

Tema 2:

Internet. HTML

2	INTERNET.....	2
2.1	INTERNET.....	2
2.1.1	El World Wide Web (WWW)	2
2.2	EL LENGUAJE HTML	3
2.2.1	Versiones del HTML.....	3
2.2.2	Mi primera página.....	4
2.2.3	Formateo básico	5
2.2.4	Caracteres de control	7
2.2.5	Enlaces	7
2.2.6	Listas.....	8
2.2.7	Imágenes	11
2.2.8	Formateo fino.....	12
2.2.9	Estructura del documento	13
2.2.10	Formularios	14
2.2.11	Controles avanzados para formularios.....	17
2.2.12	Mapas	17
2.2.13	Tablas.....	19
2.2.14	Marcos	21
2.2.15	Sonido	24
2.2.16	Guía de estilo	25
2.2.17	Ejercicios.....	27



2 Internet.

2.1 Internet

Internet se creó hace más de veinticinco años como un proyecto del Departamento de Defensa de los Estados Unidos. Técnicamente, el grupo que la creó fue la Administración de Proyectos de Investigación Avanzada de Defensa, o DARPA. Su objetivo era la creación de un método de transmisión de información y datos entre computadoras alejadas entre sí y hacer que estas comunicaciones fuesen lo más consistentes y fiables posibles. DARPA quería construir una red que fuese lo suficientemente inteligente como para sobreponerse por sí misma de problemas como fallos en el suministro de energía, interrupciones en las líneas de comunicaciones, incluso de un ataque nuclear. DARPA denominó a esta red Darpanet.

Finalmente, el gobierno descartó la idea de que su red sirviese solamente para proyectos relativos a defensa, y la red se empezó a conocer como Arpanet. A partir de este momento, el gobierno empezó a conectar numerosas universidades del país con la red. Hasta ahora, generaciones de estudiantes han estudiado, utilizado y mejorado lo que es la Internet hoy en día.

Técnicamente, la Internet no es una red de computadoras, es una red de redes. Redes locales a lo largo de todo el mundo están conectadas por cables, líneas telefónicas, cables de fibra óptica y satélites en órbita. Pero los detalles de cómo se transmiten los datos desde una computadora a otra en la Internet no son perceptibles para el usuario.

2.1.1 El World Wide Web (WWW)

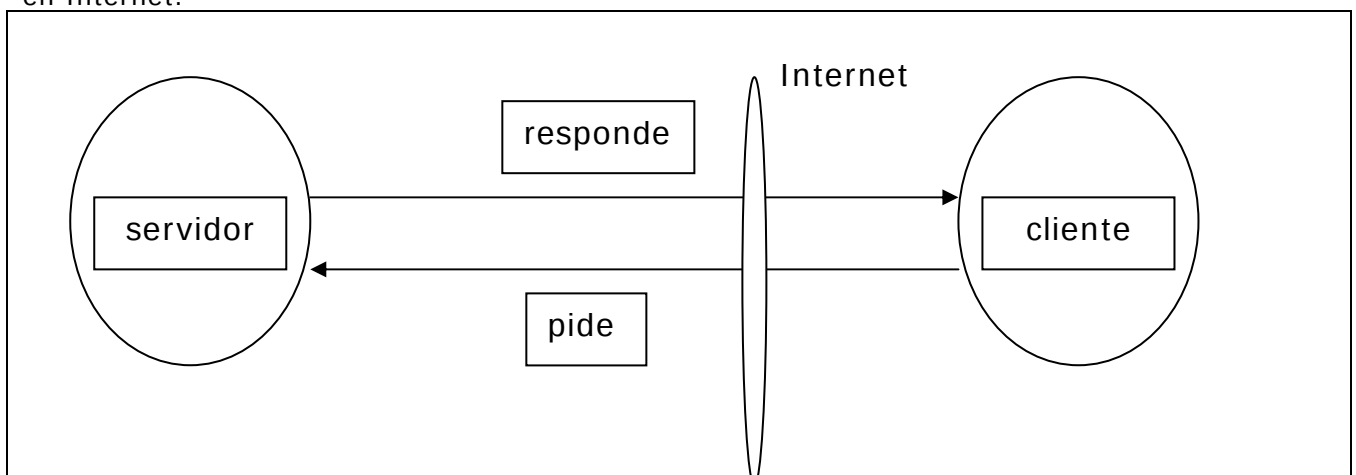
Dada su avanzada tecnología, durante años se ha considerado Internet como algo duro de aprender, difícil de utilizar y manifiestamente poco atractivo.

El WWW ha hecho que todo esto cambie. El Web se ha convertido rápidamente en la interfaz gráfica de usuario para Internet, y no tiene ninguna competencia por parte de los servicios interactivos en cuanto a términos de estética y flexibilidad se refiere.

Para acceder al Web se utiliza un programa llamado explorador Web. Un explorador web es simplemente un programa instalado en el ordenador que sabe como recuperar páginas desde otros ordenadores de Internet.

Una gran parte del éxito tan rápido y enorme que ha experimentado el Web se debe a su facilidad de manejo: es tan sencillo como pulsar un botón del ratón. Y aunque es necesario comprender algunos de los detalles, crear páginas Web es fácil y sin duda divertido.

En el esquema siguiente se pretende mostrar el funcionamiento de las páginas web que residen en Internet.





2.2 El lenguaje HTML

El lenguaje HTML es un lenguaje de etiquetas que permite generar páginas web.

2.2.1 Versiones del HTML

- HTML 2.0: Cuando se produjo la explosión de Internet el estándar de HTML que circulaba era el 2.0 (establecido en noviembre del 95), de modo que cualquier navegador que se encuentre será capaz de interpretarlo.
- HTML 3.0 y 3.2: Aunque la versión 2.0 cumplía bien el objetivo para el que se creó, carecía de herramientas para tener un control mínimamente complejo de los documentos. No se consideró necesario que lo tuviera, ya que por aquel entonces Internet era un fenómeno más bien circunscrito a la actividad académica y el contenido primaba sobre el diseño. Debido a ello, Netscape (líder del mercado de navegadores por aquel entonces) empezó a incluir etiquetas nuevas no incluidas en ningún estándar.
Por estos problemas, el IETF (el comité que suele decidir todos los estándares dentro de Internet) comenzó a elaborar el borrador del HTML 3.0, que resultó ser demasiado grande para la época, lo que dificultó su aceptación en el mercado.
Esto llevó a una serie de compañías (entre ellas Netscape, Microsoft, IBM, Sun, etc...) a crear un nuevo comité llamado W3C, que es el que actualmente elabora las nuevas versiones del HTML. Su primer trabajo consistió en crear el borrador del HTML 3.2, que incluía muchas de las mejoras que los principales navegadores (Netscape y Explorer) habían introducido en Internet, como son las tablas, los *applets*, etc..
Este borrador fue aprobado en enero de 1997 e inmediatamente el W3C se puso a trabajar en la elaboración del siguiente estándar: el 4.0.
- HTML 4.0: En julio de 1997 se presenta el borrador de este estándar. Por fin se estandarizan los marcos (*frames*), las hojas de estilo y los *scripts* (entre otras cosas). El 17 de diciembre de 1997 dicho borrador fue finalmente aprobado.



2.2.2 Mi primera página

primera.html

```
<HTML>
  <HEAD>
    <TITLE>Mi primera pagina</TITLE>
  </HEAD>
  <BODY>
    <CENTER>
      <H1>
        Mi Primera pagina
      </H1>
    </CENTER>
    <HR>
    Esta es mi primera pagina
  </BODY>
</HTML>
```

2.2.2.1 La explicación

Lo primero que conviene explicar es en qué consisten todos esos símbolos de mayor y menor que están distribuidos por ahí. El lenguaje HTML se basa en la sintaxis SGML. Esto quiere decir que cualquier cosa que hagamos en HTML estará encerrada entre dos etiquetas de esta manera:

```
<ETIQUETA parámetros> ... </ETIQUETA>
```

Hay ocasiones en que no es necesario *cerrar* la etiqueta. Mirando el código se habrán observado un par de ejemplos que ya se explicarán más adelante. Pero como lo primero que debemos indicar es que el texto que estamos componiendo es un documento HTML pues lo indicamos así:

```
<HTML> ... </HTML>
```

Un documento HTML tiene una estructura que lo separa en dos partes: cuerpo y cabecera. En la primera estará la página en sí, mientras que en la segunda se incluirán algunas cosas que no se ven al principio pero que pueden llegar a ser muy importantes. Lo primero que hay que incluir en el código es la cabecera. Se escribe:

```
<HEAD>
  <TITLE>Mi primera pagina</TITLE>
</HEAD>
```

Dentro de la cabecera sólo hay otra etiqueta. Es la única imprescindible: el título de la página. Es lo que se verá como título de la ventana en los navegadores que lo permitan.

2.2.2.2 El cuerpo del documento

Ahora se va a indicar el contenido. Lo primero será indicar que se está en el cuerpo del documento:

```
<BODY> ... </BODY>
```

La siguiente etiqueta pone el texto algo recalcado:

```
<CENTER><H1> ... </H1></CENTER>
```

Con esto se colocará el texto centrado (<CENTER>) y en formato <H1> (cabecera 1) que asegura que aumentará el tamaño del tipo de letra lo suficiente como para que se vea bastante resaltado. Luego se separará ese título que se ha puesto a la página del texto por medio de una línea horizontal:



<HR>

La línea horizontal carece de etiqueta de cierre. Esto es normal en etiquetas que no varían los atributos de un texto, sino que insertan un elemento.

2.2.3 Formateo básico

Se pueden establecer varias categorías dentro de las etiquetas usadas para formatear el texto. Se dividirán entre aquellas que sirven para cambiar párrafos enteros y las que son capaces de formatear tiras de caracteres dentro del párrafo.

2.2.3.1 Formato del párrafo

Estas son las etiquetas más importantes (excluyendo algunas que se verán más adelante):

Etiqueta	Utilidad	Resultado
<P>	Sirve para delimitar un párrafo. Inserta una línea en blanco antes del texto.	párrafo
<CENTER> ... </CENTER>	Permite centrar todo el texto del párrafo.	Texto normal Texto centrado
<DIV ALIGN=x> ... </DIV>	Permite justificar el texto del párrafo a la izquierda (ALIGN=LEFT), derecha (RIGHT), al centro (CENTER) o a ambos márgenes (JUSTIFY)	Texto a la izquierda Texto al centro Texto a la derecha Texto en todos los lados del documento
<ADDRESS> ... </ADDRESS>	Para escribir direcciones.	<i>Daniel Rodríguez Herrera C/Albalat de la Ribera 9 46002 Moixent</i>

2.2.3.2 Las 6 cabeceras

HTML ofrece seis etiquetas distintas para mostrar cabeceras. Son éstas:

Etiqueta	Resultado
<H1> ... </H1>	Cabecera de nivel 1
<H2> ... </H2>	Cabecera de nivel 2
<H3> ... </H3>	Cabecera de nivel 3
<H4> ... </H4>	Cabecera de nivel 4
<H5> ... </H5>	Cabecera de nivel 5
<H6> ... </H6>	Cabecera de nivel 6

Se usan para dividir correctamente en secciones nuestra página, tal y como se hace en un documento de texto normal.



2.2.3.3 Cambiando el tipo de letra

Todas estas etiquetas nos permiten cambiar de una manera u otra el aspecto del tipo de letra que se está utilizando y se pueden utilizar con tiras de caracteres dentro de un párrafo.

Etiqueta	Utilidad
 ... 	Pone el texto en negrita.
<I> ... </I>	Representa el texto en cursiva.
<U> ... </U>	Para subrayar algo.
<S> ... </S>	Para tachar.
^{...}	Letra superíndice.
_{...}	Letra subíndice.
<BIG> ... </BIG>	Incrementa el tamaño del tipo de letra.
<SMALL> ... </SMALL>	Disminuye el tamaño del tipo de letra.

2.2.3.4 Formato de frase

En estos elementos indicas el tipo de información que encierran las etiquetas, pero no como se representan:

Etiqueta	Utilidad
<CITE> ... </CITE>	Para citar un texto ajeno.
<CODE> ... </CODE>	Para escribir código fuente.
 ... 	La cosa es importante.
 ... 	Para dar énfasis.
<KBD> ... </KBD>	Texto tecleado por el usuario.
<VAR> ... </VAR>	Representar variables de un código.
<SAMP> ... </SAMP>	Para representar una serie de caracteres literalmente.
<ABBR> ... </ABBR>	Abreviaturas.

No son muy utilizados, ya que no permiten tener un control exacto de la manera en que la página se representará finalmente.

2.2.3.5 Otros elementos

Por último, se debe estudiar algunas cosas que no son texto y que podemos incorporar a nuestra página.

Etiqueta	Utilidad
<HR>	Inserta una barra horizontal.
 	Salto de línea.
<!-- ... -->	Comentarios.



2.2.4 Caracteres de control

En el HTML existen cuatro caracteres de control, que se usan para formar etiquetas, establecer parámetros, etc.. Para poder emplearlos sin riesgo se deberá escribir los siguiente códigos:

Código	Resultado
<	<
>	>
&	&
"	"

2.2.5 Enlaces

Las siglas HTML significan *HyperText Markup Language*, lo que para nosotros quiere decir que es un lenguaje para hipertexto. Existen múltiples formatos de hipertexto (por ejemplo, los ficheros de ayuda de Windows) y lo que tienen en común es que todos poseen enlaces.

Un enlace es una zona de texto o gráficos que si son seleccionados nos trasladan a otro documento de hipertexto o a otra posición dentro del documento actual. Siendo HTML el lenguaje de Internet, la diferencia que posee con respecto a otros tipos de hipertexto es que ese otro documento puede estar físicamente en la otra punta del planeta. Son los enlaces lo que hacen de la telaraña o *World Wide Web* lo que es.

2.2.5.1 La etiqueta <A>

Para incorporar un enlace hay que utilizar esta etiqueta. Todo lo que se encierre entre <A> y , ya sea texto o imágenes, será considerado como enlace y sufrirá dos modificaciones:

1. Se visualizará de manera distinta en el navegador. El texto aparecerá subrayado y de un color distinto al habitual, y las imágenes estarán rodeadas por un borde del mismo color que el del texto del enlace.
2. Al pulsar sobre el enlace, el usuario será enviado al documento que apuntaba el enlace.

Para que el enlace sirva para algo debemos especificarle una dirección. Lo haremos de la siguiente manera:

Pulsame

La dirección estará en formato URL (*Uniform Resource Locator*).

2.2.5.2 Las URLs

Una URL nos indica tanto una dirección de Internet como el servicio que se espera que ofrezca el servidor al que corresponde la dirección. Tiene el siguiente formato:

servicio://máquina:puerto/ruta/fichero@usuario

donde el servicio podrá ser uno de los siguientes:

http

Es el servicio invocado para transmitir páginas web y el que se usará normalmente en los enlaces.

https

Es una innovación sobre el anterior, que nos permite acceder a servidores (generalmente comerciales) que ofrecen el uso de técnicas de encriptación para proteger los datos que intercambiamos con él de terceras personas.

ftp

Permite transmitir ficheros desde servidores de *ftp* anónimo. Si no le pide un fichero sino un directorio, en general el navegador se encargará de mostrar el contenido del mismo para que se pueda escoger cómodamente. Utilizando la @ se podrá acceder a servidores privados.



mailto

Para poder mandar un mensaje.

news

Para poder acceder a foros de discusión (mal traducidos a veces como grupos de noticias). Se indica el servidor y el grupo. Por ejemplo *news://news.iber.net.es/es.comp*.

telnet

No está implementado generalmente por los navegadores, que suelen invocar un programa externo. Permite conectar con otros ordenadores y entrar en ellos como si el ordenador fuese una terminal del mismo.

La dirección de la máquina puede ser, o bien una serie de cuatro números entre 0 y 255 (123.3.5.65) o bien algo más fácil de recordar como es una serie de palabras separadas por puntos (*www.programacion.net*). El puerto generalmente no se indica, ya que el servicio predetermina uno.

La ruta es una serie de directorios separados por el símbolo /, que es el utilizado en UNIX (el sistema operativo más extendido en los servidores de Internet).

Existe otro formato de URL. Cuando se quiere acceder a un fichero situado en la misma máquina que la página web que se está creando y se puede utilizar este formato:

ruta_relativa/fichero

En la ruta relativa se pueden utilizar los dos puntos (..) para acceder al directorio padre o comenzar con la barra diagonal (/) para acceder a una ruta absoluta dentro de nuestro ordenador.

2.2.5.3 Anclas

Como se dijo, es posible acceder a una posición del documento HTML. Para hacerlo, primero se debe especificar el lugar del documento al que se quiere acceder:

`<A NAME="ancla">`

Para poder acceder a la ancla se crea de la siguiente manera:

`<A HREF="#ancla">Vamos a donde antes</A>`

También se puede acceder a anclas situadas en documentos remotos. Para ello se añadirá el siguiente enlace:

`<A HREF="enlaces.html#ancla">Otra vez</A>`

2.2.6 Listas

Existen varios tipos de listas en HTML. Todas ellas se pueden meter unas dentro de otras formando árboles muy bonitos y preciosos. Todos los tipos siguen el siguiente formato:

`<tipo_lista>`

`<LI>Primer elemento`

`<LI>Segundo elemento`

`</tipo_lista>`

tipo_lista puede ser una de las siguientes: DIR, DL, MENU, OL y UL.



2.2.6.1 Listas desordenadas

La etiqueta permite presentar listas de elementos sin orden alguno. Cada elemento de la lista irá normalmente precedido por un círculo. Por ejemplo,

Primer elemento

Segundo elemento

se verá como

- Primer elemento
- Segundo elemento

La etiqueta admite estos parámetros:

Parámetro	Utilidad	Resultado
COMPACT	Indica al navegador que debe representar la lista de la manera más compacta posible.	• Primer elemento • Segundo elemento
TYPE="disc", "circle", "square"	Indica al navegador el dibujo que precederá a cada elemento de la lista. Para mayor flexibilidad se admite también como parámetro de .	• Tipo disc - o Tipo circle ▪ Tipo square

También son listas desordenadas aquellas que utilizan las etiquetas <DIR> y <MENU>. En principio tenían como propósito representar una lista estilo directorio (multicolumna) o tipo menú (una sola columna).

2.2.6.2 Listas ordenadas

La etiqueta permite presentar listas de elementos ordenados de menor a mayor. Normalmente cada elemento de la lista irá precedido por su número en el orden. Por ejemplo,

Primer elemento

Segundo elemento

se verá como

1. Primer elemento
2. Segundo elemento



La etiqueta admite estos parámetros:

Parámetro	Utilidad	Resultado
COMPACT	Indica al navegador que debe representar la lista de la manera más compacta posible.	1. Primer elemento 2. Segundo elemento
TYPE="1", "a", "A", "i", "I"	Indica al navegador el tipo de numeración que precederá a cada elemento de la lista. Para mayor flexibilidad se admite también como parámetro de .	1. Tipo 1 b. Tipo a C. Tipo A iv. Tipo i V. Tipo I
START="num"	Indica al navegador el número por el que se empezará a contar los elementos de la lista.	3. Primer elemento 4. Segundo elemento
VALUE="num"	Atributo de , actúa como START pero a partir de un elemento predeterminado.	1. Primer elemento 4. Segundo elemento 5. Tercer elemento

2.2.6.3 Listas de definiciones

Este es el único tipo de lista que no utiliza la etiqueta . Al tener como objetivo presentar una lista de definiciones, de modo que tiene que representar de manera distinta el objeto definido y la definición. Esto se hace por medio de las etiquetas <DT> y <DD>:

<DL>

<DT>Primer elemento<DD>Es un elemento muy bonito.

<DT>Segundo elemento<DD>Este, en cambio, es peor.

</DL>

se verá como

Primer elemento

Es un elemento muy bonito.

Segundo elemento

Este, en cambio, es peor.

La etiqueta <DL> sólo admite como parámetro el ya conocido COMPACT, que tiene el mismo comportamiento que con los otros dos tipos de lista anteriores.



2.2.7 Imágenes

Para incluir gráficos e imágenes en las páginas se utilizará la etiqueta

`<IMG>`

de esta manera:

`<IMG SRC="fichero_grafico" ALT="descripcion">`

El parámetro SRC especifica el nombre del fichero que contiene el gráfico. Los formatos estándar en la red son el GIF y el JPG. Las últimas versiones de Netscape y Explorer aceptan también el formato PNG.

El parámetro ALT especifica el texto que se mostrará en lugar del gráfico en aquellos navegadores que no sean capaces de mostrarlos (como el Lynx) y en el supuesto de que el usuario los haya desactivado. Algunos navegadores lo muestran cuando pasamos el ratón por encima de la imagen. Es por eso que, aunque algunos usuarios no lo lleguen a ver nunca, conviene ponerlo siempre. De hecho, el estándar HTML 4.0 obliga a hacerlo.

Existen dos atributos que, aunque opcionales, conviene indicar siempre: la altura y la anchura del gráfico en pixels. De este modo, el navegador puede mostrar un recuadro del tamaño de la imagen mientras la va leyendo de la red y así poder mostrar el resto de la página correctamente mientras tanto.

`<IMG SRC="graficos/dwnldns.gif" ALT="Netscape 4.0" WIDTH=88 HEIGHT=31>`

WIDTH es la anchura y HEIGHT la altura.

2.2.7.1 Imágenes y enlaces

Suele ser común incluir enlaces dentro de un gráfico. En ese caso, por defecto, los navegadores le pondrán un borde al gráfico para indicar que efectivamente es un enlace. Práctico, pero la mayoría de las veces bastante poco estético. Por medio del parámetro BORDER se podrá alterar el grosor de ese borde o incluso eliminarlo poniéndolo a cero.

`<A HREF="http://www.netscape.com">`

`<IMG SRC="graficos/dwnldns.gif" ALT="Netscape 4.0" WIDTH=88 HEIGHT=31>`

`</A>`

Sin embargo,

`<A HREF="http://www.netscape.com">`

`<IMG SRC="graficos/dwnldns.gif" ALT="Netscape 4.0" WIDTH=88 HEIGHT=31`

`BORDER=0>`

`</A>`



2.2.8 Formateo fino

Lo ideal cuando se trabaja con texto sería poder cambiarlo al tamaño que te viniese bien, ponerlo de colorines y cambiar el tipo de letra. Todo esto se puede hacer gracias a la etiqueta .

2.2.8.1 Cambio de color

Para hacerlo vamos a utilizar el parámetro COLOR. La manera de especificarle el color es común a todas las etiquetas HTML: o bien indicando el nombre, si es un color normal, o bien especificando el porcentaje de rojo, verde y azul (código RGB) del mismo. Los colores reconocidos son los siguientes:

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

```
<FONT color="red">Estoy en rojo</FONT>
```

El modo de indicar el color RGB es el siguiente:

```
<FONT COLOR="#FF0000">D</FONT>
```

```
<FONT COLOR="#EF0000">E</FONT>
```

```
<FONT COLOR="#DF0000">G</FONT>
```

```
<FONT COLOR="#CF0000">R</FONT>
```

```
<FONT COLOR="#BF0000">A</FONT>
```

```
<FONT COLOR="#AF0000">D</FONT>
```

```
<FONT COLOR="#9F0000">A</FONT>
```

```
<FONT COLOR="#8F0000">D</FONT>
```

```
<FONT COLOR="#7F0000">O</FONT>
```

Lo que nos mostraría lo siguiente:

DEGRADADO

La primera componente en hexadecimal es el rojo, la segunda el verde y la tercera el azul (Red Green Blue, RGB).

2.2.8.2 Tamaños del texto

El parámetro utilizado para indicar el tamaño es SIZE. Puede utilizarse para indicar tamaños absolutos:

SIZE=1 SIZE=2 SIZE=3 SIZE=4 SIZE=5 SIZE=6 SIZE=7

El tamaño por defecto es 3. También se puede utilizar los modificadores + y - para indicar un incremento (o decremento) relativo del tamaño del tipo de letra. Así, por ejemplo, si se indica que se quiere un tamaño de -2 se estará pidiendo al navegador que nos muestre el tipo de letra dos veces más pequeño.

```
<FONT SIZE=2>
```

```
<font size="2">Tamaño 2</font>
```

```
<FONT SIZE="+3">
```

```
<font size="2"><font size="+3">Tamaño 6</font></font>
```

```
</FONT>
```

```
</FONT>
```



2.2.8.3 Tipo de letra

Por último, se puede especificar el nombre del tipo de letra que se quiere utilizar gracias al parámetro FACE. Como en principio no se tiene manera de saber que tipo de letra tiene instalado el ordenador del usuario que está viendo nuestras páginas, podemos indicar más de uno separado por comas. Si el navegador no encuentra ninguno seguirá utilizando el que tiene por defecto:

```
<FONT FACE="Helvetica,Arial,Times">
  <font face="Helvetica,Arial,Times">No sé como voy a salir exactamente</font>
</FONT>
```

De todos modos es recomendable no utilizar con fe ciega este atributo en Internet, ya que impide que todos puedan ver nuestras páginas como nosotros. E Internet, siempre que nos lo permitan Microsoft y Netscape, debe ser lo más estándar posible.

2.2.9 Estructura del documento

La estructura de un documento HTML se puede resumir así:

```
<HTML>
  <HEAD>
    <TITLE>titulo de la página</TITLE>
  </HEAD>
  <BODY parámetros>
    contenido de la página
  </BODY>
</HTML>
```

2.2.9.1 La cabecera

Suele ser el lugar más indicado para colocar aquellos elementos de la página que no alteren el contenido de la misma, aunque si la forma de presentarlo y su comportamiento. Es por eso que es el lugar más recomendable para colocar los *scripts* y las hojas de estilo, como veremos en los capítulos correspondientes.

2.2.9.2 El cuerpo

Obviamente no se va a explicar lo que entra dentro del cuerpo (prácticamente todos los capítulos del tema intentan hacerlo) sino los parámetros que admite la etiqueta <BODY>:

BACKGROUND	Permite definir un gráfico de fondo para la página.
BGCOLOR	Permite definir el color de fondo de la página.
BGPROPERTIES	Cuando es igual a FIXED el gráfico definido como fondo de la página permanecerá inmóvil aunque se utilicen las barras de desplazamiento.
TEXT	Cambia el color del texto.
LINK	Cambia el color de un enlace no visitado (por defecto azul).
VLINK	Cambia el color de un enlace ya visitado (por defecto púrpura).
ALINK	Cambia el color que toma un enlace mientras lo estamos pulsando (por defecto rojo).
LEFTMARGIN y TOPMARGIN	Especifican el número de pixels que dejará de margen entre el borde de la ventana y el contenido de la página. Se suelen utilizar para dejarlos a cero.



2.2.10 Formularios

Una de las mayores ventajas de la web es que resulta tremendamente interactiva. Los usuarios de una página no tienen más que escribir al autor de la misma para comentarle cualquier cosa de la misma. Sin embargo, si se desea que digan sólo unas cosas concretas (responder a alguna pregunta, seleccionar entre varias opciones, etc..) se deberá utilizar formularios. Por ejemplo,

```
<FORM ACTION="" METHOD=POST>
```

```
Nombre:<BR>
```

```
<INPUT NAME="nombre" TYPE=TEXT SIZE=32>
```

```
<BR>¿Cuantos son dos y dos?<BR>
```

```
<INPUT NAME="Respuesta" TYPE=RADIO VALUE="mal">3<BR>
```

```
<INPUT NAME="Respuesta" TYPE=RADIO VALUE="bien">4<BR>
```

```
<INPUT NAME="Respuesta" TYPE=RADIO VALUE="mal">5<BR>
```

```
<INPUT TYPE="Submit" VALUE="Comprobar">
```

```
</FORM>
```

se verá así:

Nombre:

¿Cuantos son dos y dos?

☐

3

☐

4

☐

5

Comprobar

El botón no hace nada porque no tiene asociada una función.

Todos los elementos de un formulario deben estar encerrados entre <FORM> y </FORM>. Como parámetros cabe destacar tres. ACTION define el URL que deberá gestionar el formulario.

2.2.10.1 Cajas de texto

Existen tres maneras de conseguir que el usuario introduzca texto en nuestro formulario. Las dos primeras se obtienen por medio de la etiqueta <INPUT>:

```
<INPUT TYPE=TEXT>
```

```
<INPUT TYPE=PASSWORD>
```

El primero dibujará una caja donde escribir un texto (de una sola línea). El segundo es equivalente, pero no se verá lo que se teclee en él. Estos son los atributos para modificarlos:

Parámetro	Utilidad
SIZE	Tamaño de la caja de texto.
MAXLENGTH	Número máximo de caracteres que puede introducir el usuario.
VALUE	Texto por defecto que contendrá la caja.



Por otro lado, puede que se necesite que el usuario pueda introducir más de una línea. En ese caso se utilizará la siguiente etiqueta:

`<TEXTAREA><br/>Por defecto<br/></TEXTAREA>`

Lo que se incluya entre las dos etiquetas será lo que se muestre por defecto dentro de la caja. Admite estos parámetros:

Parámetro	Utilidad
ROWS	Filas que ocupará la caja de texto.
COLS	Columnas que ocupará la caja de texto.

2.2.10.2 Opciones

Si lo que se desea es que el usuario decida entre varias opciones se podrá hacer de dos modos. El primero es el que se vió en el ejemplo inicial:

3<INPUT NAME="Respuesta" TYPE=RADIO VALUE="mal"> 	3 ☐
4<INPUT NAME="Respuesta" TYPE=RADIO VALUE="bien"> 	4 ☐
5<INPUT NAME="Respuesta" TYPE=RADIO VALUE="mal"> 	5 ☐

Para asociar varios botones de radio a una misma variable se pondrá a todos ellos el mismo NAME. Aparte de esto acepta los siguientes parámetros:

Parámetro	Utilidad
VALUE	Este es el valor que asignará a la variable.
CHECKED	Si lo indicamos en una de las opciones esta será la que esté activada por defecto.

Pero también se tiene una posibilidad que ocupa bastante menos: las listas desplegables. Para emplearlas se deberán utilizar dos etiquetas, SELECT y OPTION:

```
<SELECT NAME="Navegador">
  <OPTION>Netscape
  <OPTION>Explorer
  <OPTION>Opera
  <OPTION>Lynx
  <OPTION>Otros
</SELECT>
```



Los parámetros que admite SELECT son los siguientes:

Parámetro	Utilidad
SIZE	El número de opciones que se podrá ver. Si es mayor que 1 se verá una lista de selección y, si no, se verá una lista desplegable.
MULTIPLE	Si se indica se podrá elegir más de una opción.

Y OPTION estos:

Parámetro	Utilidad
VALUE	Este es el valor que asignará a la variable.
SELECTED	Si lo indicamos en una de las opciones esta será la seleccionada por defecto.

2.2.10.3 Botones del formulario

Existen dos: uno que se utiliza para mandar el formulario y otro que sirve para limpiar todo lo que haya rellenado el usuario:

<code><INPUT TYPE=SUBMIT>
</code> <code><INPUT TYPE=RESET></code>	Enviar cons <u>u</u> lta
	<u>R</u> establecer

Se podrá cambiar el texto que el navegador pone por defecto en esos botones utilizando el parámetro VALUE.

2.2.10.4 Otros elementos

Puede que necesites que el usuario sencillamente nos confirme o niegue algo. Se podrá conseguir por medio de controles de confirmación:

`<INPUT NAME="Alto" TYPE=CHECKBOX>` Me considero alto ☐ Me considero Alto _____

Si se quiere que el control esté activado por defecto se añadirá el parámetro CHECKED. El formulario asignará a la variable NAME el valor *on* u *off*.



2.2.11 Controles avanzados para formularios

El estándar HTML 4.0 ha traído varias mejoras a los formularios, que acercan los mismos a las características que tienen en lenguajes como Java o Visual Basic. Estas mejoras sólo están contempladas en Internet Explorer.

2.2.11.1 Botones

Los botones tienen su propia etiqueta como se muestra a continuación:

```
<BUTTON TYPE="button"> boton </BUTTON>
```

2.2.12 Mapas

Como ya se comentó se pueden hacer enlaces de texto y de gráficos. Pero también existe otra posibilidad: que dentro de una sola imagen se pueda acceder a varios enlaces. Se hace declarando zonas dentro de la imagen (rectángulos, círculos, etc.), siendo cada una de ellas un enlace distinto. Tradicionalmente, siempre han existido dos maneras de hacerlo:

- Mapas gestionados por el cliente (el navegador).
- Mapas gestionados por el servidor.

Los segundos fueron los primeros en desarrollarse y estaban incluidos dentro del estándar HTML 2.0. Sin embargo, nunca hubo una manera común de gestionar esos mapas. Debido a ello, Netscape elaboró un sistema propio que fue incluido en el estándar 3.2: los mapas gestionados por el navegador.

2.2.12.1 Mapas gestionados por el cliente

Para utilizarlos se necesitan dos cosas: declarar el mapa y asignarlo a una imagen. Se va primero a declarar el mapa:

```
<MAP NAME="mi_mapa">
  <AREA SHAPE=... COORDS=... ALT="Enlace a..">
  ...
</MAP>
```

Dentro de la etiqueta MAP sólo se podrá definir el nombre del mapa (algo imprescindible, por otra parte). Es dentro de cada elemento AREA donde se podrán definir cosas más interesantes:

Parámetro	Utilidad
SHAPE	Define la forma de la zona, que podrá ser rectangular, circular o poligonal.
COORDS	Coordenadas (separadas por comas) que definen la zona. El número y significado de esas coordenadas dependerá de la zona.
HREF	URL a donde irá el usuario si pulsa sobre esa zona.
NOHREF	Especifica que esa zona no tiene asignado enlace alguno.
ALT	Texto que se presentará en lugar de la imagen en navegadores no gráficos para acceder al enlace.



Para declarar correctamente una zona se necesita conocer cómo se definen exactamente las formas y coordenadas:

	SHAPE	COORDS
Rectangular	RECT	"x1,y1,x2,y2" siendo (x1,y1) la esquina superior izquierda y (x2,y2) la inferior derecha.
Circular	CIRC	"x,y,radio" siendo (x,y) el centro del círculo y radio.
Polígono irregular	POLY	"x1,y1,x2,y2,x3,y3,..." definiendo cada pareja (x,y) una esquina del polígono. La última pareja de coordenadas se unirá a la primera para cerrar el polígono.
Toda la imagen	DEFAULT	No se indican.

2.2.12.2 Cómo usar un mapa

Después de haber definido un mapa, sólo queda asignarlo a una imagen. Esto se hace del siguiente modo:

```
<IMG SRC=... USEMAP="#mi_mapa">
```

Siempre se tiene que añadir al comienzo del nombre del mapa una almohadilla (#). A continuación se muestra un ejemplo:

```
<MAP NAME="navegadores">
  <AREA SHAPE=RECT COORDS="0,0,24,31"
    HREF="http://www.netscape.com" ALT="Netscape">
  <AREA SHAPE=RECT COORDS="26,0,53,31"
    HREF="http://www.microsoft.com" ALT="Microsoft">
  <AREA SHAPE=DEFAULT NOHREF ALT="Nada">
</MAP>
<IMG SRC="nav.gif" WIDTH=53 HEIGHT=31 BORDER=0 USEMAP="#navegadores">
```

Hay que tener en cuenta que, cuando dos zonas se solapan, la que esté declarada primero es la que tiene preferencia. Por eso se declara al final una zona que no conduce a ningún URL por si el usuario pulsa con el ratón donde no debe.



2.2.13 Tablas

Las tablas son posiblemente la manera más clara de organizar la información. También es el modo más adecuado de maquetar texto y gráficos de una manera algo más controlada que con los parámetros ALIGN.

Las tablas se definen de la siguiente manera. Primero, las características de la tabla, luego las de cada fila y dentro de ésta, cada celda. Así pues, una tabla con 2 filas y 3 columnas se declarará así:

```
<TABLE>
  <TR>
    <TD>1,1</TD>
    <TD>1,2</TD>
    <TD>1,3</TD>
  </TR>
  <TR>
    <TD>2,1</TD>
    <TD>2,2</TD>
    <TD>2,3</TD>
  </TR>
</TABLE>
```

Como se puede ver (o mejor no ver) la tabla no tiene mucho aspecto de tabla. Quedaría mejor con unos bordes. Puede que tampoco le viniese mal mayor espacio entre celdas o mayor anchura. Estas son las cosas que se pueden cambiar con los atributos de TABLE:

BORDER	Especifica el grosor del borde que se dibujará alrededor de las celdas. Por defecto es cero, lo que significa que no dibujará borde alguno.
CELLSPACING	Define el número de pixels que separarán las celdas.
CELLPADDING	Especifica el número de pixels que habrá entre el borde de una celda y su contenido.
WIDTH	Especifica la anchura de la tabla. Puede estar tanto en pixels como en porcentaje de la anchura total disponible para él (se pondrá "100%" si se quiere que ocupe todo el ancho de la ventana del navegador).
ALIGN	Alinea la tabla a izquierda (LEFT), derecha (RIGHT) o centro (CENTER).

Si ahora, por ejemplo se define ahora la tabla anterior como <TABLE BORDER=1 WIDTH="50%" ALIGN=CENTER> se verá lo siguiente:

1,1	1,2	1,3
2,1	2,2	2,3



2.2.13.1 Definir las filas

Después de haber definido la tabla nos toca hacer lo mismo con las filas. Cada fila se define con una etiqueta TR, que tiene los siguientes atributos:

ALIGN	Alinea el contenido de las celdas de la fila horizontalmente a izquierda (LEFT), derecha (RIGHT) o centro (CENTER).
VALIGN	Alinea el contenido de las celdas de la fila verticalmente arriba (TOP), abajo (BOTTOM) o centro (MIDDLE).

2.2.13.2 Definir las celdas

Por último, nos queda definir cada celda gracias a la etiquetas TD y TH. Estas etiquetas son equivalentes, pero la última se utiliza para encabezados, de modo que su interior se escribirá por defecto en negrita y centrado. Estos son los atributos de ambas:

ALIGN	Alinea el contenido de la celda horizontalmente a izquierda (LEFT), derecha (RIGHT) o centro (CENTER).
VALIGN	Alinea el contenido de la celda verticalmente arriba (TOP), abajo (BOTTOM) o centro (MIDDLE).
WIDTH	Especifica la anchura de la celda. También se puede especificar tanto en pixels como en porcentaje, teniendo en cuenta que, en este último caso, será un porcentaje respecto al ancho total de la tabla (no de la ventana del navegador).
NOWRAP	Impide que, en el interior de la celda, se rompa la línea en un espacio.
COLSPAN	Especifica el número de celdas de la fila situadas a la derecha de la actual que se unen a ésta (incluyendo la celda en que se declara este parámetro). Es por defecto uno. Si se pone igual a cero, se unirán todas las celdas que queden a la derecha.
ROWSPAN	Especifica el número de celdas de la columna situadas debajo de la actual que se unen a ésta.

Posiblemente los dos últimos parámetros no puedan quedar claros sin ejemplos. De hecho, aún entendiendo perfectamente su función es habitual que se confunda a uno con otro. Pero bueno, se va a ver una tabla de 3x3 con una celda que se une a una de la derecha y otra que se une a otra de debajo:

```
<TABLE BORDER=1>
<TR>
  <TD COLSPAN=2>1,1 y 1,2</TD>
  <TD>1,3</TD>
</TR>
<TR>
  <TD ROWSPAN=2>2,1 y 3,1</TD>
  <TD>2,2</TD>
  <TD>2,3</TD>
</TR>
<TR>
  <TD>3,2</TD>
```



```
<TD>3,3</TD>
</TR>
</TABLE>
```

Como se puede ver, cuando se declara una celda con ROWSPAN o COLSPAN, no se deberá declarar las celdas "absorbidas" o la creación de la tabla se nos complicará de horrible manera.

2.2.13.3 Título de la tabla

Por último, se va a ver como definir un título a la tabla. Esto se hace por medio de la etiqueta CAPTION. Para ver cómo funciona, se va a incluir en la declaración de la tabla anterior:

```
<TABLE BORDER=1>
  <CAPTION>
    Ejemplo de tablas
  </CAPTION>
  ...
</TABLE>
```

Esta etiqueta admite sólo un parámetro: ALIGN, que es por defecto TOP. Si se define como BOTTOM el título se colocará al final de la tabla en lugar del comienzo.

2.2.14 Marcos

Un marco (o *frame*) es una ventana independiente dentro de la ventana general del navegador. Cada marco tendrá sus bordes y sus propias barras de desplazamiento. Así cada página se dividirá en la práctica en varias páginas independientes.

Para crearlos se necesitará un documento HTML específico, que se llamará documento de definición de marcos. En él se especificará el tamaño y posición de cada marco y el documento HTML que contendrá. Se va a ver un ejemplo de este tipo de documento:

```
<HTML>
<HEAD>
  <TITLE>Mi primera página con marcos</TITLE>
</HEAD>
<FRAMESET COLS="20%,80%">
  <FRAME NAME="indice" SRC="indice.html">
  <FRAME NAME="principal" SRC="introduccion.html">
</NOFRAMES>
  <P>Lo siento, pero sólo podrás ver esta página
  si tu navegador tiene la capacidad de visualizar
  marcos.</P>
</NOFRAMES>
</FRAMESET>
</HTML>
```

Se va a explicar detalladamente este ejemplo antes de investigar algo más a fondo cada una de las etiquetas. Se puede observar que la cabecera de la página es similar a un documento normal, pero el habitual BODY es sustituido por un FRAMESET. En cada FRAMESET se divide la ventana actual (sea la general o un marco) en varias ventanas definidas o por el parámetro COLS o por ROWS. En éste, separado por comas, se define el número de marcos y el tamaño de cada uno.



Dentro del <FRAMESET> se hacen dos cosas. Primero, definir cada uno de los marcos poniéndoles un nombre y especificando qué fichero HTML le corresponde mediante la etiqueta <FRAME>. Por último, se especifica lo que verá el usuario en el supuesto (cada vez más raro) de que su navegador no soporte *frames* dentro de la etiqueta <NOFRAMES>. Ahora se verán todos estos elementos en mayor detalle.

2.2.14.1 Etiqueta <FRAMESET>

Según el estándar, esta etiqueta sólo debería contener el número y tamaño de cada marco, pero las extensiones de Netscape y Explorer al estándar obligan a estudiar un par de parámetros más. En general, los navegadores dibujan un borde de separación entre los marcos. Si se desea eliminarlo se puede hacer de dos maneras: en las etiquetas <FRAME> de cada una de los marcos contiguos al borde a eliminar o incluyendo el parámetro FRAMEBORDER=0 en el <FRAMESET>.

Cuando se elimina ese borde, se podrá ver cómo el navegador deja aún un hueco entre marcos. Este se elimina añadiendo los parámetros FRAMESPACING=0 BORDER=0.

Se examinarán por último los parámetros COLS y ROWS. Se deberá asignar una lista de tamaños separada por comas. Se admiten los siguientes formatos de tamaño:

- **Con porcentajes:** Al igual que con las tablas, se puede definir el tamaño de un marco como un porcentaje del espacio total disponible.
- **Absolutos:** Si se pone un número a secas, el marco correspondiente tendrá el tamaño especificado en pixels.
- **Sobre el espacio sobrante:** Si se coloca un asterisco (*) se estará indicando que se quiere todo el espacio sobrante para ese marco. Se puede poner este símbolo en varios marcos, para que se reparta el espacio equitativamente. Si se quiere que uno tenga más se deberá poner al asterisco un número delante. Así, un marco con un espacio de 3* será tres veces más grande que su compañero, que tiene un asterisco sólo.

Por ejemplo, el siguiente código es una muestra de cómo combinar los tres métodos:

```
<FRAMESET COLS="10%,*,200,2*">
```

Se puede suponer que el ancho total de la ventana son 640 pixels. El primer marco ocupará el 10%, es decir, 64 pixels. El tercero necesita 200, luego quedan 476 para los otros dos. Como el cuarto debe tener el doble de espacio que el segundo, se tiene aproximadamente 158 pixels para este último y 316 para el cuarto marco.

Hay que tener cuidado cuando se usan valores absolutos en la definición de marcos; se debe tener al menos un marco con un tamaño relativo si se quiere estar seguro del aspecto final de la página.

Por último, indicar que las etiquetas <FRAMESET> se pueden anidar. Esto se hace poniendo otro <FRAMESET> donde normalmente se colocan las etiquetas <FRAME> tal que así:

```
<FRAMESET COLS="20%,80%">
  <FRAME NAME="indice" SRC="indice.html">
  <FRAMESET ROWS="*,80">
    <FRAME NAME="principal" SRC="introduccion.html">
    <FRAME NAME="ejemplos" SRC="ejemplo.html">
  </FRAMESET>
</FRAMESET>
```



2.2.14.2 Etiqueta <FRAME>

Esta etiqueta define tan sólo las características de un marco determinado, no de un conjunto de ellos. Estos son los parámetros que admite:

NAME	Asigna un nombre a un marco para que después pueda ser referenciado.
SRC	Indica la dirección del documento HTML que ocupará el marco.
SCROLLING	Decide si se colocan o no barras de desplazamiento al marco para que se pueda desplazar por su contenido. Su valor es por defecto AUTO, que deja al navegador la decisión. Las otras opciones que tenemos son YES y NO.
NORESIZE	Si se especifica el usuario no podrá cambiar de tamaño el marco.
FRAMEBORDER	Al igual que su homónimo en la etiqueta <FRAMESET>, si se iguala a cero se eliminará el borde con todos los marcos contiguos que tengan también este valor a cero.
MARGINWIDTH	Permite cambiar los márgenes horizontales dentro de un marco. Se representa en pixels.
MARGINHEIGHT	Igual al anterior pero con márgenes verticales.

2.2.14.3 Acceso a otros marcos

Por defecto, cuando pulsamos sobre un enlace situado dentro de un marco, la nueva página a la que se quiere acceder se verá encerrada en ese mismo marco. Es posible que se desee que esto no ocurra. Por ejemplo, si se tiene un marco que no sirve de índice y otro donde se muestran los contenidos sería deseable que los enlaces del marco índice se abrieran en el otro marco. Esto es posible hacerlo gracias al parámetro TARGET.

Este parámetro se puede colocar en tres etiquetas: <A>, <AREA> y <BASE>. En las dos primeras sirve para indicar el marco en el que se abrirá ese enlace en particular y el último modificará el marco en el que por defecto se muestran todos los enlaces.

Pero para que un parámetro funcione, es habitual que le asignemos un valor, y TARGET no es una excepción. Para indicarle el marco que se desea se le asignará el nombre del mismo. Así, en los ejemplos anteriores, si en el marco llamado índice se tiene un enlace que se quiere que se abra en el marco principal pondremos:

```
<A HREF="pagina.html" TARGET="principal">
```

También existen cuatro nombres reservados que se podrán utilizar en el parámetro TARGET:

_top

Elimina todos los marcos existentes y muestra la nueva página en la ventana original sin marcos.

_blank

Muestra la nueva página en una ventana nueva y sin nombre del navegador.

_self

Muestra la nueva página en el marco donde está declarado el enlace.

_parent

Muestra la nueva página en el <FRAMESET> que contiene al marco donde se declara el enlace. En el ejemplo que se puso de <FRAMESET> anidados, una enlace situado en el marco ejemplo cuyo parámetro TARGET fuese igual a _parent eliminaría la separación entre los marcos ejemplo y principal y mostraría en ese nuevo marco la nueva página.



2.2.15 Sonido

Tanto Netscape como Internet Explorer incorporan desde hace tiempo la capacidad de reproducir sonido. El único problema es que los archivos suelen ser grandes y, siendo algo innecesario y superfluo, poca gente incluye melodías en sus páginas.

Los formatos que se puede asegurar que los navegadores reproducirán son los archivos WAV y MID. Para poder reproducir otros necesitarán el *plug-in* o añadido necesario.

2.2.15.1 Sonido activado por el usuario

La manera más sencilla de incluir sonidos es dejando al usuario la decisión de escucharlos o no. Para hacerlo incluiremos el sonido en el parámetro HREF de un enlace, como si fuera una página HTML:

```
<A HREF="hola.wav">Si pulsas te saludo<A>
```

2.2.15.2 Sonido de fondo

Lo del sonido de fondo ya es más complicado, ya que Netscape y Explorer ofrecen soluciones propietarias, distintas e incompatibles de hacer sonar un archivo de fondo.

En Explorer, desde la versión 2.0, se pueden incluir fondos sonoros utilizando la etiqueta BGSOUND:

```
<BGSOUND SRC="musica.mid">
```

El parámetro SRC indicará el archivo a reproducir. Esta etiqueta admite también otro parámetro, LOOP, que indica el número de veces consecutivas que sonará el fichero. Si se indica LOOP="infinite", el archivo se reproducirá indefinidamente, mientras se esté en la página.

Netscape utiliza su etiqueta <EMBED>. Teóricamente, esta etiqueta debería servir para unir objetos de varios tipos a la página web, pero en la práctica sólo se utiliza para esto. Esta etiqueta tiene los siguiente parámetros:

Parámetro	Utilidad
SRC	Contiene el nombre de archivo de sonido a reproducir
WIDTH y HEIGHT	En Netscape aparece un pequeño reproductor, estos parámetros especifican su tamaño.
AUTOSTART="true"	Arranca automáticamente la reproducción.
LOOP="true"	Reproduce ininterrumpidamente el fichero hasta que salimos de la página.
HIDDEN="true"	Oculto el reproductor.

Sin embargo, y debido a algunos bugs, si se quiere reproducir infinitamente un archivo con el reproductor oculto, se deberán incluir todos los parámetros, incluyendo WIDTH y HEIGHT. Además, si el usuario tiene algún *plug-in* de sonido extraño, en lugar del que viene con Netscape, es posible que deje de funcionar correctamente.

Dado que ambas etiquetas son incompatibles entre sí, basta con incluir las dos... o, mejor dicho, bastaba. Ahora el Explorer es capaz de interpretar <EMBED>, pero no exactamente de la misma manera, lo que provoca que aparezca una ventana aparte con el reproductor. En definitiva, la mejor manera de mostrar una música de fondo es usando un pequeño *script* que averigüe en qué navegador está instalado y discrimine.



2.2.16 Guía de estilo

Este punto del tema consiste en una serie de consejos que pueden servir para hacer páginas más atractivas y de mayor calidad. Es recomendable leerla aunque luego no se siga al pie de la letra.

2.2.16.1 Contenido

Cualquier página será visitada si su contenido es interesante. Casi da lo mismo su estructura y diseño si la información es lo suficientemente interesante. Así que quizá sea conveniente detenerse un poco en qué es lo que debe tener la página web.

En Internet uno visita lo que quiere, y la mayoría de las veces lo que a la gente le interesa y lo que a nosotros nos interesa que vean no es lo mismo. Así pues, el mejor modo de conseguir que vean lo que queremos es ofrecerles lo que quieren. De todas maneras no hay que engañarse: los contenidos que te interesa que la gente conozca y que no sean "populares" no serán visitados más que por una pequeña parte de los interesados por los contenidos "populares".

El averiguar qué poner es fácil. Prácticamente todo el mundo tiene algún *hobby*, alguna pasión, o domina algún tema bastante por encima de la media. Sobre ese tema deberá estar enfocada su página. Porque, para qué engañarse: a nadie le interesan esas páginas personales que sólo contienen tres fotos del autor, una de su novia y otra de su perro, junto con la historia de su vida y un curriculum y una serie de enlaces a páginas de sus amigos.

Conviene también, sabiendo que el usuario ha acudido a tu página por el interés hacia un tema específico, facilitarle enlaces a páginas similares. Te lo agradecerá.

2.2.16.2 Navegación

Es muy importante que sea sencilla e intuitiva la navegación por las páginas que componen tu web. Dependiendo del tamaño de la página no se debería tardar más de tres toques de ratón en ir de una página a otra. Sin embargo, esto no se debe conseguir por medio de una saturación de enlaces porque el usuario podría perderse.

Hay muchas maneras de conseguirlo. La primera es utilizar un marco como índice, que siempre permitirá acudir a las secciones principales del web. También resulta útil incluir enlaces a la página principal y a la página principal de la sección que el usuario está consultando.

En definitiva, no hay receta única, pues el éxito del sistema de navegación depende en buena medida de los contenidos y estructura del web. Resulta útil ver a alguien ajeno a su desarrollo navegar por ella sin darle ninguna clase de pistas: en general se pueden ver con facilidad los mayores problemas que pueda tener.

Sin embargo, hay algo que siempre irrita al usuario: enlaces a páginas vacías o con un gráfico que señale que esa página está en construcción. Si lleva tiempo desarrollar una sección pero se quiere que los usuarios sepan que estará ahí colocala con las demás en la lista de enlaces, pero que ella no tenga. Que se vea el nombre aunque no lleve a ninguna parte.

2.2.16.3 Estructura de las páginas

Es fundamental a la hora de ponerse a crear páginas el pensar en una estructura común y consistente, porque facilita la navegación y le confiere personalidad propia. Esa estructura puede estar dada por una manera común de titular las secciones, un fondo, un logo, una manera común de colocar gráficos y texto, o una combinación de todo esto.

Hay que cuidar la longitud de las páginas: que no sean ni demasiado cortas ni demasiado largas. La página debe contener información suficiente para resultar interesante y no tanta como para desanimar al posible lector.

Se debe recordar que el título que se pone a la página es por lo que se la conocerá por el mundo. Es lo que guarda el navegador en su índice de marcadores, favoritos o bookmarks. Por ello, es conveniente que sea significativo y pueda ser entendido fuera de contexto.

Conviene que el primer párrafo o las primeras frases de todas las páginas contengan un resumen o idea del contenido del resto de la misma. Eso ayuda al usuario a localizar más fácilmente la información.



A su vez, resulta bastante útil indicar la fecha de la última modificación de la página, aunque es aún mejor si esa fecha hace referencia a la última modificación del contenido, que al fin y al cabo es lo que le interesa el usuario.

Por último, uno de los aspectos más importantes de Internet como medio es la facilidad de comunicación entre creadores y usuarios: se puede facilitar incluyendo la dirección de correo.

2.2.16.4 Accesibilidad

Hay que tener en cuenta que, en general, no se sabe con qué navegador se visitarán las páginas. Ni cómo estará configurado. Ni la resolución del monitor. Ni la potencia de su CPU. Ni siquiera si el navegador es capaz de ver gráficos. Y como se supone que la red es algo de lo que debe poder disfrutar todo el mundo conviene hacerla accesible.

Para ello hay que tener en cuenta algunas cosas. Para empezar, que las páginas siempre se ven distintas dependiendo del navegador que se use. Conviene no usar elementos que sólo funcionen con un navegador, como puede ser el texto parpadeante o las marquesinas. Y siempre será bueno probar con más de un navegador las páginas: no siempre representan igual las mismas etiquetas.

Hay gente que viaja por la red con navegadores en modo texto o que, debido a la lentitud de su conexión, configuran su navegador para que no cargue los gráficos. Por ello, conviene añadir siempre el parámetro ALT a todas las imágenes.

Es aconsejable probar las páginas a la resolución mínima de 640x480 y comprobar que se puede ver y navegar con relativa comodidad. Es desgraciadamente frecuente ver marcos índice diseñados para verse a 800x600 y que impiden cambiarlos de tamaño y se han eliminado las barras de desplazamiento.

Es conveniente no abusar de los applets Java ni hacer de ellos un elemento imprescindible en nuestra web. A gente con pocos recursos en potencia de CPU o usuarios de Windows 3.1 no les resulta muy agradable estas aplicaciones.

En definitiva, no hacerles la vida imposible a los minusválidos de la red. Eliminar barreras arquitectónicas.

2.2.16.5 Diseño

Una página atractiva, con gráficos bonitos y bien escogidos y colocados es algo siempre recomendable. Pero no si, debido a ello, la página tarda dos días y medio en terminar de cargarse. Hay que tener en cuenta que se está trabajando en un medio llamado Internet que, aunque denominado autopista de la información, en realidad suele parecerse más bien a una carretera comarcal. Así pues, a la hora de diseñar las páginas, se deberá cuidar el equilibrio entre belleza gráfica y tiempo de carga.

Recuerda que el gráfico de fondo no debe dificultar la lectura del texto. Si se utiliza uno oscuro que el color del texto sea claro y viceversa.

Los gráficos animados, en general, resultan muy entretenidos para el autor pero poco útiles para el usuario. No deberían usarse más que para cosas sobre las que realmente quieres llamar la atención.

Mientras se pueda, evita los gráficos meramente ornamentales. Que tengan algo de utilidad, sea como título o enlace o lo que sea.

Se debe procurar evitar el exceso de líneas horizontales: dan la sensación de una página inconexa y troceada.

Se debe evitar la proliferación de marcos. Si se ponen muchos es posible que el usuario se pierda y no sepa en cuál está cada cosa. Hay que tener en cuenta que sólo puede haber uno con información y todos los demás deben ser auxiliares de este. El marco principal debe destacar sobre los otros por su mayor tamaño.



2.2.16.6 Mantenimiento y pruebas

Resulta conveniente que el código sea claro y fácilmente comprensible: así se evitan los problemas de no entender cómo se hizo una página en un momento dado, o de no encontrar esa parte que se tiene que modificar entre tanto texto.

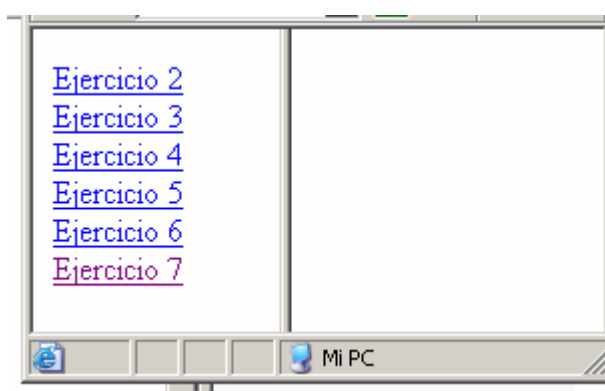
Resulta conveniente que otras personas prueben las páginas. Hay que cuidar que los probadores no conozcan previamente las páginas ni hayan tenido que ver con su desarrollo.

2.2.17 Ejercicios

1. Realiza una página en la que utilices el formato de párrafo y las cabeceras.
2. Realiza una página donde utilices todas las listas ordenadas y desordenadas.
3. Busca una imagen en el ordenador y realiza una página que contenga esa imagen y que sea un link (enlace) a la página realizada en el ejercicio 2.
4. Realiza una página que tenga una imagen de fondo y una lista ordenada que empiece en el número 4.
5. Realiza una página cuyo fondo sea #FFFF00, la letra White y de tamaño 6.
6. Realiza un formulario que tenga una caja de texto, cuyo tamaño sea 40 y el número máximo de caracteres sea 35, y un botón.
7. Realiza una tabla como la que se muestra en la figura.

1	2	3
4		
5	6	7
	9	10
8	11	12

8. Realiza una página como la que se indica en la figura, donde los links del menú hagan referencia a las páginas generadas en los ejercicios anteriores, es decir del ejercicio 2 al 7.



9. Realiza una página que al abrirla emita un sonido de forma infinita.
10. Realiza el siguiente formulario:



Apellidos:

Nombre:

Dirección:

Ciudad:

Estado/Prov:

CP:

Correo Electrónico:

11. Realiza el siguiente formulario:

Ficha de registro del alumno

Nombre y Apellidos:	
Nombre de usuario:	
Contraseña:	
Repetir contraseña:	
DNI:	
Lugar de nacimiento:	
☐ ¿Primer curso?	☐ Diseño HTML ☐ Metodología ☐ Arquitectura



Sistema operativo	Windows 2000 MacOS X RedHat Linux Free BSD
Prueba de imagen	 (Pulsar para enviar)
Enviar los datos	Borrar los datos



12. Realiza las siguientes tablas:

Encabezado colspan=3		
Rowspan=2	Colspan=2	
	Colspan=2	
Celda normal	Celda normal	Celda de 250 px

Rowspan=2	Colspan=4			
	Colspan=2		Colspan=2	
Celda	Celda	Rowspan=2	Celda	Celda
Celda	Celda		Celda	Celda
Celda	Celda	Celda	Celda	Celda

	
Enlace 1 Enlace 2 Enlace 3 Enlace 4 Enlace 5	 Imagen del logotipo .NET dentro de una celda de la tabla
Bill Gates sigue siendo considerado uno de los hombres más ricos del mundo. A pesar de las dificultades de muchas de las empresas relacionadas con Internet, Microsoft continúa en alza.	Quejas de usuarios de Windows XP: Numerosos usuarios de Windows XP se están quejando de la poca información obtenida de los frecuentes errores del tipo "userenv", que pueden ser inspeccionados desde el visor de sucesos. Estos errores suceden cuando XP no puede descargar adecuadamente el registro, pero es casi imposible averiguar la causa que los origina.



13. Realiza los siguientes frames:

