

HTML básico - I

Cómo crear una página web

En este primer capítulo, presentaremos las nociones básicas del lenguaje HTML. Con una utilización inteligente de las etiquetas HTML más sencillas, pueden lograrse documentos web de calidad suficiente. Al fin y al cabo, no debe olvidarse que lo importante, tanto en un papel como en Internet, sigue siendo el contenido.

1. Introducción
2. La estructura mínima de un documento HTML
3. El texto básico de un documento HTML
4. Los párrafos de un documento HTML
5. Los enlaces de un documento HTML

1. Introducción

- HTML¹ es un lenguaje universal, que funciona en cualquier plataforma (Windows, Macintosh, Unix, OS/2, etc.) y con cualquier navegador o browser (Netscape, Internet Explorer, Mozilla Firefox, etc.). El precio que paga por su universalidad es su poca sofisticación, puesto que no es otra cosa que el viejo formato ASCII² (y con 7 bits en lugar de ocho, por lo que ni tan siquiera tiene acentos ni otros caracteres especiales).
- Por tanto, para que un documento HTML sea algo más que simples caracteres básicos, debe contener, además de dicho texto, una serie de instrucciones para el browser que lo va a reproducir: estas instrucciones se denominan etiquetas o tags y se distinguen del texto porque van entre guiones (< >)³. Estas etiquetas contienen todo el resto de la información de la página web.
- Por tanto, si un documento HTML no es más que texto ASCII (parte sin guiones y parte entre guiones: las etiquetas), cualquier documento web, hasta el más sofisticado, puede escribirse directamente desde un sencillo programa básico de texto, como el Cuaderno de Notas (*Notepad*) de Windows, por ejemplo.
- Sin embargo, escribir un documento complejo de esta manera exige un conocimiento exhaustivo de las numerosísimas etiquetas existentes y sus normas de utilización; para evitar esta dificultad, hay una serie de programas denominados comúnmente editores de HTML (desde el Composer al Dreamweaver, pasando por tantos otros), que permiten al usuario componer una página (es decir, generar etiquetas HTML) desde un interfaz más o menos parecido al de un procesador de textos.
- A la hora de crear una página web, y aunque utilicemos un editor de HTML, es casi imprescindible entender la disposición del etiquetado de un documento y manejar siquiera las etiquetas más elementales. Son varias las razones que pueden aducirse para ello:

1. Los editores de HTML no son siempre herramientas perfectas, sobre todo cuando ha habido muchas modificaciones durante el proceso de creación. Con bastante frecuencia se hará necesario consultar el código completo, con las etiquetas HTML (lo que se suele denominar **código fuente** o **código madre**) para corregir las disfunciones.
 2. Tal vez la mejor manera de aprender a diseñar páginas web es observar las páginas de los demás, cuyos códigos fuente están siempre disponibles para ser imitados (o incluso copiados, técnicamente, al menos). Pero, para ello, es necesario entenderlos.
 3. Los mismos editores HTML manejan términos propios del lenguaje HTML (*heading*, *cellspacing*, etc.), que es preciso comprender.
 4. Es necesario un buen conocimiento de HTML para insertar otros códigos más potentes, como los de Javascript, por ejemplo.
- El objetivo de este documento es aprender a componer una sencilla página web desde el más básico de los programas de texto disponibles en Windows: el Cuaderno de Notas.⁴ De esta manera, seremos capaces de entender y manejar los códigos fuente de las páginas más complejas.



2. Esquema mínimo de un documento HTML

- Un documento HTML comienza siempre con la etiqueta `<HTML>`⁵, que indica que el documento en cuestión está construido con dicho lenguaje.
- La mayoría de las etiquetas son pareadas, es decir, `<...>` corresponde al principio de la acción y `</...>` indica el fin de dicha acción.
- Por tanto, una página web estará siempre contenida entre las etiquetas `<HTML>` y `</HTML>`.
- Por otra parte, todo documento HTML consta de dos partes: la cabecera (**head**) y el cuerpo del documento (**body**).
 1. La cabecera contiene básicamente información destinada al browser (o navegador), que queda oculta al usuario. Su etiqueta (pareada) es `<HEAD>`.
 2. El cuerpo es el documento que ve el usuario. Su etiqueta (pareada) es `<BODY>`.
- Ya estamos en condiciones de componer nuestra primera página web. Basta con escribir las siguientes etiquetas en nuestro editor de texto y salvar el resultado con la extensión ***.HTML** o ***.HTM**:

EJEMPLO 1

```
<HTML>
<HEAD>
</HEAD>
<BODY>
</BODY>
</HTML>
```

[Ver resultado](#)

- He introducido una pequeña tabulación a la hora de escribir las etiquetas de cabecera y cuerpo, simplemente para facilitar su comprensión gráfica, puesto que esta cuestión es totalmente

indiferente. El resultado no variaría de haber colocado las etiquetas de cualquier otra manera. Por ejemplo:

EJEMPLO 1: Variantes de escritura

```
<HTML>
<HEAD>
</HEAD>
<BODY>
</BODY>
</HTML>
```

```
<HTML><HEAD></HEAD><BODY>
</BODY></HTML>
```

```
<HTML>
<HEAD></HEAD>
<BODY></BODY>
</HTML>
```

[Ver resultado](#)

- Esta primera página web está, lógicamente, vacía, pero es perfectamente operativa. Para modificarla, basta con modificar sus códigos en el editor de texto. No hace falta cerrar para ello el navegador: una vez modificado –y salvado– el fichero, basta con utilizar el comando **Actualizar** (o **Refresh**) para ver los cambios.
- Al visualizar los resultados del [Ejemplo 1](#), en la barra de título de la ventana correspondiente sólo aparece el nombre del navegador (con Netscape) o la trayectoria completa del archivo (con Explorer). Esto sucede porque nuestro documento no posee un título propio. Para crearlo, debemos introducir una nueva etiqueta (pareada) en la cabecera: `<TITLE>`. Nuestra página web podría quedar, por tanto, así:

EJEMPLO 2

```
<HTML>
<HEAD>
<TITLE>Ejemplo 2</TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

[Ver resultado](#)

- Ya sabemos crear el esqueleto de una página web. La siguiente tarea será aprender a introducir texto (entre los tags `<BODY>` y `</BODY>`) y controlar sus etiquetas elementales.



3. El texto básico de un documento HTML

- El texto básico de un documento HTML puede escribirse sin etiquetas. Cada navegador lo visualiza entonces con el tipo y tamaño de caracteres escogidos en su configuración por defecto, e introduce un salto de línea cuando los caracteres alcanzan el borde de la ventana.

- Para dominar las cuestiones de formato (de caracteres, de párrafo, etc.) es necesario utilizar las etiquetas correspondientes.

3.1. La etiqueta básica

- La etiqueta básica, que controla el tipo de fuente utilizado, es ``.
- `` es una etiqueta pareada. Por tanto, afecta a los caracteres introducidos entre `` y ``. Pero sin más especificaciones, la etiqueta aún no sirve para nada.
- Se puede especificar **el tipo de letra** (es decir, la fuente de caracteres) añadiéndolo a la etiqueta, de la siguiente manera (en este caso, la fuente escogida es Arial): `...`.
- Se puede especificar **el color de la fuente** con la etiqueta `...`. Para especificar los colores, conviene saber al menos lo siguiente:
 1. Cada color posee su correspondiente código.
 2. Hay dos códigos: hexadecimal –lo más habitual– o RGB. En ambos casos se trata de expresar las proporciones de tres colores básicos: rojo, verde y azul.
 3. Los valores hexadecimales de los tres colores se expresan de la siguiente manera:
 1. Rojo: color:#ff0000;
 2. Verde: color:#00ff00;
 3. Azul: color:#0000ff;
 4. Los mismos colores, expresados con los valores RGB (red, green, blue), son los siguientes:
 1. Rojo: color:rgb(255,0,0);
 2. Verde: color:rgb(0,255,0);
 3. Azul: color:rgb(0,0,255);
 5. Todos los colores se expresan mediante combinaciones de estos tres colores básicos. Existen muchos programas que proporcionan los valores de todos los colores que el usuario puede componer. También en internet existen muchas páginas con los [Códigos de colores](#).
 6. Los colores más básicos pueden especificarse sin código, simplemente escribiendo la palabra correspondiente (por supuesto en inglés). Por ejemplo: `...`, `...`, etc.
- Se puede especificar **el tamaño de la fuente** con la etiqueta `...`. El valor del tamaño ("?", deseado puede suministrarse de manera absoluta (un número del 1 al 7) o relativa (+1, +2, -1, -2, etc., teniendo en cuenta que el valor por defecto es 3).⁶
- Todas estas etiquetas se pueden combinar, de modo que con una sola se controla, por ejemplo, el tipo, el tamaño y el color de los caracteres: `...`.

Con esta etiqueta hemos visto ya la sintaxis básica de las etiquetas HTML, que podíamos formalizar así: `<ETIQUETA [ATRIBUTO="valor"]>`.

Lo anterior quiere decir que una etiqueta HTML (por ejemplo, FONT) puede llevar uno o varios atributos (FACE, COLOR, SIZE), cuyos valores se colocan entrecomillados tras el signo igual (= "arial", = "#0000FF", etc.).

Esto será válido para todas las etiquetas HTML que veamos a partir de este momento.

- Volvamos a nuestro documento web y practiquemos con las posibilidades de la etiqueta ``. Por ejemplo:

EJEMPLO 3

```
<HTML>
<HEAD>
<TITLE>Ejemplo 3</TITLE>
</HEAD>
<BODY>
Esto es texto simple: cada navegador lo visualizará según su
configuración por defecto.
<FONT SIZE="1">Este texto es tamaño 1.</FONT>
<FONT SIZE="2">Este texto es tamaño 2.</FONT>
<FONT SIZE="4">Este texto es tamaño 4.</FONT>
<FONT SIZE="+1">Este texto es tamaño +1 (que es lo mismo
que tamaño 4).</FONT>
<FONT FACE="Arial" SIZE="5" COLOR="FFFF00">Este texto
posee varias especificaciones de formato.</FONT>
</BODY>
</HTML>
```

[Ver resultado](#)

- Nuestra página web, tal y como está diseñada en el Ejemplo 3, no posee ninguna especificación de líneas. A pesar de que las etiquetas están ordenadas en varias líneas, el navegador coloca todos los caracteres seguidos (sólo tiene en cuenta un único espacio entre palabras). Para corregir este problema podemos utilizar la etiqueta `
`, que introduce un salto de línea. Nuestro página quedaría entonces así:

EJEMPLO 4

```
<HTML>
<HEAD>
<TITLE>Ejemplo 4</TITLE>
</HEAD>
<BODY>
Esto es texto simple: cada navegador lo visualizará según su
configuración por defecto.<BR>
<FONT SIZE="1">Este texto es tamaño 1.</FONT><BR>
<FONT SIZE="2">Este texto es tamaño 2.</FONT><BR>
<FONT SIZE="4">Este texto es tamaño 4.</FONT><BR>
<FONT SIZE="+1">Este texto es tamaño +1 (que es lo mismo
que tamaño 4).</FONT><BR>
<FONT FACE="Arial" SIZE="5" COLOR="FFFF00">Este texto
posee varias especificaciones de formato.</FONT>
</BODY>
</HTML>
```

[Ver resultado](#)

Hay otra cuestión importante que debe saberse sobre la etiqueta ``: es la etiqueta más básica, y conviene empezar a aprender por ella, pero está en desuso y desaparecerá en las versiones futuras de HTML. En su lugar, existen otras etiquetas más potentes, que iremos viendo en su momento.

3.2. Etiquetas de formato de caracteres

- A partir de este momento, se trata simplemente de añadir más etiquetas a nuestra aún pequeña colección, de modo que vayamos teniendo cada vez más control sobre el formato de los caracteres. La tabla siguiente muestra las más comunes:

Código	Función
<code>...</code>	Negrita
<code><I>...</I></code>	Cursiva
<code><U>...</U></code>	Subrayado
<code><SUB>...</SUB></code>	Subíndice
<code><SUP>...</SUP></code>	Superíndice

- Hay que realizar algunos comentarios sobre las etiquetas precedentes:
 1. Existe otra etiqueta similar a ``, menos utilizada porque depende de las opciones de configuración del navegador: `...`.
 2. Existe otra etiqueta similar a `<I>`, menos utilizada porque depende de las opciones de configuración del navegador: `...`.
 3. Conviene tener cuidado a la hora de manejar el subrayado, puesto que éste se utiliza convencionalmente para indicar los hiperenlaces, y podría por tanto resultar confuso para el usuario.
- Es normal utilizar varias etiquetas para un mismo elemento de texto. En ese caso, hay que tener cuidado para encajarlas correctamente. Por ejemplo:
`<I>...</I>` es correcto, pero
`<I>...</I>` es incorrecto.

3.3. Otras etiquetas de caracteres

- Hay muchas otras etiquetas que afectan a los caracteres. Vamos a presentar varias más, que deben manejarse con cierta precaución:
 1. La etiqueta `<PRE>...</PRE>` indica que el texto afectado está preformateado, de modo que el navegador sí tiene en cuenta todos los espacios y saltos de línea definidos en la pantalla.
 2. La etiqueta `<BLINK>...</BLINK>` indica texto parpadeante. Sólo funciona con Netscape.
 3. La etiqueta `<S>...</S>` genera texto tachado.
 4. La etiqueta `<TT>...</TT>` genera texto similar al de una máquina de escribir (fuente courier de paso fijo).
 5. La etiqueta `<BIG>...</BIG>` aumenta el tamaño de letra (equivale a ``).
 6. La etiqueta `<SMALL>...</SMALL>` disminuye el tamaño de letra (equivale a ``).

EJERCICIO 1

Ya es hora de practicar con las herramientas que hemos presentado hasta ahora. ¿Cómo? Es muy sencillo: icrea una página web! Si no se te ocurre nada, te propongo que reproduzcas esta página. Para que no tengas

que teclearlo todo, *aquí* tienes el archivo de texto, de modo que, si quieres, no tienes más que pegarlo en tu editor y colocar las etiquetas correspondientes en los lugares adecuados.

(Por cierto: las fuentes que he utilizado son Arial y Comic Sans MS.)



4. Los párrafos de un documento HTML

En el apartado anterior hemos aprendido a controlar la apariencia de los caracteres. (Recuerda, de todas maneras, que la etiqueta `` está en desuso.) Vamos ahora a estudiar las etiquetas fundamentales que controlan la organización de los párrafos en un texto más complejo.

- Ya conocemos un tag que afecta a las líneas: `
`. Si deseamos que una línea no se parta al llegar al final de la ventana del navegador, debemos colocar al comienzo la etiqueta `<NOBR>`, y el navegador colocará entonces un *scroll* o barra de desplazamiento horizontal.
- Por supuesto, si queremos que haya más líneas, al final sí deberá aparecer `
`.

4.1. Los párrafos y su alineamiento

- La etiqueta `
` inserta un salto de línea, pero no un salto de párrafo.⁷ En cambio, la etiqueta `<P>...</P>` inserta un salto de línea y una línea en blanco, por lo que, en la práctica, equivale a un nuevo párrafo.
- Por supuesto, tanto `
` como `<P>...</P>` pueden repetirse varias veces seguidas para dejar más distancia entre las líneas y los párrafos.
- De todas maneras, sólo podemos separar los párrafos con líneas completas (una, dos, etc.), pero no podemos controlar el espacio entre los párrafos con la precisión que permite, por ejemplo, el programa Word (con las opciones de espaciado anterior y posterior).⁸
- Un párrafo puede alinearse introduciendo los siguientes atributos en la etiqueta habitual:

Código	Función
<code><P align="left">...</P></code>	Párrafo alineado a la izquierda
<code><P align="center">...</P></code>	Párrafo centrado
<code><P align="right">...</P></code>	Párrafo alineado a la derecha
<code><P align="justify">...</P></code>	Párrafo justificado

- Las etiquetas anteriores permiten alinear los párrafos enteros, pero siempre dejando una línea vacía delante de cada uno de ellas. ¿Hay alguna manera de alinear independientemente líneas o grupos de líneas? Sí, con las siguientes etiquetas:

Código	Función
<code><DIV align="left">...</DIV></code>	Línea(s) alineada(s) a la izquierda
<code><DIV align="center">...</DIV></code>	Línea(s) centrada(s)
<code><DIV align="right">...</DIV></code>	Línea(s) alineada(s) a la derecha
<code><DIV align="justify">...</DIV></code>	Línea(s) alineada(s) justificadas

- En definitiva, disponemos de varias etiquetas similares para la creación y la alineación de los párrafos, que serán más o menos apropiadas dependiendo de las ocasiones. Veamos cómo funcionan en una página web:

EJEMPLO 5

```
<HTML>
<HEAD>
<TITLE>Ejemplo 5</TITLE>
</HEAD>
<BODY>
<P align="left">Este párrafo está alineado a la izquierda. En la
práctica, equivale sencillamente a la misma etiqueta sin
indicación de alineación.</P>
<P align="center">Este párrafo está centrado. Hay una línea en
blanco entre este párrafo y el anterior.</P>
<P>Ahora vamos a alinear tres líneas seguidas de tres
maneras diferentes, pero sin dejar líneas vacías entre ellas, por
medio de otras etiquetas:</P>
<DIV align="left">alineado izquierdo alineado izquierdo alineado
izquierdo alineado izquierdo</DIV>
<DIV align="center">alineado centrado alineado centrado
alineado centrado alineado centrado</DIV>
<DIV align="right">alineado derecho alineado derecho alineado
derecho alineado derecho </DIV>
</BODY>
</HTML>
```

[Ver resultado](#)

- Otra etiqueta que afecta a la alineación de los párrafos es `<BLOCKQUOTE>...<BLOCKQUOTE>`, que introduce una doble sangría, derecha e izquierda. Pueden colocarse varias etiquetas seguidas para acentuar la sangría, como se observa en el siguiente ejemplo:

EJEMPLO 6

```
<HTML>
<HEAD>
<TITLE>Ejemplo 6</TITLE>
</HEAD>
<BODY>
<P>Párrafo normal Párrafo normal Párrafo normal Párrafo
normal Párrafo normal Párrafo normal Párrafo normal Párrafo
normal.</P>
<P><BLOCKQUOTE>Párrafo con una sangría izquierda-
derecha Párrafo con una sangría izquierda-derecha Párrafo con
una sangría izquierda-derecha Párrafo con una sangría
izquierda-derecha.</BLOCKQUOTE></P>
<P><BLOCKQUOTE><BLOCKQUOTE>
<BLOCKQUOTE>Párrafo con tres sangrías izquierda-derecha
Párrafo con tres sangrías izquierda-derecha Párrafo con tres
sangrías izquierda-derecha Párrafo con tres sangrías izquierda-
derecha.</BLOCKQUOTE></BLOCKQUOTE>
</BLOCKQUOTE></P>
</BODY>
</HTML>
```

[Ver resultado](#)

4.2. Los títulos

- Pueden utilizarse hasta seis tipos de títulos en un documento HTML, ordenados jerárquicamente por medio de la etiqueta `<Hn>...</Hn>`, donde $n = 1$ a 6.
- Las etiquetas de los títulos introducen una línea superior en blanco. Como lo habitual después de un título es un nuevo párrafo (`<P>`), que también introduce su propia línea, el resultado final suele ser que una línea de título tiene una línea en blanco antes y otra después.
- Las etiquetas de los títulos admiten indicaciones de alineación, de la misma manera que la etiqueta de párrafo.
- Veamos qué aspecto tienen los títulos de un documento HTML:

EJEMPLO 7

```
<HTML>
<HEAD>
<TITLE>Ejemplo 7</TITLE>
</HEAD>
<BODY>
<H1>Título 1</H1>
<H2>Título 2</H2>
<H3>Título 3</H3>
<H4>Título 4</H4>
<H5>Título 5</H5>
<H6>Título 6</H6>
<H7>Título 7 (como no existe, no supone ningún cambio)</H7>
<P>Además, los títulos pueden recibir indicaciones de
alineación, como en los siguientes ejemplos</P>
<H3 align="center">Título centrado</H3>
<H3 align="right">Título a la derecha</H3>
</BODY>
</HTML>
```

[Ver resultado](#)

- Este ejemplo permite plantear el siguiente ejercicio:

EJERCICIO 2

En el [EJEMPLO 7](#) se observa que siempre queda una línea en blanco entre un título y el párrafo siguiente (el que comienza con "Además". ¿Cómo puede suprimirse dicha línea? Elimina, por ejemplo, la que hay entre el título 7 y el párrafo siguiente.

¿Sabes solucionarlo? ¿Te parece fácil? Eso quiere decir que has entendido estupendamente todo lo expuesto hasta el momento.

¿No lo has conseguido? No pasa nada, pero tal vez deberías repasar todo lo anterior con más detalle. En cualquier caso, [aquí](#) tienes la solución.

4.3. Las listas

- Las listas son elementos que, como los títulos, permiten presentar niveles jerarquizados de texto.

- Los documentos HTML permiten dos tipos de listas:
 1. Listas no ordenadas (*bullet list*): su etiqueta principal es `...`.
 2. Listas ordenadas o numeradas (*numbered list*): su etiqueta principal es `...`.
 3. Listas de definiciones (*definition list*): su etiqueta principal es `<DL>...</DL>`.
- Una vez definido el tipo de lista deseada, los elementos que la componen se marcan de distinta manera según el tipo:
 1. En las listas ordenadas (`...`) o no ordenadas (`...`), cada elemento de dicha lista se marca con la etiqueta `...` (*list item*).
 2. En las listas de definición (`<DL>...</DL>`), se utilizan dos: cada elemento que se define se marca con `<DT>...</DT>` y el texto de cada definición se marca con `<DD>...</DD>`. Esta etiqueta se puede repetir si se necesita introducir varios elementos de definición.
- Las listas se pueden encajar unas dentro de otras, además de con el resto de las etiquetas disponibles. Vemos algún ejemplo:

EJEMPLO 8

```
<HTML>
<HEAD>
<TITLE>Ejemplo 8</TITLE>
</HEAD>
<BODY>
<H3>Los meses de primavera</H3>
<UL>
<LI>Abril</LI>
<LI>Mayo</LI>
<LI>Junio</LI>
</UL>
<BR>
<H3>Los meses de verano</H3>
<OL>
<LI>Julio</LI>
<LI>Agosto</LI>
<LI>Septiembre</LI>
</OL>
<BR>
<H3>Los meses del año</H3>
<UL>
<LI>Los meses de primavera
<OL>
<LI>Abril</LI>
<LI>Mayo</LI>
<LI>Junio</LI>
</OL>
</LI>
<LI>Los meses de verano
<OL>
<LI>Julio</LI>
<LI>Agosto</LI>
<LI>Septiembre</LI>
</OL>
```

```

</LI>
</UL>
<BR>
<H3>Refranes de los meses</H3>
<DL>
<DT>Marzo</DT>
<DD>El sol de marzo da con el mazo.
</DD>
<DT>Abril</DT>
<DD>El abril, aguas mil.</DD>
<DT>Mayo</DT>
<DD>Hasta el 40 de mayo no te quites el
sayo.</DD>
</DL>
</BODY>
</HTML>

```

[Ver resultado](#)

- Las listas admiten varios parámetros, que varían según el tipo de lista:
 1. Las tres listas admiten el parámetro **COMPACT**, que hace que el navegador presente la lista de la manera más compacta posible.
 2. El parámetro **TYPE** funciona con las dos primeras listas. Sus valores respectivos son, lógicamente, distintos:
 - En las no numeradas, controla el tipo de viñeta: con forma de disco (`<UL type="disc">`, la opción por defecto), de círculo (`<UL type="circle">`) y de cuadrado (`<UL type="square">`).
 - En las numeradas, controla el tipo de numeración: con números árabes (`<OL type="1">`, la opción por defecto), romanos (`<OL type="I">`), romanos en minúsculas (`<OL type="i">`), letras mayúsculas (`<OL type="A">`) y minúsculas (`<OL type="a">`).
 3. La etiqueta `<OL start="n">` indica el número por el que empieza la lista numerada.
 4. En una lista numerada, se puede cambiar el número de un elemento –y consiguientemente de los posteriores– con la etiqueta `<LI value="n">`.

4.4. Las rayas horizontales

- Las rayas horizontales son otro de los procedimientos habituales para separar los contenidos de un documento HTML. Se introducen simplemente con la etiqueta `<HR>`.
- Hay una serie de características de las rayas que pueden controlarse, como siempre, por medio de parámetros añadidos a la etiqueta básica:
 1. **El tamaño:** la etiqueta `<HR>` introduce una raya que ocupa todo el ancho de la pantalla. Pueden especificarse distintas anchuras añadiendo el valor deseado, expresado en píxeles (`<HR width="n">`) o en términos porcentuales (`<HR width="n%">`).⁹
 2. **La alineación:** cuando una raya ocupa sólo una parte de la pantalla, puede especificarse su alineación por medio de las etiquetas habituales:


```

<HR align="left">
<HR align="center">
<HR align="right">

```
 3. **La altura:** se expresa en píxeles por medio de la etiqueta `<HR size="n">`. Puede adoptar el valor 0 (la raya más fina).

4. El **color**, por medio del parámetro habitual: `<HR color="?">`.

5. El **sombreado**: la raya posee habitualmente un pequeño efecto de sombreado, que puede eliminarse con la etiqueta `<HR noshade>`.¹⁰

- Como siempre, lo habitual suele ser la combinación de varios parámetros en una misma etiqueta. Por ejemplo:

`<HR width="50%" size="0" align="center">`

- No queda sino practicar con las distintas posibilidades de rayas. Por ejemplo:

EJEMPLO 9

```
<HTML>
<HEAD>
<TITLE>Ejemplo 9</TITLE>
</HEAD>
<BODY>
Ésta es una línea normal:
<HR>
Ésta es una línea normal, sin sombreado:
<HR noshade>
Ésta es una línea que ocupa la mitad de la pantalla (si no se
establece la alineación, se coloca centrada por defecto):
<HR width="50%">
Ésta es más estrecha y mucho más gorda:
<HR width="10%" size="20">
Ésta es igual pero de color rojo:
<HR width="10%" size="20" color="#FF0000">
</BODY>
</HTML>
```

[Ver resultado](#)



5. Los enlaces de un documento HTML

- En los apartados anteriores hemos aprendido a manejar correctamente el elemento fundamental de todo documento: el texto. Sin embargo, hay otro elemento básico en un documento HTML, que le proporciona su mayor potencial: los hiperenlaces, que estudiaremos en este apartado.
- Un hiperenlace es un vínculo que une un elemento del documento activo (por ejemplo, una palabra) con otro, de manera que activamos ese segundo elemento simplemente con un *click* del ratón en el primero.¹¹
- La sintaxis general de la etiqueta es muy simple: `...`. (Los puntos suspensivos representan el elemento enlazado.)
- Para que el destino se abra en una ventana nueva, hay que indicar el siguiente atributo: `...`
- El destino de un hiperenlace puede ser de tres tipos:

1. **Enlace externo**: otro documento HTML situado en un ordenador remoto.

2. **Enlace local**: otro documento HTML situado en el mismo ordenador.

3. **Ancla:** otro lugar dentro del mismo documento HTML.

- Vamos a examinar separadamente cada uno de estos tipos.

5.1. Los enlaces externos

- Desde cualquier punto de un documento HTML puede establecerse un enlace con cualquier recurso de Internet.
- Como es sabido, los recursos más habituales de Internet son:

1. La WWW, es decir, las páginas web o documentos HTML. La etiqueta correspondiente es:

`...`

2. El correo electrónico. La etiqueta correspondiente es:

`...`

3. La transferencia de ficheros (servidores FTP). La etiqueta correspondiente es:

`...`

- Veamos algunos ejemplos:

EJEMPLO 10

```
<HTML>
<HEAD>
<TITLE>Ejemplo 10</TITLE>
</HEAD>
<BODY>
```

Vamos a enlazar las palabras "pincha aquí" con algunos destinos remotos.

- 1.- Con la página web de la UD:

`pincha aquí
`

- 2.- Con el servidor FTP de la sede de Ginebra de la OMS:

`pincha aquí
`

- 3.- Con mi correo electrónico:

`pincha aquí
`

```
</BODY>
```

```
</HTML>
```

[Ver resultado](#)

5.2. Los enlaces locales

- La mayor parte de las "páginas web" son, en rigor, una colección de páginas, puesto que la organización clásica de los sitios web consiste en un conjunto ordenado de documentos HTML entrelazados y residentes en un mismo ordenador.
- Dentro de un mismo ordenador, los documentos HTML pueden residir en un mismo directorio o en varios:

1. Cuando todos los documentos residen en el mismo directorio no hay ningún problema, y la etiqueta será, simplemente: `...`.
2. Cuando los ficheros residen en diferentes directorios, hay que facilitar la dirección completa del fichero, lo que puede hacerse de manera absoluta o relativa:

- Direccionamiento absoluto: se indica la trayectoria completa del archivo en cuestión:

`...`

- Direccionamiento relativo: no se indican los nombres de los directorios. Por ejemplo:

`...`

- Aun utilizando un direccionamiento relativo, es necesario indicar los nombres de los directorios "descendentes". Por ejemplo, en una web ordenada como en la figura siguiente, un enlace desde un documento situado en el directorio **<HTML>** a un recurso situado en el directorio **<Imágenes>** debería especificarse así:

`...`



5.3. Las anclas

- Especialmente cuando el documento es largo, conviene utilizar enlaces que apunten hacia otro lugar dentro del mismo documento. Estos enlaces se denominan anclas, anclajes o apuntadores.
- El procedimiento de enlace es doble:
 1. En primer lugar, hay que establecer el punto de anclaje en el lugar del documento que vaya a ser el destino del enlace. Su etiqueta es `...`.
 2. Después sólo queda enlazar los caracteres deseados con el ancla anterior. Su etiqueta es `...`.
- Algunas observaciones sobre las anclas:
 1. El nombre del ancla es arbitrario, pero lo lógico es que guarde alguna relación con el destino, especialmente cuando hay muchas en un mismo documento.
 2. Aunque la etiqueta del ancla sea pareada, lo normal es que no incluya nada entre sus dos partes (`...`), puesto que lo importante es marcar el punto de inicio del ancla, no su final.
- Vamos a practicar estas etiquetas creando una nota a pie de página como las utilizadas en este documento, que implican un doble anclaje:

EJEMPLO 11

```

<HTML>
<HEAD>

```

```
<TITLE>Ejemplo 11</TITLE>
</HEAD>
<BODY>
<P>Al final de este párrafo vamos a colocar una nota a pie de
página, de modo que al pinchar sobre la llamada, vamos a la
nota, y al pinchar sobre el número de nota en el pie de página,
volvemos a la llamada. <A NAME="llamada_1"></A><A
HREF="#nota1"><SUP>1</SUP></A><BR><P>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR> Bla,<BR>
Bla,<BR> Bla,<BR> Bla,<BR> Bla,
<HR size="0">
<A NAME="nota_1"></A><A HREF="#llamada_1">
<SUP>1</SUP></A><FONT SIZE="-1">Éste es el texto de la
nota. Si pinchamos sobre el número, volvemos a la llamada.
</FONT>
</BODY>
</HTML>
```

[Ver resultado](#)

Hemos presentado el etiquetado básico de los enlaces, su utilización y sus tipos, pero aún no somos capaces de alterar su formato predefinido. Esto lo veremos en el tercer capítulo de esta serie, dedicado a [Información Complementaria](#).



Notas

- 1 HTML es el acrónimo de *HyperText Markup Language*. Para redactar este apartado me he valido en buena medida del trabajo de Luc Van Lancker [Aprender el lenguaje Html](#).
- 2 El código ASCII (*American Standard Code for Information Interchange*) es el más extendido de los muchos sistemas de codificación informática, es decir, maneras de representar las informaciones (letras, números, etc.) con bits (esto es, ceros y unos).
- 3 Escribiremos todas las etiquetas con color, como suelen aparecer, por mera convención visual, en los programas editores de HTML.
- 4 El Cuaderno de Notas de Windows, por su propia simplicidad, genera código ASCII simple. Los procesadores de textos más potentes (Word y otros) generan códigos propios, que entorpecen en este caso su conversión a HTML.
- 5 Las etiquetas no se ven afectadas por mayúsculas o minúsculas, por lo que da igual escribir `<HTML>`, `<Html>`, `<html>`, etc.
- 6 Las etiquetas `` y `` equivalen respectivamente a las etiquetas `<BIG>` y `<SMALL>`.
- 7 Equivale a la combinación <Shift>+Intro del procesador de textos Word.
- 8 Por supuesto, sí hay procedimientos HTML para controlar estas distancias con exactitud, pero son de mayor nivel y no se presentan en este capítulo.
- 9 La información relativa a los tamaños suele facilitarse más en términos porcentuales que en píxeles, porque depende menos de las opciones de configuración de cada navegador. Así por ejemplo, una raya

de 100 píxeles ocupará más en una pantalla configurada a 800 x 600 píxeles que en otra configurada a 1024 x 768. Sin embargo, en cualquier configuración, una raya definida por ejemplo con el 75% siempre ocupará lo mismo.

- 10 Tampoco tiene efectos de sombreado con el valor `<HR size="0">`.
- 11 Este sistema de hipertexto ha sido utilizado por los ficheros de ayuda de Windows (y aun antes, en las últimas versiones del MS-Dos), y hoy en día es un procedimiento habitual en un número creciente de aplicaciones.



Alexander Iribar >> Edición digital

Comentarios: alex.iribar@deusto.es