de texto. No obstante, la infinita variedad de posibles contenidos textuales hace que no sean suficientes. Si se considera el siguiente ejemplo:

Importante: si quiere ponerse en contacto con la empresa ACME, puede hacerlo en el teléfono 900 555 555 o a través de la dirección de correo electrónico contacto@acme.org

El texto del ejemplo anterior contiene elementos de texto importantes, siglas, números de teléfono y direcciones de correo electrónico. XHTML define la etiqueta `<strong>` para marcar los elementos importantes y `<acronym>` para marcar las siglas:

`<strong>`Importante`</strong>`: si quiere ponerse en contacto con la empresa `<acronym>`ACME`</acronym>`, puede hacerlo en el teléfono 900 555 555 o a través de la dirección de correo electrónico contacto@acme.org

Desafortunadamente, XHTML no define ninguna etiqueta específica para marcar números de teléfono o direcciones de correo electrónico. De la misma forma, no define etiquetas para otros posibles elementos que se pueden encontrar en los contenidos de texto.

Por este motivo, el estándar HTML/XHTML incluye una etiqueta llamada `<span>` que se emplea para marcar cualquier elemento que no se puede marcar con las otras

etiquetas definidas. Siguiendo con el ejemplo anterior, la etiqueta `<span>` se utiliza para marcar el teléfono y la dirección de correo electrónico:

`<strong>Importante</strong>`: si quiere ponerse en contacto con la empresa `<acronym>ACME</acronym>`, puede hacerlo en el teléfono `<span>900 555 555</span>` o a través de la dirección de correo electrónico `<span>contacto@acme.org</span>`

La etiqueta `<span>` se visualiza por defecto con el mismo aspecto que el texto normal. Por tanto es habitual utilizar esta etiqueta junto con los atributos `id` y `class` para modificar posteriormente su aspecto con CSS:

`<strong>Importante</strong>`: si quiere ponerse en contacto con la empresa `<acronym>ACME</acronym>`, puede hacerlo en el teléfono `<span class="telefono">900 555 555</span>` o a través de la dirección de correo electrónico `<span class="email">contacto@acme.org</span>`

La etiqueta `<span>` sólo se puede utilizar para encerrar contenidos y etiquetas en línea. Cuando se quieren estructurar elementos de bloque, se utiliza la etiqueta `<div>`, tal y como se verá en capítulos posteriores.

## 3.5. Espacios en blanco y nuevas líneas

---

El aspecto más sorprendente del lenguaje HTML cuando se desarrollan los primeros documentos es el tratamiento especial de los "*espacios en blanco*" del texto. HTML considera *espacio en blanco* a los espacios en blanco, los tabuladores, los retornos de carro y el carácter de nueva línea (`ENTER` o `Intro`).

El siguiente ejemplo ilustra este comportamiento:

```
<html>
<head><title>Ejemplo de etiqueta p</title></head>
<body>
<p>Este primer párrafo no contiene saltos de línea ni otro
tipo de espaciado.</p>

<p>Este segundo párrafo sí que contiene saltos
de
línea
y otro tipo de espaciado.</p>
</body>
</html>
```

El anterior código HTML se visualiza en cualquier navegador de la siguiente manera:



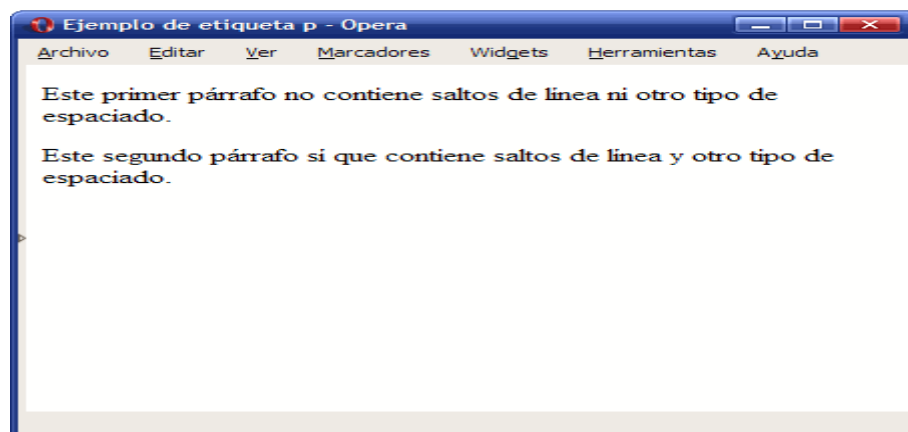


Figura 3.12. Ejemplo de comportamiento de HTML con los espacios en blanco

Los dos párrafos de la imagen anterior se ven idénticos, aunque el segundo párrafo incluye varios espacios en blanco y está escrito en varias líneas diferentes. La razón de este comportamiento es que HTML ignora todos los espacios en blanco *sobrantes*, es decir, todos los espacios en blanco que no son el espacio en blanco que separa las palabras.

No obstante, HTML proporciona varias alternativas para poder incluir tantos espacios en blanco y tantas nuevas líneas como sean necesarias dentro del contenido textual de las páginas.

### 3.5.1. Nuevas líneas

Para incluir una nueva línea en un punto y forzar a que el texto que sigue se muestre en la línea inferior, se utiliza la etiqueta `<br>`. En cierta manera, insertar la etiqueta `<br>` en un determinado punto del texto equivale a presionar la tecla **ENTER** (o **Intro**) en ese mismo punto.

La definición formal de `<br>` se muestra a continuación:

<b><code>&lt;br&gt;</code></b>	Nueva línea
<b>Atributos comunes</b>	básicos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea y etiqueta vacía
<b>Descripción</b>	Fuerza al navegador a insertar una nueva línea

La etiqueta `<br>` es una de las pocas *etiquetas especiales* de HTML. La particularidad de `<br>` es que es una etiqueta vacía, es decir, no encierra ningún texto. De esta forma, la etiqueta debe abrirse y cerrarse de forma consecutiva: `<br></br>`.

En estos casos, HTML permite utilizar un atajo para indicar que una etiqueta se está abriendo y cerrando de forma consecutiva: `<br/>` (también se puede escribir como `<br />`).

Utilizando la etiqueta `<br>` se puede rehacer el ejemplo anterior para que respete las líneas que forman el segundo párrafo:

```
<html>

<head>
<title>Ejemplo de etiqueta br</title>
</head>

<body>
<p>Este primer párrafo no contiene saltos de línea ni otro
tipo de espaciado.</p>

<p>Este segundo párrafo sí que contiene saltos <br/>
de <br/>
línea <br/>
y otro tipo de espaciado.</p>
</body>

</html>
```

El navegador ahora sí que muestra correctamente las nuevas líneas que se querían insertar:

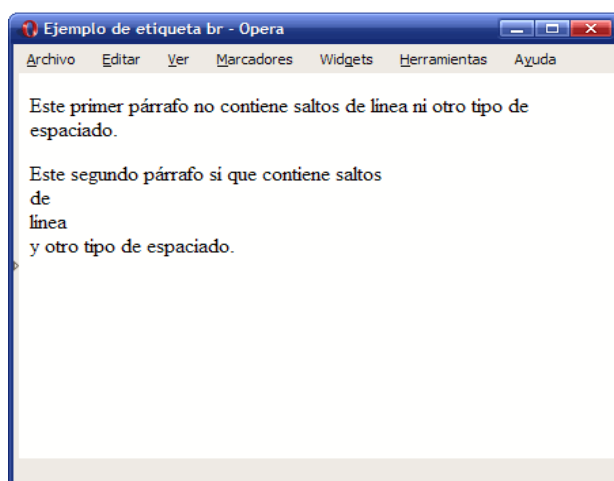


Figura 3.13. Ejemplo de uso de la etiqueta br

### 3.5.2. Espacios en blanco

La solución al problema de los espacios en blanco no es tan sencilla como el de las nuevas líneas. Para incluir espacios en blanco adicionales, se debe sustituir cada nuevo espacio en blanco por el texto `&nbsp;` (es importante incluir el símbolo `&` al principio y el símbolo `;` al final).

Así, el código HTML del ejemplo anterior se debe rehacer para incluir los espacios en blanco adicionales:

```
<html>

<head>
<title>Ejemplo de entidad &nbsp;</title>
```

```

</head>

<body>
<p>Este primer párrafo no contiene saltos de línea ni otro
tipo de espaciado.</p>

<p>Este segundo párrafo sí que contiene saltos <br/>
de <br/>
línea <br/>
y &nbsp;&nbsp;&nbsp; otro &nbsp; tipo &nbsp; de &nbsp;
espaciado.</p>
</body>

</html>

```

Ahora el navegador sí que muestra correctamente los espacios en blanco (y las nuevas líneas) del segundo párrafo:

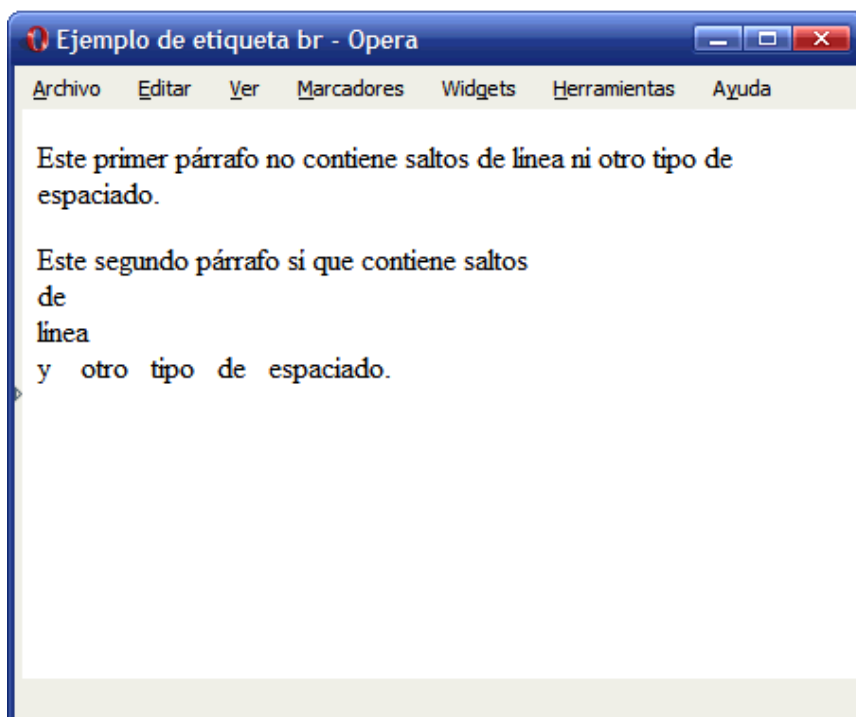


Figura 3.14. Ejemplo de uso de espacios en blanco en HTML

Cada texto `&nbsp;` solamente equivale a un espacio en blanco, por lo que se deben escribir tantos `&nbsp;` seguidos como espacios en blanco seguidos existan en el texto.

Más adelante se profundiza en el origen de `&nbsp;` y se comprenderá por qué es necesario incluir esa sucesión tan extraña de caracteres cada vez que se quiere incluir un espacio en blanco adicional.

## Ejercicio 4

Determinar el código HTML que corresponde al siguiente documento:

Nombre	Diametro relativo	Periodo orbital	Número de lunas
Mercurio	0,382	0,24 años	0
Venus	0,949	0,62 años	0
Tierra	1	1 año	1
Marte	0,532	1,88 años	2
Júpiter	11,209	11,86 años	49
Saturno	9,449	29,46 años	52
Urano	4,007	84,01 años	27
Neptuno	3,883	164,80 años	13

Figura 3.15. Texto HTML con espacios en blanco y nuevas líneas

### 3.5.3. Texto preformateado

En ocasiones, es necesario mostrar los espacios en blanco de un texto que no se puede modificar. Se trata de un caso habitual cuando una página web debe mostrar directamente el texto generado por alguna aplicación.

En estos casos, se puede utilizar la etiqueta `<pre>`, que muestra el texto tal y como se ha escrito, respetando todos los espacios en blanco y todas las nuevas líneas. La definición formal de la etiqueta se muestra a continuación:

<b><code>&lt;pre&gt;</code></b>	Texto preformateado
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Muestra el texto que encierra tal y como está escrito (respetando los espacios en blanco)

El siguiente ejemplo muestra el uso de la etiqueta `<pre>`:

```
<html>
<head><title>Ejemplo de etiqueta pre</title></head>
  <body>
    <pre>
      La etiqueta pre
```

```

    respeta los espacios en blanco
    y
muestra el texto
    tal y como
    está escrito
</pre>

<p>
    La etiqueta pre
    respeta los espacios en blanco
    y
muestra el texto
    tal y como
    está escrito
</p>
</body>
</html>

```

El ejemplo anterior incluye el mismo texto (con espacios en blanco y varias líneas) dentro de una etiqueta `<pre>` y dentro de una etiqueta `<p>`. Las diferencias visuales en un navegador son muy evidentes:

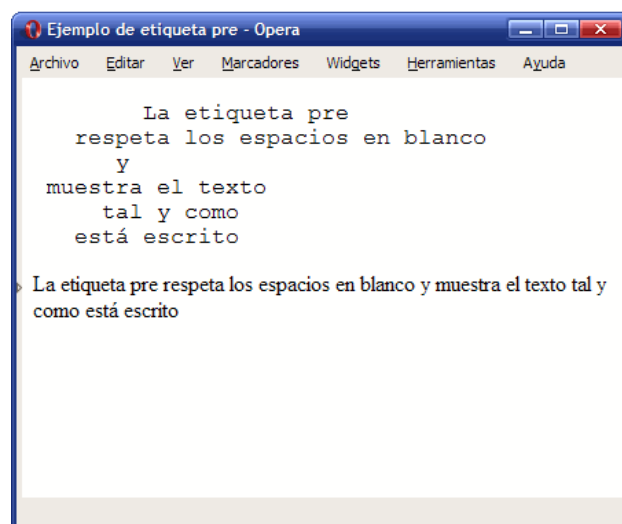


Figura 3.16. Ejemplo de uso de la etiqueta pre

El primer texto se ve en pantalla tal y como se ha escrito, respetando todos los espacios en blanco y todas las nuevas líneas. El segundo texto se ve como un párrafo normal, ya que HTML ha eliminado todos los espacios en blanco sobrantes. Los elementos `<pre>` son *especiales*, ya que los navegadores les aplican las siguientes reglas:

- Mantienen todos los espacios en blanco (tabuladores, espacios y nuevas líneas)
- Muestra el texto con un tipo de letra especial, denominado "*de ancho fijo*", ya que todas sus letras son de la misma anchura
- No se ajusta la longitud de las líneas (las líneas largas producen un *scroll* horizontal en la ventana del navegador)

Esta última característica diferencia por completo a los párrafos de los elementos `<pre>`. Como se ha visto, los navegadores ajustan la anchura de los párrafos de texto para que ocupen todo el tamaño de la ventana. Sin embargo, los elementos `<pre>` se

muestran tal y como son originalmente, por lo que una línea muy larga dentro de un elemento `<pre>` provoca que la anchura de la página sea superior a la anchura de la ventana del navegador.

Si en el ejemplo anterior se añade más texto al final de la segunda línea (para producir una línea larga), el navegador muestra un *scroll* horizontal ya que el texto completo no cabe en el tamaño de la ventana y las líneas de los elementos `<pre>` nunca se ajustan.

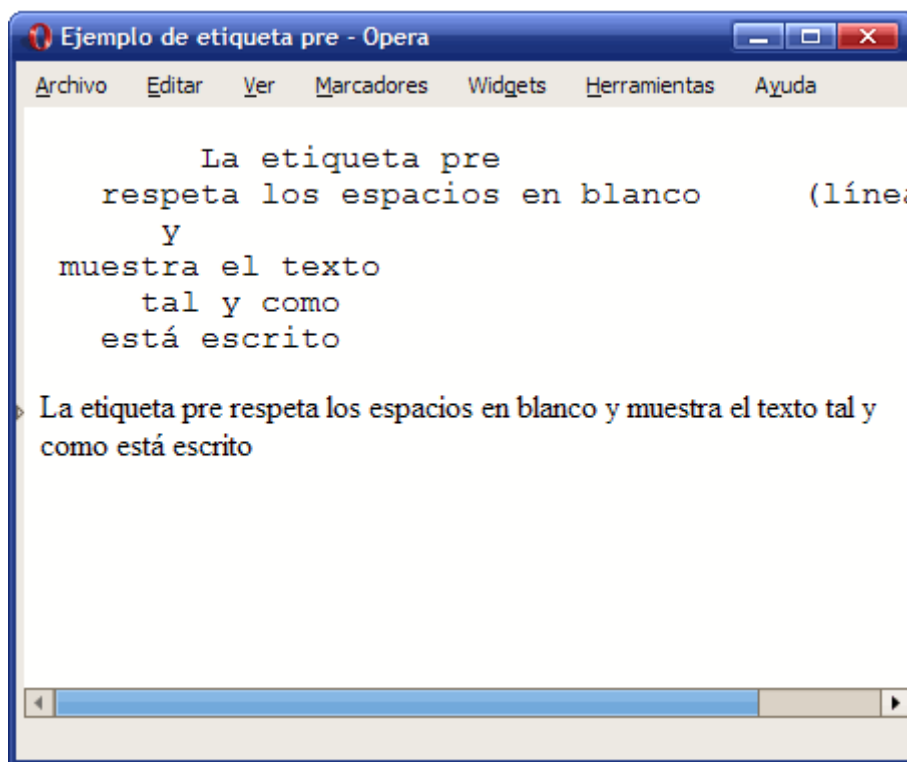


Figura 3.17. Ejemplo de aparición de scroll horizontal con la etiqueta `pre`

Otra etiqueta relacionada con `<pre>` es la etiqueta `<code>`, que se utiliza para mostrar código fuente de cualquier lenguaje de programación. La definición formal de `<code>` es la siguiente:

<b><code>&lt;code&gt;</code></b>	Código fuente
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Delimita el texto considerado un fragmento de código fuente

En la mayoría de páginas web, no tiene sentido utilizar la etiqueta `<code>`. Sin embargo, en muchas páginas web técnicas que incluyen listados de programas, trozos de código o etiquetas HTML, lo correcto es emplear la etiqueta `<code>`.

Ejemplo:

```
<html>
<head><title>Ejemplo de etiqueta code</title></head>

<body>

<code>
    La etiqueta code
    no respeta los espacios en blanco
</code>

<p>La etiqueta <code>code</code> es similar a la
etiqueta <code>pre</code>, sobre todo en el formato
del texto.</p>

</body>
</html>
```

El navegador muestra claramente el comportamiento de `<code>` y sus diferencias con `<pre>`:

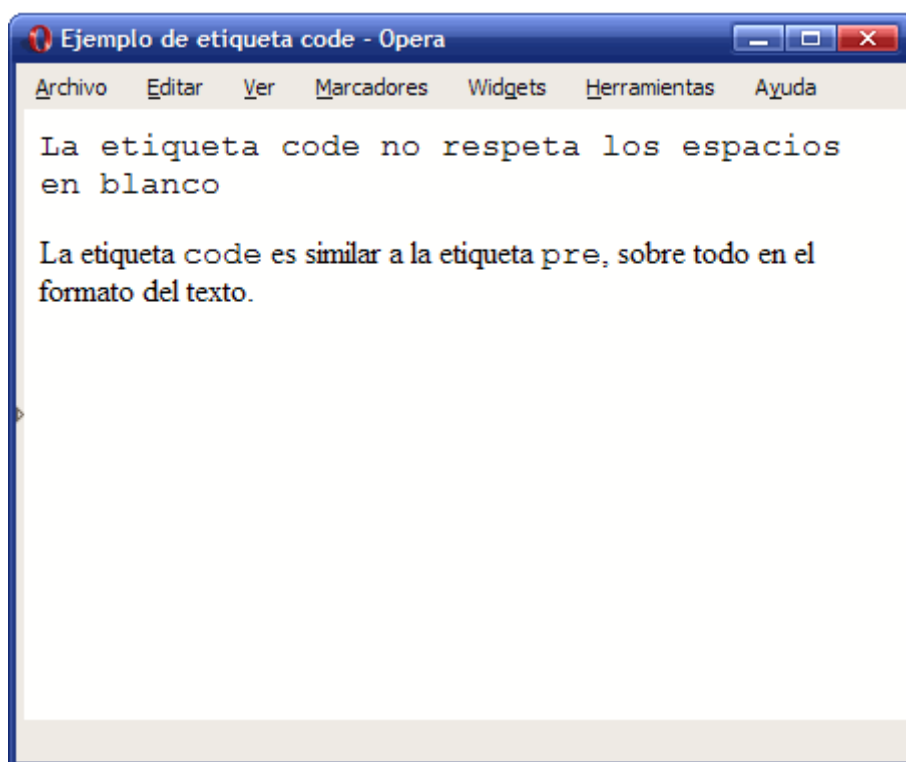


Figura 3.18. Ejemplo de uso de la etiqueta code

Al igual que sucede con los elementos `<pre>`, el texto encerrado por la etiqueta `<code>` se muestra con un tipo de letra especial de ancho fijo. Por el contrario, el elemento `<code>` no respeta los espacios en blanco ni las líneas, por lo que su comportamiento es similar a la etiqueta `<p>`. La última diferencia es que `<code>` es un elemento en línea, mientras que `<pre>` es un elemento de bloque.

## 3.6. Codificación de caracteres

---

Una consideración importante directamente relacionada con el texto de las páginas HTML es la codificación de los caracteres y la inserción de caracteres *especiales*. Algunos de los caracteres que se utilizan habitualmente en los textos no se pueden incluir directamente en las páginas web:

- Los caracteres que utiliza HTML para definir sus etiquetas (<, > y ") no se pueden utilizar libremente.
- Los caracteres propios de los idiomas que no son el inglés (ñ, á, ç, é, í, etc.) pueden ser problemáticos dependiendo de la codificación de caracteres utilizada.

La solución a la primera limitación consiste en sustituir los caracteres reservados de HTML por unas expresiones llamadas *entidades HTML* y que representan a cada carácter:

Entidad	Carácter	Descripción	Traducción
&lt;	<	less than	signo de menor que
&gt;	>	more than	signo de mayor que
&amp;	&	ampersand	ampersand
&quot;	"	quotation mark	comillas
&nbsp;	(espacio en blanco)	non-breaking space	espacio en blanco
&apos;	'	apostrophe	apóstrofo

De esta forma, si se considera el siguiente texto:

Los caracteres <, >, " y & pueden dar problemas con los textos en HTML

Para mostrar correctamente el texto anterior en una página HTML, se debe sustituir cada carácter especial por su entidad HTML:

<p>Los caracteres &lt;, &gt;, &quot; y &amp; pueden dar problemas con los textos en HTML</p>

## Ejercicio 5

Determinar el código HTML que corresponde al siguiente documento:



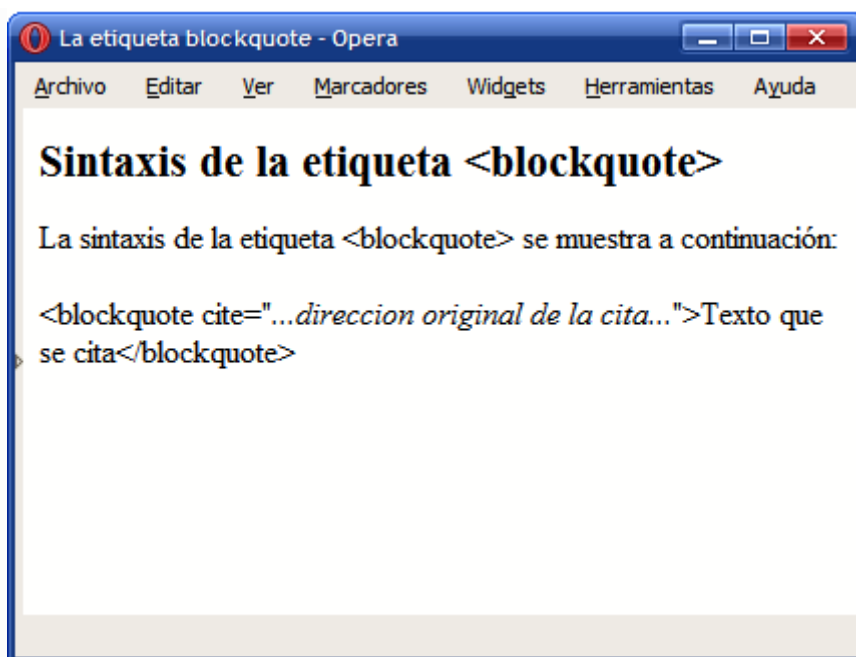


Figura 3.19. Texto HTML que incluye caracteres especiales

Por otra parte, los caracteres propios de los idiomas diferentes al inglés también pueden ser problemáticos. El motivo es que desde que se crea una página web hasta que llega al navegador del usuario, intervienen numerosos procesos:

- El diseñador crea la página web con su editor HTML (por ejemplo Dreamweaver).
- Si se trata de una aplicación dinámica, el programador recorta la página HTML del diseñador y la mezcla con el resto del código de la aplicación (por ejemplo PHP).
- El servidor web almacena las páginas HTML estáticas o el código de la aplicación web y sirve las páginas solicitadas por los usuarios.
- El usuario solicita y visualiza las páginas web a través de su navegador.

Si en todos los procesos anteriores se utiliza la misma codificación de caracteres, los caracteres propios de los idiomas se pueden escribir directamente:

`<p>`Este párrafo contiene caracteres acentuados y se almacena en formato UTF-8`</p>`

La palabra **párrafo** del ejemplo anterior incluye la letra **á**. Si el editor HTML del diseñador utiliza la codificación UTF-8, el entorno de desarrollo del programador también utiliza UTF-8, el servidor web sirve las páginas con esa codificación y el navegador del usuario es capaz de visualizar las páginas con formato UTF-8, el texto anterior se verá correctamente en el navegador del usuario.

Sin embargo, muchas veces no es posible que todos los procesos involucrados utilicen la misma codificación de caracteres. Por limitaciones técnicas o por decisiones de los diseñadores y programadores, los textos pueden pasar de codificación UTF-8 a codificación ISO-8859 en cualquier momento. Si se produce este cambio sin realizar una conversión correcta, el navegador del usuario mostrará caracteres extraños en todos los acentos y en todas las letras como la **ñ**.

La solución más sencilla para asegurar que todos estos caracteres potencialmente problemáticos se van a visualizar correctamente en el navegador del usuario consiste en sustituir cada carácter problemático por su entidad HTML:

Entidad	Carácter	Descripción oficial
&ntilde;	ñ	latin letter n with tilde
&Ntilde;	Ñ	latin capital n letter with tilde
&aacute;	á	a acute
&eacute;	é	e acute
&iacute;	í	i acute
&oacute;	ó	o acute
&uacute;	ú	u acute
&Aacute;	Á	A acute
&Eacute;	É	E acute
&Iacute;	Í	I acute
&Oacute;	Ó	O acute
&Uacute;	Ú	U acute
&euro;	€	euro

Así, el párrafo de texto del ejemplo anterior, se podría escribir de la siguiente manera:

`<p>Este p&aacute;rrafo contiene caracteres acentuados y se almacena en formato UTF-8</p>`

Si se utilizan las entidades HTML en vez de los caracteres problemáticos, es indiferente pasar de una codificación de caracteres a otra diferente. En la Wikipedia se puede consultar [la lista completa de las 252 entidades HTML definidas](#).

Los apartados siguientes tratan temas que son más eficientes utilizando Css,

## 3.7. color, tamaño y tipo de letra

---

A pesar de que por razones de homogeneidad y sencillez de código este tipo de formatos son controlados actualmente por hojas de estilo en cascada (de las cuales ya tendremos tiempo de hablar), existe una forma clásica y directa de definir color, tamaño y tipo de letra de un texto determinado.

Esto se hace a partir de la etiqueta `<font>` y su cierre correspondiente. Dentro de esta etiqueta deberemos especificar los atributos correspondientes a cada uno de estos parámetros que deseamos definir. A continuación os comentamos los atributos principales de esta etiqueta:

### **Atributo face**

Define el tipo de letra. Este atributo es interpretado por versiones de Netscape a partir de la 3 y de MSIE 3 o superiores. Otros navegadores las ignoran completamente y muestran el texto con la fuente que utilizan.

Hay que tener cuidado con este atributo ya que cada usuario, dependiendo de la plataforma que utilice, puede no disponer de los mismos tipos de letra que nosotros con lo que, si nosotros elegimos un tipo del que no dispone, el navegador se verá forzado a mostrar el texto con la fuente que utiliza por defecto (suele ser Times New Roman). Para evitar esto, dentro del atributo suelen seleccionarse varios tipos de letra separados por comas. En este caso el navegador comprobará que dispone del primer tipo enumerado y si no es así, pasará al segundo y así sucesivamente hasta encontrar un tipo que posea o bien acabar la lista y poner la fuente por defecto. Veamos un ejemplo.

```
<font face="Comic Sans MS,arial,verdana">Este texto tiene otra tipografía</font>
```

Que se visualizaría así en una página web.

## **Atributo size**

Define el tamaño de la letra. Este tamaño puede ser absoluto o relativo.

Si hablamos en términos absolutos, existen 7 niveles de tamaño distintos numerados de 1 a 7 por orden creciente. Elegiremos por tanto un valor `size="1"` para la letra más pequeña o `size="7"` para la más grande.

`<font size="4">`Este texto es más grande`</font>`

Que se visualizaría así en una página web.

Este texto es más grande

Podemos asimismo modificar el tamaño de nuestra letra con respecto al del texto mostrado precedentemente definiendo el número de niveles que queremos subir o bajar en esta escala de tamaños por medio de un signo + o -. De este modo, si definimos nuestro atributo como `size="+1"` lo que queremos decir es que aumentamos de un nivel el tamaño de la letra. Si estabamos escribiendo previamente en 3, pasaremos automáticamente a 4.


Los tamaños reales que veremos en pantalla dependerán de la definición y del tamaño de fuente elegido por el usuario en el navegador. Este tamaño de fuente puede ser definido en el Explorer yendo al menu superior, Ver/Tamaño de la fuente. En Netscape elegiremos View/Text Size. Esta flexibilidad puede en más de una ocasión resultarnos embarazosa ya que en muchos casos desearemos que el tamaño del texto permanezca constante para que éste quepa en un determinado espacio. Veremos en su momento que esta prefijación del tamaño puede ser llevada a cabo por las hojas de estilo en cascada.

## **Atributo color**

El color del texto puede ser definido mediante el atributo color. Cada color es a su vez definido por un número hexadecimal que esta compuesto a su vez de tres partes. Cada una de estas partes representa la contribución del rojo, verde y azul al color en cuestión.

Por otra parte, es posible definir de una manera inmediata algunos de los colores más frecuentemente usados para los que se ha creado un nombre más memotécnico:

## Nombre Color

Aqua	
Black	
Blue	
Fuchsia	
Gray	
Green	
Lime	
Maroon	
Navy	
Olive	
Purple	
Red	
Silver	
Teal	
White	
Yellow	

`<font color="red">Este texto está en rojo</font>`

Que se visualizaría así en una página web.

Este texto está en rojo

Con todo esto estamos ya en disposición de crear un texto formateado de una forma realmente elaborada.

## Ejercicio 6

---

página que tenga las siguientes características:

- Un titular con encabezado de nivel 1, en itálica y color verde oliva.
- Un segundo titular con encabezado de nivel 2, también de color verde oliva.
- Todo el texto de la página deberá presentarse con una fuente distinta de la fuente por defecto. Por ejemplo "Comic Sans MS" y en caso de que ésta no esté en el sistema que se coloque la fuente "Arial".

## 3.8. linea separadora horizontal

El elemento [HR](#) hace que el agente de usuario represente una línea separadora horizontal.

La cantidad de espacio vertical insertado entre el separador y el contenido que le rodea depende del usuario.

### EJEMPLO :

Este ejemplo centra los separadores, dándoles un tamaño igual a la mitad de la anchura disponible entre márgenes. El separador superior tiene el tamaño por defecto, mientras que los dos inferiores son de 5 píxeles. El separador inferior debería representarse con un color sólido sin efecto de relieve:

```
<HR width="50%" align="center">
<HR size="5" width="50%" align="center">
<HR noshade size="5" width="50%" align="center">
```

Estos separadores podrían representarse como sigue:



## 3.9. Alineacion

Es posible alinear elementos en bloque (tablas, imágenes, objetos, párrafos, etc.) en el lienzo con el atributo `align`. Aunque este atributo se puede establecer en muchos elementos HTML, su rango de valores posibles difiere en ocasiones de unos elementos a otros. Aquí sólo se considera el significado del atributo `align` para texto.

### *Definiciones de atributos*

**`align = left | center | right | justify`**

Este atributo especifica la alineación horizontal de su elemento con respecto al contexto circundante. Valores posibles:

- `left`: las líneas de texto se representan con los bordes izquierdos alineados.
- `center`: las líneas de texto son centradas.
- `right`: las líneas de texto se representan con los bordes derechos alineados.
- `justify`: las líneas de texto se justifican en ambos márgenes.

El valor por defecto depende de la dirección base del texto. Para un texto de izquierda a derecha el valor por defecto es `align=left`, mientras que para un texto de derecha a izquierda el valor por defecto es `align=right`.

Este ejemplo centra un encabezado en el lienzo.

```
<H1 align="center"> Cómo Tallar la Madera </H1>
```

Usando CSS, por ejemplo, podríamos conseguir el mismo efecto de la manera más eficiente.