CURSO DE CREACIÓN DE PÁGINAS WEB EN HTML

PARA REALIZAR LA PÁGINA DEL INSTITUTO

Realizado por: Antonio Bueno

CURSO DE CREACIÓN DE PÁGINAS WEB EN HTML

INDICE:

1.- **Software y Hardware** necesario. 3
2.- **Consideraciones** generales. 3
3.- Tipos de **formatos de transmisión** de datos por iNET. 5
4.- **Estándares** del HTML. 6
5.- **Estructura** de la página básica.
(HTML, HEAD, BODY, CENTER, H1, HR, TITLE, P) 7
6.- Los **caracteres especiales.**
(Acentos, eñes, espacios, etc) 8
7.- Formateando el **texto**. (BR, PRE, B, I, TT, BLOCKQUOTE, SUB, SUP,
UL, LI, OL, DL, DD, DT, <!--, -->) 10
8.- **Enlaces** con otras páginas. <A href=”xxx”> yy</A> 14
9.- Inserción de **Imágenes**.
<IMG SRC="portalesocom.gif" ALT="PortalESO"> 18
10.- Creación de una **página de estilos** CSS.
(Style, div, span style, style type, link) 22
11.- Inserción de **fondos**. (Bgcolor, text, link, alink, background)
..... 34
12.- Alineación y dimensionado de **imágenes**.
(Top, middle, buttom, align, clear, width, height)..... 38
13.- **Tablas**. (Table, td, tr, th, caption, border, align, valign, width, heigth, colspan,
rowspan, bgcolor, gackground, cellspacing, cellpadding) 40
14.- **Formularios**. (Form, action, enctype, method, input, textarea, radio, select, checked)
. 50
15.- **Gif animados**. 58
16.- **Menús** realizados **con Mapas**.
(Map, Name, Area, href, scr, shape, coords, usemap, nohref) 59
17.- Páginas que utilizan **Frames**.(Frame, frameset, target, src, name, marginwidth,
marginheight, scrolling, noresize, frameborder) 63
18.- Inserción de **Sonidos**.
(Bgsound, embed, loop, ahref, src) 73
19.- Inserción de **JavaScripts**. (script, language) 76
20.- Inserción de **Applets de Java**.
(Applet, code, width, height, param, name, value, codebase)..... 79
21.- Inserción de películas **Flash**. (Embed, scr, width, height, type, quality, bgcolor, scale,
border, align, noembed, div) 84
22.- **Publicación** de una Web. 86
23.- Otras cosas. 89
24.- Punteros, referencias y Bibliografía. 92

1.- Software y Hardware necesario.

Este curso esta pensado para realizarse en el siguiente equipo (**Hardware**):

- Pentium o superior.
- 64 Mb Ram o más.
- Disco Duro de 200 Mb o más
- Conexión a iNET, 56K o más. (en caso de no existir no se podrá publicar la página).

El **Software** utilizado es:

- Windows 95 /98 o superior.
- I Explorer 5.0 o superior.
- Editor ASCII, Notepad, EDITPAD (se puede obtener gratis en iNET)
- Software para realizar FTP, Windows Commander u otro.
- Software de tratamiento grafico, Photo editor de Microsoft, Paint Shop Pro, Etc.

Método de trabajo:

Con el procesador de textos crearemos un fichero con el nombre que queramos (p. Ej. *principal*), pero que debe tener necesariamente la extensión *.html* (o *.htm* si nuestro sistema operativo no soporta extensiones de más de tres letras) . Cada vez que escribamos algo nuevo en este fichero *principal.html*, debemos guardarlo antes de cargarlo en el navegador.

Para cargar nuestro documento HTML en el navegador, utilizaremos la herramienta adecuada para encontrar donde esté guardado el fichero, y abrirlo:

- En el I Explorer: Archivo/Abrir/Abrir Archivo

- En el Netscape: File/Open File

2.- Consideraciones generales.

2.1. ¿Qué es Internet?.

Internet es una malla mundial de ordenadores y redes de computadores interconectados. Se puede decir que Internet (iNET) es una red de redes. Es una red de tipo WAN que integra multitud de LANs en única red de comunicaciones por todo el mundo.

2.2. Terminología y Argot de Internet.

Con la tecnología de INet es amplia y variada, su terminología tiene la más variada procedencia:

- Informática y sistemas operativos: Directorio, Mega (Mb), Kb, password, etc.
- Tecnología de redes: Módem, servidor, cliente, etc.
- Propios de Internet: e-mail, www, URL, etc.

- Costumbres y modos de los usuarios de iNet: símbolos emoticones :(, mayúsculas para gritar, etc.

2.3. Acceso y comunicación.

¿Cómo viaja la información por iNET?.

Viaja gracias a los llamados protocolos que se instalan en el ordenador para poder enviar y recibir la información. Los procesos que se utilizan son:

- División de la información: El protocolo TCP divide la información en paquetes con una cabecera para ordenarlos a la llegada.
- El protocolo IP cataloga el paquete con la dirección. Siempre se utiliza conjunto al TCP (TCP/IP).
- Los Routers trasladan la información a la dirección indicada por el IP a lo largo de la red y cada paquete por una ruta diferente.
- En la recepción el protocolo TCP se encarga de ensamblar los paquetes ordenadamente para tener una copia fiel del original.

El Acceso.

Se ha de realizar mediante una plataforma de acceso. Es un sistema de conexión mediante el cual se puede acceder a Internet a través de una línea telefónica o red de ordenadores. En el caso de la línea telefónica el gasto solo es el consumo del teléfono.

Las plataformas telefónicas más usuales en España son: Infovía Plus, Retenet, Uni2, ONO, etc. La velocidad máxima viene determinada por la línea telefónica y el módem (56Kbits/segundo).

Las plataformas de cable en Valencia son: ONO y Telefónica (ADSL). Se puede contratar hasta un máximo de 2Mbits/segundo.

¿Qué ofrece iNET?

- Uso de recursos remotos: permite trabajar de forma remota sobre otra máquina.
- Transferencia de archivos.
- Búsqueda y localización.
- Correo electrónico.
- Integración de información.

¿Quién es el responsable de iNET?.

No existe un gobierno único y no tiene dueño. Todos somos dueños. Pero existen normas de circulación que podemos dividir en dos grupso:

a.- Control de las direcciones de dominios. Una dirección para un ordenador(IP).

b.- Adopción de estándares. Que todos utilicen un único sistema de comunicación aunque usen distintos sistemas operativos.

Entre los grupos y organismos que trabajan por el funcionamiento de iNET destacan:

a.- DNS Internacional: controla los dominios internacionales ".com, otros"
b.- Network Center de España: controla los dominios ".es"
c.- Comisión del Mercado de las Telecomunicaciones: vela por que sea correcta la competencia en el campo de las telecomunicaciones.
d.- Asociación de Internautas y Asociación de Usuarios de Internet: velan por los derechos de los usuarios en la Red.
e.- Fronteras Electrónicas de España: se encarga de velar por la libertad de expresión y la intimidad de Internet.

3.- Tipos de formatos de transmisión de datos por iNET

3.1. Acceso universal a iNET.

Internet ha roto con la mayoría de las barreras: Utiliza un lenguaje común a todos los sistemas operativos, es un medio excelente para facilitar la comunicación a las personas con ciertas discapacidades como sordos, ciegos a través de aplicaciones específicas, etc. Otras barreras que se ha roto son las de los idiomas con traductores simultáneos automáticos, y la inmediatez de la transmisión-recepción de los datos, entre cualquier lugar del mundo.

3.2. Contenido.

a.- Texto: Es su componente fundamental.
b.- Enlaces: Son textos o imágenes que conducen a otras zonas de la misma página o a otras páginas. Suelen aparecer subrayados, con otro color o cambiando de aspecto el cursor.
c.- Imágenes y animaciones.
d.- Sonido.

3..3. Dirección de acceso = URL. Consta de:

a) Protocolo: Depende del tipo de acceso que se hace a la información.

- **http**: Accede a un servidor de páginas web HTML, tan sólo se pueden leer los archivos. Ej: http://www.portaleso.com/
- **ftp**: Accede a un servidor de ficheros, se pueden traer o enviar ficheros al servidor. Ej: ftp://www.portaleso.com/index.html
- **malito**: Precede a una dirección de correo electrónico. Envía un correo electrónico a la dirección indicada. Ej: malito:portaleso@ono.com este sistemas de acceso, no se puede hacer directamente desde el navegador.
- **file**: Se accede a un archivo de nuestra máquina. Ej: file://c:/autoexec.bat
- **news o nntp**: Accede a un servidor de noticias para leerlas.

Ej: http://newsforge.com/newsforge.rdf ofrece una dirección de noticias en lenguaje rdf, para extraer de él las noticias y sus enlaces. Otro ejemplo es el de http://odisea.org/backend.php donde el programa está preparado para php.

b) Dominio: Suele tener la forma **máquina.subred.organización.país** y puede estar formado por varios subdominios separados por puntos. Ejemplo: gva.es, cult.gva.es, centres.cult.gva.es.

El último subdominio corresponde a un sentido especial:

- **.com** Comercial.
- **.edu** Educación o investigación.
- **.net** Organizaciones realacionadas con Internet.
- **.org** Organizaciones no clasificadas.
- **.gov** Gobierno de los EEUU.
- **.mil** Fuerzas armadas de los EEUU.
- **.es** España.

Y así el resto de los países.

c) Directorios o carpetas del ordenador servidor.

d) Nombre del fichero a leer.

e) Ejemplo: http://www.portaleso.com/index.html

* Protocolo: **http** * Dominio: **www.portaleso.com**
* Directorio: **/** * Archivo a leer: **index.html**

3.4. Observaciones:

Hacer notar que el símbolo de directorio en internet se expresa así / y no como habitualmente estamos acostumbrados en windows \ .

4.- Estándares del HTML.

En la actualidad el estándar HTML es el **HTML 4.01** y se puede encontrar en http://www.w3.org/TR/html401/ . Como todo lo que ocurre con la informática todo evoluciona rápidamente y así ha ocurrido.

Hasta hace poco, la última versión oficial del HTML era la **HTML 2.0**. Cumplía su función perfectamente, por lo que casi todos los navegadores lo implementan correctamente, pero muchos usuarios del HTML querían tener un control mayor sobre sus documentos, tanto en el formateo del texto como en la apariencia de la página.

Así el comité de trabajo del HTML de la IETF, propuso un borrador de un nuevo estándar, el **HTML 3.0** que incluía numerosas y útiles mejoras. Pero debido a que era excesivamente extenso y ambicioso, los navegadores sólo utilizaban una pequeña parte del nuevo estándar, lo que producía confusión y por lo que acabó finalmente siendo abandonado.

Cada vez era más evidente la necesidad de un nuevo estándar que fuera aceptado por todos. Se formó otro comité, el W3C (www.w3.org/pub/WWW), apoyado por los principales vendedores de software (entre ellos, IBM, Microsoft, Novell, Netscape, Sun, etc.), y finalmente, en Enero de 1997 se aprobo otro nuevo estándar el **HTML 3.2**.

En Julio de 1997 se presento el primer borrador público del próximo estándar oficial **HTML 4.0**. En él se incorporan las especificaciones sobre tablas, frames, scripts, hojas de estilo, accesibilidad por distintos medios, e internacionalización (incluyendo el uso de Unicode, en lugar del Latin-1).

5.- Estructura de la página básica.

(HTML, HEAD, BODY, CENTER, H1, HR, TITLE, P)

El principio esencial del lenguaje HTML (HyperText Markup Language) es el uso de las **etiquetas** (*tags*). Funcionan de la siguiente manera:

<XXX> Este es el **inicio** de una etiqueta.
</XXX> Este es el **cierre** de una etiqueta.

Las letras de la etiqueta pueden estar en mayúsculas o minúsculas, indiferentemente. Por claridad, usaremos solamente las mayúsculas.

Lo que haya entre ambas etiquetas estará influenciada por ellas. Por ejemplo, todo el documento HTML debe estar entre las etiquetas **<HTML>** y **</HTML>**:

<HTML> [Todo el documento] **</HTML>**

El documento en sí está dividido en dos zonas principales:

- El **encabezamiento**, comprendido entre las etiquetas **<HEAD>** y **</HEAD>**
- El **cuerpo**, comprendido entre las etiquetas **<BODY>** y **</BODY>**

Dentro del encabezamiento hay información del documento, que no se ve en la pantalla principal, principalmente el **título** del documento, comprendido entre las etiquetas **<TITLE>** y **</TITLE>**. El título debe ser breve y descriptivo de su contenido, pues será lo que vean los demás cuando añadan nuestra página a su *bookmark* (agenda de direcciones o favoritos).

Dentro del cuerpo está todo lo que queremos que aparezca en la pantalla principal (texto, imágenes, etc.)

Por tanto, la estructura queda de esta manera:

<HTML>
<HEAD>
<TITLE> ***Título de la página*** **</TITLE>**
</HEAD>
<BODY>
[*Aquí van las etiquetas que visualizan la página*]
</BODY>
</HTML>

Antes de crear nuestra primera página, unas consideraciones sobre el texto: Cuando escribimos en el documento el texto que queremos que aparezca en la pantalla, veremos que éste se acomoda a ella, sin que tengamos que pulsar el retorno del carro. Si queremos separar el texto en distintos **párrafos** debemos usar la etiqueta **<P>** y **</P>.**

El texto puede tener unas **cabeceras**, comprendidas entre las etiquetas **<H1>** y **</H1>**, **<H2>** y **</H2>**, etc. (hasta el número 6), siendo el número indicativo del tamaño. El tamaño mayor es el correspondiente al número 1. Puedes experimentar en el ejemplo que sigue, cambiando el número para comprobar el efecto que se logra.

Una etiqueta muy interesante es la de **centrado** **<CENTER>** y **</CENTER>** (no la soportan todos los navegadores, aunque sí la mayoría de ellos). Nos centra todo lo que esté dentro de ella, ya sea texto, imágenes, etc.

También tenemos los **separadores** (*horizontal rules*), que se consiguen con la etiqueta **<HR>** (no existe la correspondiente de cierre). Con ella se obtiene una raya horizontal tan ancha como la pantalla, y con la apariencia de estar embutida sobre el fondo, como se puede observar a continuación:

Ejemplo práctico

En el procesador de texto copiamos lo siguiente:

```
<HTML>
  <HEAD>
    <TITLE> Curso para realizar la web de centro </TITLE>
  </HEAD>
 <BODY>
    <H1> <CENTER> P&aacute;gina Inicial </CENTER> </H1>
    <HR>
    Esta es mi primera p&aacute;gina, aunque todav&iacute;a es muy sencilla. Como
el lenguaje HTML no es dif&iacute;cil, pronto estar&eacute; en condiciones de
hacer cosas mas interesantes.
    <P> Este es el segundo parrafo. </P>
  </BODY>
</HTML>
```

Este es el aspecto que tendrá:



Actividades:
1.- Cambia el texto de la página y observa como se modifica.
2.- Convierte en párrafo el texto del primer fragmento y observa las modificaciones.

6.- Los caracteres especiales.

(Acentos, eñes, espacios, etc.)

Existen algunas limitaciones para escribir el texto. Una de ellas es debido a que las etiquetas se forman como un comando escrito entre los símbolos **"<"** y **">"**. Por tanto, si se quisieran escribir estos caracteres como parte normal del texto, daría esto lugar a una ambigüedad, ya que el programa navegador podría interpretarlos como el comienzo o final de una etiqueta, en vez de un carácter más del texto.

Para resolver este problema, existen unos **códigos** para poder escribir estos caracteres y otros relacionados con las etiquetas.

<	**para**	**<**	**(less than, menor que)**
>	**para**	**>**	**(greater than, mayor que)**
&	**para**	**&**	**(ampersand)**
"	**para**	**"**	**(double quotation)**

Como se ve, estos códigos empiezan siempre con el signo **&** y acaban siempre con **;**

De una manera similar, existen códigos para escribir letras específicas de distintos idiomas. Hay muchos de ellos, pero, lógicamente, los que más nos interesan son los propios del castellano (las vocales acentuadas, la ñ y los signos ¿ y ¡)

Los códigos de las **vocales acentuadas** se forman comenzando con **&**, seguido de la vocal en cuestión, seguido de la palabra **acute** *(aguda)* y terminando con el signo **;**

á	**para**	**la**	**á**	**Á**	**para**	**la**	**Á**
é	**para**	**la**	**é**	**É**	**para**	**la**	**É**
í	**para**	**la**	**í**	**Í**	**para**	**la**	**Í**
ó	**para**	**la**	**ó**	**Ó**	**para**	**la**	**Ó**
ú	**para**	**la**	**ú**	**Ú**	**para**	**la**	**Ú**

El resto de los códigos son:

ñ	**para**	**la**	**ñ**	**Ñ**	**para**	**la**	**Ñ**
ü	**para**	**la**	**ü**	**Ü**	**para**	**la**	**Ü**
¿	**para**	**¿**		**¡**	**para**	**¡**	
** **	**para**	**el**	**espacio**				

Se puede consultar el juego completo de caracteres en el manual de HTML Tejedores del Web, de Carlos Castillo, en la sección Set de caracteres (http://www.tejedoresdelweb.com/307/article-1836.html).

Todo esto, que como se ve es muy laborioso, puede parecer inútil ya que si escribimos nuestro texto sin hacer ningún caso de estas convenciones, escribiendo las letras acentuadas y demás signos directamente, es muy posible que el resultado lo veamos correctamente en nuestro navegador, pero nunca podremos estar seguros que les ocurra lo mismo a todos los que accedan a nuestras páginas con otros navegadores distintos.

En la práctica, el problema no es tan grave, pues los programas editores de HTML suelen tener la posibilidad de escribir los códigos automáticamente.

Incluso si se escribe en un procesador de textos de forma manual, se puede escribir el texto en una primera etapa de forma normal y luego aplicarle algún de los programas que hacen la conversión adecuada.

Como ejemplo, observar como se ha escrito el texto en el ejemplo anterior, sobre todo los acentos.

Actividades:

1.- Introduce en la página anterior un tercer párrafo que diga:

"Crear una página de web es muy fácil, tan sólo es necesario fijarse un poco. Para obtener una sangría en la primera línea basta con añadir espacios al principio de la línea"

2.- Realiza la sangría de todas las primeras líneas con cinco espacios. Observa que pasa si cambias de línea en medio del párrafo.

3.- Divide el párrafo segundo en dos.

7.- Formateando el texto.

(BR, PRE, B, I, TT, BLOCKQUOTE, SUB, SUP, UL, LI, OL, DL, DD, DT, <!--, -->)

Como hemos visto en el ejemplo del capítulo anterior, cuando queremos poner un texto sin ninguna característica especial, lo ponemos directamente. Únicamente, la separación entre párrafos (dejando una línea en blanco) la conseguimos con la etiqueta **<P> </P>**.

Si queremos separar los párrafos, o cualquier otra cosa, pero sin dejar una línea en blanco, usamos una etiqueta parecida **
** (*break*, o romper) al final de la línea. No tiene etiqueta de cierre.

Si queremos obtener **múltiples líneas en blanco** basta con repetir la etiqueta **
**. Así por ejemplo, si queremos obtener cuatro líneas en blanco, pondríamos:

```
<BR/>
<BR/>
<BR/>
<BR/>
```

Con lo que se obtienen las líneas en blanco.

Al escribir el texto, si ponemos más de un **espacio en blanco** entre dos palabras observamos que el navegador sólo reconoce uno de ellos. Si queremos forzarle a que lo haga, debemos poner el código " " *(non-breaking space)*.

Para **destacar** alguna parte del texto se pueden usar:

- **<B>** y **</B>** para poner algo en **negrita** (*bold*).
- **<I>** y **</I>** para poner algo en *cursiva* (*italic*).

Otra etiqueta interesante es **<PRE>** y **</PRE>**. El texto que se encuentre entre ella estará **preformateado**, es decir que aparecerá como si hubiera sido escrito con una máquina de escribir, con una fuente de espaciado fijo (tipo Courier). Además se respetarán los espacios en blanco y retornos del carro, tal como estaban en nuestro documento HTML (lo cual no ocurre normalmente, como hemos visto anteriormente). Es muy apropiada para confeccionar **tablas** y otros documentos similares.

Con la etiqueta **<TT> y </TT>** conseguimos también que el texto tenga un tamaño menor y la apariencia de los caracteres de una **máquina de escribir** (*typewriter*). La diferencia con la anterior es que no preformatea el texto, sino que únicamente cambia su apariencia.

> La etiqueta **<BLOCKQUOTE>** y **</BLOCKQUOTE>** se utiliza para destacar una **cita textual** dentro del texto general.

En las fórmulas matemáticas puede interesar poder escribir **índices** y **subíndices**, que se consiguen con las etiquetas **** y **** respectivamente. Así, por ejemplo:

m^2 se consigue de la siguiente manera: **m²**
v_x se consigue con: **v_x**

A menudo nos interesará presentar las cosas en forma de **listas**. Podemos escoger entre tres tipos distintos:

1. Listas desordenadas (no numeradas)
2. Listas ordenadas (numeradas)
3. Listas de definición.

Las **listas desordenadas** (*unordered lists*) sirven para presentar cosas que, por no tener un orden determinado, no necesitan ir precedidas por un número. Su estructura es la siguiente:

```
<UL>
   <LI>      ALICANTE
   <LI>      CASTELL&Oacute;N
   <LI>      VALENCIA
   <LI>      Etc.
</UL>
```

Es decir, toda la lista está dentro de la etiqueta **<UL> y </UL>,** y luego cada cosa va precedida de la etiqueta **<LI>** (*list item*).

El resultado de lo anterior es el siguiente:

- ALICANTE
- CASTELLÓN
- VALENCIA
- Etc.

Se puede anidar una lista dentro de otra. Por ejemplo:

```
<UL>
<LI> Mam&iacute;feros
<LI> Peces
  <UL>
  <LI> Sardina
  <LI> Bacalao
  </UL>
<LI> Aves
</UL>
```

Que daría el siguiente resultado:

- Mamíferos
- Peces
  - Sardina
  - Bacalao
- Aves

Las **listas ordenadas** (*ordered lists*) sirven para presentar cosas en un orden determinado. Su estructura es muy similar a la anterior. La diferencia estriba en que en el resultado aparecerá automáticamente un número correlativo para cada cosa.

```
<OL>
  <LI>Primera evaluaci&oacute;n
  <LI>Segunda evaluaci&oacute;n
  <LI>Tercera evaluaci&oacute;n
  <LI>Etc.
</OL>
```

El resultado es:

1. Primera evaluación
2. Segunda evaluación
3. Tercera evaluación
4. Etc.

Al igual que las listas desordenadas, también se pueden anidar las listas ordenadas.

El tercer tipo lo forman las **listas de definición**. Como su nombre indica, son apropiadas para glosarios (o definiciones de términos). Toda la lista debe ir englobada entre las etiquetas **<DL>** y **</DL>**. Y a diferencia de las dos que hemos visto, cada renglón de la lista tiene dos partes: 1) el nombre de la cosa a definir , que se consigue con

la etiqueta **<DT>** (*definition term*) y 2) la definición de dicha cosa, que se consigue con la etiqueta **<DD>** (*definition definition*).

```
<DL>
 <DT> ¿Qu&eacute; es la electricidad?
  <DD> Es el paso ordenado de electrones a trav&eacute;s de un conductor.
 <DT> ¿Cu&aacute;l es la unidad de medida en el SI?
  <DD> La unidad de medida en el SI es el metro.
</DL>
```

Su resultado es:

¿Qué es la electricidad?
Es el paso ordenado de electrones a través de un conductor.
¿Cuál es la unidad de medida en el SI?
La unidad de medida en el SI es el metro.

Comentarios no visibles en la pantalla

A veces es muy útil escribir **comentarios** en el documento HTML sobre el código que escribimos, que nos pueden servir para recordar posteriormente sobre lo que hicimos, y que no queremos que se vean en pantalla.

Esto se consigue encerrando dichos comentarios entre estos dos símbolos: **<!--** y **-->**
Ejemplo:

```
<!-- Esto es un comentario que no se verá en pantalla -->
```

Escribe el ejemplo siguiente en el procesador de texto y guárdalo con el nombre pagina2.html, llámalo con el explorador y observa el resultado.

Ejemplo práctico

```
<HTML>
 <HEAD>
  <TITLE> Curso para realizar la web de centro - 2 </TITLE>
 </HEAD>
 <BODY>
  <CENTER>
  <H1> Departamentos del Centro </H1>
  </CENTER>
  <HR>
   Sin un orden particular, los<B> departamentos </B> son los
   siguientes:
  <UL>
    <LI> Tecnología
    <LI> Lenguas
    <UL>
      <LI> Lengua Española
```

```
    <LI> Lengua Valenciana
    <LI> Lengua Inglesa
    <LI> Lengua Francesa
   </UL>
  <LI> Matemáticas
 </UL>
  La clase que más me gusta es  <I> (en orden de preferencia): </I>
 <OL>
   <LI> Matemáticas
   <LI> Lengua Española
   <LI> Tecnología
 </OL>
</BODY>
</HTML>
```

Actividades:

1.- Modifica el archivo anterior con los correspondientes códigos, para los acentos, ñ, y espacios necesarios.

2.- Confecciona una lista con todos los departamentos y áreas que contienen cada uno de ellos, ordenándolos en primer lugar el que más profesores tiene, y así sucesivamente.

3.- Introduce un comentario, cuando cambia de tipo de lista.

8.- Enlaces con otras páginas. <A href=”xxx”> yy</A>

La característica que más ha influido en el espectacular éxito del Web (o *tela de araña*) ha sido, aparte la de su carácter multimedia, la posibilidad de unir los distintos documentos repartidos por todo el mundo por medio de **enlaces hipertexto**.

Estructura de los enlaces.

En general, los enlaces tienen la siguiente estructura:

<A HREF="xxx"> yyy </A>

donde xxx es el **destino** del enlace (Obsérvese las comillas). yyy es el **texto** indicativo en la pantalla del enlace (con un color especial y generalmente subrayado)

Tipos de enlaces

Vamos a distinguir cuatro tipos de enlaces:

1. Enlaces dentro de la misma página
2. Enlaces con otra página nuestra
3. Enlaces con una página fuera de nuestro sistema
4. Enlaces con una dirección de email

1. Enlaces dentro de la misma página

A veces, en el caso de documentos (o páginas) muy extensos, nos puede interesar dar un salto desde una posición a otra determinada.

En este caso, lo que antes hemos llamado **XXX**, es decir, el **destino** del enlace, en este caso el sitio dentro de la página a donde queremos saltar, se sustituye por **#marca** (la palabra **marca** puede ser cualquier palabra que queramos). Lo que hemos llamado antes **YYY** es la palabra (o palabras) que aparecerán en la pantalla en color (en forma de hipertexto). Su estructura es, entonces:

<A HREF="#marca"> YYY </A>

Y en el sitio exacto a donde queremos saltar, debemos poner la siguiente etiqueta:

<A name="marca"> </A>

Por ejemplo, si quiero saltar desde aquí a la pantalla final, pongo la siguiente etiqueta:

<A HREF="#siguiente"> Pulsa para ir al párrafo siguiente</A>

Que resulta como: Pulsa para ir al párrafo siguiente(*Puedes comprobar cómo salta a la pantalla final*)

Y en el párrafo siguiente del documento he puesto esta otra etiqueta:

<A NAME="siguiente"> </A>

2. Enlaces con otra página nuestra

Puede ser que tengamos una sola página. Pero lo más frecuente es que tengamos varias páginas, una inicial (o principal) y otras conectadas a ella, e incluso entre ellas mismas.

Supongamos que queremos enlazar con la página creada en el ejemplo del capítulo anterior, que la hemos llamado **pagina2.html**. En este caso, simplemente sustituimos lo que hemos llamado **XXX** (el destino del enlace) por el nombre del fichero:

<A HREF="pagina2.html"> Ejemplo del capítulo 8</A>

Que resulta como: Ejemplo del capítulo 8 (*Puedes comprobar cómo carga esa página*)

Si queremos que vaya a un **sitio concreto** de otra página nuestra en vez de ir al principio de la página, adonde va por defecto, en ese sitio tenemos que colocar una **marca** (véase la sección anterior), y completar el enlace con la referencia a esa marca.

Lo veremos con el siguiente **ejemplo**: Quiero poner un enlace desde aquí al capítulo 7, pero a un sitio concreto, donde he puesto la marca

<A NAME="alinear"></A>.

Entonces la etiqueta tiene que ser:

<A HREF="pagina2.html#alinear"> Capítulo 8 </A>

Que resulta:

Capítulo 8

(Obsérvese cómo al pulsar el enlace salta al capítulo 8, pero no al comienzo, sino adonde yo quiero que lo haga)

Una observación importante: Estoy suponiendo que la página en la que estoy escribiendo esta etiqueta y la otra página a la que quiero saltar **están en el mismo directorio**. Porque pudiera ocurrir que he organizado mi sitio del Web con un directorio principal, y otros subdirectorios auxiliares. Si la página a la que quiero saltar está, p. ej. en el subdirectorio **subdir**, entonces en la etiqueta tendría que haber puesto **"subdir/pagina2.html"**.

Y a la inversa, si quiero saltar desde una página a otra que está en un directorio **anterior**, en la etiqueta tendría que haber puesto **"../pagina2.html"**. Esos dos puntos hace que se dirija al directorio anterior. Obsérvese que se debe utilizar el símbolo **/** para indicar los subdirectorios, y no este otro ****, que es propio únicamente de Windows.

Si nos queremos evitar todas estas complicaciones, podemos tener todo junto en un único directorio, pero esto tiene el inconveniente de que esté todo más desordenado, y sean más difíciles de hacer las futuras modificaciones.

3. Enlaces con una página fuera de nuestro sistema

Si queremos enlazar con una página que esté fuera de nuestro sistema (es decir, que esté en un servidor distinto al que soporta nuestra página), es necesario conocer su dirección completa, o URL (*Uniform Resource Locator*). El URL podría ser, además de la dirección de una página del Web, una dirección de ftp, gopher, etc.

Una vez conocida la dirección (o URL), lo colocamos en vez de lo que hemos llamado anteriormente **xxx** (el destino del enlace). Si queremos enlazar con la página de PortalESO (cuyo URL es: **http://www.portaleso.com/**), la etiqueta sería:

<A HREF="http://www.portaleso.com/"> Página inicial de PortalESO </A>

Que daría como resultado:

Página inicial de PortalESO

Es muy importante copiar estas direcciones correctamente (respetando las mayúsculas y minúsculas, pues los servidores UNIX sí las distinguen)

4. Enlaces con una dirección de email

En este caso, sustituimos lo que se ha llamado antes **xxx** (el destino del enlace) por **mailto:** seguido de la dirección de email. La estructura de la etiqueta es:

<A HREF="mailto: *dirección de email*"> Texto del enlace </A>

Un ejemplo de esto está al final de la página principal de este manual. Podría haber puesto:

Comentarios a <A HREF="mailto: mailto:portaleso@ono.com **Información de PortalESO</A>**

Que resultaría:

Comentarios a Información de PortalESO

Pero hay algunos navegadores que no reconocen este tipo de enlace, y lo verían de esta manera:

Comentarios a Información de PortalESO

y no tendrían ningún medio de conocer la dirección de email. Por lo tanto, es más seguro poner algo así como:

Comentarios a Información de PortalESO en <A HREF="mailto: portaleso@ono.com"> portaleso@ono.com </A>

Que resulta:

Comentarios a Información de PortalESO en portaleso@ono.com

Es decir, es conveniente, por la razón dicha anteriormente, poner **también** en el texto del enlace la dirección de email.

Ejemplo práctico

En el procesador de textos copiamos:

```
<HTML>
<HEAD>
 <TITLE> Curso para realizar la web de centro - 3</TITLE>
</HEAD>
```

```
<BODY>
 <CENTER>
  <H1> Mis p&aacute;ginas favoritas </H1>
 </CENTER>
 <HR>
  Estas son mis p&aacute;ginas favoritas:
  <P><A HREF="http://www.portaleso.com">PortalESO  (Portal Educativo)</A>
  <BR> <A HREF="http:/www.microsoft.com"> Microsoft </A>
  <BR> <A HREF="http://www.google.com"> Google</A>
</BODY>
</HTML>
```

Guardamos el fichero de texto con el nombre **pagina3.html** *y lo cargamos en el navegador. Este es el resultado.*

Actividades:

1.- Crea una página donde aparezca un enlace a cada una de las páginas de los departamentos del instituto. El nombre de la página será **nombre_departamento.html** y se encontrará en un directorio de nombre **departamento**.

9.- Inserción de Imágenes. <IMG SRC="portalesocom.gif" ALT="PortalESO ">

La etiqueta que nos sirve para incluir imágenes en nuestras páginas del Web es muy similar a la de enlaces a otras páginas, que hemos visto en el capítulo anterior. La única diferencia es que, en lugar de indicar al programa navegador el nombre y la localización de un documento de texto HTML para que lo cargue, se le indica el nombre y la localización de un fichero que contiene una imagen.

La estructura de la etiqueta es:

<IMG SRC="portalesocom.gif">

Con el comando **IMG SRC** (*image source*, fuente de la imagen) se indica que se quiere cargar una imagen llamada **portalesocom.gif** (o el nombre que tenga).

Dentro de la etiqueta se pueden añadir otros comandos, tal como **ALT**

<IMG SRC="portalesocom.gif" ALT="PortalESO">

Con el comando **ALT** se introduce una descripción (una palabra o una frase breve) indicativa de la imagen. Este comando, que en principio se puede omitir, es en beneficio de los que accedan a nuestra página con un programa navegador en forma de texto sólo. Ya que no son capaces de ver la imagen, por lo menos pueden hacerse una idea sobre ella. Pero no es sólo por esto. Hay casos, como veremos más adelante, en los que se utiliza una imagen como enlace a otra página. Si se omitiera este comando, los que utilizan dichos navegadores no podrían de ninguna manera acceder a esas páginas.

Con respecto a la **localización** del fichero de esa imagen, se puede decir aquí lo mismo que en el capítulo anterior referente a los enlaces. Si no se indica nada especial, como en el caso que se ha expuesto, quiere decir que el fichero **portalesocom.gif** está

en el mismo directorio que el documento HTML que estamos escribiendo. Si no es así, se siguen los mismos criterios que los indicados para los enlaces.

Al igual que una página con la que queremos enlazar puede estar **fuera de nuestro sistema**, (en cuyo caso había que indicar su URL o dirección completa), podemos cargar una imagen que no esté en nuestro sistema siguiendo el mismo método, es decir, indicar en la etiqueta el URL completo de la imagen. Aunque esto no es muy aconsejable, pues alargaría innecesariamente el tiempo de carga de nuestra página.

Las imágenes deben estar guardadas en un **formato de fichero** especial llamado **GIF**. (*Hay también otro formato más avanzado, del que se hablará más adelante*). Este formato GIF almacena las imágenes con un máximo de 256 colores, en forma comprimida. Hay programas gráficos (como el Paint Shop Pro y el Photo Editor para el PC, o el Graphic Converter para el Mac) que nos permiten guardar las imágenes en este formato, además de cumplir otras muchas tareas de manipulación de las mismas.

Un aspecto muy importante a tener en cuenta es el **tamaño** de las imágenes, pues una imagen grande supone un fichero grande, y esto puede resultar en un tiempo excesivo de carga, con el consiguiente riesgo de que quien esté intentando cargar nuestra página se canse de esperar, y desista de ello.

Para elegir la **posición de la imagen con respecto al texto** hay distintas posibilidades. La más sencilla es colocarla entre dos párrafos, con un titular a un lado. Los navegadores más actuales (como el Netscape Navigator y el Microsoft Internet Explorer) permiten que el texto pueda rodear a la imagen (como se verá más adelante, en el capítulo 12).

De momento nos vamos a limitar a escoger la **posición del titular** con respecto a la imagen (si es que queremos ponerle un titular, claro está). Se puede poner arriba, en medio o abajo del lado de la imagen. Para ello se añade el comando **ALIGN** a la etiqueta, de la siguiente manera:

<IMG SRC="isla.gif" ALIGN=”TOP”> Titular alineado arriba



Titular alineado arriba

<IMG SRC="isla.gif" ALIGN=”MIDDLE”> Titular alineado en medio



Titular alineado en medio

<IMG SRC="isla.gif" ALIGN=”BOTTOM”> Titular alineado abajo

Titular alineado abajo

Véanse las extensiones de Netscape de esta etiqueta en el Capítulo 12

Otra posibilidad muy interesante es la de utilizar una **imagen como enlace a otra página**. Para estos casos se utilizan generalmente imágenes pequeñas (iconos), aunque se puede usar cualquier tipo de imagen.

Según vimos en el capítulo anterior, la estructura general de un enlace es:

<A HREF="xxx"> yyy </A>

donde **xxx** era el destino del enlace e **yyy** el texto del enlace (o más generalmente hablando, lo que aparece en la pantalla como el enlace; en el capítulo anterior era un texto, y en éste va a ser una imagen). En este caso sustituimos **xxx** por el nombre del fichero de la página a la que queremos acceder. Y en lugar de **yyy** ponemos la **etiqueta completa de la imagen** (que queda así englobada dentro de la etiqueta del enlace)

Como ejemplo vamos a utilizar la imagen 

(portalesocom.gif) para acceder al ejemplo práctico (pagina2.html):

<A HREF="pagina2.html"><IMG SRC="portalesocom.gif"></A>

que da como resultado: 

Pulsando la imagen comprobamos cómo efectivamente enlaza con la página deseada. Obsérvese además que la imagen está rodeada de un **rectángulo** del color normal en los enlaces. Si no se desea que aparezca ese rectángulo, hay que incluir dentro de la etiqueta de la imagen el atributo **BORDER=0**, es decir:

<A HREF="pagina2.html"><IMG SRC="portalesocom.gif" BORDER=”0”></A>

que da como resultado: 

Posicionando el cursor sobre esta última imagen, comprobamos que actúa también como enlace aunque carezca del rectángulo de color. Esto puede resultar más estético, pero se corre el riesgo de que el usuario no se dé cuenta de que la imagen sirve de enlace.

También podemos utilizar una **imagen para enlazar con otra imagen**. En este caso sustituimos **xxx** (el destino del enlace) con el nombre del fichero de la imagen a la que queremos acceder e **yyy** (lo que aparece en pantalla como el enlace) por la etiqueta completa de la imagen que queremos que aparezca en la pantalla como el enlace de la otra.

Supongamos que queremos enlazar con la imagen **isla.gif** por medio de esta otra imagen:  (**inicio.gif**):

<A HREF="isla.gif"><IMG SRC="inicio.gif"></A>

Que da como resultado: 

Por último, otra posibilidad es la de utilizar **un texto para enlazar con una imagen**. En este caso sustituimos **xxx** (el destino del enlace) con el nombre del fichero de la imagen a la que queremos acceder e **yyy** (lo que aparece en pantalla como el enlace) por el texto.

Supongamos que queremos enlazar con la imagen **isla.gif** por medio del texto "un paraíso tropical":

<A HREF="isla.gif"> un paraíso tropical </A>

Que da como resultado: un paraíso tropical

Se pueden **capturar las imágenes** que aparecen en pantalla, con objeto de guardarlas permanentemente en nuestro disco duro. En los navegadores más actuales se hace pulsando sobre la imagen con la tecla derecha del ratón, con lo que se obtiene un menú en el que está la posibilidad de guardar la imagen.

Un tipo de imágenes del que se hace abundante uso y que sirven para mejorar la presentación de la página son los **iconos, botones, barras separadoras**, etc. A pesar de su tamaño o forma, son imágenes como cualquier otra.

Por ejemplo, vamos a capturar el **botón** dorado que está acontinuación. Colocamos el puntero del ratón justo en el botón, y pulsamos la tecla derecha. Si nuestro navegador implementa esta característica, aparece un menú en el que está la posibilidad de guardar esta imagen (bgold.gif). Lo hacemos, en el mismo directorio en el que va a estar nuestro documento. Para utilizarlo en el comienzo de una línea, escribiríamos en el sitio correspondiente:

<IMG SRC="bgold.gif> Esta l&iacue;nea esta precedida por un botón dorado.

Que resulta:

Esta línea esta precedida por un botón dorado.

Podemos hacer lo mismo con la **barra separadora** que está en todas las páginas de este manual (bargold.gif). La capturamos, la guardamos y escribimos:

<CENTER><IMG SRC="bargold.gif>
**
Esta línea esta debajo de una barra separadora dorada.**
</CENTER>

Que resulta:



Esta línea esta debajo de una barra separadora dorada.

Para ampliar el tema de este capítulo, ver:

Capítulo 12: Alineación y dimensionado de imágenes

Actividades:

1.- Busca gráficos, que identifiquen varias páginas web y crea una página web que enlace hasta estas páginas. Por lo menos deben haber 5 páginas.

2.- Busca un gráfico que pueda servir para identificar tu asignatura. Crea un icono de 90x20 pixels que se pueda utilizar como icono identificativo de tu asignatura.
Junto con los iconos del resto de compañeros, crea una página donde aparezca un enlace a cada una de las páginas de los departamentos del instituto, donde el enlace se realice a través de los gráficos. El nombre de la página será **nombre_departamento.html** y se encontrará en un directorio de nombre **departamento**.

10.- Creación de una página de estilos CSS.

(Style, div, span style, style type, link)

Las hojas de estilo representan un gran avance con respecto a cómo los diseñadores de páginas Web pueden aumentar grandemente su capacidad de mejorar la **apariencia** de sus páginas.

En su origen el Web fue concebido en un entorno científico, como un medio de intercambio de documentación, en el que lo esencial era el **contenido**, siendo la presentación algo completamente accesorio. Pero a medida que el Web era descubierto por toda clase personas como un fantástico medio de difusión, las limitaciones del HTML con respecto al control sobre la presentación se convirtieron en una fuente continua de frustraciones.

Los nuevos diseñadores de páginas Web estaban acostumbrados a los medios tradicionales sobre papel, en los que tenían un control total sobre la apariencia. Han ido aprendiendo a utilizar diversos trucos para suplir las limitaciones estilísticas del HTML.

Pero aunque su intención ha sido buena - mejorar la presentación de las páginas - las técnicas empleadas han tenido unos efectos secundarios a menudo desafortunados. Estas técnicas o trucos suelen funcionar para algunas personas, algunas veces, pero nunca para todas las personas y en todos los casos.

Varios ejemplos de estas técnicas son:

- Utilización de etiquetas no estándar, inventadas por los creadores de algunos navegadores (como Netscape o Explorer).
- Convertir el texto en imagen.
- Usar imágenes transparentes para crear espacios en blanco.
- Uso de tablas para forzar determinadas presentaciones.
- Utilizar programas o lenguajes ajenos al HTML para conseguir determinados fines, etc.

Estas técnicas aumentan considerablemente la complejidad de las páginas del Web, tienen una flexibilidad limitada, y sobre todo, provocan problemas al ser utilizados por

distintos navegadores o plataformas.

Las hojas de estilo devuelven la facilidad de controlar la presentación y sobrepasan la limitada gama de mecanismos de presentación que se han ido añadiendo al HTML durante estos últimos años. Por ejemplo, con ellas es fácil determinar el espacio entre las líneas de texto, el tipo y tamaño de las fuentes, el grosor de los márgenes, los colores usados para el texto o el fondo, así como el fomateo a voluntad de cualquier etiqueta.

Además hay otro aspecto muy interesante de las hojas de estilo, y es que **separan** por completo la información para la **presentación** de una página de su **contenido**, con lo que se facilita mucho el diseño y revisión de las páginas, pues se puede variar la presentación de una página, o de todo el conjunto de ellas, sin cambiar una sola línea del código del HTML.

Como veremos más adelante, se pueden definir variaciones de diseño por medio de las "clases". Por ejemplo, si se necesitan cinco estilos de párrafos diferentes, se pueden definir cinco clases para la etiqueta **<P>** : P.normal, P.indentado, P.subrayado, P.rojo y P.dobleespacio.

En este capítulo se hace una **introducción** a las hojas de estilo, pues se trata de un tema muy extenso. También se verán con detalle los distintos atributos y valores así como distintos consejos para su mejor utilización.

¿Cómo funcionan?

Las hojas de estilo en cascada nos permiten **redefinir** las reglas que utiliza el navegador para presentar una página del Web.

Todos los navegadores tienen sus **reglas**. Así, por ejemplo, como vimos en el capítulo 7, el HTML define que las cabeceras del nivel uno (el texto que está entre las etiquetas **<H1>** y **</H1>**) se muestre como un texto alineado a la izquierda, un par de puntos mayor que el resto, en negrita y en la fuente Times Roman.

Ejemplo: **<H1> Cabecera de nivel 1 </H1>** se verá como:

Cabecera de nivel 1

Cada etiqueta define sus propias reglas para mostrar el texto que engloba. Con las hojas de estilo podemos **cambiar** cada una de estas reglas si lo deseamos. Es decir, podemos modificar a nuestro antojo el comportamiento de cada etiqueta.

Hay **tres** maneras de añadir estilos a nuestras páginas Web.

1. Añadiendo instrucciones de estilo sólo a **etiquetas** concretas, o a un grupo de ellas.
2. **Incluyendo** las instrucciones de estilo en el documento HTML de una página concreta - de una manera análoga a cómo se incluye un *script* de Javascript. Esto permite cambiar la apariencia de una hoja entera, cambiando unas pocas líneas.
3. **Enlazando** todos documentos HTML de todas las páginas que componen un

sitio del Web con un **fichero** de definición del estilo. De esta manera, se puede cambiar la apariencia de múltiples páginas retocando un solo fichero.

Podemos utilizar en nuestras páginas uno, dos o los tres métodos descritos, como se verá más adelante.

1. Estilo para etiquetas concretas o grupos de etiquetas

Este método es el apropiado si sólo se desea cambiar el estilo en unas secciones determinadas de una página. Con él podemos manipular las propiedades de alguna etiqueta o de algún grupo de etiquetas.

Se hace añadiendo el atributo **STYLE**, que contiene una serie de propiedades, dentro de la etiqueta correspondiente. Veamos el siguiente ejemplo:

```
<P>
"Este es un p&aacute;rrafo normal, sin m&aacute;rgenes a la izquierda ni a la derecha, con el color de su texto en negro". </P>

<P STYLE="margin-left: 0.5in; margin-right: 0.5in; color:green">
"Quiero que este otro p&aacute;rrafo tenga un margen izquierdo de media pulgada, al igual que el margen derecho. Adem&aacute;s quiero que su texto sea de color verde". </P>
```

Que se verá (en los navegadores que soportan hojas de estilo) de esta manera:

"Este es un párrafo normal, sin márgenes a la izquierda ni a la derecha, con el color de su texto en negro".

"Quiero que este otro párrafo tenga un margen izquierdo de media pulgada, al igual que el margen derecho. Además quiero que su texto sea de color verde".

Como se ve en este ejemplo, hemos variado la manera de cómo se comporta habitualmente la etiqueta **<P>**. Podíamos haber conseguido otras muchas más cosas, como se verá más adelante.

Obsérvese la **sintaxis** del atributo **STYLE**. Se ponen entre comillas una serie de propiedades (por ejemplo, **margin-left**), y después de dos puntos se pone el valor de esa propiedad (en este caso, **0.5in** o sea, 0.5 pulgadas). Las distintas propiedades deberán estar separadas por punto y coma.

Si queremos cambiar la apariencia de una **sección entera** (que agrupe un conjunto de etiquetas), se puede utilizar la etiqueta **<DIV>**, con la que definimos el estilo globalmente para esa sección.

En el siguiente ejemplo cambiamos el color (a rojo) y el tamaño de la fuente (a 16 puntos de altura) de un bloque de etiquetas por medio de la etiqueta **<DIV>**, lo que tiene el mismo efecto que si hubiéramos asignado este estilo separadamente a cada una de las etiquetas englobadas (**<P>, <UL>** y **<LI>**):

```
<DIV STYLE="font-size: 16pt; color: red">
```

```
<P>La definici&oacute;n del estilo (rojo y 16 puntos), afecta a todo el bloque de etiquetas. </P>
<P>Este texto tambi&eacute;n es rojo y tiene 16 puntos de altura. </P>
<P>Al igual que esta l&iacute;nea. </P>
</DIV>
```

Que se ve de la siguiente manera:

La definición del estilo (rojo y 16 puntos), afecta a todo el bloque de etiquetas.
Este texto también es rojo y tiene 16 puntos de altura.
Al igual que esta línea.

¿Qué ocurre si a una de las etiquetas englobadas por **DIV** le introducimos un estilo distinto? Comprobémoslo introduciendo un estilo propio a la segunda etiqueta **<P>**:

```
<DIV STYLE="font-size: 16pt; color: red">
<P> Veamos si la definici&oacute;n del estilo (rojo y 16 puntos), afecta a todo el bloque de etiquetas, hasta la etiqueta de cierre de DIV. </P>
<P STYLE="color: blue">Esta l&iacute;nea es azul, a pesar de estar englobada dentro de DIV, porque tiene su propio estilo. </P>
<P>Esta otra cumple con el estilo de DIV, porque no tiene estilo propio. </P>
</DIV>
```

Este es el resultado:

Veamos si la definición del estilo (rojo y 16 puntos), afecta a todo el bloque de etiquetas.
Esta línea es azul, a pesar de estar englobada dentro de DIV, porque tiene su propio estilo.
Esta otra cumple con el estilo de DIV, porque no tiene estilo propio.

Como se comprueba en este caso, el estilo dentro de una etiqueta concreta tiene **precedencia** sobre el estilo del bloque, impuesto por **DIV**.

Aquí comenzamos a ver la noción de precedencia de unos métodos sobre otros, que como hemos visto antes, son muy variados. Este es el motivo de que se llamen Hojas de Estilo en **Cascada** (hay una cascada de métodos para aplicar los estilos, teniendo un orden de precedencia riguroso de unos sobre otros). Se verá esto más adelante.

Hay otra etiqueta, **<SPAN>**, que tiene una misión parecida a **<DIV>**, pero que se usa para cambiar elementos más pequeños (por ejemplo, palabras o incluso letras), en lugar de bloques enteros. Por ejemplo:

```
<P> En este p&aacute;rrafo solamente queremos atribuir un estilo a esta <SPAN STYLE="font-family: courier;font-size: 15pt;color:
```

fuchsia">palabra</SPAN>, que queremos que su fuente sea courier, de 15 puntos de altura y de color fucsia. </P>

Resultado:

En este párrafo solamente queremos atribuir un estilo a esta palabra, que queremos que su fuente sea courier, de 15 puntos de altura y de color fucsia.

Este método de introducir el estilo en etiquetas concretas o en conjuntos de etiquetas es sencillo de aplicar (como hemos visto), y es el adecuado si sólo se desea hacer algún cambio puntual en una página. Sin embargo, no es un adelanto muy grande con respecto al HTML, donde existen etiquetas que hacen funciones similares.

Además, si se desean hacer cambios, hay que examinar todo el documento en busca de etiquetas dispersas. Si se desean hacer cambios **globales** a una o más páginas son más convenientes los otros dos métodos que veremos a continuación.

2. Inclusión global del estilo en un documento HTML

Se hace poniendo un bloque de instrucciones dentro de las etiquetas **<STYLE> </STYLE>**, que deberá estar colocado dentro de la cabecera del documento, después del título, entre las etiquetas **</TITLE>** y **</HEAD>** (de igual manera que se hace con los *scripts* de Javascript).

Esta etiqueta **<STYLE>** tiene un atributo, **TYPE**, que especifica el tipo de medio en que va a ser publicado en Internet, en nuestro caso será "text/css" (que permitirá a los navegadores que no soporten este tipo el ignorar la hoja de estilo). Es decir, la etiqueta queda de esta manera: **<STYLE TYPE="text/css">**

Por tanto, la estructura será la siguiente:

```
<HTML>
<HEAD>
<TITLE>[Título] </TITLE>
   <STYLE TYPE="text/css">
   [bloque de instruciones de estilo]
   </STYLE>
</HEAD>
<BODY>
[conjunto de todas las etiquetas que componen la página]
</BODY>
</HTML>
```

Veamos ahora con un ejemplo, cómo se escribe el bloque de instrucciones del estilo, incluido dentro de la etiqueta **<STYLE>** (más tarde se verán con detalle cada uno de los atributos):

```
<STYLE TYPE="text/css">
BODY {background: yellow ;font-size: 10pt; font-family: Arial; margin-left: 0.5in;
margin-right: 0.5in}
H1 {background: blue; font-size: 14pt;font-weight: bold; color: red}
```

```
H2 {font-size: 12pt;font-weight: bold; color: red}
DIV {background: URL(nubes.jpg)}
</STYLE>
```

Como se ve en este ejemplo, se ponen las distintas etiquetas a las que se quiere atribuir un estilo (en este caso **BODY, H1, H2** y **DIV**, pero podrían haber sido otras cualesquiera); a continuación del nombre de cada etiqueta, y englobadas por los signos "{" y "}" van las distintas propiedades con sus respectivos valores, separadas unas de otras por un punto y coma.

Si se coloca este bloque en la cabecera del documento HTML, donde se ha indicado antes, veremos que cada vez que se usa una de las etiquetas que tienen una definición de estilo, no se comportará de la forma habitual, sino tal como la hemos definido. El resto de las etiquetas, que no están incluidas en el bloque de definición del estilo, se comportarán de la forma habitual. Puede comprobarse lo dicho, viendo una página a la que se le ha aplicado el estilo del ejemplo anterior.

3. Enlazando distintos documentos a una hoja de estilo

Si nuestro sitio del Web está compuesto por muchas páginas, y queremos darles un estilo uniforme a unas cuantas, en vez de incluir un bloque de definición de estilo repetido en cada una de ellas (como se ha visto en la sección anterior), se puede establecer un **enlace a un fichero de texto** que contiene el bloque de definición del estilo.

Supongamos que queremos aplicar a unas cuantas páginas el estilo visto en el ejemplo anterior. Tendríamos que crear un **fichero de texto** como este:

```
BODY {background: yellow ;font-size: 10pt; font-family: Arial; margin-left: 0.5in;
margin-right: 0.5in}
H1 {background: blue; font-size: 14pt;font-weight: bold; color: red}
H2 {font-size: 12pt;font-weight: bold; color: red}
DIV {background: URL(nubes.jpg)}
```

Obsérvese que no tiene **ninguna etiqueta**, pues no es un documento normal HTML, sino que es un fichero de texto que sólo contiene el bloque de definición del estilo.

Se guarda con el nombre que se quiera, pero tiene que tener necesariamente la extensión **.css** Supongamos que lo guardamos con el nombre de **miestilo.css**

En todas las páginas que queremos que tengan este estilo concreto, solamente deberemos de añadir (en el mismo sitio de la cabecera que para el caso anterior, es decir entre **</TITLE>** y **</HEAD>**, la siguiente etiqueta:

```
<LINK REL=stylesheet HREF="miestilo.css" TYPE="text/css">
```

Aquí se ha supuesto que el fichero **miestilo.css** se ha dejado en el mismo directorio donde están los documentos HTML. Si no es así, es decir, si está en otro subdirectorio en el servidor, habría que indicar la ruta (por, ejemplo: **HREF="subdirectorio/miestilo.css"**).

Mezclando los métodos

Como se ha visto anteriormente, hay tres métodos de aplicar las hojas de estilo en cascada.

¿Se pueden aplicar más de uno a la vez en una misma página?

Sí.

¿Qué ocurre si hay información contradictoria entre ellos?

Para evitar conflictos entre los distintos métodos usados simultáneamente, existe un orden de **precedencia**, es decir, cuál prevalecerá sobre el otro si dan órdenes contradictorias sobre un aspecto concreto.

El orden de precedencia es el siguiente (de mayor a menor):

1. **Estilo dentro de una etiqueta.**
2. **Bloque de estilo en la cabecera del documento.**
3. **Enlace a un fichero que contiene la hoja de estilo.**

Esto quiere decir que una orden de estilo, por ejemplo el color del texto, puesta dentro de una etiqueta, prevalecerá sobre la que esté indicada en los otros dos métodos, si es que se usan en esa página.

Lo más práctico es utilizar el tercer método (enlace a una hoja de estilo), para dar una apariencia consistente a todas las páginas, y si es necesario modificar un aspecto concreto en alguna de ellas, utilizar el primero o el segundo.

Atributos de las hojas de estilo.

A continuación se muestra en una tabla el resumen de los atributos que se pueden incluir en las hojas de estilo.

Atributo	Descripción	Valores	Ejemplo
font-size	Establece el tamaño de texto.	puntos (pt) pulgadas (in) centímetros (cm) pixels (px)	{font-size: 12pt}

font-family	Establece la fuente.	nombre de la fuente nombre de la familia de la fuente	{font-family: courier}
font-weight	Establece el espesor de la fuente.	extra-light light demi-light medium demi-bold bold extra-bold	{font-weight: bold}
font-style	Convierte el texto a cursiva.	normal italic	{font-style: italic}
line-height	Establece la distancia entre líneas.	puntos (pt) pulgadas (in) centímetros (cm) pixels (px) porcentaje (%)	{line-height: 24pt}
color	Establece el color del texto.	nombre del color valores RGB	{color: blue}
text-decoration	Subraya o remarca el texto.	none underline italic line-through	{text-decoration: underline}
margin-left	Establece el márgen izquierdo de la página.	puntos (pt) pulgadas (in) centímetros (cm) pixels (px)	{margin-left: 1in}
margin-right	Establece el márgen derecho de la página.	puntos (pt) pulgadas (in) centímetros (cm) pixels (px)	{margin-right: 1in}
margin-top	Establece el márgen superior de la página.	puntos (pt) pulgadas (in) centímetross (cm) pixels (px)*	{margin-top: -20px}
text-align	Establece la justificación del texto.	left center right	{text-align: right}
text-indent	Establece la indentación del texto.	puntos (pt) pulgadas (in) centímetros (cm) pixels (px)	{text-indent: 0.5in}
background	Establece la imagen o el color del fondo.	URL, nombre del color valor RGB	{background: #33CC00}

Explicación de los atributos

---

font-size

El atributo **font-size** establece el **tamaño del texto** en puntos (pt), pulgadas (in), centímetros (cm), o pixels (px). Ejemplos:

{font-size: 12pt}
{font-size: 1in}
{font-size: 5cm}
{font-size: 24px}

---

font-family

el atributo **font-family** establece la **fuente del texto**. Se puede especificar una única fuente, como por ejemplo:

{font-family: Arial}

u otras fuentes alternativas, separadas por una coma, como por ejemplo:

{font-family: Arial, Helvetica}

En el ejemplo anterior, nos aseguramos que los sistemas que no soporten la fuente Arial, uticen la fuente Helvetica. Es muy aconsejable especificar, como útimo recurso, un nombre genérico de familia de fuentes. Ejemplo:

{font-family: Arial, Helvetica, sans-serif}

Estos nombres genéricos de familia de fuentes (serif, sans-serif, cursive, fantasy, o monospace) tienen la ventaja de que son representados como las fuentes que tenga instaladas el usuario.

Si se hace referencia a una fuente cuyo nombre consiste en varias palabras (separadas por espacios en blanco), hay que englobarla entre comillas. Ejemplo:

{font-family: "Courier New"}

---

font-weight

El atributo **font-weight** establece el **espesor de la fuente**:

{font-weight: medium}
{font-weight: bold}

Los valores aceptados (extra-light, light, demi-light, medium, demi-bold, bold, y extra-bold) dependen en las fuentes que tenga instaladas el usuario. (Por ejemplo, el sistema del usuario puede que sólo permita medium y bold para una determinada fuente).

---

font-style

El atributo **font-style** establece la **fuente como cursiva**:

{font-style: italic}

En el borrador de la W3C se contemplan además otros posibles estilos (minúsculas, oblicuas, etc.)

---

line-height

Este atributo establece la **separación entre líneas**, que se puede expresar en puntos (pt), pulgadas (in), centÌmetros (cm), pixels (px) o porcentaje (%). Ejemplo:

{line-height: 20pt}

También se puede expresar como un porcentaje del valor por defecto:

{line-height: 150%}

En el Explorer 3.0, el espaciado se añade antes de las líneas, no después de ellas. Además este atributo se comporta de forma impredecible con texto que usa diferentes tamaños de texto en la misma línea.

---

color

Este atributo establece el **color del texto** de acuerdo con su valor hexadecimal (véase el Cap. 7), o usando los nombres de colores:

{color: #33CC00}
{color: red}

Los **nombres de los colores** son los siguientes:

black	silver	gray	white
maroon	red	purple	fuchsia
green	lime	olive	yellow
navy	blue	teal	aqua

---

text-decoration

Este atributo permite **remarcar el texto**. Los valores soportados son **underline** (subrayado), **line-through** (tachado), **none** (ninguno) e **italic** (cursiva). Ejemplos:

{text-decoration: underline}
{text-decoration: line-through}

---

margin-left, margin-right, y margin-top

Estos atributos establecen los **márgenes** (izquierdo, derecho y superior respectivamente) en el ámbito de una etiqueta. Se pueden especificar los márgenes en puntos, pulgadas (inches), centímetros o pixels. Por ejemplo:

```
BODY {margin-left: 0.5in;
      margin-right: 0.5in;
      margin-top: 1in}
```

Se pueden usar valores negativos. Entonces, en vez de contraerse el magen, se extenderá en la dirección opuesta.

---

text-align

Este atributo permite **justificar** los elementos HTML a la izquierda, al centro o a la derecha. Ejemplos:

{text-align: left}
{text-align: center}
{text-align: right}

---

text-indent

Además de establecer los márgenes, se puede provocar una **indentación** del texto (es decir, que la primera palabra del párrafo se desplace una cantidad deseada). Se puede expresar en puntos, pulgadas, centímetros o pixels. Por ejemplo:

P {text-indent: 0.5cm}

hace que los párrafos que se crean con la etiqueta **<P>**, empiecen con su primera línea indentada 0.5 centímetros.

Se pueden usar valores negativos, que sacan el texto hacia los márgenes.

---

background

Se utiliza este atributo para destacar secciones de una página, estableciendo un **color de fondo o una imagen de fondo**.

Para establecer un **color** de fondo, se especifica su valor hexadecimal, o un nombre de color (ver el atributo **color** visto anteriormente). Ejemplos:

{background: red}
{background: #6633FF}

También se puede colocar una **imagen de fondo** en el ámbito de la etiqueta. Es decir, se puede poner, por ejemplo, una imagen de fondo en un párrafo determinado.

Para colocar una imagen, se especifica el URL entre paréntesis (no entre comillas, como es lo habitual). Por ejemplo:

{background: URL(http://nubes.jpg)}

En este caso se ha puesto el URL absoluto, es decir, la referencia completa en el servidor. Pero es más conveniente hacerlo de forma relativa, es decir con respecto al documento HTML. Si el fichero de imagen y el documento HTML están en el mismo directorio, no hay que poner ningún URL, sino el nombre del fichero de imagen directamente. En el ejemplo anterior sería:

{background: URL(nubes.jpg)}

Vamos a ver una aplicación práctica: poner esta imagen de fondo a un párrafo, utilizando el método de incluir el estilo en la etiqueta **<P>**:

<P STYLE="background: URL(nubes.jpg)"> Este párrafo tiene un fondo con imágenes, al igual que con el HTML convencional se consigue para toda una página, pero en este caso está limitado al ámbito de esta etiqueta. </P>

<P> En cambio, este otro párrafo no tiene imagen de fondo, porque no se le ha incluido este atributo. </P>

Este es el resultado:

Este párrafo tiene un fondo con imágenes, al igual que con el HTML convencional se consigue para toda una página, pero en este caso esta limitado al ámbito de esta etiqueta.

En cambio, este otro párrafo no tiene imagen de fondo, porque no se le ha incluido este atributo.

Actividades:

1.- Copia cinco párrafos del texto de este curso en una página web. Modifica el tipo de letra de cada uno de los párrafos, el color y el tamaño de la letra, haz que por lo menos uno de ellos tenga un gráfico de fondo.

2.- Repite la página anterior, pero ahora debes hacerlo por medio de una hoja de estilos, donde se encuentren todos los estilos que has utilizado.

3.- Con ayuda de la hoja de estilos cambia las propiedades de los párrafos eligiendo el estilo que desees.

4.- A continuación tienes una hoja de estilos que puedes implementar. Prueba a escribir un párrafo, y observa como se modifica según el estilo elegido.
Puedes copiar la hoja de estilo y nombrarla como iesxirivella.css.

```
H1 {CLEAR: both; FONT-SIZE: 140%; MARGIN-BOTTOM: 0.5em }
H2 {CLEAR: both; MARGIN-TOP: 1.5em; FONT-SIZE: 130%; MARGIN-BOTTOM: 0.5em }
H3 {CLEAR: both; MARGIN-TOP: 1.2em; FONT-SIZE: 120%; MARGIN-BOTTOM: 0.5em }
H4 {CLEAR: both; MARGIN-TOP: 1.2em; FONT-SIZE: 110%; MARGIN-BOTTOM: 0.5em; MARGIN-LEFT:
1em }
H5 {CLEAR: both; MARGIN-TOP: 1.2em; FONT-SIZE: 100%; MARGIN-BOTTOM: 0.5em; MARGIN-LEFT:
2em }
H6 {CLEAR: both; MARGIN-TOP: 0.2em; FONT-SIZE: 70%; MARGIN-BOTTOM: 0.5em }
P {MARGIN-TOP: 1.5em; MARGIN-BOTTOM: 0.6em }
P.indent {MARGIN-LEFT: 3em}
P.note {CLEAR: both; MARGIN-TOP: 1em; FONT-WEIGHT: bold; MARGIN-BOTTOM: -0.25em }
P.tip {CLEAR: both; MARGIN-TOP: 1em; FONT-WEIGHT: bold; MARGIN-BOTTOM: -0.25em }
P.important {CLEAR: both; MARGIN-TOP: 1em; FONT-WEIGHT: bold; MARGIN-BOTTOM: -0.25em }
P.caution {CLEAR: both; MARGIN-TOP: 1em; FONT-WEIGHT: bold; MARGIN-BOTTOM: -0.25em }
P.warning {CLEAR: both; MARGIN-TOP: 1em; FONT-WEIGHT: bold; MARGIN-BOTTOM: -0.25em }
P.proclabel {CLEAR: both; FONT-WEIGHT: bold; FONT-SIZE: 100% }
OL {MARGIN-TOP: 0.5em; PADDING-LEFT: 0em; MARGIN-BOTTOM: 0px; MARGIN-LEFT: 2.1em }
UL {MARGIN-TOP: 0.6em; PADDING-LEFT: 0em; MARGIN-BOTTOM: 0px; MARGIN-LEFT: 1.5em; LIST-
STYLE-TYPE: disc }
LI {CLEAR: both; MARGIN-BOTTOM: 0.7em }
DD { MARGIN-BOTTOM: 0px; MARGIN-LEFT: 1.5em }
BODY  {color: #000000; font-size: 11px; font-family: Verdana, Arial, Helvetica, "Sans Serif"; background-
color: #ffffff }
TABLE {FONT-SIZE: 100%; BACKGROUND: #ffffff; COLOR: #000000; FONT-FAMILY: Verdana, Arial,
Helvetica, Sans Serif }
A:link {COLOR: #0066cc; TEXT-DECORATION: none}
A:visited {COLOR: #0066cc; TEXT-DECORATION: none }
A:hover {COLOR: #ff0000; TEXT-DECORATION: none }
A:active {COLOR: #ff0000; TEXT-DECORATION: none }
HR {HEIGHT: 1px}
DIV P {MARGIN-LEFT: 2em }
a.finePrint {font-size: 85%;}
```

11.- Inserción de fondos.

(Bgcolor, text, link, alink, background)

Se puede cambiar el fondo de dos maneras distintas:

1. Con un color uniforme
2. Con una imagen

11.1 Fondos con un color uniforme

Se consigue añadiendo el comando **BGCOLOR** a la etiqueta **<BODY>** (situada al principio del documento), de la siguiente manera:

<BODY BGCOLOR="#XXYYZZ">

donde:

XX es un número indicativo de la cantidad de color **rojo**
YY es un número indicativo de la cantidad de color **verde**
ZZ es un número indicativo de la cantidad de color **azul**

Estos números están en numeración **hexadecimal**. Esta numeración se caracteriza por tener 16 dígitos (en lugar de los diez de la numeración decimal habitual). Estos dígito son:

0 1 2 3 4 5 6 7 8 9 A B C D E F

Es decir, que en nuestro caso, el número menor es el 00 y el mayor el FF. Así, por ejemplo, el color rojo es el #FF0000, porque tiene el máximo de rojo y cero de los otro dos colores .

Los colores primarios son:

#FF0000 - Rojo
#00FF00 - Verde
#0000FF - Azul

Otros colores son:

#FFFFFF - Blanco
#000000 - Negro
#FFFF00 - Amarillo

Para hacer un color **más oscuro**, hay que reducir el número de su componente, dejando los otros dos invariables. Así, el rojo #FF0000 se puede hacer más oscuro con #AA0000, o aún más oscuro con #550000.

Para hacer que un color tenga un tono más suave (más **pastel**), se deben variar los otros dos colores haciéndolos más claros (número más alto), en una cantidad igual. Así, podemos convertir el rojo en rosa con #FF7070.

Podemos hacer nuestros propios experimentos, pero si desea, existen **páginas del Web** en las que se pueden elegir los colores directamente.

Colores del texto y de los enlaces

Si no se variasen los colores habituales del texto y de los enlaces (negro y azul, respectivamente), podría ocurrir que su lectura contra un fondo oscuro fuese muy dificultosa, o incluso imposible, si el fondo fuese precisamente negro o azul.

Para evitar esto, se pueden escoger los colores del texto y de los enlaces, añadiendo a la etiqueta (si se desea) los siguientes comandos:

TEXT - color del texto
LINK - color de los enlaces
VLINK - color de los enlaces visitados
ALINK - color de los enlaces activos (el que adquieren en el momento de ser pulsados)

Los códigos de los colores son los mismos que los que se han visto anteriormente.

La etiqueta, con todas sus posibilidades, sería:

< BODY BGCOLOR="#XXYYZZ" TEXT="#XXYYZZ" LINK="#XXYYZZ" LINK="#XXYYZZ" ALINK="#XXYYZZ">

Ejemplo

```
<HTML>
<HEAD>
<TITLE> Ejemplo de cambio de fondo</TITLE>
</HEAD>
<BODY BGCOLOR="#000000" TEXT="#0000FF" LINK="#FF0000" VLINK="#FFFF00"
ALINK="#FF7070">
<CENTER><H1>Ejemplo de cambio de fondo</H1>
<H5>&lt;BODY BGCOLOR="#000000" TEXT="#0000FF" LINK="#FF0000"
VLINK="#FFFF00" ALINK="#FF7070"&gt;</H5></CENTER>
<FONT SIZE=+1><P><STRONG>
El fondo est&aacute; en negro, el texto en azul y <A HREF="pagina3.html"> el enlace de
retorno en rojo</A> , el enlace visitado en amarillo y el enlace activado en rosa.
<P>Obs&eacute;rvese el curioso efecto de <FONT COLOR="#FF0000"> relieve </FONT>
 que se obtiene.</FONT>
</STRONG>
</BODY>
</HTML>
```

<BODY BGCOLOR="#000000" TEXT="#0000FF" LINK="#FF0000" VLINK="#FFFF00" ALINK="#FF7070">

El fondo está en negro, el texto en azul y el enlace de retorno en rojo , el enlace visitado en amarillo y el enlace activado en rosa.

Obsérvese el curioso efecto de relieve que se obtiene.

Cambio del color de una parte del texto

El comando **TEXT** explicado anteriormente (que va englobado dentro de la etiqueta **<BODY>**) cambia el color de la totalidad del texto de la página.

Tanto el Netscape Navigator 2, como el Microsoft Explorer soportan una etiqueta de color de la fuente con la que se puede cambiar sólo una parte del texto:

<FONT COLOR="#FF0000"> Este texto es de color rojo </FONT>

Que resulta:

Este texto es de color rojo

11.2. Fondos con una imagen

El fondo de una página puede ser también una **imagen**, ya sea en formato **GIF** o **JPEG**. Esta imagen se repite por toda la página, de una manera análoga al tapiz de Windows. La estructura de la etiqueta es:

<BODY BACKGROUND="imagen.gif">

o bien:

<BODY BACKGROUND="imagen.jpg">

No todos los navegadores soportan este formato

Se pueden añadir también a esta etiqueta todos los comandos para cambiar los colores del texto y de los enlaces, vistos anteriormente. Esto es imprescindible a veces para conseguir que el texto sea legible, en contraste con el fondo.

Dos ejemplos de fondos

brickz.gif



nubes.jpg

Hay que prever la posibilidad de que quien acceda a nuestra página haya **deshabilitado la carga automática de imágenes**, en cuyo caso tampoco cargaría la imagen que sirve como fondo y sólo vería el fondo estándar de color gris. Esto podría ser muy perjudicial si hemos escogido unos colores para el texto y los enlaces que no contrastan bien con ese fondo gris. La solución a este problema es poner dentro de la etiqueta **<BODY>** los dos comandos **BACKGROUND** y **BGCOLOR** (en este orden), teniendo cuidado en escoger un color uniforme de fondo parecido al de la imagen.

Por ejemplo, supongamos que queremos poner como fondo la imagen **nubes.jpg**. Escogemos entonces un color de fondo azul claro, (p. ej. #CCFFFF). La etiqueta quedaría así:

<BODY BACKGROUND="nubes.jpg" BGCOLOR="#CCFFFF">

Esto tiene la ventaja adicional de que, incluso aunque no se deshabilite la carga automática de imágenes, al cargar la página, lo primero que se ve es ese fondo de color uniforme, que luego es reemplazado por el de la imagen.

¿Cómo conseguir un fondo para nuestra página?

Teóricamente, cualquier imagen puede servir como fondo, pero unas son más apropiadas que otras. Además, podemos querer crear un fondo nosotros mismos, o capturarlo de otras páginas.

En WebStore encontrarás una coleción de fondos, preparados para ser capturados

Actividades.

1.- Pon un fondo gráfico en la página de estilos anterior, carga también un color de fondo que sea parecido al fondo.

12.- Alineación y dimensionado de imágenes.

(Top, middle, buttom, align, clear, width, height)

La alineación de las imágenes fue, en su día, el primer golpe de efecto del programa Navigator de Netscape. Permitió alinear una imagen a la izquierda o a la derecha de la página y hacer que el **texto la rodee completamente**, consiguiéndose así una apariencia similar a la de una revista.



Es el caso de este párrafo con respecto a la imagen de la derecha. Obsérvese cómo las líneas, cuando rebasan su parte inferior, continúan normalmente hasta el margen derecho de la página. De manera análoga, se puede alinear la imagen a la izquierda, comportándose el texto de una forma equivalente. Esto se consigue con las extensiones de la etiqueta **ALIGN**, (que ya se vio en el capítulo 9 con los comandos **TOP, MIDDLE** y **BOTTOM**).

Pero entonces sólo se hacía referencia al **titular** de la imagen, es decir, a un texto explicativo, pero menor que una línea completa, pues en caso de rebasarla, el texto



saltaba a la parte inferior de la imagen, dejando un hueco en blanco, con lo que su utilidad era muy limitada. Este inconveniente queda solventado con los comandos de Netscape **RIGHT** (derecha) y **LEFT** (izquierda). La imagen de arriba, "doom.gif", **alineada a la derecha**, se ha obtenido con la etiqueta:

<IMG SRC="doom.gif" ALIGN=”RIGHT”>

y la imagen de abajo, "doom2.gif", **alineada a la izquierda**, se ha obtenido con la etiqueta:

<IMG SRC="doom2.gif" ALIGN=”LEFT”>

Si se quiere **interrumpir** el proceso de rellenado del texto a los lados de la imagen, para que salte hasta debajo de ella, es decir, dejar un espacio en blanco parcialmente, se pueden emplear las siguientes extensiones de la etiqueta **
**:

<BR CLEAR=”LEFT”> Busca el primer margen libre *(clear)* a la izquierda.
<BR CLEAR=”RIGHT”> Busca el primer margen libre a la derecha.
<BR CLEAR=”ALL”> Busca el primer margen libre a ambos lados.

Un ejemplo para aclarar esto:

<IMG SRC="isla.gif" ALIGN=”LEFT”> Este texto esta a un lado de la imagen.
**
 Este también esta a un lado de la imagen, en la línea siguiente.**
<BR CLEAR=”LEFT”> Este otro texto, en cambio, ha buscado el primer margen libre a la izquierda.

Que resulta:



Este texto esta a un lado de la imagen.
Este también esta a un lado de la imagen, en la línea siguiente.

Este otro texto, en cambio, ha buscado el primer margen libre a la izquierda.

Dimensionando la imagen

Los programas navegadores cuando cargan un documento HTML y encuentran una etiqueta de una imagen, **interrumpen** el proceso de carga y solicitan al servidor que le envíe dicha imagen, quedando a la espera hasta que se complete el envío, repitiéndose este proceso con cada una de las imágenes.

Esto es especialmente molesto cuando, como ocurre frecuentemente, en la cabecera de la página se encuentra una imagen grande, ya que durante un tiempo relativamente largo no se verá nada en la pantalla.

Para evitar este inconveniente existen unas extensiones de la etiqueta de la imagen **<IMG SRC="imagen.gif">** que sirven para indicar al navegador cuáles son sus **dimensiones en pixels**. (Este dato lo habremos obtenido previamente de algún programa gráfico).

En este caso, el navegador actúa de una forma más favorable, ya que entonces, como conoce las dimensiones de las imágenes les **reserva un espacio** en la pantalla y va colocando el texto de forma apropiada, sin ninguna interrupción, a la vez que va rellenando esos espacios reservados a las imágenes.

Estos comandos son **WIDTH** *(ancho)* y **HEIGHT** *(alto)*.
Por ejemplo, para la imagen **isla.gif** situada más arriba:

<IMG SRC="isla.gif" WIDTH=”120” HEIGHT=”94”>

Es conveniente hacer esto con **todas** las imágenes, incluso con las más pequeñas (iconos, botones, etc), para que no haya ninguna interrupción en el proceso de carga del documento.

Se pueden también, si se quiere, dimensionar las imágenes con unos valores **distintos** a los que realmente tienen, variando el tamaño, la anchura o la altura. Esto es muy conveniente, por ejemplo para poner en la página un ***thumbnail*** (reproducción en pequeño de una imagen), que hace de enlace a la imagen en su verdadero tamaño. De esta manera no recargamos demasiado una página, y el usuario será quien decida qué imágenes desea cargar.

He aquí un ejemplo de *thumbnail*:



La imagen, "nytimes.gif", tiene realmente unas dimensiones de 575x300 pixels (datos obtenidos de un programa gráfico). Para

dimensionar el *thumbnail* a 150x75 (guardando unas proporciones parecidas al original, de 2:1), lo conseguimos con:

<IMG SRC="nytimes.gif" WIDTH=”150” HEIGHT=”75”>

Para hacer que esta imagen reducida sea el enlace con la imagen en su tamaño original, lo conseguimos con:

<A HREF="nytimes.gif"> <IMG SRC="nytimes.gif" WIDTH=”150” HEIGHT=”75”> </A>

También se puede conseguir esto de otra manera, más correcta aunque más laboriosa.

Es la de **reducir en un programa gráfico** esta imagen a 150x75, guardarla con otro nombre, y luego hacer que la pequeña sea el enlace de la grande. Es más correcta esta otra solución porque no todos los navegadores reconocen los comandos **WIDTH** y **HEIGHT**, incluso tampoco algunas de las versiones más antiguas de Netscape.

Actividades.

1.- Crea una colección de fotografías, donde este la misma repetida tres veces con tamaños diferentes. Cuando seleccionemos cualquiera de ellas hará aparecer la foto en su tamaño original.

13.- Tablas.

(Table, td, tr, th, caption, border, align, valign, width, heigth, colspan, rowspan, bgcolor, gackground, cellspacing, cellpadding)

Vamos a ver ordenadamente (de fuera hacia dentro) las etiquetas necesarias para confeccionar las tablas.

1. La **etiqueta general**, que engloba a todas las demás es **<TABLE>** y **</TABLE>**. Es decir:

<TABLE>
[resto de las etiquetas]
</TABLE>

Con esto se presentarían los datos tabulados, pero faltaría la característica que hace más atractivas a las tablas, y es que estos datos vayan **dentro de unos cajetines formados por un borde**. Para esto tenemos que añadir el atributo **BORDER** a la etiqueta, es decir:

<TABLE BORDER>
[resto de las etiquetas]
</TABLE>

2. En el siguiente nivel, dentro de la anterior, están las etiquetas para formar cada **fila** *(row)* de la tabla, que son **<TR>** y **</TR>**. Hay que **repetirlas** tantas veces como

filas queremos que tenga la tabla. Es decir, para una tabla con dos filas, sería:

<TR>
[etiquetas de las distintas celdas de la primera fila]
</TR>
<TR>
[etiquetas de las distintas celdas de la segunda fila]
</TR>

3. En el último nivel (dentro de las anteriores) están las etiquetas de cada **celda**, que son **<TD>** y **</TD>**, que engloban el contenido de cada celda concreta (texto, imágenes, etc.). Hay que **repetirla** tantas veces como celdas queremos que haya en esa fila.

Veamos un **ejemplo** de una tabla con dos filas. Cada fila va a tener tres celdas. Dentro de cada celda vamos a poner un texto indicativo de la posición de dicha celda:

```
<TABLE BORDER>
<TR>
<TD>fila1-celda1</TD> <TD>fila1-celda2</TD> <TD>fila1-celda3</TD>
</TR>
<TR>
<TD>fila2-celda1</TD> <TD>fila2-celda2</TD> <TD>fila2-celda3</TD>
</TR>
</TABLE>
```

Que resulta:

fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	fila2-celda3

Filas con desigual número de celdas

En este ejemplo hemos puesto dos **filas con igual número de celdas**. ¿Qué pasa si ese número es distinto? Pues el navegador forma el número de filas y columnas que haga falta, dejando **espacios en blanco** en las filas que tengan menos celdas.

Si en el ejemplo anterior eliminamos la tercera celda de la segunda fila, es decir si borramos **<TD>fila2-celda3</TD>**, resultará:

fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	

Titular de la tabla

Se puede añadir un **titular** *(caption)* a la tabla, es decir un texto situado encima de la tabla que indica cuál es su contenido. Se consigue con la etiqueta **<CAPTION> y </CAPTION>**.

Si en el ejemplo anterior añadimos la siguiente línea:

<CAPTION> Ejemplo de filas desiguales </CAPTION>, resulta:

Ejemplo de filas desiguales		
fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	

Variando el espesor de los bordes

El atributo **BORDER** (visto anteriormente) pone por defecto un borde de espesor igual a la unidad. Pero se puede hacer que este borde sea tan grueso como queramos, poniendo:

<TABLE BORDER=”número deseado”>

Si en el ejemplo anterior ponemos:

<TABLE BORDER=”5”>

Resultará:

Ejemplo de filas desiguales		
fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	

Celdas de cabecera

Además de las celdas que contienen datos normales, podemos poner, si nos conviene, **celdas de cabecera** *(header)*, que se distinguen por estar el texto de dichas celdas en negrita y centrado.

Esto se consigue con la etiqueta **<TH> y </TH>** (en vez de la normal **<TD> y </TD>**)

Vamos a añadir, en el ejemplo anterior, una fila de estas celdas de cabecera, antes de las otras dos que ya existían:

```
<TR>
  <TH>Columna 1</TH> <TH>Columna 2</TH> <TH>Columna 3</TH>
</TR>
```

Que resulta:

Ejemplo de filas desiguales		
Columna 1	**Columna 2**	**Columna 3**
fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	

Se pueden colocar en el sitio que se quiera, aunque lo normal es que vayan en los bordes, encabezando las columnas o las filas.

Contenido de las celdas

Hasta ahora, en todos los ejemplos se ha puesto un texto normal dentro de las distintas celdas. Pero se puede poner en ellas cualquier otro elemento de los que van en un documento HTML, como **imágenes, enlaces**, etc. No hay más que poner dentro de la etiqueta de la celda la etiqueta correspondiente a una imagen, un enlace, etc.

Ejemplo con una imagen:

```
<TABLE BORDER=”2”>
<TR><TD><IMG SRC="babe.gif"></TD></TR>
</TABLE>
```



Ejemplo con un enlace:

```
<TABLE BORDER=”2”>
<TR><TD><A HREF="index.html">P&aacute;gina principal </A></TD></TR>
</TABLE>
```

Página principal

Posicionamiento del contenido dentro de la celda

Normalmente, el contenido de una celda está **alineado a la izquierda**. Pero se puede cambiar esto añadiendo dentro de la etiqueta de la celda los siguientes atributos:

<TD ALIGN=”CENTER”> Al centro </TD>
<TD ALIGN=”RIGHT”> A la derecha </TD>
<TH ALIGN=”LEFT”> Cabecera a la izquierda </TH> *(Recuérdese que por defecto están centradas)*

Ejemplos de alineamiento horizontal dentro de la celda
Al centro
A la derecha
Cabecera a la izquierda

El alineamiento por defecto en el sentido **vertical** es en el **medio**. También se puede cambiar, añadiendo dentro de la etiqueta de la celda los siguientes atibutos:

<TD VALIGN=”TOP”> Arriba </TD>
<TD VALIGN=”BOTTOM”> Abajo </TD>

	Arriba	Abajo

Variando las dimensiones de la tabla

El navegador se encarga normalmente de dimensionar el tamaño total de la tabla de acuerdo con el número de filas, de columnas, por el contenido de las celdas, espesor de los bordes, etc.

A veces nos puede convenir forzarle para que la tabla tenga unas **dimensiones totales mayores** que las que le corresponden, tanto en anchura como en longitud. Esto se consigue añadiendo dentro de la etiqueta de la tabla los atributos **WIDTH y HEIGHT** igual a un porcentaje de la dimensión de la pantalla, o a una cifra que equivale al número de pixels.

Por ejemplo, si en el último ejemplo ponemos:

<TABLE WIDTH=”60%”>

	Arriba	Abajo

O, si en ese mismo ejemplo, ponemos:

<TABLE HEIGHT=”200”>

	Arriba	Abajo

Celdas que abarcan a otras varias

A veces puede interesarnos que una celda **se extienda sobre otras varias**. Esto se consigue añadiendo dentro de la etiqueta de la celda los atributos **COLSPAN=número** para extenderse sobre un número determinado de columnas, o **ROWSPAN=número** para extenderse verticalmente sobre un número determinado de filas.

Por ejemplo, en la primera tabla de este capítulo vamos a añadir una fila con una sola celda, que abarca a dos columnas:

<TR> <TD COLSPAN=”2”> Celda sobre 2 columnas </TD> <TR>

Celda sobre 2 columnas		
fila1-celda1	fila1-celda2	fila1-celda3
fila2-celda1	fila2-celda2	fila2-celda3

O, en la misma tabla, vamos a añadir una celda en la primera fila. pero que abarque también a la siguiente:

<TD ROWSPAN=”2”> Celda junto a 2 filas </TD>

Celda junto a 2 filas	fila1-celda1	fila1-celda2	fila1-celda3
	fila2-celda1	fila2-celda2	fila2-celda3

Color de fondo en las tablas

Podemos conseguir que las tablas tengan un **color de fondo**, siguiendo un procedimiento totalmente análogo al empleado para que una página tenga un color de fondo uniforme (según vimos en el capítulo 11). Para ello debemos utilizar el atributo **BGCOLOR="#XXYYZZ"**, visto en dicho capítulo.

Se puede conseguir:

1. Que la **totalidad de la tabla** tenga un color de fondo. Para ello, colocamos el atributo dentro de la etiqueta **TABLE**. Por ejemplo, vamos a hacer que la tabla tenga un fondo verde (#00FF00):

   <TABLE BORDER BGCOLOR="#00FF00">

```
<TR> <TD>fila1-celda1</TD> <TD>fila1-celda2</TD> </TR>
<TR> <TD>fila2-celda1</TD> <TD>fila2-celda2</TD> </TR>
</TABLE>
```

Que resulta:

fila1-celda1	fila1-celda2
fila2-celda1	fila2-celda2

2. Que sólamente **una celda determinada** tenga un color de fondo. Para ello, colocamos el atributo dentro de la etiqueta de la celda correspondiente. Por ejemplo, vamos a hacer que sólo la celda 1 de la fila 1 tenga un color verde:

```
<TABLE BORDER>
<TR><TD BGCOLOR="#00FF00">fila1-celda1</TD><TD>fila1-
celda2</TD></TR>
<TR> <TD>fila2-celda1</TD> <TD>fila2-celda2</TD> </TR>
</TABLE>
```

Que resulta:

fila1-celda1	fila1-celda2
fila2-celda1	fila2-celda2

3. Que la generalidad de las celdas tenga un color, pero que alguna celda tenga uno particular. El atributo del color general se coloca en la etiqueta **TABLE**, y el del color particular en la etiqueta de la celda en cuestión (una **combinación de los dos casos anteriores**). por ejemplo, vamos a hacer que la generalidad de la tabla sea de color rojo (#FF0000), pero que la celda 1 de la fila 1 sea de color verde (#00FF00):

```
<TABLE BORDER BGCOLOR="#FF0000">
<TR><TD BGCOLOR="#00FF00">fila1-celda1</TD><TD>fila1-
celda2</TD></TR>
<TR> <TD>fila2-celda1</TD> <TD>fila2-celda2</TD> </TR>
</TABLE>
```

Que resulta:

fila1-celda1	fila1-celda2
fila2-celda1	fila2-celda2

Imágenes de fondo en las tablas

El **Explorer** de Microsoft soporta la colocación de **imágenes de fondo** en el interior de las tablas, de una manera análoga a como se hace en una página (según vimos en el

capítulo 11). Para ello debemos utilizar el atributo **BACKGROUND="imagen.gif"** o **BACKGROUND="imagen.jpg"**, visto en dicho capítulo.

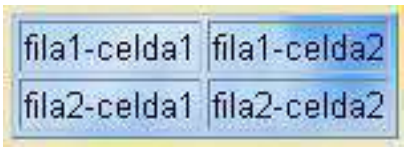
Si se utiliza dentro de la etiqueta **<TABLE>** la imagen en cuestión se multiplicará detrás de todas las celdas.

Por ejemplo, si ponemos:

<TABLE BORDER BACKGROUND="nubes.jpg">

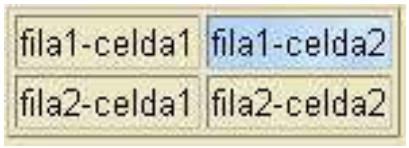
(Se omite el resto de las etiquetas de la tabla)

Obtenemos:



(Con el Netscape no se verá ninguna imagen de fondo, sólo con el Explorer)

Si, por el contrario, sólo se pone este atributo dentro de la etiqueta de una **celda concreta** (**<TD>** o **<TH>**), entonces la imagen de fondo se verá sólo en esa celda, como por ejemplo:



Separación entre las celdas de una tabla

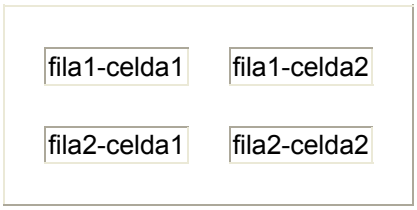
Por defecto, la **separación** entre las distintas **celdas** de una tabla es de dos pixels. Pero se puede variar esto con el atributo **CELLSPACING**, que se pone dentro de la etiqueta **TABLE**.

Por ejemplo, para obtener una separación de 20 pixels entre celdas ponemos:

<TABLE BORDER CELLSPACING=”20”>

(Se omite el resto de las etiquetas de la tabla)

Con lo que se obtiene:



A primera vista parece como si esto fuera lo mismo que si hubiéramos aumentado el espesor de los bordes. Pero para comprobar que no es así, hagamos que en el caso anterior, tenga además unos bordes de 5 de espesor:

<TABLE BORDER=”5” CELLSPACING=”20”>

Con lo que se obtiene:

fila1-celda1	fila1-celda2
fila2-celda1	fila2-celda2

Separación entre el borde y el contenido dentro de las celdas

Por defecto, la separación entre el **borde** y el **contenido** dentro de las celdas es de un pixel. Se puede cambiar esto con el atributo **CELLPADDING**, que se pone dentro de la etiqueta **TABLE**.

Por ejemplo, para obtener una separación de 20 pixels entre el contenido y los bordes, dentro de cada celda:

<TABLE BORDER CELLPADDING=”20”>

(Se omite el resto de las etiquetas de la tabla)

Con lo que se obtiene:

fila1-celda1	fila1-celda2
fila2-celda1	fila2-celda2

Se puede **combinar** este atributo con **CELLSPACING** (visto en el apartado anterior).
Por ejemplo, una tabla con bordes de 5 de espesor, separación entre celdas de 15 y separación del contenido con respecto a los bordes de las celdas de 20, lo obtendríamos con:

<TABLE BORDER=”5” CELLSPACING=”15” CELLPADDING=”20”>

(Se omite el resto de las etiquetas de la tabla)

Con lo que se obtiene:



Ejemplo práctico

Vamos a crear una tabla de dos filas, con una celda cada una. La primera, como cabecera, con el texto "Un lugar ideal para mis vacaciones", y la segunda con la imagen isla.gif. Además vamos a alargar a lo ancho la tabla al 75% de la pantalla.

Para ello copiamos:

```
<HTML>
<HEAD>
<TITLE> Práctica para realizar la web del centro con tablas</TITLE>
</HEAD>
<BODY BACKGROUND="nubes.jpg" BGCOLOR="#CCFFFF" TEXT="#AA0000">
<CENTER>
<H1> Mi p&aacute;gina del Web </H1>
</CENTER>
<HR>
Esta es mi p&aacute;gina del Web. No es muy extensa, pero tiene todos los
elementos b&aacute;sicos. Espero que os guste. Poco a poco le a&ntilde;adiremos m&aacute;s
cosas interesantes.
<P> <A HREF="pagina2.html"> <IMG SRC="hombre.gif" WIDTH=”29” HEIGHT=”27”> </A> <FONT
SIZE=+3>Mis aficiones</FONT>
<P> <A HREF="pagina3.html"> <IMG SRC="inicio.gif" WIDTH=”30” HEIGHT=”29”> </A> <FONT
SIZE=+3>Mis p&aacute;ginas
favoritas</FONT>
<CENTER>
<P><TABLE BORDER=”5” WIDTH=”75%”><TR><TH> Un lugar ideal para mis vacaciones </TH></TR>
<TR><TD ALIGN=CENTER><IMG SRC="isla.gif" WIDTH=”120” HEIGHT=”94”></TD><TR>
</TABLE>
</CENTER>
<P><HR>
<P><IMG SRC="doom.gif" WIDTH=”160” HEIGHT=”100” ALIGN=”left”>Una de mis aficiones favoritas
son los juegos tipo "Doom", con los que paso horas incontables.
<BR CLEAR=LEFT>
<P><HR>
<P><CENTER>
<H3>"Buscando a Mulder"</H3>
<IMG SRC="3mulder.gif" WIDTH=”119” HEIGHT=”160”></CENTER>
</BODY>
</HTML>
```

Actividades

1.- Copia el texto anterior en un fichero y llámale pagina12.html, modifica este fichero y crea uno de estilos donde se pueda obtener distintas configuraciones de tablas con solo llamarlas y observa como modificas todo.

14.- Formularios.

(Form, action, enctype, method, input, textarea, radio, select, checked)

La manera general para que los lectores de nuestra página se puedan comunicar con nosotros es por medio de un enlace a nuestra dirección de email, con lo que recibiríamos un email convencional.

Pero puede ser que lo que necesitemos sea solamente una **respuesta concreta** a unas opciones que presentaremos nosotros mismos, o un **comentario** del usuario, para lo que le suministraremos un espacio en donde introducirlo.

Se puede hacer todo esto, además de otras cosas, utilizando los formularios, con los que se pueden confeccionar páginas que contengan los **elemento**s necesarios para ello, tal como botones de radio, listas de selección, cajetines de introducción de texto y de control, etc., como vamos a ver.

Los formularios permiten que los demás nos envíen la información **directamente a nosotros** o bien a nuestro servidor, en donde hemos instalado un **programa** que procese esta información.

Por ejemplo, vamos a suponer que queremos crear una lista de correo. Los usuarios pueden introducir sus nombres y direcciones de email y pulsar un botón de envío. Esos datos los podemos recibir **"en bruto"** en nuestro correo, con los que confeccionaríamos manualmente dicha lista de correo, sin necesitar ningún programa para ello. Este proceso es el que vamos a comentar en este capítulo.

La otra posibilidad, de la que únicamente se va a hacer esta mención, es que hubiéramos **instalado en nuestro servidor** un programa especial para procesar esos datos y añadirlos a la lista de correo, y que incluso pudiera devolver automáticamente al usuario algún tipo de información. Para conseguir esto, los formularios necesitan ejecutar programas o *scripts* por medio del **CGI** (Common Gateway Interface). El CGI permite a los formularios ser procesados por programas escritos en cualquier lenguaje, aunque los más usados en Internet son el Perl y el C.

Para saber más sobre el CGI:

- En castellano:
  Manual de CGI, por Diego Sebastián Quiroga
  Ejecutables CGI-BIN, en Tejedores del Web, por Carlos Castillo
  FAQ de la lista HTML de la UBA
  LibrerÌa CGI 3.0

- En inglés:
  Web Developper's Virtual Library: CGI
  CGI Index (Yahoo!)

NOTA:
El tipo de formulario que se describe a continuación podrá ser utilizado por la mayoría de los navegadores, **con la notable excepción del Explorer** (incluso la versión 3.0), por lo que es conveniente suministrar al final del formulario, como una alternativa para estos casos, un enlace de email ordinario (ver el Capítulo 8)

Estructura de un formulario

La estructura general de un formulario es:

1. Etiqueta de inicio:
   <FORM ACTION="mailto:*dirección_de_email*" METHOD="POST" ENCTYPE="TEXT/PLAIN">
2. Cuerpo del formulario, con los distintos elementos para poder introducir los datos.
3. Botones de envío y de borrado.
4. Etiqueta de cierre **</FORM>**

Etiqueta de inicio

El atributo **ACTION** indica la acción que se debe efectuar y que es que los datos sean **enviados por email** a la dirección indicada. (Si hiciéramos uso del CGI, sería precisamente aquí donde indicaríamos su localización en el servidor, que habitualmente es el directorio cgi-bin, para que procese los datos).

El atributo **METHOD=POST** indica que los datos sean **inmediatamente** enviados por correo a la dirección de email, nada más pulsar el usuario el botón de envío.

Con el atributo **ENCTYPE="TEXT/PLAIN"** se consigue que las respuestas las recibamos como un **fichero de texto, perfectamente legible y sin codificar**.

Elementos para introducir los datos

Los vamos a dividir en tres clases:

1. Introducción por medio de **texto**
2. Introducción por medio de **menús**
3. Introducción por medio de **botones**

La introducción de los datos se consigue por medio de la etiqueta:

<INPUT TYPE="xxx" NAME="yyy" VALUE="zzz">

En donde:

xxx es la palabra que indica el tipo de introducción.
yyy es el nombre que le asignamos nosotros a la variable de introducción del dato.
zzz es la palabra asociada a un elemento.

Todo esto, que de momento parece muy confuso, se aclarará al ir viendo los distintos casos.

Introducción por medio de texto (una línea)

En este caso es **xxx=text**, es decir, **INPUT TYPE="text"**. El atributo **VALUE** no procede en este caso.

Vamos a poner un ejemplo: solicitamos el apellido del usuario.

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
Escribe tu apellido:
<BR><INPUT TYPE="text" NAME="Apellido">
</FORM>
```

Que resulta:

Escribe tu apellido:

Si el usuario introduce su apellido, p. ej. Ruiz, y pulsa el botón de envío (que veremos más adelante), recibiremos un email suyo con el siguiente texto:

Apellido=Ruiz

La **longitud** de este formulario es por defecto de 20 caracteres. Se puede variar incluyendo en la etiqueta el atributo **SIZE="número"**. Por otra parte, sea cual sea la longitud del formulario, si no se indica nada, el usuario puede introducir el **número de caracteres** que quiera. Se puede limitar esto, incluyendo en la etiqueta el atributo **MAXLENGTH="número"**.

En el caso que hemos visto, si hubiéramos cambiado la etiqueta correspondiente por:
<INPUT TYPE="text" NAME="Apellido" SIZE="10" MAXLENGTH="12">

Habríamos obtenido:

Escribe tu apellido:

(Se puede comprobar cómo no se pueden introducir más de 12 caracteres).

También se puede hacer que el texto introducido **no sea reconocible**, es decir que todos los caracteres se representen por asteriscos. Basta con cambiar en la etiqueta **INPUT TYPE="text"** por **INPUT TYPE="password"**. En el último ejemplo, si cambiamos la etiquea correspondiente por:

<INPUT TYPE="password" NAME="Apellido" SIZE="10" MAXLENGTH="12">

Obtenemos:

Escribe tu apellido:



(Se puede comprobar cómo los caracteres introducidos se representan por asteriscos)

Introducción por medio de texto (múltiples líneas)

Cuando el texto a introducir puede alcanzar una **gran longitud**, por ejemplo un comentario, es conveniente utilizar un formulario de texto de **múltiples líneas**.

Esto se consigue con la etiqueta de inicio:

<TEXTAREA NAME="yyy" ROWS="número" COLS="número">

*(en donde no se utiliza **INPUT TYPE** y donde **ROWS** representa el número de filas, y **COLS** el de columnas).*

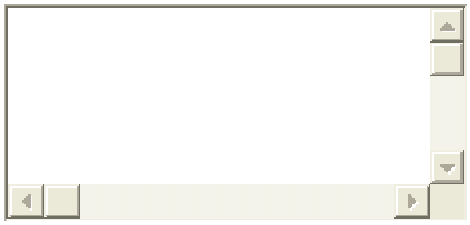
y la de cierre: **</TEXTAREA>**

Ejemplo: un formulario solicitando los comentarios del usuario:

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
Introduce tus comentarios:
<BR><TEXTAREA NAME="Comentarios" ROWS="6" COLS="40">
</TEXTAREA>
</FORM>
```

Que resulta:

Introduce tus comentarios:



(El salto de línea del texto introducido no se efectúa automáticamente).

Una vez que el usuario haya escrito sus comentarios dentro del formulario, y haya pulsado el botón de envío, recibiremos un email suyo con el siguiente texto:

Comentarios=Bla bla bla...

Introducción por medio de menús

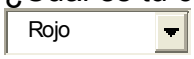
Si queremos que el usuario, en vez de introducir un texto, como hemos visto en los casos anteriores, **escoja entre varias opciones** que le presentamos nosotros, haremos uso de un formulario en forma de menú.

Se consigue con la etiqueta de inicio **<SELECT NAME="yyy">** y la de cierre **</SELECT>**.
Las distintas opciones a escoger se consiguen con la etiqueta **<OPTION>**.
Ejemplo: Pedimos al usuario que elija su color preferido:

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
¿Cu&aacute;l es tu color preferido?
<BR><SELECT NAME="ColorPreferido">
<OPTION>Rojo
<OPTION>Verde
<OPTION>Azul
<OPTION>Amarillo
</SELECT >
</FORM>
```

Que resulta:

¿Cuál es tu color preferido?



Si el usuario ha escogido, p. ej. Azul y ha pulsado el botón de envío, recibiremos un email suyo con el texto: **ColorPreferido=Azul**.

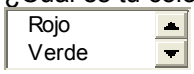
En el ejemplo anterior, sólo es visible en el formulario una opción. Si queremos que sean **visibles múltiples opciones a la vez**, añadimos en la etiqueta los atributos **MULTIPLE SIZE="número"**, donde especificamos el número de opciones visibles.

Si cambiamos en el ejemplo anterior la etiqueta correspondiente por:

<SELECT NAME="ColorPreferido" MULTIPLE SIZE="2">

Obtenemos:

¿Cuál es tu color preferido?



Formulario de confirmación *(checkbox)*

Si queremos que el usuario confirme una opción determinada, podemos hacer uso de un formulario de confirmación, o *checkbox*, que se consigue con la etiqueta:

<INPUT TYPE="checkbox" NAME="yyy">

Ejemplo: Solicitamos al usuario que confirme su inclusión en una lista de correo:

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<INPUT TYPE="checkbox" NAME="Lista">
S&iacute;, deseo ser incluido en la lista de correo.
</FORM>
```

Que resulta:

☐ Sí, deseo ser incluido en la lista de correo.

Si el usuario marca este formulario y pulsa el botón de envío, recibiremos un email suyo con el texto: **Lista=On**.

Si queremos que el formulario aparezca **inicialmente como marcado** (el usuario no necesitará hacerlo), basta con añadir el atributo **CHECKED** dentro de la etiqueta. En el ejemplo anterior si sustituimos la etiqueta equivalente por:

```
<INPUT TYPE="checkbox" NAME="Lista" CHECKED>
```

Obtenemos:

☑ SÌ, deseo ser incluido en la lista de correo.

Botones de radio

Cuando queremos que el usuario elija **una única opción entre varias**, podemos hace uso de los botones de radio, que se consiguen con la etiqueta:

<INPUT TYPE="radio" NAME="yyy" VALUE="zzz">

Donde **yyy** es el nombre que le ponemos a la variable que se trata de elegir, y **zzz** es el nombre de cada una de las opciones en concreto.

Ejemplo: solicitamos al usuario que defina cuál es su sistema operativo preferido:

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
¿Cu&aacute;l es tu sistema operativo preferido?
<BR>
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="PC" CHECKED> PC
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="Mac"> Mac
```

```
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="Unix"> Unix
</FORM>
```

Que resulta:

¿Cuál es tu sistema operativo preferido?
PC Mac Unix

Obsérvese el atributo opcional **CHECKED** que se ha añadido en la primera etiqueta. Esa será la opción que aparece marcada por defecto.

Obsérvese también que no es posible escoger más de una opción.
Si el usuario ha escogido la opción PC y pulsa el botón de envío, recibiremos un email suyo con el texto: **SistemaOperativo=PC**.

Botones de envío y de borrado

Hasta ahora, en todos los ejemplos que hemos visto, faltaba **un elemento esencial** en cualquier formulario, y es el botón de envío de los datos, que se consigue con la etiqueta:

```
<INPUT TYPE="submit" VALUE="zzz">
```

En donde **zzz** es el texto que queremos que aparezca en el botón.
Vamos a añadirlo al primer ejemplo, en el que se solicitaba el apellido del usuario:

```
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
Escribe tu apellido:
<BR><INPUT TYPE="text" NAME="Apellido">
<P><INPUT TYPE="submit" VALUE="Enviar datos">
</FORM>
```

Que resulta:

Escribe tu apellido:

Enviar datos

Otro botón interesante es el de **borrado** de los datos introducidos, muy conveniente en un formulario con muchos elementos. Es muy similar al de envío, pues se consigue con la etiqueta:

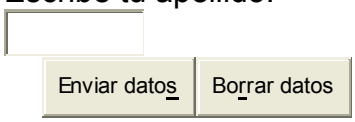
```
<INPUT TYPE="reset" VALUE="zzz">
```

En donde **zzz** es el texto que queremos que aparezca en el botón.
Si añadimos al ejemplo anterior la etiqueta:

```
<P><INPUT TYPE="reset" VALUE="Borrar datos">
```

resulta:

Escribe tu apellido:



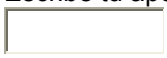
Se puede comprobar su funcionamiento, escribiendo algo en el formulario y pulsando luego el botón de borrado.

Consideraciones finales

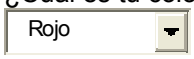
Hasta ahora hemos visto uno a uno los diferentes elementos que se pueden utilizar. Pero no hay ningún inconveniente en usar, dentro del mismo formulario, **distintos tipos** de introducción de datos. Al pulsar el usuario el botón de envío recibiríamos en email suyo con las distintas parejas **NAME=VALUE** de cada elemento, encadenadas con el símbolo &.

Por ejemplo, con este formulario, con dos elementos distintos:

Escribe tu apellido:



¿Cuál es tu color preferido?



recibiríamos un email con el texto (p. ej.):
Apellido=Ruiz
ColorPreferido=Azul.

Actividades:

1.- Crea una página en la que aparezca un libro de visitas donde nos envíe un correo con los datos de quien nos ha visitado. Puedes ayudarte del ejemplo siguiente.

```
<CENTER>
<H2>Libro de visitas</H2>
<BR><FORM ACTION="mailto:portaleso@ono.com" METHOD="POST" ENCTYPE="TEXT/PLAIN">
Tu nombre:
<BR><INPUT TYPE="text" NAME="Nombre">
<P>Escribe tus comentarios: </P>
<BR><TEXTAREA NAME="Comentarios" ROWS="6" COLS="40">
</TEXTAREA>
<P><INPUT TYPE="submit" VALUE="Enviar datos">
<INPUT TYPE="reset" VALUE="Borrar datos"></P>
</FORM>
</CENTER>
<HR>
```

15.- Gif animados

En 1989 se estableció la posibilidad de la existencia de colores transparentes, así como que un fichero GIF pudiera contener internamente **múltiples imágenes**, visibles durante un tiempo determinado, con lo que se podía obtener una **imagen animada**. Pero esta interesante característica permaneció prácticamente en el olvido hasta finales de 1995, cuando Netscape la implementó en su navegador, adquiriendo muy pronto una gran popularidad.

Hay **otros tipos** de animaciones, como las obtenidas con:

- Quicktimes, FLIC, AVI, etc., que tienen el gran inconveniente de que requieren, para poder ser contempladas, de programas auxiliares previamente instalados.
- Java, que requiere un nivel de conocimientos de este lenguaje de programción que no está al alcance de cualquiera.
- Las animaciones basadas en los servidores, que tienen el inconveniente de que consumen el ancho de banda e incrementan el tráfico de datos, debido a que deben estar alimentando continuamente la animación desde el servidor.

Las animaciones basadas en el formato GIF 89a **eliminan todos estos problemas**, pues:

- No requieren de ningún programa auxiliar, siempre que se utilice el navegador adecuado para poder contemplarlas.
- Son tan fáciles de utilizar en una página del Web como cualquier imagen GIF fija.
- No incrementan el tráfico de datos, pues el usuario la carga de una sola vez y se ejecutan luego desde el caché del navegador.

Para insertar un fichero Gif animado, se hace igual que cualquier otro un ejemplo es:

```
<P></P><HR>
<H2>¡Sitio del Web en obras!</H2>
<IMG SRC="workanim.gif">
<P></P><HR>
```

Actividades.

1.- Crea una página donde aparezca un gif animado que ocupe la parte central de la página para indicar que esa sección está en obras.

16.- Menús realizados con Mapas

(Map, Name, Area, href, scr, shape, coords, usemap, nohref)

Hemos visto anteriormente dos maneras distintas de **enlazar una página con otra**: por medio de un enlace de texto o por medio de una imagen. Pero en este último caso se hacía uso de una imagen para enlazar a **una única página**.

Se puede utilizar una única imagen para enlazar con **varias** páginas, yendo a una u otra según la zona en donde se pulse el ratón. Este tipo de imágenes se llaman **mapas**.

Veremos cómo se crea un mapa partiendo de una imagen, y cómo se implementa dicho mapa con el lenguaje HTML en nuestra página.

Hay **dos tipos** de mapas:

1. Mapas gestionados por el servidor
2. Mapas gestionados por el cliente

Mapas gestionados por el servidor

En este tipo de mapas al desplazar el cursor sobre la imagen que forma el mapa se van obteniendo las distintas **coordenadas**. Cuando se pulsa el ratón en un punto determinado, el navegador envía esas coordenadas al servidor, y éste comprueba en un fichero MAP (situado en en su directorio cgi-bin) cuál es la página que se corresponde con estas coordenadas, envía esta información al navegador, y éste solicita a su vez al servidor que le enlace con dicha página.

Estos mapas, que fueron los que aparecieron inicialmente, tienen una **ventaja**: que pueden ser utilizados por **todos los navegadores**, incluso por las versiones más antiguas.

Pero tienen varios **inconvenientes**: No todos los servidores tienen implementado el programa CGI necesario para que pueda funcionar un mapa de este tipo. Además hay que conocer el tipo de servidor (NCSA, el más común, o CERN), pues el código a aplicar es distinto en un caso u otro.. Por otra parte, al requerir un trasiego de información entre el navegador y el servidor, el tiempo de respuesta es mayor que en el otro tipo, como veremos.

Mapas gestionados por el cliente

Estos mapas son gestionados por el cliente (es decir, **nuestro navegador**) y no por el servidor, como en el caso anterior.

En este tipo de mapas, al ir desplazando el cursor se observa que en ciertas zonas se convierte en el símbolo normal para enlazar con otras páginas (una mano, habitualmente). Si se pulsa en estas **zonas activas** *(hotspots)*, el navegador consulta con el documento HTML que ha recibido, y decide a qué página o dirección corresponde esa zona activa.

Es decir, no necesita enviar la información al servidor y esperar la respuesta de éste, con lo que el proceso es **más rápido** que en el caso anterior, reduciéndose además el tráfico de datos y la sobrecarga al servidor. Otras **ventajas** son que al pasar por una zona activa se muestra la dirección a la que se corresponde (al contrario que en el caso anterior, en la que sólo se ven unas coordenadas). Además, estos mapas se pueden utilizar *off-line*, al contrario también que en el otro tipo, ya que no requieren la ayuda del servidor.

Pero tiene el **inconveniente** de que estos mapas no son implementados por todos los navegadores, como en el caso anterior, aunque sí por la mayoría de los utilizados actualmente, desde las versiones siguientes: Netscape 2.0, Microsoft Explorer 2.0, Mosaic 2.1, etc. Pero se puede obviar este inconveniente suministrando **enlaces alternativos** para los navegadores que no implementen este tipo de mapas, como veremos.

En lo que sigue, vamos a ver **exclusivamente** cómo implementar este tipo de mapas, es decir, **mapas gestionados por el cliente** *(client-side image maps)*.

Confección de la imagen del mapa

Antes que nada, es necesario confeccionar la **imagen** que va a servir como mapa. Se hará con un programa gráfico, y pueden utilizarse los formatos GIF o JPG.

Vamos a crear una mapa para utilizarlo en el ejemplo práctico de este capítulo, con dos zonas activas rectangulares que enlacen con los ejemplos prácticos de capítulos anteriores.

Se confecciona con un programa gráfico la imagen que nos va a servir como mapa, tal como la situada a la izquierda, que todavía no es un mapa.

Para definir un **área activa rectangular**, necesitamos conocer las coordenadas de su ángulo superior izquierdo y las de su ángulo inferior derecho.

Estas coordenadas las obtenemos con el programa gráfico con el que hemos confeccionado la imagen. Estos valores son los siguientes:

Para la primera zona activa "Mis aficiones": (48,45) y (250,80)

Para la segunda zona activa "Mis páginas favoritas": (48,103) y (250,135)

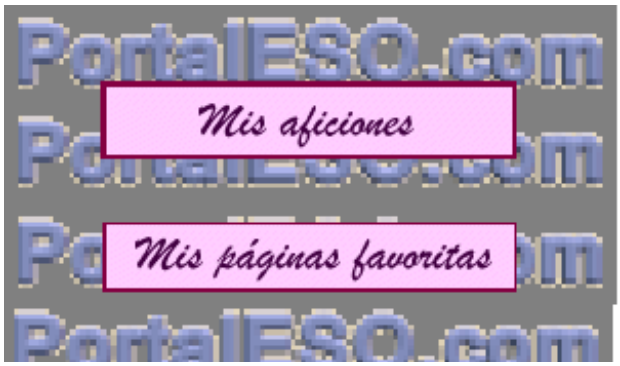
Ya tenemos los datos necesarios para escribir el código HTML que convierta esta imagen en un mapa.

Código HTML para definir un mapa gestionado por el cliente

En este caso, el código que debemos escribir es el siguiente:

```
<MAP NAME="mi_mapa">
<AREA SHAPE="RECT" COORDS="47,38,248,73" HREF="pagina1.html">
<AREA SHAPE="RECT" COORDS="49,107,249,137" HREF="pagina2.html">
<AREA SHAPE="DEFAULT" NOHREF>
</MAP>
<IMG SRC="mapa1.gif" USEMAP="#mi_mapa">
```

Que da como resultado:



Veamos ahora cada una de las etiquetas:

<MAP NAME="mi_mapa">

Esta es la etiqueta de apertura del mapa, y en donde se define su nombre.

<AREA SHAPE="RECT" COORDS="47,38,248,73" HREF="pagina1.html">

Esta etiqueta define la primera zona activa. Se indica que su forma (*shape*, en inglés) es rectangular, y que las coordenadas de su ángulo superior izquierdo son (47,38), y que las del ángulo inferior derecho son (248,73). A continuación se indica cuál es enlace que se desea obtener, en este caso con la página **pagina1.html**.

La siguiente etiqueta es análoga, y se refiere a la segunda zona activa.

<AREA SHAPE="DEFAULT" NOHREF>

Con esta etiqueta se define **el área completa del mapa**, indicando que su forma es la que tiene por defecto (*default*), o sea rectangular. Si se quisiera que el mapa no abarcara la totalidad de la imagen, o incluso que tuviera una forma distinta, habría que indicarlo aquí, en lugar de la instrucción **DEFAULT**, haciendo uso del mismo tipo de instrucciones que las de las zonas activas.

A continuación se indica, con **NOHREF**, que el área completa de la imagen es una **zona no activa**, excepto en las áreas definidas anteriormente como activas.

</MAP>

Etiqueta de cierre del mapa.

<IMG SRC="mapa1.gif" USEMAP="#mi_mapa">

Con esta etiqueta se solicita primero al servidor que envíe una imagen llamada **mapa1.gif**. Con **USEMAP="#mi_mapa"** se indica que esta imagen es el mapa definido anteriormente con ese nombre, y que debe actuar en consecuencia a las pulsaciones del ratón, solicitando un enlace determinado en las zonas activas, o no haciendo nada en el resto.

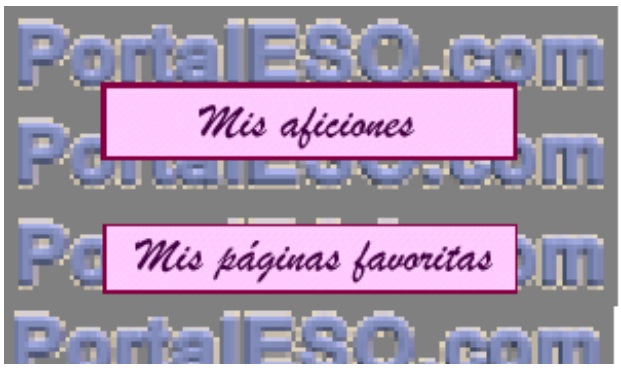
Enlaces alternativos de texto

Como se ha dicho anteriormente, no todos los navegadores implementan este tipo de mapas gestionados por el cliente. Por ello, es necesario suministrar unos **enlaces alternativos convencionales de texto** , para obviar este inconveniente.

Por ejemplo, a continuación del mapa, se pueden poner estos dos enlaces de texto, correspondientes a las dos zonas activas (reduciendo la fuente, para hacerlo menos antiestético):

```
<FONT SIZE=”-1”>
<A HREF="pagina1.html">Mis aficiones</A> |
<A HREF="pagina2.html">Mis p&aacute;ginas favoritas</A>
</FONT>
```

Que, una vez de colocado a continuación del código del mapa y centrado todo, resulta como:



Mis aficiones | Mis páginas favoritas

Zonas activas en forma de círculos o polígonos

En el ejemplo que se ha visto anteriormente se han definido las dos zonas activas en forma de rectángulos. Pero se pueden definir también estas zonas activas como **círculos** o como **polígonos**.

Para definir una **zona activa circular** es necesario conocer las coordenadas de su centro (x,y) y la longitud de su radio (r). Una vez sabidos estos datos, la etiqueta es como sigue:

```
<AREA SHAPE="CIRCLE" COORDS="x,y,r" HREF="dirección_de_la_página">
```

Para definir una **zona activa poligonal** hay que conocer las coordenadas de los distintos puntos **(x1,y1), (x2,y2), (x3,y3)**, etc. Con estos datos, la etiqueta se escribe como sigue:

```
<AREA SHAPE="POLYGON" COORDS="x1,y1,x2,y2,x3,y3,..."
HREF="dirección_de_la_página">
```

Programas editores de mapas

Hemos visto cómo se debe escribir el código HTML para convertir una imagen en un mapa. Pero hay **programas** que lo hacen de una manera **automática**.

Partiendo de una imagen (confeccionada con un programa gráfico normal), con este tipo de programas no hay más que señalar con el ratón qué áreas se desean que sean activas (y qué forma deben tener) e indicar los enlaces para cada zona activa. Entonces, el programa escribe todo el código HTML de una forma automática, sin necesidad de tener que suministrar los datos de las coordenadas de cada zona activa.

Programas:

Para PC y Unix: Mapedit. (http://www.boutell.com/mapedit/)
Para el Mac: WebMap. (http://www.webcrawler.com/info.wbcrwl/)

Actividades.

1.- Obtén una imagen y divídela en cuatro zonas, dos rectangulares, otra redonda, y otra con forma de polígono irregular. Haz que se acceda a cada una de las páginas siguientes respectivamente: pagina1.html, pagina2.html, pagina3.html y pagina12.html

2.- Crea un gráfico donde aparezcan todas las áreas del instituto. A continuación crea una tabla donde aparezcan las coordenadas inicial y final del texto de cada una de las áreas. Por último crea el código html que es necesario para obtener el enlace a cada una de las áreas. El nombre de la página se llamará: nombre_area.html. Guarda el fichero con el nombre: mapa_del_ies.html

17.- Páginas que utilizan Frames.

(Frame, frameset, target, src, name, marginwidth, marginheight, scrolling, noresize, frameborder)

Frames (en inglés, marcos o cuadros) es un procedimiento del lenguaje HTML para dividir la pantalla en diferentes zonas, o ventanas, que pueden actuar independientemente unas de otras, como si se trataran de **páginas diferentes**, pues incluso cada una de ellas pueden tener sus propias barras deslizadoras. Los navegadores que lo implementan son el Netscape 2.0, y el Explorer 2.0 en adelante. Ojo esta opción no esta incluida en los estándares anteriores al HTML 3.2.

Una de sus características más importantes es que pulsando un enlace situado en un frame, se puede cargar en otro frame una página determinada. Esto se utiliza frecuentemente para tener un frame estrecho en la parte lateral (o superior) con un **índice** del contenido en forma de diferentes enlaces, que, al ser pulsados cargan en la ventana principal las distintas páginas. De esta manera **se facilita la navegación** entre las páginas, pues aunque se vaya pasando de unas a otras, siempre estará a la vista el índice del conjunto.

Para comprender los distintos conceptos vamos a desarrollar un **ejemplo**, creando una página con dos frames. El de la izquierda va a servir de índice de lo que veamos en el de la derecha, y en éste veremos inicialmente una página de presentación.

Documento de definición de los frames

Lo primero que tenemos que hacer es crear un documento HTML en el que **definiremos** cuántas zonas va a haber, qué distribución y tamaño van a tener, y cuál va ser el contenido de cada una de ellas.

En el ejemplo que vamos a desarrollar, la página va a tener dos frames distribuidos en **columnas** (es decir, uno al lado del otro, en vez de uno encima del otro, lo que sería una distribución en filas).

Con respecto al **tamaño**, haremos que el primero (el de la izquierda) ocupe el 20% del ancho de la pantalla, y el otro, el 80% restante.

Y con respecto al **contenido**, el frame de la izquierda va a contener un documento HTML que va a servir de índice de lo que veamos en el otro (y que vamos a llamar **indice.html**), y el de la derecha otro documento HTML que va a servir de página de presentación (al que llamaremos **presentacion.html**).

Todo lo anterior se refleja en el siguiente documento HTML:

```
<HTML>
<HEAD>
<TITLE>P&acute;gina con frames</TITLE>
</HEAD>
<FRAMESET COLS="25%, 75%">
<FRAME SRC="indice.html">
<FRAME SRC="presentacion.html" NAME="principal">
</FRAMESET>
</HTML>
```

Obsérvese lo siguiente:

- Es un documento parecido a los que conocíamos hasta ahora. La **diferencia** está en que en vez de utilizar la etiqueta **BODY**, que sirve normalmente para delimitar lo que se va a ver en la pantalla, se hace uso de la etiqueta **FRAMESET** (*definir los frames*).

- En este caso, con la etiqueta **<FRAMESET COLS="25%, 75%">** se define que va a haber dos frames y que van a ir en **columnas**. Si hubiéramos querido que fueran en **filas**, habríamos puesto **ROWS** (filas, en inglés). También se define el espacio en anchura que van a ocupar cada uno de ellos en la pantalla. Se ha puesto como porcentajes del total, pero se podría también haber puesto una cifra absoluta, que representaría el número de pixels a ocupar.

- Ya se ha definido el número de frames, su distribución y su tamaño, pero falta por

definir el **contenido** de cada frame. Esto se hace con las etiquetas:

```
<FRAME SRC="indice.html">
<FRAME SRC="presentacion.html" NAME="principal">
```

Con esto se define que el contenido del primer frame (el de la izquierda) sea el documento HTML **indice.html** y el del segundo (el de la derecha) sea el documento HTML **presentacion.html**.

- Obsérvese que en la etiqueta del segundo se ha incluido el atributo **NAME="principal"**, pero no así en el primero. El motivo es que se necesita dar un **nombre** al segundo frame, pues, como veremos a continuación, en el documento del primer frame va a haber unos enlaces que van a ir dirigidos hacia él. En este caso sólo tenemos dos frames, pero podría haber más, y es necesario distinguirlos unos de otros. Y el primero no necesita nombre, pues no va a haber enlaces en el segundo dirigidos hacia él.

  A este documento le vamos a llamar **frames.html**, pero todavía no lo vamos a guardar, pues falta por añadir algo que veremos más adelante.

Documentos HTML de cada frame

Necesitamos ahora confeccionar el documento HTML de cada uno de los frames. Recuérdese que son como páginas independientes, que pueden tener cada una su propio fondo, etc., y todo lo que queramos añadir en ellos y que hemos aprendido hasta ahora.

Documento del frame de la izquierda

Va a tener un fondo amarillo, y va a contener dos enlaces dirigidos al frame de la derecha. Además, como muestra de que se puede añadir cualquier cosa en un frame, vamos a incluir este logo animado

(**logo.gif**): 

```
<HTML>
<HEAD>
<TITLE> &lacute;ndice </TITLE>
</HEAD>
<BODY BGCOLOR="#FFBB00">
<P><IMG SRC="logo.gif">
<P><A HREF="presentacion.html" TARGET="principal">
Presentaci&oacute;n </A>
<P><A HREF="pagina12.html" TARGET="principal"> Mi p&aacute;gina </A>
</BODY>
</HTML>
```

Dentro de las etiquetas de los enlaces podemos observar algo nuevo, y es el atributo **TARGET** (*en inglés: objetivo, blanco*), que sirve para hacer que al ser activado el enlace **no se cargue en el propio frame, sino en otro**, precisamente en el que hayamos llamado con ese nombre en el documento de definición de los frames.

En nuestro caso, le hemos dado el nombre de "principal" al frame de la derecha, y es por tanto ahí donde se van a cargar los documentos HTML.

Guardamos este documento con el nombre de **indice.html**. Además, capturamos la imagen **logo.gif**, y la guardamos junto con el documento.

Documento del frame de la derecha

Va a tener un fondo negro, y va a contener solamente un texto. Pero puede contener cualquier cosa.

```
<HTML>
<HEAD>
<TITLE> Presentaci&oacute;n </TITLE>
</HEAD>
<BODY BGCOLOR="#000000" TEXT="#0000FF">
<CENTER>
<FONT SIZE= +3 ><STRONG>
<P>ESTA ES LA VERSI&Oacute ;N</P>
<BR><FONT COLOR="#FF0000">CON FRAMES</FONT>
<BR>DE MI P&Aacute;GINA
</STRONG>
</FONT>
</CENTER>
</BODY>
</HTML>
```

Guardamos este documento con el nombre de **presentacion.html**

Alternativa para los navegadores que no soportan frames

Con lo visto anteriormente, ya tenemos las tres piezas que necesitamos:

- Documento de definición de los frames
- Documento del contenido del primer frame
- Documento del contenido del segundo frame

En teoría no necesitamos nada más. Sólo queda un detalle: ¿Qué ocurriría con los **navegadores que no soportan** los frames? Pues que al desconocer las etiquetas

FRAMESET y **FRAME** del documento de definición, no podría ejecutarlo y no podría cargar la página.

Para estos casos está prevista la etiqueta **<NOFRAMES>** y **</NOFRAMES>**. Se colocan a continuación de las distintas etiquetas **<FRAME>**, quedando la estructura de esta manera:

```
<HTML>
  <FRAMESET ....>
    <FRAME....>
    <FRAME....>
    <NOFRAMES>
      <BODY>
      [Aquí va lo que queremos que vean los usuarios con navegadores
      que no soportan frames]
      </BODY>
    </NOFRAMES>
  </FRAMESET>
</HTML>
```

Obsérvese que **dentro** de la etiqueta **<NOFRAMES>** se han colocado las etiquetas **<BODY>** y **</BODY>**, pero téngase en cuenta que esto es **sólo** para esa página de advertencia.

En nuestro ejemplo, vamos a poner un mensaje advirtiendo de esta circunstancia, y dirigiendo al usuario, con un enlace normal, hacia la página **pagina12.html**.

(*También podríamos no decir nada sobre frames, y colocar aquí el código de la página de presentación, pero añadiendo un enlace a **pagina12.html**, para que se pueda pasar de página)*

```
<NOFRAMES>
<BODY>
Estas utilizando un navegador que no soporta frames.
<P><A HREF="pagina12.html">Pulsa para visitar mi  p&aacute;gina</A>.
</BODY>
</NOFRAMES>
```

Lo incluimos dentro del documento de definición de los frames, que hemos visto antes, y que queda de esta manera:

```
<HTML>
<HEAD>
<TITLE>P&aacute;gina con frames</TITLE>
</HEAD>
<FRAMESET COLS="25%, 75%">
<FRAME SRC="indice.html">
<FRAME SRC="presentacion.html" NAME="principal">
<NOFRAMES>
<BODY>
```

```
Estas utilizando un navegador que no soporta frames.
<P><A HREF="pagina12.html">Pulsa para visitar mi  p&aacute;gina</A>.
</BODY>
</NOFRAMES>
</FRAMESET>
</HTML>
```

Ahora sí podemos guardar este documento con el nombre de **frames2.html**.

Si lo cargamos, este es el **resultado con frames** (para los navegadores que los soportan).

Se puede comprobar cómo cambiamos de página en el frame de la derecha, según el enlace que pulsemos en el frame de la izquierda.

Los que **no los soportan** verían esto **otro** (*Está forzado para que lo vean todos los navegadores, incluso los que soportan frames*).

Con este ejemplo práctico hemos aprendido la estructura básica de los frames, pero no hemos utilizado todas sus posibilidades. A continuación se detallan los distintos atributos que se pueden añadir a las etiquetas.

Atributos de la etiqueta FRAMESET

La etiqueta **FRAMESET**, como hemos visto en el ejemplo, es la que define la distribución, el número y tamaño de los frames. Tiene dos atributos: **COLS** (columnas) y **ROWS** (filas):

```
<FRAMESET COLS="xx, yy, zz, ..">
<FRAMESET ROWS="xx, yy, zz, ..">
```

- Define la **distribución** de los frames en columnas o en filas, según se use uno u otro atributo.
- Define el **número** de frames que habrá, pues, por ejemplo:

  <FRAMESET COLS="xx, yy"> (habrá dos frames en columnas)
  <FRAMESET COLS="xx, yy, zz"> (habrá tres frames en columnas)

- Define el **tamaño** de los frames, según el valor que demos a **xx, yy, zz...**. Este valor se puede expresar en:
  - Un **porcentaje** del ancho del pantalla (para las columnas), o del alto de la pantalla (para las filas). Así, por ejemplo:

    <FRAMESET COLS="%20, %80"> (la columna de la izquierda ocupará el 20% del ancho de la pantalla, y la de la derecha el 80% restante)

<FRAMESET ROWS="%10, %70, %20"> (la fila superior ocupará el 10% del alto de la pantalla, la del medio el 70%, y la inferior el 20%)

- Un **número absoluto** que representa el número de pixels que ocupará cada frame a lo ancho o a lo alto (según sean filas o columnas). Así, por ejemplo:

<FRAMESET COLS="40, 600"> (la columna de la izquierda tendrá 40 pixels de ancho y la de la derecha 600). Pero es **peligroso** utilizar sólo valores absolutos, pues el tamaño de la pantalla varía de un usuario a otro. Si se va a usar algún valor absoluto para un frame, es mejor mezclarlo con alguno relativo, como los que vamos a ver a continuación, para que se **ajuste** el total a la pantalla del usuario.

- Un **valor relativo** que se consigue poniendo un asterisco (*), en vez de un número. Esto se interpreta como que ese frame debe tener el espacio **restante**. Por ejemplo:

<FRAMESET ROWS="100,*,100"> (Habrá tres filas, la superior y la inferior de una altura fija de 100 pixels, y la del medio obtendrá el espacio restante). Si hay más de un frame con asterisco, ese espacio restante se dividirá por igual entre ellos.
Si hay un número antes del asterisco, ese frame obtiene esa cantidad más de espacio relativo. Así "2*,*" daría dos tercios para el primer frame y un tercio para el otro.

Frames sin bordes

Si se desea que no haya un **borde de separación** entre los frames, se deben incluir el atributo **FRAMEBORDER=”0”** dentro de la etiqueta **FRAMESET**. No todas las versiones de los navegadores lo implementan.

Para que también desaparezcan los huecos de separación entre frames hay que añadir otros dos atributos (el primero es para el Explorer y el segundo para el Netscape): **FRAMESPACING=”0”** y **BORDER=”0”**. con lo que la etiqueta completa quedaría:

<FRAMESET FRAMEBORDER=”0” FRAMESPACING=”0” BORDER=”0” COLS="xx, yy">

Atributos de la etiqueta FRAME

Esta etiqueta define las cararacterísticas de un frame **concreto**, no del conjunto de los frames, como era el caso con la etiqueta **<FRAMESET>**. Puede tener los siguientes posibles atributos, que van **dentro de la etiqueta <FRAME>**:

- **SRC="*dirección*"**. Esta dirección puede ser la de un documento HTML, o cualquier otro recurso del Web (o URL). Con este atributo se indica lo que se cargará

inicialmente en el frame.
Si no se le pone este atributo a la etiqueta <**FRAME**>, entonces dicho frame aparecerá inicialmente **vacío**, aunque tendrá las dimensiones asignadas.

- **NAME="*nombre_de_la_ventana*"**. Este atributo se usa para asignar un nombre a un frame. De esta manera se podrá "dar en el blanco" (en inglés, *target*) en esta página, desde un enlace situado en otra página. Es decir, que pulsando en otra página un enlace, se cargará precisamente en ésta, tal como hemos visto en el ejemplo.
  El atributo **NAME** es opcional. Por defecto, todas las ventanas carecen de nombre. Los nombres que se escojan deben comenzar por un carácter **alfanumérico** (una letra o un número, pero no otro tipo de símbolo)

- **MARGINWIDTH="*número*"**. Se utiliza este atributo cuando se quiere controlar la el **ancho** de los **márgenes** dentro de un frame. El número que se ponga representa los pixels de los márgenes. Este atributo es opcional.

- **MARGINHEIGHT="*número*"**. Igual que el anterior, pero referido a los márgenes en **altura**.

- **SCROLLING="yes|no|auto"**. Este atributo se utiliza para decidir si el frame tendrá o no una **barra deslizadora**. Si se escoje **"yes"** tendrá siempre una barra deslizadora. Si se escoje **"no"** no la tendrá nunca, y si se escoje **"auto"**, será el navegador quien decida si la tendrá o no. Este atributo es opcional. Su valor por defecto es **"auto"**.

- **NORESIZE**. A este atributo no se le asigna un valor numérico, como a los demás. Es un indicador para que la ventana **no se pueda re-dimensionar** (en inglés, *resize*) por parte del usuario. Se puede comprobar en el ejemplo con frames que si se coloca el cursor del ratón entre los dos frames, al pulsarlo y arrastrarlo en un sentido u otro, varían las dimensiones prefijadas de los frames. Este atributo impide que ocurra esto en un frame. Es un atributo opcional. Por defecto, todos los frames son re-dimensionables.

- **FRAMEBORDER="no"**. Este atributo elimina el borde en un frame, pero si se quiere que se elimine completamente, también hay que ponérselo al frame contiguo. Si se quiere eliminar los bordes de todos los frames, se debe colocar en la etiqueta **FRAMESET**, como hemos visto anteriormente.

El atributo TARGET

En el ejemplo hemos visto que, como queríamos que los enlaces situados en el frame de la izquierda surtieran efecto no en él mismo, sino en otro frame, teníamos que poner dentro de cada enlace el atributo **TARGET="principal"**, siendo **"principal"** el nombre que habíamos dado al segundo frame, en el documento de definición de frames. Es decir, hemos utilizado este atributo de esta manera:

TARGET="*nombre_dado_a_otro_frame*".

Estos nombres, que los escogemos nosotros, pueden ser cualquiera, pero con la condición que el primer carácter sea alfanumérico (letra o número).

Pero hay unos **nombres reservados** (es decir, que no se pueden usar para denominar a un frame), que hacen que este atributo efectúe unas **funciones especiales**. Para que cumplan su cometido, es imprescindible escribir estas palabras reservadas (blank, self y top) en **minúsculas**.

TARGET="_blank". Hace que se abra una **nueva copia** del navegador, y el enlace activado se carga en ella, a pantalla completa. Es decir, tendríamos dos copias del navegador (Netscape, Explorer, etc.) funcionando a la vez.

TARGET="_self". Hace que el enlace se cargue en el propio frame.

TARGET="_top". Hace que el enlace se cargue a pantalla completa, suprimiendo todos los frames, pero sin que se cargue una nueva copia del navegador. Este es **particularmente útil**. Un error muy común es olvidarse de poner este atributo en los enlaces que están en un frame, con lo que resulta que al ser activados, la página llamada se carga dentro del propio frame, lo cual es muy molesto si esa página pertenece a otro sitio del Web, y aún más grave si esa página tiene a su vez frames. Este inconveniente se evita poniendo este atributo dentro de las etiquetas de los enlaces.

Frames anidados dentro de otros frames

Hasta ahora hemos contemplado sólo la posibilidad de tener una distribución de los frames bien en filas o bien en columnas, (dependiendo que se utilice el atributo **ROWS** o **COLS** en la etiqueta **FRAMESET**), pero no ambos a la vez.

Se pueden obtener distribuciones más complejas **anidando** filas dentro de una columna, o a la inversa, columnas dentro de una fila.

Supongamos que queremos la siguiente distribución:

- Un frame estrecho en horizontal en la parte superior, de lado a lado de la pantalla (altura, el 15%)
- Otro frame estrecho en vertical en la parte izquierda, debajo del anterior (anchura, el 20%)
- Un tercero ocupando el resto de la pantalla.

Vemos que, en realidad, esto equivale a la siguiente distribución:

- Dos filas. La superior ocupa el 15% y la inferior el resto.
- La fila inferior está a su vez dividida en dos columnas. La primera (la de la izquierda) ocupa el 20% y la otra, el resto.

El documento de definición de las dos filas (olvidémonos de momento que la de abajo está subdividida), sería:

```
<HTML>
<HEAD>
<TITLE>P&aacute;gina con dos filas</TITLE>
</HEAD>
<FRAMESET ROWS="15%, *">
<FRAME SRC="documento_fila_superior">
<FRAME SRC="documento_fila_inferior">
</FRAMESET>
</HTML>
```

Como la fila inferior, en realidad, son dos columnas (con una distribución del 20% y resto), sustituimos (**anidando**) la etiqueta **<FRAME SRC="documento_fila_inferior">** por:

```
<FRAMESET COLS="20%, *">
<FRAME SRC="documento_columna_izqda">
<FRAME SRC="documento_columna_dcha">
</FRAMESET>
```

Con lo que queda el documento definitvo así:

```
<HTML>
<HEAD>
<TITLE>P&aacute;gina con fila superior y dos columnas inferiores</TITLE>
</HEAD>
<FRAMESET ROWS="15%, *">
<FRAME SRC="documento_fila_superior">
    <FRAMESET COLS="20%, *">
    <FRAME SRC="documento_columna_izqda">
    <FRAME SRC="documento_columna_dcha">
    </FRAMESET>
</FRAMESET>
</HTML>
```

Actividades:

1.- Crea una página con tres frames, el primero estará en la parte izquierda con el 20%, y los otros dos en la parte derecha uno encima del otro con el 50% cada uno. El frame de la izquierda contendrá un índice con cuatro opciones. Las opciones pares se cargaran en el frame de la derecha superior, y las opciones impares en el frame de la derecha inferior.

18.- Inserción de Sonidos.

(Bgsound, embed, loop, ahref, src)

Una página del Web puede tener sonidos incorporados, bien sea como un **fondo sonoro** que se ejecuta automáticamente al cargar la página, o como una **opción** para que la active el propio usuario.

Capacidades sonoras de los navegadores

Para poder escuchar los sonidos es necesario disponer, como es lógico, de una tarjeta de sonido con sus correspondientes altavoces. Pero esto no es suficiente, pues no todos los programas navegadores están capacitados en la misma medida.

Explorer de Microsoft

Es el que está mejor adaptado para el sonido, pues a partir de la versión 2.0 es capaz de reproducir fondos sonoros sin necesidad de añadir nada, y no hay ninguna complicación con los servidores, como ocurre con el Netscape. Además, a partir de la versión 3.0 del Explorer, es incluso compatible con los *plug-ins* del Netscape.

Netscape

- Las versiones anteriores a la 2.0 no son capaces de reproducir fondos sonoros que se ejecuten automáticamente, sino que requerirá que se activen los programas auxiliares asociados a los formatos .wav o .mid.
- La versión 2.0 sí es capaz de reproducir un fondo sonoro, pero es necesario que tenga instalado un *plug-in* llamado Crescendo que se obtiene en LiveUpdate *(Una vez obtenido hay que instalarlo manualmente en el directorio 'plugins' del Netscape 2.0)*.
- La versión 3.0 lleva implícito el *plug-in* Live Audio *(pero sólo la versión completa, no la reducida)*. En caso afirmativo, es capaz de reproducir un fondo sonoro.

  Pero a todas estas complicaciones de las distintas versiones de los **navegadores** de los usuarios, hay otra más y es que el **servidor** donde esté alojada la página del Web debe tener **configurados como MIME** los formatos .mid y .wav. Si esto no es así, aunque depositemos en el servidor nuestro documento HTLM acompañado por el correspondiente fichero de sonido, éste no se ejecutará. En dicho caso, es necesario ponerse en contacto con los administradores del servidor para que configuren como MIME los formatos .mid y .wav.

Este problema no existe, sin embargo para el Explorer, con el que no nos tenemos que preocupar por este tema.

Hemos visto hasta aquí las distintas capacidades de los navegadores. Ahora vamos a ver las **etiquetas** que debemos poner en nuestro documento HTML para que se puedan oir los sonidos en uno y otro navegador, y que lamentablemente no son iguales, pero que se pueden combinar, como veremos más adelante.

Fondo sonoro para el Microsoft Internet Explorer

Para las versiones 2.0 en adelante, se utiliza la etiqueta:

<BGSOUND SRC="*fichero_de_sonido*" LOOP=”n”>

El fichero de sonido puede estar en formato .mid o .wav.

El atributo **LOOP** (en inglés, lazo) sirve para especificar el **número** (n) de veces que se debe ejecutar el fichero de sonido. Si se escoje el número **n=”-1”** o se pone **LOOP=”infinite”**, el sonido se ejecutará indefinidamente. Se puede omitir este atributo, y entonces el fichero se ejecutará una sola vez.

Ejemplo 1

La etiqueta para que se ejecute el fichero **canon.mid** dos veces en el Explorer es:

<BGSOUND SRC="canon.mid" LOOP=”2”>

Fondo sonoro para el Netscape

La etiqueta básica para el Netscape es:

<EMBED SRC="*fichero_de_sonido*" WIDTH=”xxx” HEIGHT=”yy”>

donde **WIDTH** es la anchura y **HEIGHT** la altura de una **consola** que aparece, y que tiene diferentes teclas (play, stop, pausa, etc.). Más adelante veremos los valores **xxx** e **yy** que debemos atribuirles.

El fichero de sonido puede estar en **formato** .mid o .wav, pero recuérdese la advertencia hecha anteriormente, de que estos formatos deben estar configurados como MIME por el **servidor** donde esté alojada la página.

Dentro de la etiqueta se pueden añadir los siguientes **atributos** opcionales:
AUTOSTART="true" (arranca automáticamente)
LOOP="true" (se ejecuta ininterrumpidamente)

Ejemplo 2

Vamos a ver una etiqueta para que aparezca en el Netscape una **consola** con sus teclas. Pulsando la tecla *play* se ejecutará el fichero canon.mid.

Con respecto a las **dimensiones** de la consola, tenemos el problema de que son distintas para los diferentes *plug-ins* que existen para el Netscape, y no sabemos cuál de ellos estarán usando los distintos usuarios de nuestra página. En este ejemplo se van a poner las dimensiones de la consola del *plug-in* Crescendo del Netscape 2.0 (200 de ancho por 55 de alto), pero es posible que si se está utilizando otro distinto se vean zonas en blanco.

La etiqueta queda de esta manera:

<EMBED SRC="canon.mid" WIDTH=”200” HEIGHT=”55”>

Ejemplo 3

Vamos a crear una etiqueta para el Netscape, en la que la consola sea **invisible** y el fichero se ejecute **automáticamente** al cargar la página.

Para conseguir que la consola sea invisible hay que añadirle el atributo **HIDDEN="true"**.

<EMBED SRC=" canon.mid " HIDDEN="true">

Fondo sonoro combinado para el Explorer y el Netscape

Se pueden **combinar** los dos tipos distintos de etiquetas para conseguir que un fondo sonoro sea escuchado por usuarios que utilicen tanto el Explorer como el Netscape (siempre que éste último esté preparado para ello). En este caso, el Explorer **ignorará** la etiqueta del Netscape, y a la inversa.

Ejemplo 4

Vamos a ver las dos etiquetas necesarias para que se ejecute el fichero prv89.mid como sonido de fondo, tanto por el Explorer como por el Netscape (lo hará **una sola vez**):

<BGSOUND SRC=" canon.mid ">
<EMBED SRC=" canon.mid " HIDDEN="true">

Ejemplo 5

Ahora vamos a las etiquetas para que el fichero prv89.mid se ejecute como sonido de fondo, tanto por el Explorer como por el Netscape y que se ejecute **indefinidamente**.

A la etiqueta del Explorer basta con añadirle el atributo **LOOP=”infinite”**, como se ha visto anteriormente.

Y con respecto a la etiqueta del Netscape, teóricamente debería también bastar añadirle el atributo **LOOP="true"**, pero curiosamente esto no es así, porque hay que poner las dimensiones de la consola (que no se va a ver).

Las dos etiquetas quedan de esta manera:

<BGSOUND SRC=" canon.mid " LOOP=”infinite”>
<EMBED SRC=" canon.mid " WIDTH=”200” HEIGHT=”55” AUTOSTART="true" LOOP="true" HIDDEN="true">

Activación del sonido por el propio usuario

Hasta aquí hemos visto cómo poner un sonido de fondo en una página. Hay otra opción, mucho más sencilla, y es la de poner un **enlace** a un fichero de sonido, de tal manera,

que al pulsarlo se ejecute el fichero. *(Ver el Capítulo 8, en el que se explica cómo crear enlaces)*.

Por ejemplo, vamos a poner un enlace al fichero canon.mid:
Escucha esta <A HREF="canon.mid">música</A>

Que resulta como: Escucha esta música

Al pulsar el enlace se activa, en una ventana aparte, el programa que ejecuta el sonido.
Esto es válido para **todos** los navegadores, incluso las versiones más antiguas, con la única condición de que se haya configurado un **programa auxiliar** capaz de ejecutar ficheros .mid o .wav.

Se puede hacer el enlace con un **icono** *(Ver el Capítulo 9)*

Ejemplo 6

Vamos a hacer un enlace con el icono **sound1.gif** al fichero de sonido **hola.wav**:
<A HREF="hola.wav"><IMG SRC="sound1.gif"></A>
Que resulta: 

Actividades:

1. Inserta como sonido de fondo en el interior de la página de html pagina12.html el sonido canon.mid.
2. Crea un enlace en la página pagina12.html con una imagen de un disco de música de manera que cuando se pulse diga hola para ello utiliza el fichero hola.wav

19.- Inserción de JavaScripts.
(script, language)

JavaScript, es un lenguaje de programación, el cual se ejecuta en nuestra máquina en el momento de ser cargado por el navegador. Este lenguaje se puede emplear para realizar pequeños programas que serán ejecutados por todos los navegadores.

Recordemos que la estructura de un documento HTML es:

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

El *script* (documento) en Javascript lo tenemos que **colocar** dentro de la cabecera,

después del título. Es decir, entre las etiquetas **</TITLE>** y **</HEAD>**, contenido dentro de la etiqueta:

<SCRIPT LANGUAGE="JavaScript"> </SCRIPT >

Es decir, que queda de esta manera:

```
<HTML>
 <HEAD>
  <TITLE>
  </TITLE>
  <SCRIPT LANGUAGE="JavaScript">
   [Aquí debe ir colocado el script]
  </SCRIPT >
 </HEAD>
 <BODY>
 </BODY>
</HTML>
```

Además de esto, se debe **añadir** algo dentro de la etiqueta **<BODY>**, como se indicará en cada caso.

Ejemplo: Scroll en la barra de estado

El **script** es el siguiente:

```
<SCRIPT LANGUAGE="JavaScript">
<!--
///   Scroll en la barra de estado, ©PortalESO software.
var texto="Esta es una línea de texto que se desplaza "
 + " y esta es otra, puedes poner todas las"
 + " que quieras !                    ";

function scroll()
{
  window.status = texto;
  texto = texto.substring(1, texto.length) + texto.charAt(0);
  window.setTimeout("scroll()",160);
}
//-->
</SCRIPT>
```

Observación sobre el texto**: Se puede poner el texto en una sola línea, tan larga como se quiera, siempre que esté entre **"** y **";

*Pero como no debe de haber **ningún salto de línea**, si el texto es demasiado largo, es mejor dividirlo en distintas líneas, tal como está en el ejemplo.*
Al final de la última palabra del texto conviene dejar una serie de espacios en blanco, para que no esté encadenado el comienzo con el final.

Dentro de la etiqueta **<BODY>** se debe añadir el atributo:
onLoad="scroll();"

quedando así:

<BODY onLoad="scroll();">
Este es el resultado.

Scroll en una ventana

Ahora vamos a ver cómo conseguir que se desplace un texto dentro de una ventana de un formulario. Con esto se consigue un efecto parecido al de las **marquesinas** (pero en este caso sí es implementado por el Netscape 2.0 o superior).

El **script** es el siguiente:

```
<SCRIPT LANGUAGE="JavaScript">
<!--
///   Scroll en una ventana, ©PortalESO software.
var texto="Esta es una línea de texto que se desplaza "
 + " y esta es otra, puedes poner todas las"
 + " que quieras !                    ";

function scroll()
{
  document.frm.w.value = texto;
  texto = texto.substring(1, texto.length) + texto.charAt(0);
  window.setTimeout("scroll()",160);
}
//-->
</SCRIPT>
```

Dentro de la etiqueta **<BODY>** se debe añadir el atributo:
onLoad="scroll();"

quedando así:
<BODY onLoad="scroll();">

Las etiquetas para la **ventana** del formulario son:

```
<FORM NAME='frm'>
 <INPUT TYPE='text' NAME='w' SIZE=64>
</FORM>
```

Como la ventana está situada dentro de la **página**, debemos colocar estas etiquetas en el sitio que nos convenga, dentro de la seción **<BODY> </BODY>**, mezclado con el resto del documento HTML.

Este es el resultado.

20.- Inserción de Applets de Java

(Applet, code, width, height, param, name, value, codebase)

Java™ es un lenguaje de programación orientado a objetos que ha sido desarrollado por la compañía Sun Microsystems, basándose en el lenguaje C++. Aunque es también un lenguaje de carácter general, su principal característica es la de ser **independiente** de cualquier plataforma, lo que le hace ser muy adecuado para ser utilizado en **Internet**, ya que puede ejecutarse en cualquier ordenador que tenga un navegador compatible con Java.

Esto quiere decir que el mismo programa se puede utilizar, sin ningún cambio, en un PC, Mac o máquina Unix. Esto es así porque un programa Java no corre directamente en el ordenador, sino que lo hace en una **máquina virtual** (la Máquina Virtual Java), quedando además confinado dentro de ella, lo que hace que sea un lenguaje **seguro**, ya que no puede acceder a otros recursos de nuestro ordenador.

Aunque el lenguaje Java es más fácil que el C++, en el que está basado, se trata de un lenguaje de programación de alto nivel, y por tanto es muchísimo más difícil (para quien no tiene conocimientos previos) que el **HTML**, que en realidad no es más que una sencilla manera de formatear el texto de las páginas del Web.

Pero una cosa es **programar** en Java y otra mucho más sencilla es **utilizar** programas ya existentes, como veremos. Estos programas se llaman **applets**, y se pueden incrustar dentro de una página del Web para realizar todo tipo de tareas.

El propósito de este capítulo es mostrar cómo hacer uso de estos programas y dónde encontrarlos, y no el de aprender a programarlos.

¿Qué son los applets?

Los applets son pequeños programas ejecutables escritos en lenguaje Java, que podemos colocar en nuestro **servidor**, junto con el resto de ficheros que componen un sitio del Web (documentos HTML, ficheros de imagen, sonido, etc.)

Si en una de nuestras páginas hemos **incrustado** un applet (de la manera que se verá más adelante), cuando un usuario cargue la página, recibirá del servidor dicho applet (al igual que ocurre con una imagen o sonido), y se ejecutará en su navegador.

Pero para que esto ocurra así, es condición indispensable que el usuario esté utilizando un navegador **compatible** con Java: **Netscape 2.x** o superior, **Explorer 3.0** o superior, o **HotJava** (navegador desarrollado por Sun, y escrito totalmente en Java). Quien utilice un navegador no compatible, no verá la ejecución del programa (ni siquiera lo cargará), aunque podemos informarle de este hecho con un mensaje.

¿Para qué sirven los applets?

Con los applets se pueden conseguir efectos visuales y sonoros (incluso ambos a la vez), textos en movimiento, utilidades (por ej., relojes), pequeños programas educativos, juegos interaractivos, presentaciones multimedia, etc. En definitiva, cualquier cosa que se pueda conseguir con un programa pequeño.

No es que haya ninguna limitación teórica en el tamaño de estos programas. La razón de que sean pequeños es la misma por la que no es razonable colocar en una página gigantescos ficheros de imagen o sonido, es decir, no hay que alargar en exceso su tiempo de carga.

¿Cómo se coloca un applet en una página?

Lo primero de todo, es necesario tener el **programa** del applet. Un ejemplo de applet que vamos a examinar a continuación con detalle, se llama **SineText.class**

Se trata de un applet que hace que un texto (elegido por el usuario) se desplace formando una onda sinusoidal, en dirección reversible con un click del ratón. Se pueden escoger los colores del fondo y de las letras, elegir entre algunos tipos de fuentes, su tamaño, velocidad de desplazamiento, etc.

Su autor es Mattias Tollet y proviene de The Java(TM) Boutique
Para conseguirlo ahora mismo, pulsa el enlace:


SineText.class (4,6 kb)

Algunas observaciones:

- Estos programas tienen la extensión **.class**
- Hay que respetar escrupulosamente la disposición de las mayúsculas y minúsculas en su nombre (en este caso se debe poner **SineText.class** y no **sinetext.class**, por ejemplo).
- Es un verdadero programa binario, y por tanto no se puede leer en un procesador de textos, como podemos hacer con un documento HTML.

Para colocar el applet en la página, debemos utilizar las siguientes **etiquetas** (que se colocan donde queramos dentro del cuerpo del documento HTML, es decir entre las etiquetas **<BODY>** y **</BODY>**):

<APPLET CODE=SineText.class WIDTH=”500” HEIGHT=”100”>
</APPLET>

Con esto se le indica al navegador que cargue un applet, cuyo código compilado es **SineText.class**, que se encuentra en el mismo directorio que el propio documento HTML, y que le reserve un área de 500x100 pixels.

Pero en este caso concreto, nos falta añadir los **parámetros** que **personalizan** el applet a nuestro gusto (sobre todo el texto que queremos que aparezca). Para ello se añaden etiquetas de este tipo:

<PARAM NAME=*nombre_del_parámetro* VALUE="*contenido_del_parámetro*">

Para escoger el texto, será, por ejemplo:

<PARAM NAME=Text VALUE="¡Curso para realizar una web!">

De momento no vamos a definir más parámetros. El resto de ellos serán los que ha escogido su autor por defecto. Las etiquetas del applet quedan de momento así:

```
<APPLET CODE=SineText.class WIDTH=”500” HEIGHT=”100”>
<PARAM NAME=Text VALUE="¡Curso para realizar una web!">
</APPLET>
```

Pulsando aquí se puede ver el resultado: sinetext.

¿Cómo sabemos qué parámetros hay, cuáles son sus nombres y cuáles son los que hay por defecto? Pues sencillamente, leyendo la **documentación** del applet, ya que son variables que se le han ocurrido al autor al confeccionar el programa. Esto quiere decir que para otro applet distinto los parámetros serán también distintos.

En este applet, su autor ha definido los siguientes parámetros:

Text - Texto que se va a ver.
Rate - Número de imágenes por segundo.
Font - Fuente: Helvetica (defecto), TimesRoman o Courier.
FontSize - Tamaño de la fuente.
TextStyle - Estilo de la fuente: Plain, Bold (defecto) o Italic.
Speed - Velocidad de desplazamiento 1-10 (5 es por defecto)
Direction - Dirección inicial del desplazamiento. Left (defecto) o Right.
Background - Color del fondo. 000000 (negro) es por defecto.
ForeGround - Color del texto. FF0000 (rojo) es por defecto.

Vamos a añadir algunos parámetros al ejemplo anterior (el orden en el que se colocan es indiferente). Variaremos también el área del applet a 600x75, y el texto del mensaje.

```
<APPLET CODE=SineText.class WIDTH=”600” HEIGHT=”75”>
<PARAM NAME=Text VALUE="¡Curso para realizar una web!">
<PARAM NAME=Rate VALUE="4">
<PARAM NAME=Font VALUE="Arial">
<PARAM NAME=Background VALUE="FFFFFF">
<PARAM NAME=Foreground VALUE="0000FF">
</APPLET>
```

Pulsando aquí se puede ver el resultado: sinetext2.

Mensaje para los navegadores no compatibles con Java

En los dos últimos ejemplos, el usuario con un navegador no compatible no verá absolutamente nada, ya que el applet no se carga ni ejecuta.

Se puede poner un **mensaje** para que lo lean los que usan un navegador no compatible, y que sin embargo, es invisible para los que sí están viendo ejecutarse el applet. No hay más que añadirlo, sin ningún tipo de etiqueta, a continuación de los parámetros. Por ejemplo:

```
<APPLET CODE=SineText.class WIDTH=”600” HEIGHT=”75”>
<PARAM NAME=Text VALUE="¡Curso para realizar una web!">
<PARAM NAME=Rate VALUE="4">
<PARAM NAME=Font VALUE="Arial">
<PARAM NAME=Background VALUE="FFFFFF">
<PARAM NAME=Foreground VALUE="0000FF">
Lo siento, pero no puedes ver este applet funcionando.
</APPLET>
```

Quien use un navegador compatible sólo podrá ver el mensaje si inhabilita la carga de programas de Java.

Otro ejemplo: reloj

Hay veces que el programa del applet necesita unos **ficheros adicionales** para funcionar (generalmente ficheros de imagen o sonido). Es el caso del ejemplo que vamos a ver, que es un reloj que necesita las **imágenes** de los dígitos.

Por tanto, aparte de instalar el fichero del applet, que en este caso se llama **DigClk.class**, hay que instalar los ficheros de imagen de los dígitos en un subdirectorio que se debe llamar necesariamente **dcimages** (porque así lo ha dispuesto el programador del applet).

Su autor es William G. Ogle Jr. y proviene de The Java(TM) Boutique.
Puedes obtener el applet **DigClk.class** y los ficheros de imagen agrupados en su subdirectorio **dcimages**, comprimidos en un fichero .zip:


dclk100.zip (5 kb)

El código HTML necesario para el applet es:

```
<APPLET CODE="DigClk.class" WIDTH=”122” HEIGHT=”29”>
Aqu&iacute; habr&iacute;a un reloj si tu navegador soportase Java
</APPLET>
```

Pulsando aquí se puede ver el resultado: reloj.html.

El atributo CODEBASE

En todos los ejemplos anteriores hemos supuesto que el fichero del applet (el que tiene la extensión **.class**), está en el mismo **directorio** que el documento HTML donde queremos incrustarlo. Pero puede ser que lo hayamos colocado en otro directorio distinto.

En estos casos hay que añadir el atributo **CODEBASE** dentro de la etiqueta **<APPLET>** indicando el directorio donde se encuentra.

Si en el último ejemplo hubiéramos colocado el fichero **DigClk.class** en un subdirectorio llamado **java** (por ejemplo), la etiqueta hubiera quedado así:

<APPLET CODEBASE=”java” CODE="DigClk.class" WIDTH=”122” HEIGHT=”29”>
[*el resto de las etiquetas quedarían igual*]

¿Donde se pueden conseguir applets de Java?

En español:
Catálogo de Applets (http://sunsite.unam.mx/applets/) (México)

En inglés:

The Java(TM) Boutique (http://redir.internet.com/java/) en donde se pueden encontrar más de cien applets, distribuidos en categorías, y con con sus correspondientes instrucciones.

The Java Place (http://www.thejavaplace.com/) Similar al anterior, con mas de cien applets.

Otro sitio es Gamelan (http://www.developer.com/java/), que es un catálogo de recursos muy extenso.

Actividades

1.- Busca el Applet, anuncio.class e insértalo en tú pagina. Si deseas ver un ejemplo de su funcionamiento pulsa: anuncio.html.

Dispone de los siguientes parámetros:

```
<PARAM NAME="width" VALUE="490">
<PARAM NAME="height" VALUE="25">
<PARAM NAME="bgcolor" VALUE="fef1ce">
<PARAM NAME="border_color" VALUE="0000a2">
<PARAM NAME="cabecera_color" VALUE="0000a2">
<PARAM NAME="mensaje_color" VALUE="000000">
<PARAM NAME="highlight_cabecera_color" VALUE="d20000">
```

```
   <PARAM NAME="highlight_mensaje_color" VALUE="d20000">
   <PARAM NAME="cabecera_font" VALUE="Arial,1,14">
   <PARAM NAME="mensaje_font" VALUE="Arial,0,12">
   <PARAM NAME="text_alignment" VALUE="center">
   <PARAM NAME="left_margin" VALUE="10">
   <PARAM NAME="base_line_ypos" VALUE="18">
   <PARAM NAME="xspace" VALUE="10">
   <PARAM NAME="scroll_delay" VALUE="60">
   <PARAM NAME="pause" VALUE="5000">
   <PARAM NAME="target_frame" VALUE="_parent">
<!--  Lineas y cabeceras de los anuncios  -->
   <PARAM NAME="cabecera1" VALUE="¡HOLA A TODOS!">
   <PARAM NAME="mensaje1" VALUE="Bienvenidos otra vez">
   <PARAM NAME="link1" VALUE="">
   <PARAM NAME="status_bar_msg1" VALUE="¡HOLA A TODOS!">
   <PARAM NAME="cabecera2" VALUE="WEB DE ENSEÑANZA">
   <PARAM NAME="mensaje2" VALUE=" www.portaleso.com">
   <PARAM NAME="link2" VALUE="http://www.portaleso.com">
   <PARAM NAME="status_bar_msg2" VALUE="Visita la web ## PortalESO ##">
<!--  Fin de las lineas y las cabeceras  -->
```

Tiene la posibilidad de crear enlaces en las distintas líneas, y se pueden poner varias líneas. Modifica los valores de los parámetros y observa como se comporta.

21.- Inserción de películas Flash

(Embed, scr, width, height, type, quality, bgcolor, scale, border, align, noembed, div)

Flash es tanto una herramienta de dibujo como un editor de películas. Flash, es muy versátil. Te permite crear animaciones interactivas, y añadir sonido a tus páginas. También se pueden crear películas e interfaces insertadas en páginas Web que ocupan pequeña cantidad de Bytes de 5 KB a 50 KB.

Otra gran ventaja de Flash es que es virtualmente independiente del navegador. Los archivos de Flash solo se pueden ver con un plug-in, todos los que tengan una versión 3 o mayor, pueden ver las películas casi idénticas. El lado malo de esto es el plug-in, los usuarios necesitan cargar e instalarlo antes de que puedan ver cualquier cosa. Pero esto a perdido importancia desde que muchos de los últimos navegadores vienen con Flash ya incluido.

De cualquier modo, Flash se está viendo como una gran solución y es más popular cada día.

¿Cómo se coloca una película Flash en una página?

Lo primero de todo, es necesario tener la película Flash. Un ejemplo de película que vamos a examinar a continuación, se llama **boton.swf**

Se trata de un banner de promoción de la página web **PortalESO.com**, realizado por Antonio Bueno.

Para colocar la película en la página, debemos utilizar las siguientes **etiquetas** (que se colocan donde queramos dentro del cuerpo del documento HTML, es decir entre las etiquetas **<BODY>** y **</BODY>**):

```
< EMBED SRC="boton.swf"  WIDTH=”100” HEIGHT=”36” TYPE="application/x-shockwave-flash">
</ EMBED>
```

Con esto se le indica al navegador que cargue una película, cuyo código compilado es **boton.swf**, que se encuentra en el mismo directorio que el propio documento HTML, que le reserve un área de 100x36 pixels se le indica que se trata de una aplicación x-shockwave-flash.

Pero en este caso concreto, nos falta añadir los **parámetros** que **personalizan** la película a nuestro gusto. Para ello se añaden más parámetros:

```
< EMBED SRC="boton.swf"  WIDTH=”100” HEIGHT=”36” TYPE="application/x-shockwave-flash" QUALITY=”high” BGCOLOR=”#FFCC99” SCALE="exactfit" BORDER="0" ALIGN="center">
</ EMBED>
```

Donde:
QUALITY, indica la calidad de la imagen que queremos.
BGCOLOR, el color de fondo que queremos.
SCALE, indica la escala a la que queremos que se reproduzca.
BORDER, indica el grosor del borde.
ALIGN, indica el lugar de alineación.

Aún así puede ocurrir que no tengamos instalado el plug-in de Flash, se puede insertar un enlace de búsqueda automática del plug-in.

```
PLUGINSPAGE="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash"
```

Y a demás podemos colocar un mensaje que se mostrará en caso de no poderse ver la película Flash.

```
<NOEMBED>
 <DIV align="center">Si lee este mensaje es porque necesita instalar el complemento de Macromedia Flash 5.0</DIV>
</NOEMBED>
```

De manera que quedaría así:

```
< EMBED SRC="boton.swf"  WIDTH=”100” HEIGHT=”36” TYPE="application/x-shockwave-flash" QUALITY=”high” BGCOLOR=”#FFCC99” SCALE="exactfit" BORDER="0" ALIGN="center" PLUGINSPAGE="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash">
</EMBED>
<NOEMBED>
 <DIV align="center">Si lee este mensaje es porque necesita instalar el complemento de Macromedia Flash 5.0</DIV>
</NOEMBED>
```

Por último nos queda colocar un enlace alternativo, para aquellos casos que no puedan acceder al enlace facilitado a través del banner.

```
<DIV align="center"><a href="http://www.portaleso.com">PortalESO.com</a></DIV>
```

Pulsando aquí se puede ver el resultado: flash.

Actividades

1.- Realiza una página web completa que incluya el banner de **PortalESO**, así como el resto de opciones anteriores.

2.- Modifica los parámetros de la película Flash, y observa como varían.

22.- Publicación de una Web

El propósito que tenemos al confeccionar una página del Web es, por supuesto, el de publicarla y ponerla a disposición de todo el mundo.

No podemos colocarla directamente en la red, pues necesitaríamos disponer para nosotros solos de un **servidor**, es decir, un ordenador conectado permanentemente con Internet y dotado de un programa específico, lo que supone una inversión económica y unos elevados conocimientos técnicos.

La solución es instalarla en un servidor conectado a la **WWW** (World Wide Web: *la telaraña mundial*), la red de servidores interconectados entre sí que nos permite acceder a cualquier página en cualquier parte del mundo, y navegar a través de ellas.

Los distintos canales para colocar una página en la red son:

- Los servidores de las **instituciones oficiales y académicas** a sus miembros, para la difusión de información relacionada con esos organismos.
- Los **proveedores comerciales de acceso a Internet**, o ISPs (Internet Service Providers) que ceden frecuentemente a sus usuarios un espacio determinado en sus servidores, para que puedan colocar sus páginas personales, bien sea gratuitamente o por una tarifa determinada.

- Las **empresas comerciales** que, sin ser proveedores de acceso a Internet, se dedican a alquilar espacio para la colocación de páginas, de carácter personal o comercial.
- Ciertos servidores que conceden de una manera **gratuita** espacio para la colocación de páginas personales. Uno de los más populares es Geocities, que concede un espacio de 2 megas. Sus páginas explicativas están en inglés.

¿Cómo se envían las páginas al servidor?

Una vez de haber confeccionado en nuestro disco duro la página Web, y estando ya lista para ser colocada en el servidor, surge la cuestión de cómo proceder para **enviarla**.

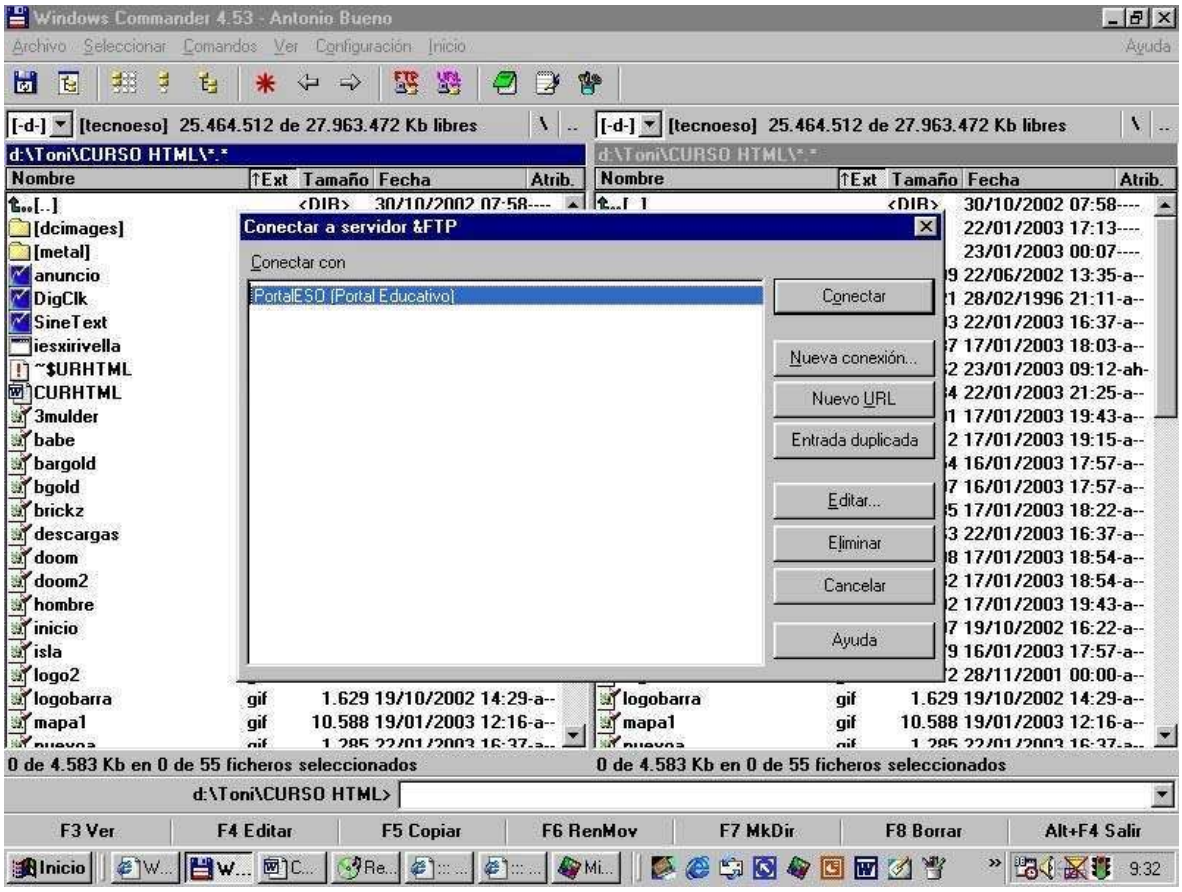
No se pueden dar aquí unas instrucciones precisas, pues depende de la manera establecida para ello por cada servidor. Generalmente se hace utilizando un programa de **FTP** (programa de envío y recepción de ficheros). Pero antes habrá que consultar en las páginas del servidor si existen las instrucciones pertinentes para el envío, o en su defecto realizar la consulta por email o por teléfono.

En ocasiones es tan sencillo como copiar los ficheros de nuestro disco duro (HD) al disco duro del servidor.

Por ejemplo si utilizamos el **Windows Commander**, es necesario configurar una sesión FTP.

Para ello seleccionamos la opción **FTP**.

A continuación seleccionamos nueva: **Nueva conexión**.



E introducimos los parámetros que configuran la nueva conexión.

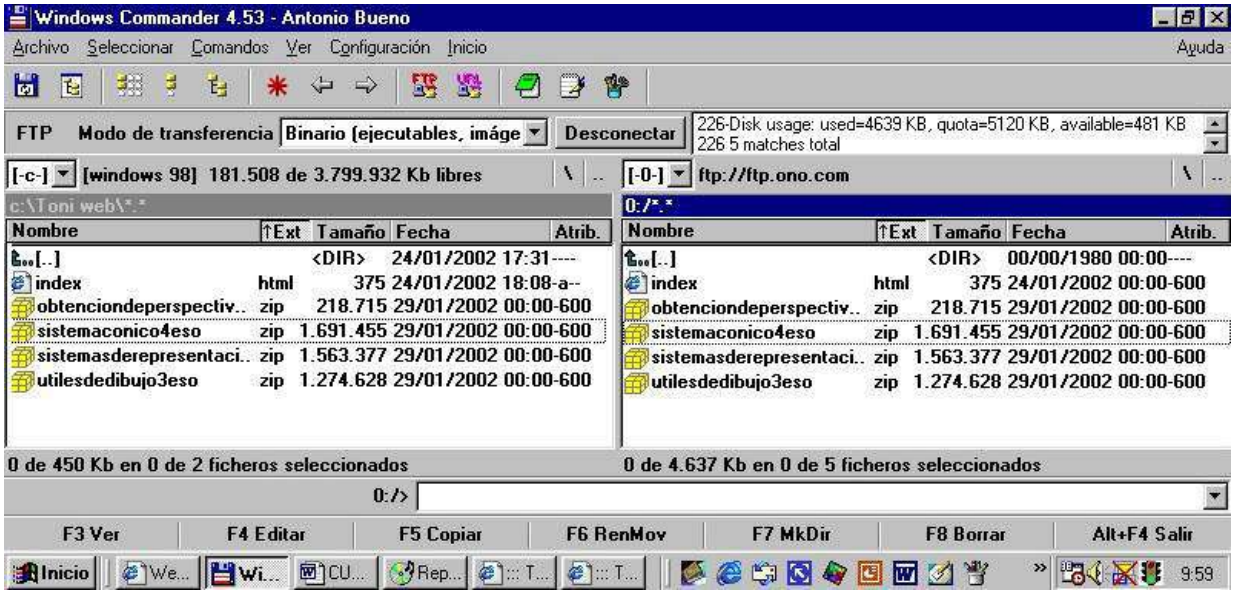




Una vez configurada la conexión seleccionamos **Conectar**, y la conexión se habrá producido.

Para **enviar** (subir) nuestra página Web hasta el servidor será suficiente con seleccionar los ficheros y arrastrarlos hasta el directorio del servidor.

Para **traernos** (bajar) los ficheros del servidor, también es suficiente con seleccionarlos desde el servidor y arrastrarlos hasta el directorio de nuestro PC.



23.- Otras cosas.

Promoción de una página

No basta con colocar nuestra página en la red. Por muy interesante que sea su contenido, y por bien diseñada que esté, para que los demás puedan acceder a ella tienen que tener conocimiento de su existencia. Es una labor nuestra la de **promocionar** la página, es decir, darla a conocer por todos los medios posibles.

Para comenzar, es conveniente incluir la dirección (o URL) de la página en la **firma** de nuestro programa de *email*, así como también en el de lectura de *newsgroups*, y mejor aún si además se incluye su título, o una frase que indique el contenido de la página. De esta manera se incita a visitarla al que esté interesado en ese tema.

Pero el paso más eficaz es **dar de alta** a nuestra página en sitios del Web especializados en almacenar y organizar direcciones. Estos sitios sirven como bases de datos a donde acude la gente en búsqueda de información sobre dónde encontrar las páginas del Web deseadas. Además, dar de alta una página en los buscadores es totalmente **gratuito**.

En líneas generales, se pueden distinguir dos tipos de estos sitios:

- Los que están organizados como **directorios**, es decir, que catalogan las páginas por su contenido en categorías y sub-categorías. Para darse de alta en ellos, es necesario situarse primero en la categoría apropiada al contenido de nuestra página. Estos sitios sólo contienen las páginas de quienes se hayan dado de alta

en ellos de manera voluntaria.

- Otros sitios, los llamados **motores de búsqueda** (*search engines*) actúan de una forma completamente distinta. Utilizan unos programas (llamados comúnmente *robots* o *arañas*) que tienen la misión de rastrear continuamente el Web en búsqueda de páginas nuevas o renovadas. Para ello, van navegando de URL en URL a través de los enlaces que encuentran en las páginas, con la intención de catalogar el número máximo de ellas.

  Si uno de estos robots visita nuestra página del Web, **grabará el texto completo** de cada una de las páginas (la principal y las sub-páginas). De esta manera, todas las palabras de todas las páginas de nuestro sitio son incorporadas a su base de datos. Cuando luego alguien haga una consulta en estos motores de búsqueda introduciendo una palabra que coincida con alguna de ellas, presentará nuestra página del Web como un resultado de la búsqueda.

  Aparte de esto, también se dedicará a visitar todos los enlaces que vaya encontrado por las distintas páginas. Es de esta manera cómo catalogan páginas que no han sido dadas de alta de manera voluntaria en ellos. Sin embargo, es conveniente que registremos nosotros mismos nuestra página en estos motores de búsqueda para **acelerar** el proceso, y no tener que esperar a que la encuentren ellos por medio de enlaces de otras páginas a la nuestra.

  Además, una vez que los robots han localizado un sitio del Web, lo visitarán periódicamente para renovar la información grabada.

Consejos para facilitar la labor a los motores de búsqueda

Hay ciertas cosas que podemos hacer para conseguir que nuestra página sea catalogada de la manera más adecuada en estos motores de búsqueda:

1. Utilizar palabras clave dentro de la etiqueta <TITLE>, haciendo que sea lo más descriptivo posible, porque cuando un motor de búsqueda presenta una página concreta como resultado de una búsqueda, lo hará reproduciendo las palabras que ha encontrado dentro de la etiqueta **<TITLE>** de esa página.
Por ejemplo, en vez de titular una página como **<TITLE> PortalESO </TITLE>**, es más eficaz hacerlo como **<TITLE> PortalESO – Portal Educativo </TITLE>**.

2. Utilizar etiquetas <META>
Son unas etiquetas que se colocan en la cabecera de la página (en la zona entre **<HEAD>** y **</HEAD>**), que sirven para suministrar una información detallada del contenido de una página, con lo que se obtiene un control mayor de cómo será catalogada la página. No todos los motores de búsqueda hacen uso de estas etiquetas, pero si las ponemos, las haremos mucho más accesibles a los motores de búsqueda que sí las utilizan.

Hay diferentes tipos de esta etiqueta, pero las que nos interesan ahora son la que hace referencia a la **descripción** (*description*) de la página y la que presenta las **palabras clave** (*keywords*) con las que la gente buscará una página como la nuestra en los motores de búsqueda.

Veamos por ejemplo las utilizadas para la página www.portaleso.com/ :

<META NAME ="description" CONTENT="Portal educativo para la Enseñanza Secundaria Obligatoria y Bachilleres, con diversos trabajos, exámenes, juegos, enlaces, chat y otros recursos útiles para alumnos y profesores. ">

En este caso, lo que está incluido en el atributo **CONTENT** (contenido) es lo que presentará el motor de búsqueda, además del título de la página.

<META NAME="keywords" CONTENT="Portaleso, Enseñanza, Secundaria, ESO, Bachiller, Oposiciones, Apuntes, Profesores, Alumnos, Recursos Educativos, Educación, Deportes, Móviles, Juego, Chat, Matemáticas, Lengua, Física, Tecnología, Historia, Química, Idiomas, Educación Física, Foro, Biología, Música, Latín Plástica, Electrónica, Mecánica, Electricidad, Boletines Oficiales">

En este otro caso, se incluyen en el atributo **CONTENT** las palabras claves para la búsqueda de nuestra página. Se pueden poner tantas como se crea oportuno, incluso sus plurales.

La utilización de estas etiquetas **META** es especialmente **conveniente** para las páginas que hacen uso de **frames** (ya que la página inicial es la de definición de los frames, que no tiene ninguna indicación del contenido de las otras páginas). También es muy conveniente para los que utilizan **Javascript** en el comienzo de sus páginas, ya que el código empleado puede tener cientos de palabras, y los robots están programados para dar mayor énfasis a las palabras que encuentran al principio que las situadas al final.

3. Poner un resumen del contenido de la página en el comienzo del texto

Hay motores de búsqueda que utilizan **las primeras 25 palabras** del texto de una página a modo de presentación de su contenido. Conviene, por tanto, hacer un breve resumen al inicio de la página, lo que por otra parte es siempre una práctica muy aconsejable.

4. Utilizar el atributo ALT en las imágenes iniciales

Como vimos en el Capítulo 9, dentro de la etiqueta de las imágenes se puede añadir el atributo **ALT** que sirve para poner un texto a la imagen, que será visto únicamente por quienes utilizan los navegadores en forma de sólo texto.

Hay motores de búsqueda que toman en cuenta el texto que encuentran de esta manera en las imágenes (sobre todo las iniciales) para hacer una descripción del sitio o para suministrar las palabras clave.

24.- Punteros, referencias y Bibliografía.

Bibliografía.

Este manual ha sido extraído en su mayor parte de la web Webmaestro (http://www.ldc.usb.ve/~vtheok/webmaestro/) de **F.Arocena**.

No pretende ser un plagio, y el autor tan solo pretende cambiar el formato a fichero de Word, para que pueda ser impreso en papel de forma coherente.

Por otra parte el autor lo ha personalizado en algunos aspectos. Sin embargo reconoce a **F.Arocena** como verdadero autor del curso.

El objeto de este manual es facilitar el aprendizaje del HTML, y la realización de páginas web por parte de profesores y alumnos de la ESO.

RECONOCIMIENTO y AGRADECIMIENTO a F. Arocena por su labor en la divulgación de la tecnología en internet.

Referencias de consulta

Manuales de HTML

- Tejedores del Web:, (http://www.tejedoresdelweb.com/)
- HTMLPOINT.com (http://www.htmlpoint.com/)
- Manual de HTML, (http://www.geocities.com/SiliconValley/2915/manual.htm) por Luis Mariñas.

Manuales de JavaScript

- Curso de JavaScript, (http://www.redestb.es/javaaula/cursjava.htm) curso de introducción en castellano.
- **Netscape**, el creador de este lenguaje, tiene unos completos tutoriales sobre este tema (en inglés):
  - JavaScript 1.5 Guide (http://devedge.netscape.com/library/manuals/2000/javascript/1.5/guide/)
  - JavaScript 1.5 Reference (http://devedge.netscape.com/library/manuals/2000/javascript/1.5/reference/)

Manuales de Java

- Tutorial de Java, (http://usuarios.lycos.es/froufe/) por AgustÌn Froufe
- Lecciones de Java, en formato zip, (http://infotutoriales.iespana.es/infotutoriales/lenguajes/lenguajes.htm) de Bourdette.

Direcciones útiles relacionadas con la creación de páginas Web

- Agenda de direcciones de HTML, (http://www.areas.net/agenda/home.htm) por Manuel Barberán.

- Lanzador Internet, (http://usuarios.iponet.es/lubrin/html.htm) por Daniel López Avellaneda

FAQs (recopilaciones de preguntas frecuentes) sobre creación de páginas Web

- En inglés
  The Web Design Group's Web Authoring FAQ
  (http://www.htmlhelp.com/faq/html/all.html)

Editores de HTML

Para Windows

- En castellano:
  Web Scope, (http://villaquintus.sarovi.com/scope.htm) por Jaime Saz Rodríguez-Vigil

- En inglés:
  HotDog (http://www.sausage.com/)
  HomeSite (http://www.macromedia.com/software/homesite/)
  HoTMetal (http://www.softquad.com/top_frame.sq)
  HTMLed Professional (http://www.ist.ca/)

- *Estos programas, y otros muchos, se pueden encontrar en el excelente Stroud's Consummate Winsock Applications, (http://cws.internet.com/)*

Para Macintosh

En inglés

html-helper-mode for EMACS (http://www.santafe.edu/~nelson/tools/)
HTML Editor (http://dragon.acadiau.ca/~giles/HTML_Editor/Documentation.html/)

Grupos de noticias *(newsgroups)* relacionados con el HTML

- En castellano:
  es.comp.infosistemas.www

- En inglés:
  comp.infosystems.www.authoring.html
  comp.infosystems.www.authoring.site-design

Páginas interesantes para visitar

- PortalESO, (http://www.portaleso.com/)

- http://www.iesrodrigobotet.com/

- http://www.latorretaonline.com/ (página muy buena)

- http://www.pntic.mec.es/centros/index.html (el pntic es una de las mejores páginas)