

[3] Etiquetas básicas HTML

[3.1] contenido y contenedores

En HTML los elementos que muestran algo en la página, se colocan dentro de la sección body.

Además, hay que tener en cuenta que hay elementos pensados para contener grandes cantidades de información (como div) y elementos que contienen poco texto (como strong). Es correcto que aparezca un elemento strong dentro de un elemento p, pero no al revés:

Hola</p> -->¡¡Incorrecto!!

Hola</p> -->Correcto

La mayoría de elementos en HTML son contenedores de otros textos y elementos. Hay que conocer que tipos de elementos tenemos. Los tipos están relacionados con la propiedad CSS display, que se encarga de decir cómo tiene que mostrarse un elemento. Por defecto los elementos funcionan de esta forma:

Elementos de tipo inline. Son elementos que se muestran seguidos en la misma línea. Los elementos de este tipo son los que marcan texto simple. Todas las etiquetas de marcado de texto tienen este display: strong, em, span, mark, abbr, etc.

Ejemplo, el código:

a Hola

Se muestra así:

Hola *Hola*

Ilustración 3-3. Elementos de tipo inline: strong y em.

Elementos de tipo block. Son contenedores más grandes. Los elementos que marcan párrafos enteros (como p, h1, h2) y los contenedores de secciones (como div) tienen este display. Su característica es que, por defecto, tienen forma rectangular y abarcan toda la anchura de su contenedor. Si indicamos para ellos un color de fondo, este se muestra en forma de rectángulo (en los inline, el color de fondo abarca exactamente al texto, como si lo hubiéramos pintado con un rotulador). Ejemplo:

```
<h1> <p>Hola</p>
```

Ese código se muestra así:

Hola

Hola

Ilustración 3-4. Elementos de tipo **block**: h1 y p.

Elementos de tipo inline-block. Son rectangulares como los block, pero no abarcan todo su contenedor. Se adaptan a su contenido. Ejemplos son los elementos button o image:

```
<strong> <button>Hola</button>
```

El resultado muestra un botón (elemento rectangular) seguido del texto de tipo strong. Si el botón tuviera un display de tipo block, saltaría a la línea siguiente.

Hola

Ilustración 3-5. Elemento **button** de tipo **inline-block**

En todo caso, el display de un elemento se puede modificar desde lenguaje CSS, pero no conviene hacer combinaciones anti naturales ya que se pierde el sentido semántico del código.

[3.1.2] div

Se trata de un elemento clásico, ya estaba presente en las primeras versiones de HTML, que se utiliza como contenedor de otros elementos a fin de poder manipularlos todos a la vez. Dentro de **div** se pueden colocar todo tipo de etiquetas (tablas, párrafos, imágenes,...).

Desde hace años se ha convertido en el elemento utilizado, combinado con CSS, para crear **capas**. Las capas son elementos visuales que se pueden posicionar a voluntad en la página web. De esta forma, a través de las capas se consiguen maquetaciones muy artísticas.

Por defecto, los navegadores no dan formato al contenido de un elemento *div*, con lo que visualmente no se diferencia del resto de elementos, salvo por el hecho de que un elemento *div* tiene un *display* de tipo *block*. Es decir, el contenido de estos elementos, por defecto, se muestra de forma separada al resto.

[3.1.3] span

Sirve para marcar un trozo de texto de forma inespecífica, sin indicar semántica alguna. En definitiva, selecciona un trozo de texto por necesidades técnicas (por ejemplo para indicar un formato CSS diferente a ese texto).

La recomendación es utilizar un elemento más semántico siempre que sea posible.

[3.2] listas

Las listas permiten crear párrafos agrupados y alineados mediante símbolos como viñetas o números para facilitar la lectura y organización de las ideas del documento.

[3.2.1] listas desordenada

Se deben englobar dentro de un elemento **ul** (acrónimo de *unordered list*, lista no ordenada), después cada párrafo de la lista estará dentro de elementos de tipo **li** (de *list item*, elemento de lista).

Ejemplo:

```
de la compra</p>
```

```
agua</li>
```

```
vino</li>
```

```
cerveza</li>
```

Resultado:

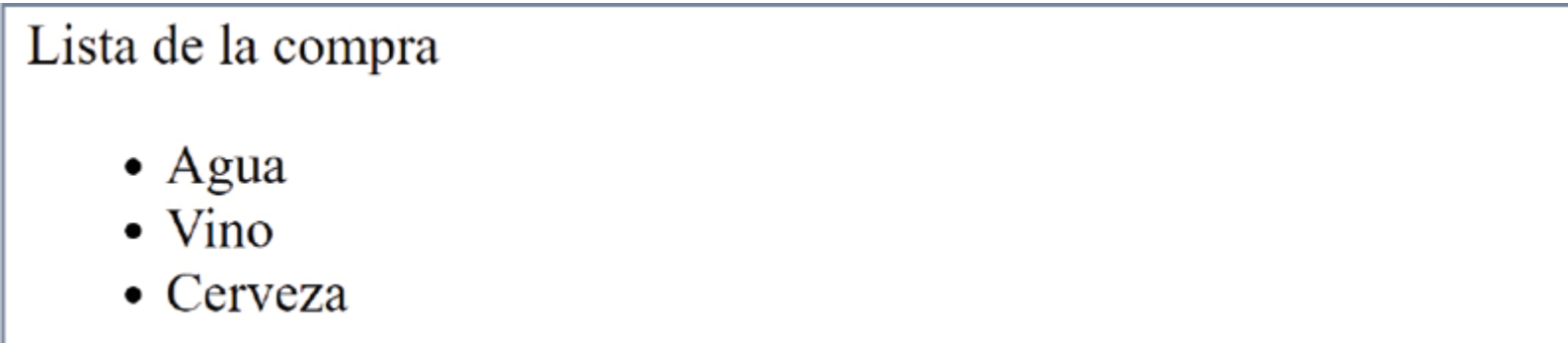


Ilustración 4-13. Lista de viñetas

[3.2.2] lista ordenada

Aparecen dentro del elemento **ol** (de *ordered list*, lista ordenada), después cada párrafo de la lista estará dentro de elementos de tipo **li**, al igual que las anteriores. La diferencia ahora es que cada párrafo con **li**, aparece con un número y no con una viñeta.

Ejemplo:

```
de la compra</p>
```

```
agua</li>
```

```
vino</li>
```

```
cerveza</li>
```

Resultado:

Lista de la compra

1. Agua
2. Vino
3. Cerveza

Ilustración 4-14. Lista numerada

[3.2.3] anidamiento de listas

Es posible meter una lista dentro de otra mezclando listas ordenadas y desordenadas, por ejemplo:

de la compra</p>

alcohólicas

.l>

Agua

ul>

cohólicas

.l>

Vino

Cerveza

ul>

Con el resultado:

Lista de la compra

1. No alcohólicas
 - Agua
2. Alcohólicas
 - Vino
 - Cerveza

Ilustración 4-16. Lista anidada de viñetas y números

[3.2.4] listas de definiciones

Permite crear una lista de definiciones de términos. En ellas se indica el término a definir y su significado. Ejemplo:

ndows</dt>

stema operativo de Microsoft disponible en versiones de 32 y 64 bits y para servidores y ordenadores e incluso tabletas y móviles.

última versión es la versión 10 y la 2016 para servidores.

nux</dt>

stema operativo de código abierto disponible en numerosas distribuciones gratuitas y de pago. La base del sistema Android.

c OS</dt>

stema operativo de los ordenadores de la empresa Microsoft.

La última versión es Mojave.

Resultado:

Windows

Sistema operativo de **Microsoft** disponible para PC disponible en versiones de 32 bits y para servidores, ordenadores e incluso tabletas y móviles.
La última versión es la versión 10 y la 2016 para servidores.

Linux

Sistema operativo de código abierto disponible en numerosas distribuciones gratuitas de pago. Es la base del sistema **Android**.

Mac OS

Sistema operativo de los ordenadores de la empresa **Apple**
La última versión es **Mojave**

Ilustración 4-17. Lista anidada de viñetas y números

[3.3] etiquetas básicas de párrafo

Desde hace años todo texto en un documento HTML debe ir dentro de una etiqueta que sirva para decir qué tipo de texto es. Y eso significa, que el texto debe de ir dentro de una etiqueta de párrafo. Se comentan las etiquetas disponibles para marcar párrafos.

[3.3.1] párrafo simple

La norma HTML estándar establece que el texto de la página debe estar contenido dentro de un elemento. Es decir, no se puede poner texto directamente dentro de la etiqueta *body*, ha de haber un contenedor para ese texto.

El elemento más sencillo lo marca la etiqueta **p**, que indica un elemento de párrafo normal. Todo texto dentro de una etiqueta *p*, queda marcado como texto dentro de un párrafo normal.

Normalmente los navegadores utilizan fuentes tipo *Times* de tamaño *11pt* para la letra de párrafo normal. Ejemplo de uso:

Párrafo con un poco de texto

un párrafo completo englosado dentro de una etiqueta
xto se escriba
odos estos saltos de línea y espacios
strará seguido,
altos de
interiores
y entre cada palabra

En el navegador saldría:

Párrafo con un poco de texto

Este es un párrafo completo englosado dentro de una etiqueta *p*, aunque el texto escriba con todos estos saltos de línea y espacios se mostrará seguido, sin saltos de línea interiores y entre cada palabra dejando un solo espacio

Ilustración 4-1. Texto de dos párrafos, no se consideran los espacios múltiples, saltos de línea y saltos de línea interiores a un párrafo. El texto se adapta automáticamente al contenedor

En la imagen se observa el modo en el que el navegador muestra el párrafo. Por defecto los párrafos tipo *p* dejan espacio arriba y abajo (normalmente media línea en cada dirección).

[3.3.2] Encabezados o títulos

Hay una serie de seis etiquetas que comienzan con la letra *h* a la que le sigue un número del 1 al 6. Sirven para marcar párrafos de forma que se considerarán títulos del texto. De modo que el elemento **h1** marcará títulos de primer nivel, **h2** de segundo nivel,... y así hasta **h6** (sexto nivel).

Sistema Solar

Planetas

Mercurio

Mercurio es el planeta del Sistema Solar más próximo al Sol y el más pequeño (a excepción de los planetas enanos). Parte de los denominados planetas interiores o rocosos y carece de satélites.

Venus

Venus es el segundo planeta del Sistema Solar en orden de distancia desde el Sol, y el tercero en cuanto a tamaño menor a mayo.

Sol

El Sol es la estrella del sistema planetario en el que se encuentra la Tierra; por tanto, es la más cercana a la Tierra con mayor brillo aparente.

Satélites

Luna

La Luna es el único satélite natural de la Tierra y el quinto satélite más grande del Sistema Solar.

Ío

Fue descubierto por Galileo Galilei en 1610 y recibió inicialmente el nombre de Júpiter I como primer satélite de Júpiter.

Otros Sistemas

Fomalhaut

Estrella conocida desde la prehistoria, ahora se le han descubierto planetas.

Vega

Antigua estrella polar, muy venerada. Posee un disco de polvo que podría contener planetas o bien formarse pronto.

Ejemplo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Estrellas</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <h1>Sistema Solar</h1>
    <h2> Planetas</h2>
    <h3>Mercurio</h3>
    <p>
      Mercurio es el planeta del Sistema Solar más próximo al Sol
y el más pequeño (a excepción de los planetas enanos). Forma parte de
los
denominados planetas interiores o rocosos y carece de satélites
    </p>
    <h3>Venus</h3>
    <p>
      Venus es el segundo planeta del Sistema Solar en orden de
distancia
desde el Sol, y el tercero en cuanto a tamaño, de menor a mayo
    </p>
    <p>...</p>
    <h2>Sol</h2>

Sol es la estrella del sistema planetario en el que se encuentra la
r tanto, es la más cercana a la Tierra y el astro con mayor
rente

Satélites</h2>
Luna</h3>

Luna es el único satélite natural de la Tierra y el quinto
ás grande del Sistema Solar.

    <h3>Ío</h3>
    <p>
Fue descubierto por Galileo Galilei en 1610 y recibió inicialmente
el nombre de Júpiter I como primer satélite de Júpiter.
    </p>
    <p>...</p>
tros Sistemas</h1>
```

```
omalhaut</h2>
```

```
onocida desde la prehistoria ahora se le han descubierto planetas  
</p>
```

```
<h2>Vega</h2>
```

```
<p>Antigua estrella polar, muy venerada. Posee un disco de polvo que  
podría  
contener planetas o bien formarse pronto
```

El resultado en el navegador es el mostrado en la [Ilustración 65](#)

líneas y saltos

[3.3.3]salto de línea

A veces es necesario dentro del texto de un determinado párrafo hacer un salto de línea. El elemento que lo realiza no tiene cierre y se llama **br**. Ejemplo:

```
la línea <br>Segunda línea</p>
```

Aparecerá cada texto en una línea distinta.

Primera línea
Segunda línea

Ilustración 4-3. Párrafo con dos líneas, separadas por un salto de línea

[3.3.4]línea horizontal

Otra posibilidad es hacer un salto pero dejando una línea horizontal en el hueco de las palabras. Esto lo hace la etiqueta **hr** (que tampoco tiene cierre):

```
 párrafo</p>  
o párrafo</p>
```

Resultado:

Primer párrafo

Segundo párrafo

Ilustración 4-4. Uso de un separador de párrafos mediante `hr`

[3.4] elementos de marcado de texto simple

En este caso, detallaremos los elementos que se emplean para señalar texto interior a un párrafo.

[3.4.1] texto importante

En HTML 5 el texto importante se marca con el elemento **strong**. En general todos los navegadores marcan el texto *strong* en negrita (letra más gruesa), pero que se muestre en negrita es un acuerdo visual que se puede modificar con el lenguaje CSS. Es decir, no es lo mismo texto en negrita que texto importante.

[3.4.2] texto enfatizado

El elemento **em** marca texto como enfatizado. El resultado visual con que le dan formato por defecto los navegadores es texto en cursiva.

La idea semántica es marcar con **em** texto con énfasis de segundo nivel (dejando a **strong** el primer nivel) .

[3.4.3] texto resaltado

Se trata del tercer elemento de HTML5 para remarcar texto. Se hace a través el elemento **mark**. En este caso, su efecto visual, suele ser aplicar en el texto así marcado un fondo de color vivo que simula el uso de un subrayador. Ejemplo:

```
texto normal, <mark>y yo estoy remarcado  
</mark>
```

Obtendría el resultado:

Yo soy texto normal, y yo estoy remarcado

Ilustración 4-5. Ejemplo de texto remarcado

Este elemento no es tan antiguo como los anteriores porque apareció con HTML5 (los navegadores antiguos no lo reconocen). La idea semántica es marcar texto que queremos realzar de forma muy llamativa indicando texto a recordar o sobre el que profundizar.

[3.5] otros elementos de marcado de caracteres

[3.5.1] subíndices

Permite que el texto aparezca por debajo de la línea base y en un tamaño más pequeño. Lo hace el elemento **sub**, ejemplo:

```
mula del agua es H<sub>2</sub>O</p>
```

Mostraría: H₂O

[3.5.2] superíndices

Parecida al anterior, pero ahora el texto marcado con el elemento **sup** aparecerá por encima y en pequeño. Ejemplo:

```
mula es x<sup>2</sup></p>
```

Obtendría: x²

[3.5.3] elemento *small*

Marcado de letra pequeña. Se usa, por ejemplo en textos que marquen copyright, derechos de uso, comentarios, notas anexas y letra pequeña (tipo la de los contratos) en general. El efecto visual es que el texto sale con un tamaño más pequeño. Es posible incluso anidar un elemento **small** dentro de otro, de modo que cada vez sale más pequeño el texto. Ejemplo:

```
mal <small>Pequeño  
n más pequeño</small></small>
```

Resultado:

Texto normal Pequeño Aún más pequeño

Ilustración 4-6. Ejemplo de uso de small

[3.5.4] definiciones

El elemento **dfn** se usa de forma similar al anterior. Sirve para indicar definiciones para el término o palabras que encuadra.

Ejemplo:

```
e="usar dos conceptos de significado opuesto en una s  
>oxímoron
```

Los navegadores suelen mostrar el contenido del elemento en cursiva y, como siempre, en un cartelito muestran el contenido del atributo **title**:

oxímoron

usar dos conceptos de significado opuesto en una sola expresión

Ilustración 4-8. Texto marcado como definición

[3.5.5] escritura de código o texto informático

escritura de código

HTML proporciona un elemento llamado **code** que se utiliza cuando en un documento deseamos escribir ejemplos de código escritos en un lenguaje de programación. Por ejemplo:

s un ejemplo de código en JavaScript:</p>

```
window.onload=function(){
    alert("Hola");
}
```

Lo cual obtendría este resultado:

Este es un ejemplo de código en JavaScript:

```
window.onload=function(){ alert("Hola"); }
```

Ilustración 4-9. Resultado de aplicar `code`

Se puede observar como los navegadores modifican el tipo de letra (utilizan un tipo de letra monoespaciada) pero no respetan los espacios y tabuladores utilizados para indicar el código. Como es muy normal querer respetar los espacios y tabuladores cuando se escriben ejemplos de código, lo habitual es hacer esta combinación:

s un ejemplo de código en JavaScript:</p>

<pre><code>

```
window.onload=function(){
    alert("Hola");
}
```

pre>

Y así obtenemos este otro resultado:

Este es un ejemplo de código en JavaScript:

```
window.onload=function(){  
    alert("Hola");  
}
```

Ilustración 4-10. Resultado de aplicar `code` dentro de un contenedor de tipo `pre`.

resultados de operaciones en un ordenador

Con intenciones similares al elemento anterior, tenemos el elemento **samp**. Se utiliza cuando queremos en un documento indicar ejemplos de resultados de operaciones en un ordenador. Ejemplo:

```
<p>Al ejecutar el comando obtuve el mensaje  
<samp>Comando o nombre de archivo incorrecto </samp>
```

Los navegadores suelen mostrar este elemento en letra monoespaciada. La salida del código anterior suele ser:

```
Al ejecutar el comando obtuve el mensaje Comando o nombre  
de archivo incorrecto
```

Ilustración 4-11. Resultado de aplicar `code` dentro de un contenedor de tipo `pre`.

[3.5.6] etiquetas obsoletas

Son etiquetas que quedaron en desusos y no se recomiendan en la versión html5.

<s> Se trata de un antiguo elemento de las páginas web que permitía indicar un subrayado. En la versión de HTML 4 quedó obsoleto, no se recomendaba su uso

ya que el subrayado se reserva para los enlaces. Con HTML 5 se ha redefinido su uso para marcar texto obsoleto que, visualmente, se muestra con efecto de tachado.

Ejemplo:

La capital de Alemania es `<s>Bonn</s>` Berlín

Resultado:

La capital de Alemania es ~~Bonn~~ Berlín

Ilustración 4-7. Texto obsoleto marcado con el elemento `s`

`` Se trata del elemento histórico (de las primeras versiones de HTML) utilizado para marcar texto en negrita. En HTML 4 quedó como obsoleto. En HTML 5 se volvió a permitir, pero solo si necesitamos un nivel de marcado de texto más allá del uso de **strong**, **em** o **mark**. En definitiva, elemento de uso muy poco aconsejable que se mantienen solo por la gran cantidad de creadores que lo siguen utilizando.

`<i>` Tiene el mismo problema que el anterior. Es el elemento histórico (de las primeras versiones de HTML) utilizado para marcar texto en cursiva. En HTML 4 quedó como obsoleto. En HTML 5 se puede utilizar para marcar texto tipo alternativo si no encaja con otros elementos de marcado de caracteres.

[3.6] enlaces

[3.6.1]enlaces globales

Los enlaces permiten colocar un texto (u otro elemento, como una imagen o un botón) resaltado de forma especial, de modo que cuando se le hace clic, el navegador web nos llevará al destino de la URL (si está disponible).

La etiqueta que permite realizar enlaces es la etiqueta **a**. El atributo **href** permite indicar la URL a la que se realiza el salto. Ejemplo:

A Augusto le sucedió el emperador

```
<a href="http://es.wikipedia.org/wiki/Tiberio">Tiberio
```

La palabra Tiberio estará remarcada de modo que al hacer clic en ella saltaremos a la URL <http://es.wikipedia.org/wiki/Tiberio>.

Ese ejemplo muestra un salto absoluto, es decir el enlace nos lleva a una dirección URL global usando la notación explicada en el punto anterior.

[3.6.2]enlaces locales relativos

La mayor parte de enlaces de un sitio web en Internet no saltan a otros sitios, sino que son saltos a recursos del mismo servicio web. Es decir se salta muy a menudo a recursos del mismo servidor web.

En ese caso no se indica una URL global tal cual se explicó anteriormente, sino que se indica un salto local. Es decir, un salto a un recurso en nuestro propio servidor web que toma como punto de partida el directorio en el que se encuentra la página web que realiza el salto.

Por ejemplo imaginemos que estamos creando una página web relacionada con la URL <http://www.miservidor.com/noticias/n1.html> y queremos en esa página hacer un enlace a la URL indicada mediante <http://www.miservidor.com/noticias/n2.html> Podríamos hacerlo con el código:

```
<a href="http://www.miservidor.com/noticias/n2.html">  
Ver segunda noticia</a>
```

El enlace funcionaría perfectamente. Pero hay que tener en cuenta que la página a la que saltamos está en el mismo directorio. Si algún día movemos todo nuestro sitio web al dominio *nuevodominio.com* resulta que tendremos que cambiar todo el código para hacer frente a la nueva situación. Por ello lo lógico es usar rutas locales.

El enlace usando una ruta local sería:

```
n2.html">Ver segunda noticia</a>
```

Hay que observar que no se ha indicado protocolo ni ruta (tampoco se indicaría puerto), simplemente se empieza indicando la ruta. Se asume que la ruta

Ejemplos de enlaces con URL local son:

```
<!-- Salto a la página tiberio.html que estará en el mismo
directorio
que la actual -->
<a href="tiberio.html">Tiberio</a>
<!-- Salto a la página tiberio.html que estará dentro del
directorio emperadores que estará, a su vez, dentro del
directorio actual -->
emperadores/tiberio.html">Tiberio</a>
<!-- Salto a la página tiberio.html que estará en el
directorio padre, es decir el directorio que contiene al
actual -->
<a href="../tiberio.html">Tiberio</a>
<!-- Salto a la página tiberio.html que estará en el
directorio emperadores, dentro del directorio padre del
actual-->
<a href="../emperadores/tiberio.html">Tiberio</a>
```

[3.6.3]enlaces locales absolutos

Hay otra posibilidad de enlace local. Por ejemplo:

```
<a href="/tiberio.html">Tiberio</a>
```

Ese enlace (*/tiberio.html*) buscaría la página *tiberio.html* en el directorio raíz del servidor ya que la ruta comienza con el símbolo */*.

Permiten ahorrar mucho texto y son más fáciles de usar que los enlaces relativos, pero en la práctica no se usan mucho, ya que, si movemos de sitio nuestro sitio web, por ejemplo a un directorio interno del servidor, la raíz ya no sería la misma y el enlace no funcionaría.

[3.6.4]enlaces a una posición interna dentro del propio documento

Hay un tipo de enlace que permite posicionarnos en un punto concreto del documento. Este tipo de enlace se denomina interno, ya que salta dentro del propio documento.

Este salto requiere primero marcar la posición a la que deseamos saltar. Antes de HTML 5 se hacía usando el atributo **name** del propio elemento **a**. Pero con HTML

id es un atributo presente en cualquier elemento de la página web. Sirve para identificar a dicho elemento dentro de la página, por ello no podemos repetir valor para ese atributo en ningún otro elemento.



```
<h1 id="tituloPrincipal">Curso de HTML</h1>
<p>
```

Este es un curso que versa sobre cómo escribir páginas web mediante...

...

El párrafo de tipo **h1** está identificado con el identificador *títuloPrincipal*. Gracias a ello si, en otra parte de la página, tenemos el código:

```
<p><a href="#títuloPrincipal">
Volver al título principal</a></p>
```

Al hacer clic en ese enlace (*Volver al título principal*), se usa el identificador como un marcador de posición y así nos colocaremos en dicho párrafo.

[3.6.5]enlaces a una posición interna dentro de otro documento

También podemos realizar un salto a una posición interna a otro documento. Por ejemplo si este enlace:

```
<a href="manual.html">Abrir manual</a>
```

Nos permite abrir una página web llamada **manual.html** (que estará en el mismo directorio que la página actual). Siempre que se abre un documento se muestra el inicio del mismo. Sin embargo este otro enlace:

```
<a href="manual.html#comentarios">
Abrir manual por la zona de comentarios</a>
```

Abre el mismo documento, pero se intenta colocar en la zona marcada por el identificador *comentarios*. Lógicamente, para que esto sea posible, debe existir un elemento marcado con ese identificador en el documento.

Es posible indicar una posición interna en una URL absoluta. Por ejemplo este enlace permite abrir la página dedicada a la ciudad de Palencia de la Wikipedia y colocarse en la zona dedicada a la historia:

```
<a
href="http://es.wikipedia.org/wiki/Palencia#Historia">
Historia de Palencia</a>
```

Atributo download en un enlace

Nuevo atributo de HTML 5 (no funciona ni en Internet Explorer ni en Safari, ni en cualquier versión de navegador que no sea bastante moderna) que permite indicar que al hacer clic en el enlace, no se abrirá en el navegador, sino que se descargará en nuestro directorio predeterminado de descargas. Ejemplo:

```
<a href="documentos/manual.pdf" download>
Descargar manual</a>
```

Se puede indicar un valor para el atributo que se tomará como el nuevo nombre del archivo descargado:

```
<a href="documentos/manual3232001.pdf" download="manual">Desca
manual</a>
```

Aunque el nombre original del archivo que se descarga en el servidor es *manual3232001.pdf*, al descargar el nombre del archivo será *manual.pdf*

[3.7] imágenes

Las imágenes se colocan mediante la etiqueta **img**. El atributo **src** indica la URL (relativa o absoluta) a la imagen. Ejemplo:

```

```

img es una etiqueta sin cierre, la imagen se coloca directamente en la posición de esta etiqueta. Se la considera como texto al maquetarla. De modo que aparece entre el texto si ponemos la etiqueta entre el texto. Colocar de forma adecuada las imágenes respecto al texto es todo un arte que se consigue (una vez más) con ayuda de CSS.

[3.7.1] atributos de la etiqueta *img*

| atributo | |
|---------------|--|
| Alt | <p>Obligatorio. Indica un texto alternativo. Ese texto aparece cuando la imagen no se ha podido cargar (o durante la carga). También suele aparecer cuando arrimamos el cursor a la imagen a fin de informarnos sobre ella.</p> <p>Es un texto también tenido en cuenta por los buscadores a fin de identificar que muestra la imagen.</p> <p>Deberíamos tomarnos este atributo como obligatorio</p> |
| width | <p>Anchura de la imagen. No es aconsejable su uso, ya que si la ampliamos se verá en buena calidad y si la reducimos estaremos cargando una imagen grande para luego mostrarla en pequeño; sería más inteligente reducir primero con un editor de imágenes.</p> <p>En cualquier caso es importante utilizar este atributo (junto con height) para que el navegador sepa de antemano el tamaño de la imagen y así que pueda cargarla correctamente. De este modo si la imagen no se carga, al menos veremos el rectángulo que la misma ocuparía y la página no se desbarataría.</p> |
| height | <p>Altura de la imagen. Tiene las mismas connotaciones que el atributo anterior.</p> |