**Etiquetas y sus atributos**

Bueno, todo eso fue demasiado fácil. Obviamente, esos especialistas informáticos pesados tenían que meterse y complicar las cosas. No basta «solo» con teclear el texto en el editor también tenéis que darle instrucciones al ordenador. Para hacer esto en HTML se usan **etiquetas**.

**Etiquetas**

Las páginas HTML están llenas de las llamadas etiquetas. Aunque estas etiquetas no las pueden ver aquellos que visiten vuestro sitio, ellas permiten que el ordenador entienda que es lo que tiene que mostrar.  
Las etiquetas son fáciles de identificar. Están rodeadas por «corchetes angulares» en otras palabras los símbolos < y >, como este: <etiqueta>

¿Para qué sirven? Indican el tipo de texto al que rodean. Quieren decir por ejemplo: «esto es el título de la página», «esto es una foto», «esto es un párrafo», etc.

Hay dos tipos de etiquetas: etiquetas en pares y etiquetas huérfanas.

**Etiquetas en pares**

Se abren, contienen texto y se cierra posteriormente. Esta es la apariencia que tienen:

<p>Esto es un párrafo</p>

Vais a ver una etiqueta que abre (<párrafo>) y una etiqueta que cierra ( <párrafo>) que indica el final del párrafo. Para el ordenador, esto significa que todo lo que *no* esté entre esas dos etiquetas no es un párrafo.

Esto no es un párrafo <p>Esto es un párrafo</p> Esto no es un párrafo

**Etiquetas huérfanas**

Estas son las etiquetas que se usan con más frecuencia para insertar un elemento en un punto determinado (por ejemplo una imagen). No es necesario definir el comienzo y el final de la imagen. Solo tenéis que decirle al ordenador que «inserte una imagen aquí»

Una etiqueta huérfana se escribe así:

<img></code>

Tomad en cuenta que el final / no es obligatorio. Podríais unicamente teclear <imagen>. Para evitar confundirlas con el primer tipo de etiqueta, sin embargo, los administradores web recomiendan que esto / (*barra*) se debería añadir al final de las etiquetas huérfanas. Así que veréis que añado una /a las etiquetas huérfanas y os recomiendo que hagáis lo mismo; es una buena práctica.

**Atributos**

Los atributos son como las opciones de la etiqueta. Las complementan para proporcionar información adicional. El atributo se coloca después del nombre de la etiqueta que abre y frecuentemente es un valor como este:

<NombreDeLaEtiqueta AtributoDeLaEtiqueta="valor">

¿Para qué sirve? Vamos a usar la etiqueta <imagen /> que acabamos de ver. Por sí misma no tiene mucha utilidad. Podemos añadir un atributo que indique el nombre de la imagen a mostrar:

<img src="foto.jpg" ></code>

El ordenador entonces entiende que tiene que mostrar la imagen que contiene el archivo foto.jpg.

En el caso de una etiqueta que funciona como una «pareja», los atributos únicamente se colocan en la etiqueta de apertura y no en la de cierre. Por ejemplo este código indica que la cita es de Neil Armstrong y data del 21 de julio de 1969:

<blockquote author="Neil Armstrong" date="2107/07/1969">

Es un pequeño paso para el hombre, pero un gran paso para la humanidad.</blockquote>

Todas las etiquetas que acabamos de ver son imaginarias. Vamos a descubrir las etiquetas reales en unos instantes.

**Estructura básica de una página HTML5**

Volvamos a nuestro editor de texto (en mi caso Sublime Text) Únicamente teclead o copiad y pegad el código fuente que está a continuación dentro de Sublime Text. Este código corresponde a los fundamentos de una página web en HTML5:

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Título</title>

</head>

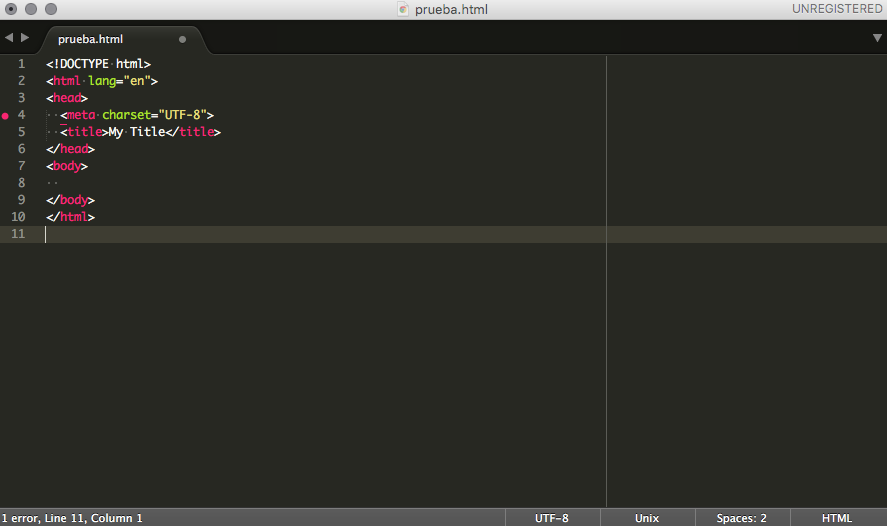
<body>

</body>

</html>

He añadido espacios al comienzo de algunas líneas para «cambiar» las etiquetas. Aunque esto no es obligatorio y no afecta a la forma en la que se muestra la página hace que el código fuente sea más legible. Esto se llama **indentation (sangría)**. En vuestro editor simplemente presionad la tecla Tabpara obtener el mismo resultado.

El resultado que se obtiene cuando se copió dentro del Sublime Text se muestra en la figura siguiente.



El código HTML5 mínimo en Sublime Text

Notaréis que las etiquetas se abren y se cierran en un orden específico. Por ejemplo, la etiqueta <html> es la primera que se abre y también la última que se cierra (al final del código con </html>). *Las etiquetas se tienen que cerrar en el orden inverso en el que se abrieron*. Por ejemplo:

* <html><body></body></html>: **correcto**. Una etiqueta que se abrió dentro de otra etiqueta también se tiene que cerrar dentro de esa etiqueta.
* <html><body></html></body> : **incorrecto**, las etiquetas están entrecruzadas.

¿Eh?, ¿podrías explicarnos todas las etiquetas que acabamos de copiar dentro del editor?

Por supuesto, ya que me lo pedís con tanta educación.  
¡No os asustéis viendo todas esas etiquetas de un golpe, os voy a explicar lo que hacen!

**doctype**

<!DOCTYPE html>

La primera línea se denomina el**doctype**. Es fundamental porque indica que esto es efectivamente una página web HTML.  
No es realmente una etiqueta como las otras (empieza con un signo de exclamación) y se puede considerar como una excepción que confirma la regla.

Esta línea doctype antiguamente era extremadamente compleja. Memorizarla era imposible. En XHTML 1.0 se tenía que escribir:  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">.  
Para el deleite de los administradores web se decidió simplificarla en el HTML5. Cuando veáis una etiqueta doctype corta ( <!DOCTYPE html>) significa que la página está escrita en HTML5.

**La etiqueta</html>**

<html>

</html>

Esta es la etiqueta principal del código. Incluye el contenido completo de la página. ¡Como podéis ver la etiqueta </html> de cierre está situada a la derecha del final del código!

**La cabecera <head> y el cuerpo <body>**

Una página web se compone de dos partes:

* La cabecera <head>: esta sección proporciona información general sobre la página como su título, la codificación (para la gestión de caracteres especiales), etc. Esta sección suele ser bastante corta. La información que contiene la cabecera no se muestra en la página y es simplemente información general concebida para el ordenador. ¡Sin embargo, es muy importante!
* El cuerpo<body>: es aquí donde se encuentra la parte principal de la página. Todo lo que tecleemos aquí se mostrará en la pantalla. La mayor parte de nuestro código se tecleará en el cuerpo.

Por ahora el cuerpo está vacío (volveremos a esto más adelante). Pero vamos a echar un vistazo a las dos etiquetas que contiene la cabecera...

**La codificación ( charset)**

<meta charset="utf-8" ></code>

Esta etiqueta indica la codificación que se usa en el archivo .html.

Sin entrar en detalles porque se complica muy rápido, la codificación especifica cómo se guarda el archivo. Es lo que determina la forma en la que se van a mostrar los caracteres especiales (acentos, caracteres japoneses y chinos, caracteres arábigos, etc.)

Hay varias técnicas de codificación con nombres raros que se usan en función del idioma: ISO-8859-1, OEM 775, Windows-1253, etc. No obstante, actualmente solo se debería usar una: UTF-8. ¡Este método de codificación permite que todos los símbolos de todos los idiomas del mundo se muestren virtualmente sin ningún problema! Esta es la razón por la que indiqué utf-8 en esta etiqueta.

Vuestro archivo se tiene que guardar efectivamente en UTF-8. Podréis ver la codificación en la fila de abajo del editor.

En Sublime Text, si queréis cambiar la codificación seleccionad la codificación asignada en la fila de abajo, haced click en Set Encoding y elegid el formato que prefiráis para que vuestro archivo se guarde en en ese formato desde el principio. Esto únicamente afecta al archivo que esté abierto en el momento.

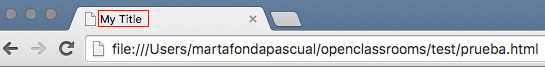
Si tenéis un problema más adelante cuando se muestren los acentos en vuestra página web significa que hay un problema de codificación. Aseguraos de que la pestaña efectivamente indica UTF-8 y que vuestro archivo se guardó en formato UTF-8 (esto lo puede indicar vuestro editor de texto).

**El título principal de la página**

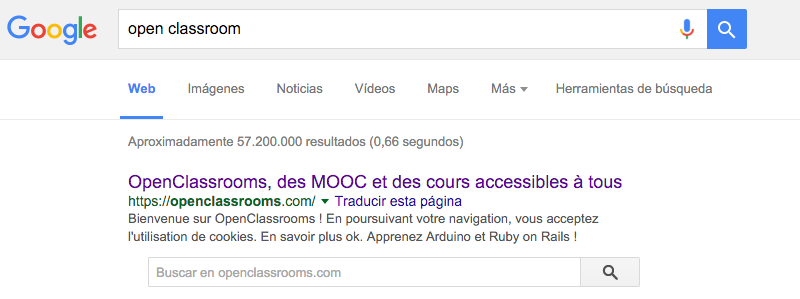
<title>

Este es el título de vuestra página, ¡lo cual es probablemente el elemento más importante! Todas las páginas tienen que tener un título que describe lo que contienen.  
Es aconsejable que el título se mantenga bastante corto (normalmente menos de 100 caracteres).

El título no se muestra en la página sino en su parte superior (habitualmente en la pestaña del navegador). Guardad vuestra página web y abridla en el navegador. Veréis que el título se muestra en la pestaña como se ve en la figura siguiente.

El título de la página figura en la parte superior del navegador

Deberíais saber que el título también aparece en los resultados de búsqueda, como en Google (figura siguiente).

El título de la página figura en las búsquedas en Google

¡Así que elegir el título es importante!

**Comentarios**

En este capítulo aprendimos cómo crear nuestra primera página HTML *real*. Antes de concluir me gustaría añadir una o dos observaciones.

Un **comentario**en HTML es un texto que es simplemente una nota. No se muestra, el ordenador no lo interpreta y no modifica lo que se visualiza en la página.

En pocas palabras, ¿no sirve para nada?

¡Bueno, claro que sí!  
Lo usáis *vosotros* y aquellos que lean el código fuente de la página. Podéis usar los comentarios para dejar instrucciones sobre la forma en la que funciona la página.

¿Qué finalidad tiene esto? Os ayudará a recordar cómo funciona la página cuando volváis al código fuente después de que haya pasado un tiempo sin verlo. No es broma, esto le pasa a todos los administradores web.

**Insertar un comentario**

Un comentario es una etiqueta HTML con un formato muy especial:

<!-- Esto es un comentario -->

Lo podéis poner en la parte que queráis en el código fuente, no afecta a la página, pero lo podéis usar como ayuda para que os situéis en el código especialmente si es largo.

<!DOCTYPE html>

<html>

<head>

<!-- Cabecera de la página -->

<meta charset="utf-8" />

<title>Título</title>

</head>

<body>

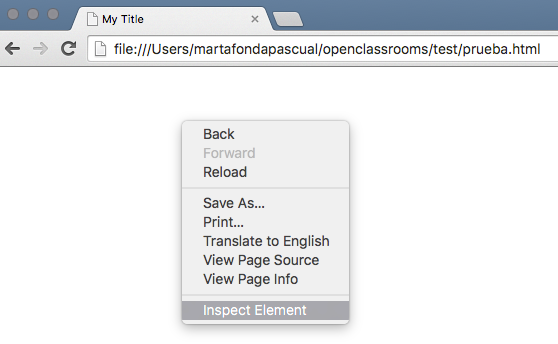
<!-- Cuerpo de la página -->

</body>

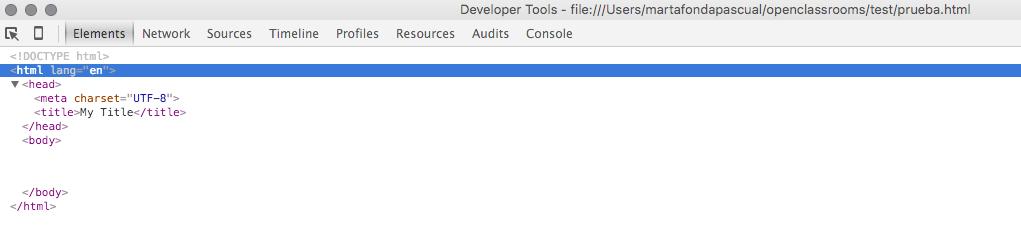
</html>

**¡Todo el mundo puede ver vuestros comentarios... y vuestro código HTML completo!**

Voy a concluir con una observación importante: *todo el mundo puede ver el código HTML de vuestra página*,ya que está *online* en la Red. Simplemente haced clic derecho en la página y seleccionad «Inspeccionar elemento» (en Chrome, ya que el texto exacto puede variar en función del navegador) como se muestra en la figura siguiente.

Menú ver código fuente de la página

A continuación aparece el código fuente (figura siguiente).

Visualizar el código fuente

Podéis probar esto en cualquier sitio web, ¡funciona! 100 % garantizado. Esto tiene una explicación sencilla, el navegador *necesita*el código HTML para saber que mostrar. El código HTML de todos los sitios por consiguiente es público.

¿La moraleja de la historia? Todo el mundo puede ver vuestro código fuente HTML y no podéis hacer nada al respecto. En consecuencia, no incluyáis información sensible en los comentarios, como contraseñas... ¡y encargaros de vuestro código fuente porque puedo comprobar en cualquier momento si habéis seguido el curso al pie de la letra!

Cuando echéis un vistazo al código de algunos sitios web no os desaniméis si os parece largo o no sigue las mismas reglas que he propuesto en este libro. No todos los sitios web están programados en HTML5 ni mucho menos, ¡y el código fuente que algunos administradores web han mal escrito no es un buen ejemplo a seguir!

**En resumen**

* Un editor de texto (nosotros usamos Sublime Text) se usa para crear un archivo con la extensión .html (p. ej. prueba.html). Esto será nuestra página web.
* Este archivo se puede abrir en el navegador web simplemente haciendo doble clic sobre él.
* Tecleamos el contenido de la página junto con las etiquetas HTML dentro del archivo.
* Las etiquetas pueden tener varias formas:
  + <etiqueta> </etiqueta>: se abren y se cierran para delimitar el contenido, por ejemplo el comienzo y el final de un título.
  + <etiqueta />: las etiquetas huérfanas (solo se introduce una etiqueta) se usan para insertar un objeto en un punto determinado, por ejemplo una imagen.
* Las etiquetas van acompañadas algunas veces con atributos que proporcionan información adicional (p. ej. <img src=«foto.jpg» />).
* Una página web se compone de dos secciones principales: una cabecera ( <head>) y un cuerpo ( <body>).
* El código fuente de cualquier página web se puede visualizar al hacer clic derecho y seleccionando «Inspeccionar Elemento»en Google Chrome. En el resto de los navegadores el texto varía pero se accede de la misma forma.

## Organizar el texto

Bueno, la página en blanco es bonita, pero si dejáis vuestra página web así es probable que no tenga mucho éxito.

En este capítulo vamos a conocer un buen número de etiquetas HTML. Algunas etiquetas se usan desde la primera versión de HTML y otras se han introducido más recientemente en HTML5.

En este capítulo vamos a ver en orden sucesivo:

* cómo teclear párrafos;
* cómo organizar la página con títulos;
* cómo resaltar palabras determinadas en el texto;
* cómo organizar la información en listas.

¿Estáis preparados? Entonces, ¡vamos! Veréis que no es complicado.

### Párrafos

El texto en una página web se teclea habitualmente en párrafos. En el lenguaje HTML la etiqueta <p> se usa para delimitar los párrafos.

<p>¡Hola y bienvenidos a mi sitio!</p>

* <p> significa "comienzo del párrafo";
* </p>significa "fin del párrafo";

Como os conté en el capítulo anterior tecleamos el contenido del sitio web entre las etiquetas <body></body>. Así que tenemos que poner nuestro párrafo dentro de estas dos etiquetas y ¡finalmente tendremos nuestra primera página web con texto!

Así que voy a coger exactamente el mismo código del capítulo anterior y lo voy a añadir al párrafo:

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Párrafos</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio!</p>

</body>

</html>

¡Probadlo, veréis los resultados!  
Vale, no es la panacea, pero es un buen comienzo.

Pero no vamos a dejarlo cuando las cosas están yendo tan bien. Ahora vamos a ver algo que es un pelín especial en HTML: el salto de línea. Parece simple, pero realmente no funciona como un procesador de texto común...

#### Crear una línea nueva

En HTML si presionáis la tecla Enter no se crea una línea nueva en la forma en la que estáis acostumbrados. Así que probad este código:

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Pruebas de salto de línea</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio web!

Esta es mi primera prueba así que por favor, tomadlo en cuenta, estoy aprendiendo poco a poco cómo funciona esto.

De momento está un poco vacío, ¡pero volved en 2 o 3 días cuando haya aprendido algunas cosas más y os puedo asegurar que os sorprenderéis!</p>

</body>

</html>

¡¡Todo el texto se muestra en la misma línea aunque haya un salto de línea en el código!! Darle a la tecla del Enterfrenéticamente en el editor de texto, por lo tanto, no sirve de nada.

Como podríais esperar, sin embargo, existe por supuesto una forma de empezar una línea nueva en HTML.

De hecho, si queréis teclear un párrafo nuevo solo necesitáis usar una segunda etiqueta <p>. ¡Así que al final vuestro código HTML debería estar lleno de etiquetas de párrafos!  
¡Así que al final vuestro código HTML debería estar lleno de etiquetas de párrafos!

Un ejemplo:

<html>

<head>

<meta charset="utf-8" />

<title>Párrafos</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio!

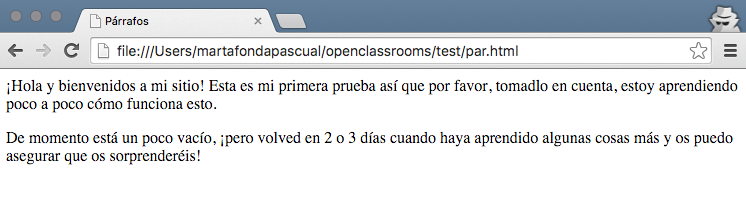
Esta es mi primera prueba así que por favor, tomadlo en cuenta, estoy aprendiendo poco a poco cómo funciona esto.</p>

<p>De momento está un poco vacío, ¡pero volved en 2 o 3 días cuando haya aprendido algunas cosas más y os puedo asegurar que os sorprenderéis!</p>

</body>

</html>

El resultado se muestra en la figura siguiente.

Dos párrafos con 2 etiquetas <p>

Sí, ¿pero qué pasa si solo quiero pasar a la línea siguiente en un párrafo dando comienzo a una línea nueva?

Bueno, ¿adivinad qué? ¡Hay una etiqueta que se llama «salto de línea»!  
Es una etiqueta **huérfana** que solo se usa para indicar el salto de línea: <br />. La tenéis que colocar dentro de un párrafo.

Aquí tenemos la forma de usarla en un código:

<html>

<head>

<meta charset="utf-8" />

<title>Saltos de línea</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio web!<br />

Esta es mi primera prueba así que por favor, tomadlo en cuenta, estoy aprendiendo poco a poco cómo funciona esto.</p>

<p>De momento está un poco vacío, ¡pero volved en 2 o 3 días cuando haya aprendido algunas cosas más y os puedo asegurar que os sorprenderéis!</p>

</body>

</html>

Teóricamente podéis poner varias etiquetas <br />una detrás de la otra para hacer varios saltos de línea, pero esto se considera una mala práctica porque dificulta el mantenimiento del código. Para cambiar un texto con más precisión se usa CSS, que es un lenguaje que complementa el HTML y os hablaré de él un poco más adelante.

Entonces, ¿habéis entendido?

* <p> </p>: para organizar el texto en párrafos;
* <br />: para hacer un salto de línea.

Ahora ya sabemos cómo teclear párrafos, vamos a ver cómo crear títulos.

### Títulos

Ya que el contenido de vuestra página incluye varios párrafos, va a ser difícil que vuestros visitantes se sitúen. Es aquí donde los títulos ganan utilidad.

En HTML se os mima, podéis seleccionar seis niveles de títulos diferentes. Con esto quiero decir que podéis decir «este es un título muy importante», «este es un título un poco menos importante», «este es un título mucho menos importante», etc. Por consiguiente, tenéis seis etiquetas de título diferentes:

* <h1> </h1>: significa «título muy importante". En general se usa para mostrar el título de la página al comienzo de la página.
* <h2> </h2>: significa "título importante".
* <h3> </h3>: asimismo, esto significa un título un poco menos importante (podéis decir «subtítulo» si queréis).
* <h4> </h4>: el título es incluso menos importante.
* <h5> </h5>: el título no es importante.
* <h6> </h6>: el título no es nada importante.

Advertencia: ¡no confundáis esto con la etiqueta <title>! La etiqueta <title> muestra el título de la página en la barra de título del navegador como hemos visto. Los títulos <h1> y similares se usan para crear títulos que se mostrarán en la página web.

Qué no os desconcierten todas estas etiquetas. De hecho, seis niveles de títulos son muchos. En la práctica yo personalmente uso solo las etiquetas <h1>, <h2> y <h3>y muy raras veces las otras (normalmente no necesito seis niveles de títulos diferentes). Vuestro navegador muestra el título realmente importante en caracteres muy grandes, los títulos que son un poco menos importante en caracteres menos grandes, etc.

¡No seleccionéis la etiqueta del título en función del tamaño que se le dé al texto! Es fundamental que estructuréis vuestra página correctamente y que empecéis con un título nivel 1 ( <h1>), luego un título nivel 2 ( <h2>), etc. ¡No tiene que haber ningún subtítulo sin un título principal!  
Si queréis cambiar el tamaño del texto, vamos a aprender cómo se hace esto en CSS más adelante.

Intentad crear una página web con títulos para ver cómo se ve:

<html>

<head>

<meta charset="utf-8" />

<title>Niveles de títulos</title>

</head>

<body>

<h1>Título muy importante</h1>

<h2>Título importante</h2>

<h3>Título un poco menos importante (subtítulo)</h3>

<h4>No es un título importante</h4>

<h5>Título sin importancia</h5>

<h6>Título sin ninguna importancia</h6>

</body>

</html>

Vale, os daré un ejemplo de cómo usar los títulos en una página web (veréis que no uso todas las etiquetas):

<html>

<head>

<meta charset="utf-8" />

<title>Presentación de mi sitio web</title>

</head>

<body>

<h1>¡Hola y bienvenidos a mi sitio web!</h1>

<p>¡Hola y bienvenidos a mi sitio web: SdZ.<br /> ¿Cuál es el contenido del sitio web SdZ?</p>

<h2>Cursos de formación para principiantes</h2>

<p>El sitio web SdZ proporciona cursos de formación (tutoriales) para principiantes; ¡no se requiere ningún conocimiento para hacer estos cursos!</p>

<p>¡Por lo tanto, podéis aprender, sin tener ningún conocimiento previo, a crear un sitio web y a programar y construir entornos 3D!</p>

<h2>Una comunidad activa</h2>

<p>¿Tenéis algún problema o hay algo en el curso que no entendáis? ¿Necesitáis ayuda con la creación de vuestro sitio web?<br /> ¡Visitad los foros! Descubriréis que no estáis solos en esto e indudablemente encontraréis a alguien que os ayudará encantado a solucionar vuestro problema.</p>

</body>

</html>

¡Esta es una página web que está cogiendo forma!

¡Sí, pero quiero centrar el título, escribirlo en rojo y subrayarlo!

Haremos todo eso cuando aprendamos CSS en la segunda parte del curso. Me gustaría indicar que <h1> no significa «Times New Roman, tamaño 16 pt.», sino « título importante ».

Cuando uséis el lenguaje CSS podréis decir: «quiero de los títulos importantes estén centrados, sean rojos y estén subrayados». De momento solo estamos estructurando la página en HTML. Estamos escribiendo el contenido antes de pasarlo bien dándole formato.

### Resaltar

Algunas palabras en los párrafos son algunas veces más importantes que otras y querréis que resalten. El HTML proporciona varias formas para resaltar el texto de la página.

#### Enfatizar

Para enfatizarel texto, tenéis que usar la etiqueta <em> </em>.  
Es muy simple de usar, ¡todo lo que tenéis que hacer en encerrar las palabras a enfatizar dentro de estas etiquetas! Voy a volver a usar el ejemplo de hace un momento y a enfatizar algunas palabras:

<html>

<head>

<meta charset="utf-8" />

<title>Enfatizar</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio web!<br />

Esta es mi primera prueba así que por favor, <em>tomadlo en cuenta</em>, estoy aprendiendo poco a poco cómo funciona esto.</p>

</body>

</html>

Como podéis ver al usar la etiqueta <em>esta tiene un efecto, coloca el texto en cursiva. De hecho, es el navegador él que elije la forma en la que se muestran las palabras. Le diremos que las palabras son bastante importantes y para enfatizar esta información cambia la apariencia del texto con el uso de cursiva.

#### Enfatizar considerablemente

Para enfatizar considerablemente un texto, podéis usar la etiqueta <strong> que significa «énfasis fuerte» o si preferís «importante». Se usa exactamente de la misma forma que <em>:

<html>

<head>

<meta charset="utf-8" />

<title>Énfasis fuerte</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio web!<br />

Esta es mi primera prueba así que por favor, <strong>tomadlo en cuenta</strong>, estoy aprendiendo poco a poco cómo funciona esto.</p>

</body>

</html>

Sin lugar a dudas veréis que el texto aparecerá en negrita. También en este caso la negrita es un solo una consecuencia . El navegador eligió mostrar las palabras importantes en negrita para hacer que destaquen más.

La etiqueta <strong> no significa «negrita» sino «importante». Podéis decidir más adelante en CSS si queréis mostrar las palabras «importantes» de una forma diferente a la negrita.

#### Marcado de texto

La etiqueta <mark> se usa para resaltar visualmente una parte del texto. El extracto no se tiene que considerar necesariamente importante, pero queréis que destaque del resto del texto. Esto puede ser útil para resaltar texto que sea relevante, por ejemplo después de buscar en vuestra página web.

<html>

<head>

<meta charset="utf-8" />

<title>Marcado de texto</title>

</head>

<body>

<p>¡Hola y bienvenidos a mi sitio web!<br />

Esta es mi primera prueba así que por favor, <mark>tomadlo en cuenta</mark>, estoy aprendiendo poco a poco cómo funciona esto.</p>

</body>

</html>

Por definición <mark> tiene el efecto de resaltar el texto. Podemos cambiar en CSS la forma en la que se muestra (decidir resaltar en otro color, poner el texto en un cuadro, etc.) Es el mismo principio como el que os conté para las etiquetas anteriores: indican la relevancia de las palabras y no la forma en la que deberían mostrarse.

#### No os olvidéis: HTML para el contenido, CSS para el estilo

Puede parecer que estoy rizando el rizo, pero es muy importante que entendáis esto correctamente, ya que los principiantes a menudo cometen este gran error en esta etapa. Ven las etiquetas<em>, <strong>, <mark>… y se cuentan a sí mismos: «¡Genial, ya sé cómo poner el texto en cursiva, negrita y subrayarlo en HTML!»

Y aún así... ¡esto no es para lo que se usan estas etiquetas! Ya sé, ya sé, me vais a decir: «sí, pero cuando uso <strong>el texto aparece en negrita, entonces se tiene que usar para poner en negrita». Aún así, es un error creer que esta etiqueta se usa para hacer eso.

La finalidad de las etiquetas es indicar la relevanciadel texto. Por tanto, <strong> le dice al ordenador que «este texto es importante». Eso es todo.  
Y para mostrar que el texto es importante, el ordenador decide ponerlo en negrita, ¡pero podría estar escrito igualmente en rojo! Aunque la mayoría de los navegadores muestren el texto importante en negrita nada les exige que lo hagan así.

No lo entiendo. ¿Qué sentido tiene que el ordenador sepa que un texto es importante?  
¡No es lo suficientemente inteligente para entender!

¡Pensadlo otra vez! Muchos programas analizan el código fuente de las páginas web y aún más importante los robots de los motores de búsqueda. Estos robots rastrean la Red y leen el código HTML de todos los sitios. Este es el caso de los robots de Google y Bing por ejemplo. Como ellos lo ven, las palabras clave «importantes» suelen tener más valor, así que si alguien hace una búsqueda con estas palabras es más probable que se encuentre con vuestro sitio web.  
Esto es por supuesto una explicación superficial y no deberíais concluir que el uso descontrolado de la etiqueta <strong> de alguna forma va a mejorar vuestro ranking de búsqueda. Solo tenéis que confiar en los ordenadores, ellos entienden lo que significa un texto «importante» y pueden usar esa información.

Entonces, ¿cómo ponéis concretamente un texto en negrita o lo escribís en rojo y todo eso?

Todo eso se hace en CSS. Recordad:

* HTML define el contenido (contenido lógico de elementos);
* CSS define el estilo (apariencia).

Veremos el CSS más adelante de momento nos centraremos en el HTML y sus etiquetas, cada una de ellas tienen un significado concreto.

### Listas

Las listas normalmente nos permiten organizar el texto y ordenar la información.  
Vamos a descubrir dos tipos de listas aquí:

* listas no ordenadas o listas con viñetas;
* listas ordenadas o listas enumeradas.

#### Listas no ordenadas

Una lista no ordenada tiene esta apariencia:

* Fresas
* Frambuesas
* Cerezas

Es un sistema que nos permite crear una lista sin ningún concepto de orden (no hay «primero» o «último»). Crear una lista no ordenada es muy sencillo. Simplemente usad la etiqueta <ul> cerrada posteriormente con </ul>.  
Entonces, empezad introduciendo esto:

<ul></ul>

Y ahora esto es lo que vamos a hacer: vamos a escribir cada elemento de la lista dentro de las etiquetas <li></li>. Cada una de estas etiquetas tienen que estar situadas entre <ul> y </ul>. Vais a entender esto de inmediato con este ejemplo:

<ul>

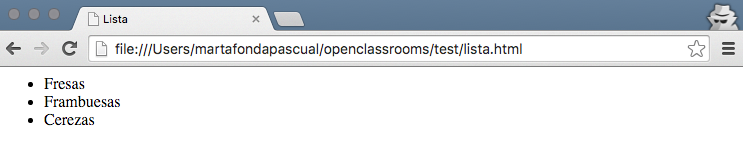
<li>Fresas</li>

<li>Frambuesas</li>

<li>Cerezas</li>

</ul>

El resultado se muestra en la figura siguiente.

Una lista no ordenada

Tened en cuenta que la lista no se ha colocado dentro de <body></body>. Para mantener el código de la página legible de ahora en adelante no voy a incluir en él todo lo que veamos.

Así que recordad estas dos etiquetas:

* <ul></ul> delimita la lista completa;
* <li></li> define un elemento de la lista (una viñeta).

Podéis incluir todos los elementos que queráis, no únicamente tres.

¡Así que ahora sabéis cómo crear una lista con viñetas! No es difícil una vez que entendáis cómo anidar las etiquetas.

Para aquellos que tengáis que crear listas complejas, recordad que podéis anidar listas no ordenadas (crear una lista no ordenada **dentro**de una lista no ordenada). Si queréis hacer esto abrid una segunda etiqueta <ul> y poned **dentro** un elemento<li></li>.  
Si cerráis las etiquetas en el orden correcto no tendréis ningún problema. Sin embargo, tened cuidado porque esta técnica no es fácil dominar.

#### Lista ordenada

Una lista ordenada funciona de la misma forma únicamente cambia la etiqueta: tenéis que sustituir <ul></ul> por <ol></ol>.  
No cambiéis nada dentro de la lista: siempre tenéis que usar las etiquetas <li></li> para delimitar los elementos de la lista.

El orden en el que colocáis los elementos en la lista es importante. El primero <li></li> será el elemento n.º 1, el segundo será el elemento n.º 2, etc.

Como es especialmente intuitivo os voy a dejar que admiréis la simplicidad de este ejemplo (los resultados se muestran en la figura de abajo):

<h1>Mi día</h1>

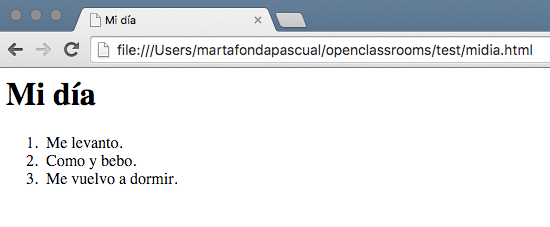
<ol>

<li>Me levanto.</li>

<li>Como y bebo.</li>

<li>Me vuelvo a dormir.</li>

</ol>

Una lista ordenada

Comparado con el ejemplo anterior todo lo que ha cambiado es la etiqueta <ol>.

Para vuestra información existe un tercer tipo de lista mucho menos frecuente: la lista de definiciones. Consiste en el uso de las etiquetas <dl> para delimitar la lista, <dt> para delimitar el término y <dd> para delimitar la definición del término.

#### En resumen

* HTML tiene muchas etiquetas que nos permiten organizar el texto de la página. Estas etiquetas proporcionan indicaciones, como «este es un párrafo», «esto es un título», etc.
* La etiqueta <p> </p> define los párrafos y la etiqueta <br /> el salto de línea.
* Hay seis niveles de títulos, desde <h1> </h1> hasta <h6> </h6>, los cuales se usan en función de la importancia del título.
* Podéis resaltar determinadas palabras con las etiquetas <strong>, <em> y <mark>.
* Para crear listas, tenéis que usar la etiqueta <ul> (lista no ordenada con viñetas) o la etiqueta <ol> (lista ordenada). Los elementos se incluyen dentro de la lista con la etiqueta <li> por cada elemento.

## Crear enlaces

En el capítulo anterior habéis aprendido cómo crear una página HTML sencilla. Vale, en realidad no era para tanto, pero era una página HTML real a pesar de todo.

Como sabréis un sitio web está formado por varias páginas. ¿Cómo vais de una página a otra? ¡Usando enlaces por supuesto! Entonces, la forma en la que se crean los enlaces entre las páginas es lo que vais a aprender en este capítulo.

Supongo que todos sabéis lo que es un enlace: es un texto en el que hacéis clic para ir a otra página.  
Podéis crear un enlace para ir de la página a.html a la página b.html, pero también podéis crear un enlace a otro sitio web, por ejemplo http://www.google.com. Veremos que ambos casos funcionan de la misma forma.

### Un enlace a otro sitio web

Estos enlaces se reconocen fácilmente en una página: están escritos de una forma diferente (de forma predeterminada en azul y subrayados) y un cursor que se parece a una mano aparece cuando se pone el ratón sobre ellos.

Os sugiero que intentéis codificar un enlace que os lleve a Google como se muestra en la figura siguiente.

Enlace a GoogleEnlace a Google

Para insertar un enlace, la etiqueta que vamos a usar es muy fácil de recordar: <a>. Sin embargo, necesita un atributo, href, que indica la página a la que debería llevar el enlace.

Por ejemplo el código que está a continuación lleva a Google, localizado en la dirección http://www.google.com:

<a href="http://www.google.com">Google</a>

Vamos a incluir este enlace en un párrafo. Entonces, así es cómo reproducís el ejemplo en la figura siguiente:

<p>Hola. ¿Queréis visitar el sitio web <a href="http://www.google.com">Google</a>?<br />

¡Es un buen sitio web! ;-)</p>

Por defecto el enlace se resalta en azul. Si ya habéis abierto la página el enlace se muestra en morado.  
Veremos cómo cambiar este aspecto cuando estudiemos CSS.

Si queréis crear un enlace a otra página podéis simplemente copiar y pegar su dirección (llamada URL) en http://. Tomad en cuenta que algunas veces algunos enlaces empiezan con https:// (sitios web seguros) u otros prefijos ( ftp://, etc.).

Si creáis un enlace a un sitio web con una dirección un poco extraña  
no veréis la diferencia, pero es necesario que creéis una página web correctamente en HTML5.

Los enlace que hemos visto se denominan **enlaces absolutos** debido a que muestran la dirección completa. Ahora vamos a ver que podemos escribir los enlaces de algún modo diferente, lo que será útil para crear enlaces entre las páginas de nuestro sitio web.

### Un enlace a otra página de vuestro sitio web

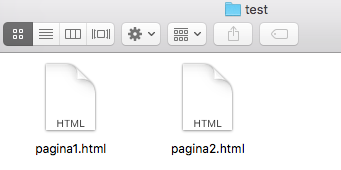
Acabamos de aprender la forma de crear enlaces a sitios web que existen. Pero estoy seguro de que os gustaría crear enlaces entre las páginas de vuestro sitio web, ¿verdad?

Sí, ¿qué tengo que hacer exactamente para crear un enlace a otra página de mi sitio web? ¡No sé su dirección http://…… ya que acabo de empezar a crear mi sitio web aquí! No tengo una dirección.

De hecho, de momento habéis creado vuestro sitio web en vuestro ordenador. Sois los únicos que podéis verlo y aún no tiene una «dirección web» que empiece con http:// como la mayoría de los sitios web. Afortunadamente esto no impedirá que trabajemos.

#### Dos páginas ubicadas en la misma carpeta

Para empezar vamos a crear dos archivos que correspondan a dos páginas HTML diferentes. Como estoy inspirado os propongo que los llamemos pagina1.html y pagina2.html. Por tanto, tendremos estos dos archivos en nuestro disco duro en la misma carpeta(ver figura a continuación).

Varios archivos HTML en la misma carpeta

¿Cómo hacéis un enlace de la página 1 a la página 2 sin que tengáis una dirección en http://? De hecho, si ambos enlaces están ubicados en la misma carpeta es fácil, simplemente escribid el nombre del archivo al que queráis moveos como el enlace de destino. Por ejemplo: <a href="pagina2.html">. Esto se denomina **enlace relativo**.

Aquí tenemos el código que vamos a usar en nuestros archivos página1.html y página2.html.

##### página1.html

<p>Hola. ¿Queréis ver la <a href="pagina2.html">página2</a> ?</p>

##### página2.html

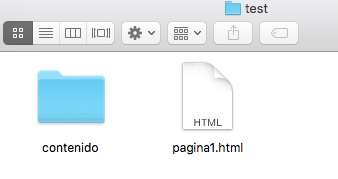
La página 2 (página de llegada) simplemente mostrará un mensaje que indicará que efectivamente habéis llegado a la página 2:

<h1>¡Bienvenidos a la página 2 !</h1>

#### Dos páginas ubicadas en carpetas diferentes

Las cosas se ponen un poco más difíciles si las páginas se encuentran en carpetas diferentes. Lo ideal sería que no estuvieran muy lejos las unas de las otras (por ejemplo en una subcarpeta).

Vamos a imaginar que la pagina2.htmlse encuentra en una subcarpeta denominada contenido como en la figura siguiente.

El archivo página2.html se encuentra en la carpeta contenido

En este caso el enlace debería estar escrito de esta forma:

<a href="contenido/pagina2.html">

Si hay varias carpetas deberíais escribir esto:

<a href="contenido/otracarpeta/pagina2.html">

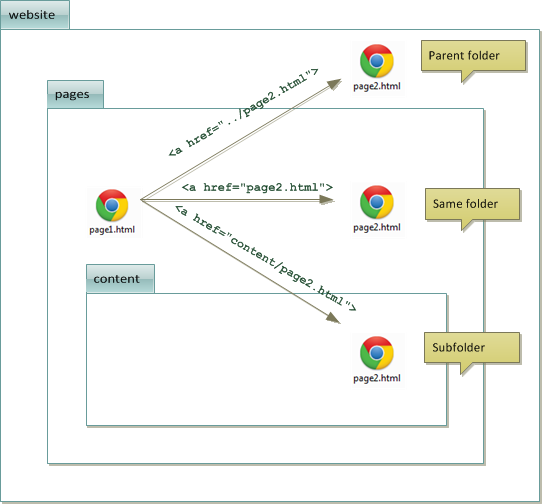
¿Y qué ocurre si el archivo no está ubicado en una subcarpeta sino en una carpeta raíz? ¿Qué hacemos entonces?

Si tu archivo de destino se encuentra en una carpeta ubicada "más arriba" en la estructura de árbol, tienes que escribir dos puntos como estos:

<a href="../pagina2.html">

#### Resumen gráfico

Los enlaces relativos no son muy difíciles de utilizar una vez que hayas entendido el procedimiento. Para saber cómo escribir tu enlace, solo tienes que mirar el "nivel de carpeta" en el que se encuentra el archivo de destino. La siguiente imagen resume los diferentes enlaces relativos posibles.

Enlaces relativos

### Un enlace a un hipervínculo

Un **hipervínculo** es una especie de punto de referencia que puedes incluir en tus páginas HTML cuando son muy largas.  
De hecho, pueden ser útiles para crear un enlace a un punto situado más abajo en la misma página para permitir a los visitantes saltar directamente a la parte que les interesa.

Para crear un hipervínculo nuevo, Puede ser cualquier etiqueta, por ejemplo un título.  
Para dar un nombre al hipervínculo, se utiliza el atributo id. Entonces podremos usarlo para crear un enlace a este hipervínculo. Por ejemplo:

<h2 id="my\_anchor">Título</h2>

Ahora solo tienes que crear un enlace como siempre, pero esta vez el atributo href incluirá una almohadilla (#) seguida del nombre del enlace. Ejemplo:

<a href="#my\_anchor">Ir al hipervínculo</a>

Normalmente, si haces clic en el enlace, te llevará a un punto más abajo de la misma página (siempre que la página contenga texto suficiente para que las barras de desplazamiento se muevan automáticamente).  
Este es un ejemplo de página con mucho texto que utiliza hipervínculos (he utilizado texto antiguo para rellenar la página):

<h1>>Mi gran página</h1>

<p>

Ir directamente a la parte que trata sobre:<br />

<a href="#cocina">La cocina</a><br />

<a href="#rodillos">Rodillos</a><br />

<a href="#tiro con arco">Tiro con arco</a><br />

</p>

<h2 id="cocina">La cocina</h2>

<p>... (mucho texto) ...</p>

<h2 id="rodillos">Rodillos</h2>

<p>... (mucho texto) ...</p>

<h2 id="tiro con arco">Tiro con arco</h2>

<p>... (mucho texto) ...</p>

Si no ocurre nada cuando hagamos clic sobre los enlaces, esto significa que no hay suficiente texto. En este caso, puedes optar por añadir texto antiguo a la página para que tenga (incluso) más texto, o bien reducir el tamaño de la ventana del navegador para que aparezcan las barras de desplazamiento en el lateral.

Para dar un nombre "único" a una etiqueta, se utiliza el atributo id. Y créeme, todavía no lo has oído todo sobre este atributo. Aquí se utiliza para crear un enlace a un hipervínculo, pero en CSS, nos será muy útil para "marcar" una etiqueta específica. Ya lo veremos.  
Sin embargo, evita crear ids con espacios o caracteres especiales. Siempre que sea posible, utiliza solo letras y número para que el valor sea reconocido por todos los navegadores.

#### Enlace a un hipervínculo situado en otra página

Ok, estáis avisados. ¡Esto va a ser un auténtico megamix!

La idea es crear un enlace que abra otra página Y que te lleve directamente a un hipervínculo situado más abajo de esta página.  
En la práctica es muy fácil hacerlo: escribe el nombre de la página, seguido por una almohadilla ( #), seguido del nombre del hipervínculo.

Por ejemplo: <a href="anchors.html#rodillos">

... te llevará a la página anchors.html, directamente al hipervínculo llamado rodillos.

Esta página contiene tres enlaces, cada uno de los cuales lleva a uno de los hipervínculos de la página del ejemplo anterior:

<h1>Megamix</h1>

<p>

Referencia a otro lugar de la misma página:<br />

<a href="anchors.html#kitchen">La cocina</a><br />

<a href="anchors.html#rollers">Rodillos</a><br />

<a href="anchors.html#arc">Tiro con arco</a><br />

</p>

### Ejemplos prácticos del uso de enlaces

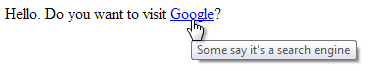
Ahora voy a intentar mostraros varios ejemplos prácticos del uso de enlaces. Por ejemplo, ¿sabías que es muy fácil crear enlaces que inician una descarga? ¿Que crean un nuevo correo electrónico? ¿Que abren una ventana nueva?

¿No? Bueno, pues vamos a ver todo esto.

#### Un enlace que muestra una descripción emergente

Puedes utilizar el atributo title que muestra una descripción emergente cuando coloques el curso sobre el enlace. Este atributo es opcional

Debes obtener un resultado similar al de la imagen siguiente.

Una descripción emergente

La descripción emergente puede ser útil para proporcionar información a los visitantes incluso antes de que hayan hecho clic sobre el enlace.  
Aquí tenemos la forma de reproducir este resultado:

<p>Hola. ¿Queréis visitar <a href="http://www.google.com" title="Dicen que es un motor de búsqueda">Google</a> ?</p>

#### Un enlace que abre una ventana nueva

Un enlace se puede "obligar" a abrir una nueva ventana. Para hacer esto, se añade target="\_blank" a la etiqueta <a>.

<p>Hola. ¿Queréis visitar <a href="http://www.google.com" target="\_blank">Google</a> ?<br />

Este sitio web se mostrará en otra ventana.</p>

Dependiendo de la configuración del navegador, la página se abrirá en una ventana nueva o en una pestaña nueva. No puedes elegir entre que se abra en una ventana nueva o en una pestaña nueva.

Sin embargo, no se recomienda abusar de esta técnica porque afecta a la navegación. Los visitantes pueden decidir si quieren abrir el enlace en una ventana nueva. Pueden pulsar Shift + Clic sobre el enlace para abrirlo en una ventana nueva o Comando + Clic para abrirlo en una pestaña nueva.

#### Un enlace para enviar un correo electrónico

Si quieres que tus visitantes puedan enviarte un correo electrónico, puedes usar enlaces del tipo mailto. A nivel de etiqueta no cambia nada. Solo cambias el valor del atributo href así:

<p><a href="mailto:nombre@email.com">¡Envíame un correo electrónico!</a></p>

Así que solo tienes que empezar el enlace con mailto: e introducir la dirección de correo electrónico a la que quieres que te escriban. Si haces clic en el enlace, se abre un nuevo mensaje en blanco, listo para ser enviado a tu dirección de correo electrónico.

#### Un enlace para descargar un archivo

Muchos de vosotros os estaréis preguntando cómo funciona esto para descargar un archivo... En realidad, el procedimiento es exactamente el mismo que para enlazar a una página web, pero esta vez tenemos que indicar el nombre del archivo que queremos que se descargue.

Por ejemplo, imaginad que queréis descargar myfile.zip. Solo tenéis que colocar este archivo en la misma carpeta que vuestra página web (o en una subcarpeta) y crear un enlace a este archivo:

<p><a href="myfile.zip">Descargar el archivo</a></p>

¡Eso es todo! Cuando el navegador vea que no hay ninguna página web que mostrar, comenzará el proceso de descarga cuando el usuario haga clic sobre el enlace.

#### En resumen

* Los enlaces sirven para moverse de una página a otra y por defecto son texto subrayado de color azul.
* Para introducir un enlace, utiliza la etiqueta <a> con el atributo href para indicar la dirección de la página de destino. Ejemplo: <a href="http://www.google.com">.
* Puedes crear un enlace a otra página de tu sitio web con tan solo escribir el nombre del archivo: <a href="pagina2.html">.
* Los enlaces también se pueden usar para saltar a otras secciones de la misma página. Tienes que crear un hipervínculo con el atributo id para "marcar" una sección de la página y después crear un enlace al hipervínculo como este: <a href="#anchor">.

## Imágenes

¿Insertar una imagen en el sitio web? Ya veréis que no es nada complicado... Bueno, casi nada. En los sitios web se pueden utilizar varios **formatos** de imágenes diferentes y no debemos elegirlos al azar. De hecho, a veces las imágenes son grandes y tardan mucho en descargarse, lo que ralentiza el tiempo de carga de la página (¡mucho más que el texto!).

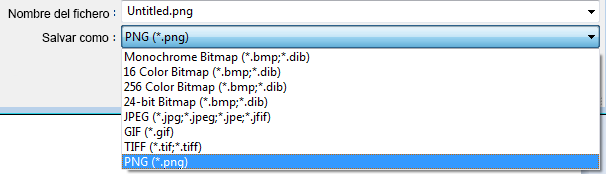
Entonces, para asegurarte de que tus páginas son legibles y rápidas de descargar, sigue mi consejo.

### Formatos de imagen

¿Sabes qué es el formato de imagen?

Cuando tienes "acceso" a una imagen, puedes guardarla en varios "formatos" diferentes. El tamaño de la imagen (en KB o incluso MB) variará dependiendo del formato seleccionado y la calidad de la imagen cambiará.

Por ejemplo, el programa de dibujo Paint (aunque por supuesto no sea el mejor) te permite elegir entre diferentes formatos cuando guardas una imagen (imagen siguiente).

Formatos de imagen que ofrece Paint

Algunos formatos son mas adecuados que otros según la imagen (foto, dibujo, imagen en movimiento, etc.). Nuestro objetivo es repasar los formatos utilizados en la Red para que tengáis una idea de ellos y sepáis elegir el más adecuado para vuestra imagen. No os preocupéis. No existen muchos formatos diferentes, así que no tardaremos mucho.

Todas las imágenes publicadas en Internet tienen un punto en común: están **comprimidas**. Esto significa que el ordenador hace cálculos para ampliarlas y que se carguen más rápido.

#### JPEG

Las imágenes en formato JPEG (Joint Photographic Expert Group) son muy comunes en la Red. Este formato está diseñado para reducir el tamaño de almacenamiento de las fotos (en otras palabras, el tamaño del archivo asociado), que puede contener más de 16 millones de colores diferentes. La siguiente ilustración es una imagen guardada en formato JPEG.

La foto de una montaña en JPEG

Las imágenes JPEG se guardan con la extensión.jpgo.jpeg.

Tened en cuenta que el formato JPEG reduce la calidad de la imagen ligeramente, de un modo generalmente imperceptible. Esto es lo que le hace tan efectivo en la reducción del tamaño de almacenamiento de las fotos.  
Cuando se trata de una foto, normalmente no se detecta la pérdida de calidad. Si no es una foto, la imagen puede parecer que está un poco borrosa. En este caso, es mejor usar el formato PNG.

#### PNG

PNG ( Portable Network Graphics) es el formato más nuevo de todos. Este formato es adecuado para la mayoría de gráficos (me atrevería a decir que para "todo lo que no sea una foto"). El formato PNG tiene dos grandes ventajas: puede ser transparente y no afectar a la calidad de la imagen.

El formato PNG se inventó para competir con otro formato, GIF, en tiempos en los que había que pagar derechos de autor por el uso de GIF. Desde entonces, el formato PNG ha cambiado mucho y se ha convertido en el formato más potente para guardar la mayoría de las imágenes.

PNG está disponible en dos versiones, dependiendo del número de colores que incluya la imagen:

* PNG de 8 bits: 256 colores;
* PNG de 24 bits: 16 millones de colores (como una imagen JPEG).

La siguiente ilustración es una imagen PNG de 24 bits, que representa a nuestra famosa mascota Zozor.

Zozor en formato PNG

Las versión antigua de Internet Explorer (IE6) no muestra correctamente imágenes PNG de 24 bits transparentes. Aunque este navegador cada vez se utiliza menos, hay que tener en cuenta esta información.

De hecho, si una imagen PNG de 24 bits puede mostrar el mismo número de colores que una imagen JPEG, y también puede hacerse más transparente sin afectar a la calidad de la imagen... ¿para qué queremos el formato JPEG?

La compresión de fotos es mas potente con JPEG. Una foto guardada en JPEG se cargará siempre más rápido que si se guarda en PNG. Por eso os recomiendo que siempre mantengáis el formato JPEG en vuestras fotos.

#### GIF

Aunque este formato es bastante antiguo, se ha utilizado mucho (y todavía se usa). El formato PNG es mucho mejor que el GIF en general: normalmente las imágenes tienen un tamaño de almacenamiento mucho menor y su transparencia es de mejor calidad. Por eso recomiendo que uséis PNG todo lo posible.

GIF está limitado a 256 colores (mientras que PNG puede llegar hasta varios millones de colores).

Sin embargo, el formato GIF tiene una ventaja que no tiene el PNG: puede ser animado (ver ilustración siguiente).

Una imagen GIF animada

#### Hay un formato apropiado para cada imagen

Para resumir, el formato se debe escoger dependiendo de la imagen que tengas, tal y como se indica a continuación:

* **Una foto**: usar JPEG.
* **Cualquier gráfico que no tenga muchos colores** (menos de 256): usar PNG de 8 bits o posiblemente GIF.
* **Cualquier gráfico con muchos colores** (más de 256): usar formato PNG de 24 bits.
* **Una imagen animada**: usar GIF animado.

#### Errores a evitar

##### Evitar otros formatos

Otros formatos que no se nombran aquí, tales como el formato BITMAP (\*.bmp), deben evitarse porque normalmente no están comprimidos y, por tanto, son demasiado grandes. No todos son apropiados para la Red. Aunque puedes incluirlos en tu sitio web, tardarán muchísimo tiempo en cargarse.

##### Elige bien el nombre de tu imagen

Si quieres evitarte problemas, acostúmbrate a guardar tus archivos con nombres en minúsculas, sin espacios ni tildes, por ejemplo:my\_image.png.

Puedes sustituir los espacios por un guión bajo ("\_") como he hecho yo aquí.

### Insertar una imagen

Ahora volvamos al código HTML para descubrir cómo incluir imágenes en nuestras páginas web.

#### Insertar una imagen

¿Cuál es la famosa etiqueta que utilizamos para insertar una imagen? ¡Es…<img />!

Es una etiqueta **huérfana** (como<br />). Esto quiere decir que no tienes que introducir dos etiquetas como en el caso de la mayoría de etiquetas que hemos visto hasta ahora. De hecho, no tenemos que definir una porción de texto; lo único que queremos es colocar una imagen en un punto específico.

La etiqueta debe ir acompañada de dos atributos obligatorios:

* src: que indica la ubicación en la que quieres insertar la imagen. Puedes incluir una ruta absoluta (e.g.http://www.website.com/flor.png) o una ruta relativa (que es lo que suele hacerse). Así que si la imagen está en la subcarpetaimagenes, tienes que introducir:src="imagenes/flor.png"
* alt: que significa "texto alternativo". Un texto alternativo a la imagen, en otras palabras, un texto corto que describe lo que contiene la imagen, debe indicarse siempre. Este texto se mostrará en lugar de la imagen si esta última no se puede descargar (esto ocurre), o en los navegadores de personas con discapacidad visual que lamentablemente no pueden "ver" la imagen. También ayuda a los robots de motores de búsqueda a buscar imágenes. Para la flor, por ejemplo, se debería escribir lo siguiente:alt="Una flor"..

Las imágenes deben colocarse obligatoriamente dentro de un párrafo (<p></p>). Aquí tenéis un ejemplo de imagen insertada:

<p>

Esta es una foto que hice durante mis pasadas vacaciones en las montañas:<br />

<img src="imagenes/monte.jpg" alt="Foto de la montaña" />

</p>

Resumiendo, es muy fácil insertar una imagen siempre y cuando puedas indicar donde está ubicada, como aprendimos a hacer con los enlaces.  
La mayor "dificultad" (si es que podemos llamarlo dificultad) es seleccionar el formato de imagen adecuado. En este caso es una foto, por lo que obviamente se utiliza el formato JPEG.

Lo diré de nuevo: debemos evitar las letras mayúsculas y los espacios a toda costa en los nombres de archivos y carpetas. Esta ruta causaría problemas:  
"Website imagenes/Mi súper imagen.jpg".  
Los espacios se deben borrar (o reemplazados por el símbolo "\_") y todo debe ir en letras minúsculas como aquí:  
"website\_imagenes/mi\_super\_imagen.jpg".  
No lo escribas mal, porque si la imagen no se muestra, es casi seguro porque la ruta es incorrecta. Elige los nombres más simples posibles para los nombres de tus carpetas y archivos, y todo irá bien.

#### Añadir una descripción emergente

El atributo utilizado para mostrar una descripción emergente es el mismo que para los enlaces:title. Este atributo es opcional (al contrario quealt).

Aquí tenemos un ejemplo:

<p>

Esta es una foto que hice durante mis pasadas vacaciones en las montañas:<br />

<img src="imagenes/monte.jpg" alt="Foto de la montaña" title="¡Me gusta la nieve!" />

</p>

Coloca el puntero del ratón sobre la foto para que aparezca la descripción emergente.

#### Miniatura clicable

Si la imagen es muy grande, es recomendable mostrar su vista en miniatura en tu sitio web. Después añade un enlace a la vista en miniatura para que los visitantes puedan ver la imagen en su tamaño original.

Existen millones de programas para crear miniaturas de imágenes. Yo personalmente utilizo [Easy Thumbnails](http://www.fookes.com/ezthumbs/). Por tanto, tendré dos versiones de mi foto (como se muestra en la siguiente ilustración): la vista en miniatura y la imagen original.

La vista en miniatura y la imagen real

Meto las dos en una carpeta llamada, por ejemplo,img. Muestro la versiónmonte\_peque.jpgen mi página y creo un enlace amonte.jpgpara que se muestre la versión ampliada al hacer clic sobre la vista en miniatura.

Aquí está el código HTML que utilizo para esto:

<p>

¿Quieres ver la imagen en su tamaño original? ¡Haz clic sobre ella!<br />

<a href="img/monte.jpg"><img src="img/monte\_peque.jpg" alt="Foto de la montaña" title="Clic para ampliar" /></a>

</p>

A veces, algunos buscadores muestran un marco azul (o morado) muy poco atractivo alrededor de la imagen clicable.  
Afortunadamente, podemos quitar rápidamente este marco con CSS.

### Figuras

Al leer este libro, ya habrás visto varios ejemplos de **figuras**. Son elementos que realzan el texto para complementar la información de la página.

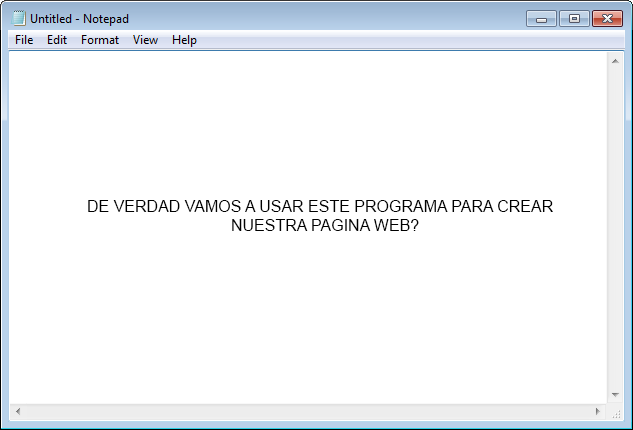
Las figuras pueden ser de diferentes tipos:

* imágenes
* código fuente;
* comillas;
* ...

Resumiendo, cualquier cosa que ilustre el texto es una figura. Aunque en este caso estamos usando imágenes, las figuras, al contrario de lo que muchos estaréis pensando, no tienen por qué ser imágenes necesariamente: un código fuente también ilustra el texto.

#### Crear una figura

Tomemos de nuevo el ejemplo de la captura de pantalla que vimos en el primer capítulo, que se muestra en la siguiente ilustración:

El programa Notepad

En HTML5, la etiqueta usada es<figure>. Aquí tenemos la forma de usarla:

<figure>

<img src="images/notepad.png" alt="Notepad" />

</figure>

Una figura suele ir acompañada la mayoría de las veces de una leyenda. Para añadir una leyenda, utiliza la etiqueta<figcaption>dentro de la etiqueta<figure>, así:

<figure>

<img src="images/notepad.png" alt="Notepad" />

<figcaption>El programa Notepad</figcaption>

</figure>

#### Entender la finalidad de las figuras

Anteriormente en este capítulo, os dije que las imágenes tenían que ir colocadas en párrafos (situadas dentro de una etiqueta<p></p>). No es totalmente cierto.

Si realizas una figura de tu imagen, la imagen se puede colocar fuera del párrafo.

<p>¿Conocéis el programa Notepad? ¡Se pueden hacer páginas web con él!</p>

<figure>

<img src="images/notepad.png" alt="Notepad" />

<figcaption>El programa Notepad</figcaption>

</figure>

No veo la diferencia. ¿Cuándo debo poner la imagen en un párrafo y cuándo debo ponerla en una figura?

¡Buena pregunta! Todo depende de lo que tu imagen proporcione al texto:

* Si no proporciona información (es simplemente una ilustración con fines decorativos): coloca la imagen en un párrafo.
* Si proporciona información: coloca la imagen en una figura.

La etiqueta<figure>tiene un papel principalmente **semántico**. Esto significa que le dice al ordenador que la imagen tiene un significado y que es importante para la comprensión del texto. Esto puede permitir a un programa recuperar todas las figuras del texto mencionarlas en una tabla de figuras, por ejemplo.

Por último, ten en cuenta que una figura puede tener varias imágenes. Este es un caso que lo justifica:

<figure>

<img src="images/internetexplorer.png" alt="Logo de Internet Explorer" />

<img src="images/firefox.png" alt="Logo de Mozilla Firefox" />

<img src="images/chrome.png" alt="Logo de Google Chrome" />

<figcaption>Logos de varios navegadores</figcaption>

</figure>

#### En resumen

* Existen varios formatos de imagen apropiados para su uso en la Red:
  + JPEG: para fotos;
  + PNG: para todas las demás ilustraciones;
  + GIF: similar a PNG, con un número de colores más limitado, pero puede ser animado.
* Las imágenes se insertan con la etiqueta<img />. Debe incluir al menos dos atributos:src(nombre de la imagen) yalt(descripción corta de la imagen).
* Si una imagen ilustra el texto (y no tiene una finalidad simplemente decorativa), es mejor ponerla dentro de una etiqueta<figure>. La etiqueta<figcaption>se utiliza para introducir la leyenda de la imagen.