

## PRESENTACIÓN

Este manual ha sido elaborado por profesores de la Dirección General de Servicios de Cómputo Académico y tiene como propósito servir como material de apoyo para el curso de "Páginas para Internet", no se espera de ninguna manera que los usuarios tengan experiencia alguna en la creación de páginas antes de consultar este texto.

Se presentan desde los conceptos más elementales como: ¿qué es una página?, o ¿qué herramientas necesitamos para crear una página?, hasta cuestiones más avanzadas como el uso de formularios que nos permiten crear páginas un poco más sofisticadas, es importante señalar que para poner en práctica la mayoría de ejemplos que presentan los diversos capítulos, no es necesario tener acceso a Internet.

El desarrollo de los contenidos se ha hecho, en la medida de lo posible, considerando básicamente la parte práctica, se pretende que a través de los distintos ejemplos se asimilen los conceptos que se van presentando a lo largo de cada capítulo.

Se ha hecho un esfuerzo importante por tratar temas que pocas veces se encuentran en otros materiales de consulta, por ejemplo, un capítulo está dedicado a cubrir las consideraciones del diseño de páginas, y en otro se plantea la manera de publicar páginas en Internet.

Siempre se tuvo presente, que aprender a crear páginas no sólo implica aprender el lenguaje HTML, sino también saber utilizar herramientas que nos ayuden a crear páginas de una manera más fácil, es por ello que se dedica también un capítulo para tratar estos temas; se cree, sin embargo, que el manejo básico o elemental del lenguaje HTML nos facilitará la manera de trabajar con dichas herramientas.

## 1. CONCEPTOS GENERALES

### 1.1 Páginas

Una página es un documento electrónico que se utiliza para difundir información en Internet, y en el cual podemos colocar texto, imágenes y sonido, sin que llegue a convertirse en un multimedia.

Las páginas se distribuyen globalmente a través de miles de sitios, de esta forma muchos usuarios pueden ver alguna página desde cualquier parte del mundo.

La utilidad de las páginas es inmensamente amplia, ya que las podemos utilizar como medio educativo, informativo, comercial, etc.; que se pueden presentar de una forma atractiva y dinámica, esto es, la información se puede actualizar con facilidad.

No generan costos de materiales, como los medios impresos, ni por número de copias.

Posiblemente hemos visto en alguna ocasión una página, y hemos notado que al colocar el puntero del ratón sobre alguna palabra subrayada o imagen cambia su apariencia, esto nos indica que podemos ir a otra página que puede estar en cualquier parte del mundo, lo cual se conoce como ligas.

### 1.2 Requerimientos

En realidad necesitamos un mínimo de herramientas que nos permitan generar y visualizar nuestras páginas. La computadora que se utilice puede tener instalado Windows, Macintosh, Unix, etc.

Con cualquier editor de texto se puede crear una página o documento fuente. En cualquier computadora podemos encontrar un editor de texto, como el Bloc de Notas de Windows.

Para que podamos visualizar las páginas que vamos creando y sus cambios necesitamos tener instalado un navegador como Internet Explorer o Netscape Communicator, que se encarga de interpretar el documento fuente.

### **1.3 Lenguaje HTML**

Las páginas se crean utilizando un lenguaje conocido como HTML (Lenguaje para Marcado de Hipertexto).

- *Lenguaje*. No es estrictamente un lenguaje de programación sino un lenguaje para definir (formatear) documentos.
- *Marcas*. Un documento HTML es un archivo de texto cuyo contenido es enriquecido con instrucciones que un programa navegador es capaz de interpretar. Estas instrucciones se introducen mediante marcas conocidas como <etiquetas>.
- *Hipertexto*. Una forma no lineal de leer la información textual. En un hipertexto se pueden enfatizar palabras y frases importantes.

HTML es en realidad una serie de instrucciones que interpreta el navegador para poder presentar una página en la forma adecuada.

Cualquier página consta de: el texto del documento y las etiquetas que indican los elementos, estructura y formato del mismo.

### **1.4 Etiquetas y sus modificadores**

Las etiquetas representan estructuras mediante las que organizaremos el contenido del documento o acciones que se ejecutan cuando el navegador interpreta el documento.

Son órdenes o instrucciones que informan al navegador sobre cómo debe de mostrar el documento.

Para distinguir una etiqueta es necesario observar que las etiquetas se encuentran entre los símbolos "<" ">".

Por lo general veremos en el documento una etiqueta inicial y una final que tienen el mismo nombre sólo que la etiqueta final o de cierre incluye una "/". La etiqueta inicial asigna una característica, y con la etiqueta final la concluye.

Cada etiqueta consta generalmente de tres partes: etiqueta de inicio, contenido y etiqueta final o de cierre.

Ejemplo:

<Nombre de etiqueta>

Contenido

</Nombre de etiqueta>

No todas las etiquetas se utilizan en pares (inicial y final), algunas no tienen etiqueta de cierre.

Otras etiquetas inclusive no tienen contenido: se les llama etiquetas vacías y no deben llevar una etiqueta final.

Las etiquetas pueden llevar asociadas atributos o modificadores. Los modificadores pueden tener valores y se colocan antes del final de la etiqueta de inicio (es decir, antes del signo ">").

Cada etiqueta tiene un número específico de atributos, los cuales pueden utilizarse en cualquier orden, separados por un espacio y la mayoría de ellos de manera opcional.

El nombre del modificador va seguido del signo "=" y del valor del modificador, el cual se delimita por comillas dobles.

Tanto los nombres de las etiquetas como el de los modificadores pueden ir indistintamente en mayúsculas o minúsculas, no así algunos valores de ciertas etiquetas los cuales deben respetar mayúsculas y minúsculas.

Ejemplo de etiqueta con modificador, en el cual sí es importante respetar mayúsculas y minúsculas:

```
< A href="http://entren.dgsca.unam.mx"> Entren </A>.
```

### **1.5 Estructura básica de un documento HTML**

Un documento HTML está definido por una etiqueta de apertura y de cierre llamada <HTML>.

Dentro de estas etiquetas se distinguen dos partes fundamentales, la cabecera, delimitada por la etiqueta <HEAD> y el cuerpo del documento delimitado por la etiqueta <BODY>.

Dentro de las etiquetas <HEAD> se coloca información como: créditos, nombre del archivo, fecha de publicación o información que sólo es relevante para los buscadores de páginas, la cual no será mostrada en el navegador.

Con la etiqueta <TITLE> especificaremos el título del documento HTML, es importante que el título guarde relación con el contenido del documento y sea definido de forma general, su tamaño no está limitado aunque se recomienda que no sea demasiado extenso. Como todos los elementos que se incluyen dentro de la etiqueta <HEAD>, el título no forma parte del documento en sí, es decir no se muestra dentro del documento, éste se muestra en la parte superior de la ventana del navegador.

Dentro de la etiqueta <BODY> se incluirá toda la información que en realidad será mostrada dentro de la ventana del navegador. A través de los modificadores de esta etiqueta se especifican algunos valores que afectan de forma global a todo el documento, como son los colores del texto y del fondo del documento, así como los colores para las ligas.

Ninguna de estas etiquetas es obligatoria, pudiendo crear páginas que se muestren sin ningún problema aun con la falta de alguna de éstas. Lo que sí es importante es tener cuidado de escribir correctamente cada una de las etiquetas que se incluyan.

Ejemplo:

```
<HTML>
```

```
<HEAD>
```

```
<TITLE> Páginas en Internet</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
Contenido del Documento
```

```
</BODY>
```

```
</HTML>
```

## **2. ETIQUETAS BÁSICAS**

### **2.1 <HTML>**

Primera etiqueta en cada documento. Indica que el contenido del documento se encuentra en lenguaje HTML, todo el texto y etiquetas (comandos) del documento deben de estar dentro de las etiquetas <HTML> inicial y </HTML> final.

Ejemplo:

```
<HTML>
```

Documento

```
</HTML>
```

### **2.2 <HEAD>**

Especifica que el texto contenido dentro del inicio y fin de la etiqueta son el prólogo del documento.

El texto que se escribe dentro de la etiqueta <HEAD> no se muestra en el navegador, por lo que hay que procurar no poner contenido del documento dentro de ella.

Dentro de esta etiqueta colocamos etiquetas como <TITLE>, <ISINDEX>, <BASE>, y <META> que son elementos opcionales. Elementos que contienen información general o meta-información acerca de un documento.

### **2.3 <TITLE>**

Dentro de la etiqueta <TITLE> colocamos el título o tema del documento. La etiqueta <TITLE> siempre va dentro de la etiqueta <HEAD>.

Al escribir el título del documento debemos ser claros y no utilizar otras etiquetas, ni títulos muy extensos ya que los navegadores muestran el texto contenido entre esta etiqueta en la barra de título, además de utilizarlo para agregarla a la lista de marcadores favoritos.

El contenido de la etiqueta <TITLE> es utilizado por los buscadores como Yahoo, Infoseek, Altavista, etc. para localizar la página.

Ejemplo:

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>
```

Páginas en Internet

```
</TITLE>
```

```
</HEAD>
```

```
</HTML>
```

### **2.4 <BODY>**

Cuerpo del documento, es en donde se incluirá toda la información que en realidad será

mostrada dentro de la ventana del navegador.

El cuerpo del documento puede contener una gran variedad de elementos los cuales pueden soportar una gran cantidad de atributos.

La etiqueta <BODY> contiene varios atributos o modificadores, los cuales podemos utilizar y actúan sobre el contenido del cuerpo del documento en forma global.

Ejemplo:

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>
```

Páginas en Internet

```
</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

Esta es mi primera página

```
</BODY>
```

```
</HTML>
```

### 3. GUARDAR UN DOCUMENTO

Ya creado un archivo HTML es necesario guardarlo en disco. Para realizar esto, se selecciona del menú **Archivo** la opción **Guardar Como**, aparecerá el cuadro de diálogo en donde se elegirá primero la unidad de disco y carpeta en donde se desea guardar el archivo.

Una vez seleccionada la ubicación se escribirá un nombre para el archivo. Al elegir un nombre para el archivo, se deben seguir ciertas reglas:

- El nombre del archivo debe de tener la extensión .html o .htm (en los sistemas DOS).
- Utilizar nombres cortos y sencillos, evitar el uso de espacios, caracteres especiales, palabras acentuadas, etc., y tomar en cuenta que las mayúsculas y las minúsculas se consideran diferentes en la mayoría del software para Web.

Tener cuidado que en el cuadro **Guardar como archivos de tipo** esté seleccionada la opción **Todos los archivos (\*.\*)**.

La acción se finaliza presionando el botón **Guardar**.

Si se utiliza un procesador de texto debe guardarse como archivo sólo de texto o como archivo HTML, si es que se encuentra disponible esta opción.

### 4. ABRIR UN DOCUMENTO EN FORMA LOCAL

Una vez guardado el documento, podrá observarse en forma local utilizando el navegador para ver los resultados de lo escrito en HTML.

Para abrir el documento en forma local, hay que seleccionar en el menú **Archivo** la opción **Abrir**.

Si se conoce la ubicación correcta del documento puede teclearse en el cuadro de texto en donde se encuentra el cursor.

Por ejemplo:

C:\Html\inicio.html

Si no se recuerda la ubicación presionar el botón **Examinar**, con lo cual se abrirá una ventana para que se localice la unidad de disco, carpeta y archivo.

Una vez localizado el archivo seleccionarlo para que aparezca el nombre en el cuadro **Nombre del archivo y presionar el botón Abrir**.

Al regresar a la ventana inicial de **Abrir**, el nombre de archivo aparece en el cuadro **Abrir** sin necesidad de haberlo tecleado. Para finalizar presionar en el botón de **Aceptar** para poder ver el documento en el navegador.

No es necesario abrir el documento cada vez que se le haga un cambio en el editor y se guarde, para ver el documento con cambios en el navegador contamos con el botón **Actualizar** de la barra de herramientas o desde el menú **Ver**, elegir la opción **Actualizar**. Como su nombre lo indica recarga el archivo que se encuentra en la ventana actualizando la página.

No resulta raro que al comenzar a escribir nuestras páginas tengamos problemas al querer visualizar nuestra página. Hay dos errores que son los más comunes, debemos revisar antes que nada que la extensión del archivo esté correctamente escrita, ya que la extensión que reconocen los navegadores es "html" o "htm", así como tener cuidado al escribir las etiquetas verificando siempre que se encuentren correctamente escritas y que las etiquetas de cierre o finales estén presentes siempre que sea necesario.

## 5. FORMATO DE PÁGINAS

### 5.1 Encabezados

Las cabeceras o encabezados proporcionan un medio sencillo pero útil para dividir el cuerpo del documento en secciones.

Existen seis niveles de encabezados <H1> </H1> a <H6> </H6>. Los números indican los niveles de encabezado, el texto de acuerdo con el nivel se despliega con diferente tipografía ya sea más grande o más negra. Provocan una línea en blanco después del mismo.

Cuando se despliegan en el navegador no aparecen numerados, pueden tener cualquier longitud inclusive varias líneas aunque esto no es conveniente ya que puede resultar cansador para el lector.

No deben de usarse los encabezados para resaltar texto en negritas, o resaltar ciertas partes del documento.

Ejemplo:

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Páginas 1 en Internet</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<h1>Encabezado H1</h1>
```

```
<h2>Encabezado H2</h2>
```

```
<h3>Encabezado H3</h3>
```

```
<h4>Encabezado H4</h4>
<h5>Encabezado H5</h5>
<h6>Encabezado H6</h6>
</BODY>
</HTML>
```

Permite atributos como align (left/ center/ right/ justify).

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>Páginas en Internet</TITLE>
</HEAD>
<BODY>
<h3 align="center">Encabezado H3 con alineación centrada</h3>
</BODY>
</HTML>
```

## **5.2 Párrafos**

Cuando se escribe un documento se tiene la costumbre de estructurarlo por medio de párrafos.

Probablemente se está acostumbrado al uso de procesadores de texto en los que un retorno de carro provoca el comienzo de un nuevo párrafo. En HTML esto no funciona, para identificar un fragmento del documento como párrafo es necesario delimitarlo con la etiqueta <P> </P>. La etiqueta de cierre puede ser omitida.

Dependiendo del navegador se pueden visualizar los párrafos con una línea en blanco antes y después de cada párrafo.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&aacute;ginas 2 en Internet</TITLE>
</HEAD>
<BODY>
```

```
<P>El Imperio Antiguo se inicia con gran esplendor político y artístico. Es este el período de la construcción de las colosales pirámides. En la III dinastía, Zóser construye la pirámide escalonada de Saqqara, prototipo arquitectónico de las edificadas posteriormente. Keops erigió la Gran pirámide de Gizeh, junto a la cual edificarán las suyas los faraones Kefrén y Micerinos. A finales de la IV dinastía comenzó a desestabilizarse el poder político y aumentó notablemente la influencia de los gobernadores provinciales. La consecuencia fue una gran crisis de Estado que desembocaría en el desmoronamiento del Imperio Antiguo.</P>
```

<P>Cómo y por qué momificaban a los muertos.</P>

<P>Para los egipcios la muerte no era un punto final a su existencia, sino que significaba el inicio de una nueva vida.</P>

</BODY>

</HTML>

En HTML una secuencia de espacios en blanco, introducidos mediante la tecla espaciadora o la tecla de tabulador en el documento, serán visualizados en el navegador como un solo espacio en blanco.

## 5.3 Caracteres

### 5.3.1 Fuente

Para controlar la forma en que el navegador presenta las fuentes del documento, contamos con las

etiquetas <BASEFONT> y <FONT>.

<BASEFONT>

Permite establecer estándares para la fuente de todo el documento utilizando los atributos de tipo, tamaño y color de fuente (face, size, color).

Ejemplo face="helvética"

· *Size*. Permite valores de 1 a 7 en el tamaño de la fuente, si no se ha definido un tamaño el valor por defecto es 3.

Ejemplo size="5"

· *Color*. Permite cambiar el color del texto, el cual se define con el nombre del color en inglés o en notación hexadecimal RGB (Red, Green, Blue) indicado por el signo de # (número) y un número hexadecimal de seis dígitos. Los dos primeros corresponden al rojo de 00 (no rojo) a.

FF (rojo intenso), los siguientes dos dígitos corresponden al verde y los dos últimos al azul.

Negro es ausencia de color, #000000 y blanco son todos los colores #FFFFFF.

Ejemplo: color="navy" o color="#000080"

Se define al inicio del cuerpo del documento, aunque no es indispensable. No obstante si dentro de la etiqueta <BASEFONT> se encuentran encabezados éstos permanecerán invariables.

Ejemplo:

<HTML>

<HEAD>

<TITLE>P&acute;ginas en Internet</TITLE>

</HEAD>

<BODY>

<basefont size="5" color="navy" face="helvetica">

Fuente Helv&acute;tica con atributos



```
<h2>Fuente Helv&eacute;tica con encabezado H2</h2>
</basefont>
</BODY>
</HTML>
```

Figura 4. Texto con etiqueta `<BASEFONT>` visto en el navegador.

### **<FONT>**

Permite cambiar el tamaño, color y estilo de fuente (size, color, face), del texto que se encuentre ubicado entre las etiquetas `<FONT></FONT>`.

Contiene los atributos: size, face y color.

- *Size*. Define un tamaño de fuente que puede ir de 1 a 7, conocido como tamaño absoluto. Si se define con el signo (+ o -) el tamaño es relativo al definido como tamaño base utilizando la etiqueta `<basefont>` o al tamaño predefinido en el navegador.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&acute;ginas en Internet</TITLE>
</HEAD>
<BODY>
<font size="6">
Tama&ntilde;o de fuente absoluto en 6</font><br>
3
<font size="+2">Tama&ntilde;o de fuente relativo en +2</font>
</BODY>
</HTML>
```

- *Face*. Permite modificar el estilo o tipo de fuente mostrada por el navegador. El navegador analiza los nombres de las fuentes instaladas en la máquina del usuario, hasta localizar la definida; si ésta no se encuentra instalada en el sistema se despliega la definida como predeterminada en el navegador.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&acute;ginas en Internet</TITLE>
</HEAD>
<BODY>
<font face="helvetica">Fuente helv&eacute;tica</font><br>
<font face="bookman old style">Fuente bookman old style</font>
```

</BODY>

</HTML>

· *Color*. Establece el color del texto contenido dentro de la etiqueta <FONT>. Puede definirse con el nombre del color en inglés como red (rojo), blue (azul), etc.; o en valor hexadecimal RGB indicado por el signo de # (número) y un número hexadecimal de seis dígitos. Los dos primeros corresponden al rojo de 00 (no rojo) a FF (rojo intenso), los siguientes dos dígitos corresponden al verde y los dos últimos al azul. Negro es ausencia de color, #000000 y blanco son todos los colores #FFFFFF.

Ejemplo:

<HTML>

<HEAD>

<TITLE>P&aacute;ginas en Internet</TITLE>

</HEAD>

<BODY>

<font color="blue">Color azul</font><br>

<font color="#FF00FF">Magenta</font>

</BODY>

</HTML>

### 5.3.2 Caracteres especiales

No todos los caracteres de texto que utilizamos y tenemos en el teclado pueden ser mostrados por el navegador. Algunos caracteres tienen significados especiales como < > que delimitan las etiquetas, que podríamos utilizar como menor o mayor en una ecuación matemática y que confunden al navegador.

HTML permite incluir cualquiera de estos caracteres, los cuales comienzan con un ampersand (&) seguido de su nombre o contracción en inglés y terminan con un punto y coma ( ; ).

Alternativamente también es posible utilizar el número de posición del carácter en la tabla de caracteres ASCII, precedida por el signo de #.

#### TABLA DE ALGUNOS CARACTERES ESPECIALES

**Entidad Numérica Entidad Nombrada Símbolo**

&#034; &quot; ; "

&#060; &lt; ; <

&#160; &nbsp; ; Espacio

&#193; &Aacute; ; Á

&#201; &Eacute; ; É

&#205; &Iacute; ; Í

&#209; &Ntilde; ; Ñ

&#211; &Oacute; ; Ó

&#218; &Uacute; Ú  
&#225; &aacute; Á  
&#233; &eacute; É  
&#237; &iacute; Í  
&#241; &ntilde; Ñ  
&#243; &oacute; Ó  
&#250; &uacute; Ú

#### **5.4 Alineación**

La etiqueta <CENTER> </CENTER> permite centrar porciones enteras de la página dentro de la ventana del navegador.

Genera un salto de línea automático después de lo contenido dentro de la etiqueta, sin agregar espacio antes o después. Permite centrar texto, imágenes, etcétera.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&aacute;ginas en Internet</TITLE>
</HEAD>
<BODY>
<center>P&aacute;ginas en Internet</center>
```

Manual realizado en la Direcci&oacute;n General de Servicios de C&oacute;mputo Acad&eacute;mico UNAM

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

#### **5.5 Saltos de línea**

El navegador muestra el texto del documento del margen izquierdo de la ventana hasta llenar cada una de las líneas, pasando automáticamente el texto a la siguiente línea cuando éste ya no puede mostrarse en la ventana. Ajustándose cada vez que se modifica el tamaño de la misma.

La etiqueta <BR> inserta un salto de línea en donde se coloque, enviando el texto que se encuentra después de ella a la siguiente línea sin generar espacios antes o después. No requiere etiqueta de cierre.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&aacute;ginas en Internet</TITLE>
</HEAD>
<BODY>
```

```
P&aacute;ginas en Internet<br>
Manual realizado en la<br>
Direcci&oacute;n General de Servicios de C&oacute;mputo
Acad&eacute;mico<br>
UNAM
</BODY>
</HTML>
```

## 5.6 Separador horizontal

Una línea horizontal trazada o regleta permite separar visualmente la página para hacer notar algún cambio de tema o como adorno.

Para realizar esto se utiliza la etiqueta <HR> la cual no tiene etiqueta de cierre y permite dibujar una línea horizontal en el ancho de la página, que puede ser de diferente largo, grueso, color, alineación y con o sin sombra.

Si deseamos modificar la línea contamos con los modificadores **Width**, **Color**, **Size**, **Align** y **Noshade**.

Cuando se requiere que la línea aparezca con relleno utilizamos el modificador **Noshade**.

Al utilizar la etiqueta HR, podemos utilizar uno o todos los modificadores. No es recomendable colocar demasiadas líneas horizontales en una página ya que pueden ser distractores para el lector.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&aacute;ginas en Internet</TITLE>
</HEAD>
<BODY>
P&aacute;ginas en Internet
<hr>
Manual realizado en la
<hr align="RIGHT" size="5" width="75%">
Direcci&oacute;n General de Servicios de C&oacute;mputo Acad&eacute;mico
<hr align="left" size="5" width="25%" color="gray"noshade>
UNAM
</BODY>
</HTML>
```

## 5.7 Texto con preformato

Cuando escribimos una página y colocamos cualquier espacio en blanco adicional (tabuladores, retorno de carro o espacios), éstos son eliminados por el navegador por lo

que al ver nuestra página observaremos que las líneas se unieron y que el texto quedó sin espacios.

Para que el navegador respete lo antes mencionado utilizamos la etiqueta <PRE>.

Esta etiqueta se utiliza con el objeto de que lo que se encuentre dentro de la etiqueta se observe igual en el navegador: alinear o sangrar texto utilizando el tabulador, o generar líneas en blanco con retornos de carro.

La desventaja de utilizar esta etiqueta es que se pierde el formato que se le haya aplicado al texto, sólo presenta el texto con fuente monoespaciada como Courier, las líneas deben de ser de 60 caracteres o menos ya que el navegador no podrá ajustar el largo de la línea, ni dar presentación de encabezado.

Este tipo de etiqueta es útil cuando necesitamos presentar ejemplos de códigos.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE>P&aacute;ginas en Internet</TITLE>
</HEAD>
<BODY>
<PRE>
import java.awt.*;
public class Ejemplo1 {
public static void main( String args[] ) {
Frame f = new Frame ("Ejemplo 1" );
f.show();
}
}
</PRE>
</BODY>
</HTML>.P&aacute;ginas para Internet
```

## 6. LISTAS

Además de los encabezados, párrafos y saltos de línea, el lenguaje HTML proporciona algunas otras estructuras basadas en texto, como son: las listas, que nos permiten entre otras cosas, representar elementos en un determinado orden de una manera sencilla.

Dentro de una lista se pueden incluir prácticamente cualquiera de los demás elementos HTML, e igualmente una lista puede incluirse dentro de formularios, tablas, etc.

A pesar de que tenemos la posibilidad de definir varios tipos de listas, cada una de éstas tendrá elementos en común. Cada una de ellas estará determinada por sus correspondientes etiquetas de inicio y de fin (por ejemplo, < UL> y </UL> o <O > <O/> que variará según el tipo de lista de que se trate, y de la misma manera, cada elemento de la lista tendrá su propia etiqueta, ya sea <DT> y <DD> en las listas de definición o <LI> en los otros tipos de listas.

En HTML existen tres tipos distintos de listas:

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

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

```
<UL>
<LI>Punto 1
<LI>Punto 2
<LI>Punto 3
<LI>Punto 4
</UL>
```

La lista está dentro de la etiqueta <UL> y </UL>, y luego cada término va precedido de la etiqueta <LI> (list ítem). El aspecto de la lista anterior en el navegador es el siguiente:

- Punto 1
- Punto 2
- Punto 3
- Punto 4

Las listas se pueden anidar una dentro de otra. Por ejemplo:

```
<UL>
<LI> Cine
<UL> Star Wars
<LI> Psicosis
</UL>
<LI> Música.
<UL>
<LI> U2
<LI> Santana
</UL>
</UL>
```

Quedaría el siguiente resultado:

- Cine
- Star Wars
- Psicosis
- Música
- U2
- Santana

Las listas ordenadas (ordered lists) sirven para listar cosas en un orden determinado y se identifican con números. Su estructura es muy similar a la anterior, sólo que en el resultado aparecerá automáticamente un número para cada término listado.

```
<OL>
```

```
<LI> Primer ítem
```

```
<LI> Segundo ítem
```

```
<LI> Tercer ítem
```

```
<LI> Etc.
```

```
</OL>
```

El resultado es:

1. Primer ítem.

2. Segundo ítem.

3. Tercer ítem.

4. Etc.

La versión 4.0 de HTML permite usar el atributo TYPE para definir el tipo de numeración en la lista.

```
<OL TYPE="i">
```

```
<LI>Primer ítem
```

```
<LI>Segundo ítem
```

```
<LI>Tercer ítem
```

```
<LI>Etc.
```

```
</OL>
```

El resultado sería el siguiente:

i. Primer ítem.

ii. Segundo ítem.

iii. Tercer ítem.

iv. Etc.

Los valores aceptados por el atributo TYPE son los siguientes:

**1** - Números por omisión: 1, 2, 3, etcétera.

**A** - Letras mayúsculas: A, B, C, etcétera.

**a** - Letras minúsculas: a, b, c, etcétera.

**I** - Números romanos en mayúsculas: I, II, III, etcétera.

**i** - Números romanos en minúsculas: i, ii, iii, etcétera.

Igual que las listas desordenadas, también se pueden anidar las listas ordenadas.

Listas de definición. Estas listas son apropiadas cuando es necesario incluir una definición del punto listado (como un glosario). Toda la lista debe ir englobada entre las etiquetas <DL> y </DL>. Y a diferencia de las listas anteriores, cada renglón de la lista consta de dos partes: el nombre del término a definir, que se consigue con la etiqueta <DT> (definition term); y la definición del término, que se consigue con la etiqueta <DD>

(definition definition).

Por ejemplo:

<DL>

<DT> Término a definir

<DD> La definición de este término

<DT> Otro término a definir

<DD> La definición de este otro término

</DL>

Su resultado es:

Una cosa a definir.

La definición de esta cosa.

Otra cosa a definir.

La definición de esta otra cosa.

Como ya vimos antes, las listas se pueden anidar, e incluso combinar. Podemos combinar no ordenadas con ordenadas, ordenadas y de definición y no ordenadas, etcétera.

## 7. LIGAS

### 7.1 A un documento local

Una **liga** es una zona de texto o gráficos que al ser seleccionados nos trasladan a otra página Web o a otro lugar dentro de la página actual. También es común llamarles *Hiperligas* debido al propio lenguaje html.

HTML es el lenguaje con el que se desarrollan las páginas para Internet, y el *Hipertexto* es una de sus características más fuertes. El *Hipertexto* es la capacidad que posee una sección de texto, para enviar al usuario a otro documento. La diferencia con otros tipos de *Hipertexto* es que los documentos a los que hace referencia pueden estar físicamente en otra parte del mundo o en el disco duro local.

Para colocar una **liga** hay que utilizar la etiqueta: <A> y su correspondiente etiqueta de cierre </A>.

La letra A de la etiqueta proviene de la palabra ANCHOR que en español significa Ancla.

Todo lo que esté dentro de ambas etiquetas (<A> y </A>), ya sean texto o imágenes, serán considerados como ligas. En ambos casos la apariencia del texto o imagen seleccionado como liga sufrirá dos modificaciones:

1. Se visualizará de manera distinta en el navegador (Browser). El texto aparecerá subrayado y de un color distinto al del resto del texto, y las imágenes estarán rodeadas por un borde del mismo color que el determinado para el texto. Este último borde puede ser eliminado con los atributos de la imagen.
2. Al pasar el apuntador del mouse encima de la liga el gráfico del apuntador cambiará a una pequeña mano que apunta con el dedo índice. Al hacer clic sobre la liga, el navegador mostrará la página a donde hacía referencia la liga, o a una sección diferente de la página actual. Por ejemplo, al inicio, al final o a la mitad de la página.

Como las otras etiquetas de HTML que se han revisado, la etiqueta <A> también cuenta con un grupo de modificadores o atributos que permiten adaptar el comportamiento de la



liga (hiperliga).

## ETIQUETA PARA DEFINIR HIPERLIGA (LIGA)

**<A modificadores >..... </A>**

Modificador Descripción del Modificador Href = *ubicación*

Especifica la dirección de la página a la que hace referencia la liga, es decir, aquella que será desplegada por el navegador después de hacer clic sobre la *hiperliga*. Generalmente la *ubicación* es un URL. Si se trata de una liga que apunta a una sección diferente de la página actual, la *ubicación* será el nombre de ese fragmento de la página (nombre del ancla), más el símbolo # al inicio.

Name = *cadena*

Identifica una sección de la página actual. Es usado en caso de hacer referencia a la página actual. La *cadena* puede ser cualquier combinación de caracteres alfanuméricos sin espacios en blanco.

Target = *nombre* Con este modificador se define la presentación del marco (frame) o ventana (window) en donde será desplegada la página a la que se hace referencia.

Title = *cadena* La *cadena* almacena un título para la página o fragmento de página que se desplegará.

En caso de que la liga haga referencia a una página local, se debe especificar la ubicación del archivo que contiene el código HTML de la página en cuestión. La sintaxis mínima para colocar una liga es la siguiente:

**<A HREF ="ubicación"> Haz clic aquí para ver la página a la que se hace referencia </A>**

La *ubicación* puede corresponder a la **ruta de un archivo** (página) local, o a una *ubicación* con formato **URL** (Uniform Resource Locator) que puede estar en cualquier parte del mundo.

Ejemplo:

El siguiente fragmento de código HTML muestra una liga que hace referencia a una página local.

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Mi primera liga</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<P> Esta es la primera página que elaboro con una liga. Para ver a donde hace referencia la liga se debe hacer clic más abajo.
```

```
<A HREF = "../especiales/pagina.html"> Haz click aquí para ver la página </A>
```

```
</BODY>
```

```
</HTML>
```

Existen dos tipos de rutas para los archivos: rutas absolutas y rutas relativas.

a) *Las rutas absolutas* proporcionan la ubicación del archivo comenzando desde la unidad de disco (o raíz para sistemas operativos como UNIX), pasando por las carpetas (directorios) y subcarpetas (subdirectorios) en donde el archivo se encuentra.

Ejemplo:

**C:\windows\system\nombre del archivo.html** (en Windows 95/98/NT)

**/usr/local/httpd/html/nombre\_del\_archivo.html** (en cualquier versión de UNIX).

b) *Las rutas relativas* describen la ubicación de un archivo tomando como base el directorio actual de trabajo, es decir, proporcionan el camino que hay que seguir entre carpetas para poder llegar desde la carpeta actual (normalmente la carpeta que almacena la página que se está elaborando) hasta la carpeta en donde se encuentra la página que se desea visualizar.

Ruta Relativa de un archivo:

**../directorio/subdirectorio/nombre\_del\_archivo.html** (en Unix)

Ejemplo:

**<A AREF="nombre\_del\_archivo.html"> liga a un archivo en el mismo directorio </A>**

Ejemplo:

**<A AREF="subdirectorio/nombre\_del\_archivo.html"> liga en un subdirectorio </A>**

**NOTA:** no es recomendable el uso de rutas absolutas para representar la ubicación exacta de los recursos de cómputo que serán utilizados.

### **7.1.1 Dentro del mismo documento**

Al principio de este capítulo se comentó que la etiqueta **<A>** proviene de la palabra ANCHOR que significa *Ancla*. De modo que **ancla** será el nombre de la sección de la página que se desea desplegar. Es posible acceder a cualquier parte de la página HTML; para hacerlo, primero debemos especificar las secciones en las que se divide la página. El parámetro *name* sirve para marcar el inicio de una sección dentro de nuestra página.

**<A NAME = "seccion1"> Aquí esta el texto de la sección 1. </A>**

La sección se llamará *sección1*. Para hacer un enlace a esta sección dentro de nuestra página lo haríamos de la siguiente forma.

**<A HREF = "#seccion1"> Haz clic aquí para ir a la primera sección de la página.</A>**

También podemos acceder a **anclas** situadas en páginas remotas. Para ello añadiremos el nombre del **ancla** al URL:

**<A HREF = "http://www.unam.mx/manual\_html/enlaces.html#seccion1"> Ir a la primera sección de la página remota.</A>**

### **7.2 A un documento en Internet**

Es posible acceder a cualquier página Web, si se conoce el URL donde se localiza. Para acceder a una página remota por medio de una *liga* se debe proporcionar la dirección completa de la misma dentro de la etiqueta **<A>** utilizando el modificador **href**.

Ejemplo:

**<A HREF = "http://www.servidor.unam.mx/directorio/archivo.html">Este es el texto de la liga a un sitio remoto. </A>**

Es posible indicar al navegador, la ubicación que deberá de tener la página a la que se hizo referencia, eso se logra usando el modificador *target*.

Ejemplo:

**<A HREF = "http://www.servidor.unam.mx/directorio/archivo.html" TARGET= "\_blank"> Este es el texto de la liga.</A>**

En el último ejemplo, el navegador abre una nueva ventana de sí mismo, para desplegar la página Web contenida en "archivo.html". Existen más valores para el modificador *target* que permiten cargar la página en la misma ventana, en un marco (frame) distinto al actual, etcétera.

Los valores predefinidos para el modificador *target* son los siguientes:

- *\_top*

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

- *\_blank*

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

- *\_self*

Muestra la nueva página en el marco donde está declarada la hiperliga.

- *\_parent*

Muestra la nueva página en el <FRAMESET> que contiene al marco donde se declara la hiperliga.

- *Nombre\_del\_marco*

Muestra la nueva página en el marco que se menciona en el parámetro "*nombre\_del\_marco*".

### 7.2.1 URLs

Un URL(Uniform Resource Locator) o Localizador Uniforme de Recursos, nos indica tanto una dirección de Internet como el servicio que esperamos nos ofrezca el servidor al que corresponde la dirección. Tiene el siguiente formato:

*servicio://máquina:puerto/ruta\_del\_archivo/nombre\_archivo* donde el **servicio** puede ser uno de los siguientes:

- *http*

Es el servicio usado para transmitir páginas Web y el que usaremos normalmente en las *ligas*.

- *https*

Es similar al anterior, y nos permite acceder a servidores (generalmente comerciales) que nos ofrecen el uso de técnicas de encriptación para proteger los datos que serán transmitidos a través de la red.

- *ftp*

Permite transmitir archivos desde servidores de **ftp** anónimo. Si solicitamos un directorio y no un archivo, en general el navegador se encargará de mostrarnos el contenido del mismo para que podamos escogerlo cómodamente. Utilizando la @ podremos acceder a servidores privados.

- *mailto*

Para poder mandar un mensaje. Por ejemplo, el URL "mailto:juan@servidor.unam.mx" enviaría un mensaje al usuario "juan" del equipo "servidor" de correo de la UNAM.

- *news*

Para poder acceder a foros de discusión (mal traducidos a veces como grupos de noticias). Se indica el servidor y el grupo.

La dirección de la máquina puede ser, o bien una serie de cuatro números entre 0 y 255 (132.248.120.29) o bien algo más fácil de recordar como es una serie de palabras separadas por puntos (www.unam.mx). El puerto generalmente no se indica, ya que el servicio proporcionado determina el número del puerto.

La **ruta\_del\_archivo** es una cadena de directorios separados por el símbolo "/", que es utilizado en UNIX (el sistema operativo más extendido en los servidores de Internet).

Existe otro formato de URL. Cuando queremos acceder a un archivo situado en la misma máquina que la página Web que estamos creando podemos utilizar este formato:

*ruta\_relativa/archivo.html*

En la *ruta relativa* podemos utilizar los dos puntos (..) para acceder al directorio padre.

Otra forma es comenzar con diagonal "/" para acceder a una *ruta absoluta* en la estructura de directorios de nuestro equipo local.

## 8. IMÁGENES

Para incluir gráficos e imágenes en nuestras páginas, utilizaremos la etiqueta **<IMG>**. La sintaxis mínima de la etiqueta <IMG> es la siguiente:

**<IMG SRC = "archivo\_gráfico" ALT = "descripción">**

Cuando el navegador interpreta el documento fuente y llega a la etiqueta **<IMG>** recupera el archivo cuya ubicación es indicada por el atributo SRC (source, origen o fuente) y reemplaza el elemento por la imagen. Observe que **<IMG>** es una etiqueta vacía que, por tanto, se utiliza siempre sin etiqueta de cierre. El atributo SRC siempre es requerido y puede ser una ruta absoluta o relativa de un archivo, incluyendo un URL.

La etiqueta **<IMG>** tiene varios parámetros:

### ETIQUETA PARA DEFINIR IMAGEN

**<IMG SRC = "archivo\_gráfico" ALT = "descripción" ..... modificadores.... >**

#### Modificador Descripción del Modificador

src = "imagen" Indica el nombre del archivo gráfico que desea mostrar.

alt = "texto"

Mostrará el texto indicado en el caso de que el navegador utilizado para ver la página no sea capaz de visualizar la imagen. En la mayoría de los navegadores modernos se muestra un mensaje emergente al momento de pasar el apuntador del mouse sobre la imagen.

lowsrc ="imagen"

Muestra una segunda imagen "superpuesta" sobre la primera una vez que se carga la página. Este parámetro no es reconocido por la totalidad de los navegadores ya que está en estudio su aplicación, así que en la mayoría de los casos será ignorado, mostrando sólo la primera imagen (src). En Netscape muestra la imagen indicada por lowsrc en primer lugar, y posteriormente muestra la imagen indicada por src. Si las imágenes son iguales pero tienen distinta "resolución" se conseguirá un efecto tipo "Fade". Si las imágenes son de distinto tamaño la imagen indicada en src se redimensionará al tamaño indicado por la imagen indicada en lowsrc.

align = TOP / MIDDLE / BOTTOM

Indica cómo se alineará la primera línea de texto que siga a la imagen. TOP alinea el texto con la parte superior de la imagen, MIDDLE con la parte central, y BOTTOM con la parte inferior.

border = "tamaño" Indica el tamaño del "borde" de la imagen. A toda imagen se le asigna un borde que será visible cuando la imagen forme parte de una hiperliga.

### **Modificador Descripción del Modificador**

height = "tamaño" Indica el alto de la imagen en puntos o en porcentaje. Se usa para modificar el tamaño de la imagen original.

width = "tamaño" Indica el ancho de la imagen en puntos o en porcentaje. Se usa para modificar el tamaño de la imagen original.

hspace = "margen" Indica el número de espacios horizontales, en puntos, que separarán la imagen del texto que la siga y la anteceda.

Vspace = "margen" Indica el número de puntos verticales que separarán la imagen del texto que le siga y la anteceda.

ismap / usemap Indica que la imagen es un MAPA sensitivo.

## **8.1 Formatos**

Hay dos formatos de imágenes que todos los navegadores modernos reconocen. Son las imágenes **GIF** y **JPG**. Las últimas versiones de Netscape y Explorer aceptan también el formato PNG.

Cualquier otro tipo de archivo gráfico o de imagen (BMP, PCX, CDR, etc...) no será mostrado por el navegador, a no ser que disponga de un programa externo que permita su visualización.

## **8.2 Modificadores**

### **8.2.1 Tamaño**

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

```
<IMG SRC="../../Capitulo8/imagenes/interrogacion.gif" ALT="Esto es una imagen"  
WIDTH="80" HEIGHT="60">
```

En este ejemplo se muestra una imagen que tiene un ancho de 88 píxeles y una altura de 31 píxeles.

El texto que aparecerá antes de que se cargue la imagen será el que está asignado al parámetro ALT.

### **8.2.2 Alineación**

Para poder colocar texto e imágenes en una misma página, HTML proporciona un parámetro muy útil para controlar la posición de estos elementos. Por medio del parámetro ALIGN, es posible alinear una imagen respecto del texto que la acompaña. La siguiente tabla muestra los valores posibles para el parámetro ALIGN.

### **Modificador Descripción del Modificador**

TOP Coloca el punto más alto de la imagen coincidiendo con el más alto de la línea de texto actual.

MIDDLE Alinea el punto medio (en altura) de la imagen con la base del texto.

BOTTOM Alinea el punto más bajo de la imagen con la base del texto.

LEFT Coloca la imagen a la izquierda del texto.

RIGHT Coloca la imagen a la derecha del texto.

### **8.2.3 Texto alternativo**

El parámetro ALT especifica el texto que se mostrará en lugar del gráfico en aquellos navegadores que no sean capaces de mostrarlos (como el Lynx) y en el supuesto de que el usuario los haya desactivado. Algunos navegadores lo muestran cuando pasamos el ratón por encima de la imagen.

Es por eso que, aunque algunos usuarios no lo lleguen a ver nunca, conviene ponerlo siempre. De hecho, el estándar HTML 4.0 obliga a hacerlo.

### **8.2.4 Imagen de fondo**

Hasta el momento se ha explicado cómo utilizar la etiqueta **<IMG>** para mostrar imágenes en nuestras páginas Web. Ahora bien, es posible utilizar una imagen como "fondo" para una página.

Sin embargo, la etiqueta **<IMG>** no permite colocar la imagen de fondo para una página Web. Por lo tanto, debemos apoyarnos en los parámetros que admite la etiqueta **<BODY>**. Esta última etiqueta define una serie de características respecto a la apariencia de toda la página Web. La siguiente tabla muestra algunos de los parámetros que acepta la etiqueta **<BODY>**.

### **Modificador Descripción del modificador**

BACKGROUND Permite definir un gráfico de fondo para la página.

BGCOLOR Permite definir el color de fondo de la página.

TEXT Cambia el color del texto.

LINK Cambia el color de un enlace no visitado (por omisión azul).

VLINK Cambia el color de un enlace ya visitado (por omisión púrpura).

ALINK Cambia el color que toma un enlace mientras lo estamos pulsando (por omisión rojo).

No resulta recomendable cambiar los colores del texto y ligas a no ser que exista alguna dificultad para leerlos por haber cambiado el fondo, ya que en muchas ocasiones el usuario ha podido cambiarlos en las opciones de su navegador y estarán ya a su gusto.

En la siguiente imagen se muestra un signo de interrogación como imagen de fondo en una página de Web. El texto que aparece de color gris no forma parte de la imagen. La imagen original mide 80 píxeles de ancho por 60 píxeles de alto. De modo que, por sí sola no cubre todo el fondo de la página. Por lo que el propio navegador repite la misma imagen tantas veces como sea necesario, pues debe cubrirse toda el área de fondo visible.

Sin embargo, esta útil forma de colocar una imagen de fondo no está disponible con la etiqueta **<IMG>**. Para colocar la imagen de fondo es necesario utilizar la etiqueta **<BODY>** con su modificador *background*.

Ejemplo:

```
<BODY BACKGROUND=" ../Capitulo8/imagenes/interrogacion.gif">
```

*Por omisión, la imagen se repite en toda el área de la página.*

### **8.2.5 Imagen como liga**

Suele ser común incluir ligas dentro de un gráfico. En ese caso, por defecto, los navegadores colocarán un borde al gráfico para indicar que efectivamente es una hiperliga. Esto es práctico, pero la mayoría de las veces poco estético. Por medio del parámetro **BORDER** se puede alterar el grosor de ese borde o incluso eliminarlo poniendo su valor a cero.

Ejemplo:

```
<A HREF = "http://www.unam.mx" >
```

```
<IMG SRC="graficos/interrogacion.gif" ALT="Ir a la página de la UNAM" WIDTH=88  
HEIGHT=31>
```

```
</A>
```

El resultado de este ejemplo es una página con la imagen anterior rodeada por un marco. Al pasar el puntero del *mouse* por encima de la imagen se puede observar que éste último cambia de forma y adquiere la apariencia de "una mano que apunta" con el dedo índice. Eso significa que esa imagen es una hiperliga.

Para quitar el marco que acompaña a la imagen, debemos colocar el modificador *border* con un valor de cero. Como se muestra a continuación.

```
<A HREF="http://www. unam.mx ">
```

```
<IMG SRC="graficos/interrogacion.gif" ALT=" Ir a la página de la UNAM" WIDTH=88  
HEIGHT=31 BORDER=0>
```

```
</A>.
```

## **9. TABLAS**

Las tablas son posiblemente la manera más clara de organizar la información. Las tablas nos permiten representar cualquier elemento de nuestra página (texto, listas, imágenes, etc.) en diferentes renglones y columnas. Es una herramienta muy útil para "ordenar" contenidos de distintas partes de nuestra página. Todas las tablas son arreglos rectangulares de renglones y columnas. Primero se definen las características de toda la tabla, después se definen las características de cada renglón y finalmente se definen las celdas, que son los componentes mínimos de una tabla.

La tabla se define mediante la etiqueta **<TABLE>** ..... **</TABLE>**. Cada uno de los renglones que componen la tabla se define mediante la etiqueta **<TR></TR>**. Estas etiquetas son colocadas entre las etiquetas **<TABLE>** ..... **</TABLE>**. Por último, cada renglón se encuentra compuesto por un determinado número de celdas. Cada celda se define mediante la etiqueta **<TD></TD>**. Estas etiquetas son colocadas entre las etiquetas **<TR></TR>**. Por lo tanto, todo aquello que se encuentre dentro de las etiquetas **<TD></TD>**, estará dentro de la celda en cuestión. La etiqueta **<TH>** indica

una celda de "encabezado", es decir, el contenido será resaltado en negritas y en un tamaño ligeramente superior al normal.

En el siguiente cuadro se explican el uso de cada uno de los modificadores para la tabla en general, los renglones y las celdas.

## **ETIQUETA PARA DEFINIR TABLA**

**<TABLE *modificadores* >..... </TABLE>**

### **Modificador Descripción del Modificador**

**Border=** número Indica el ancho del borde de la tabla en puntos.

**Cellspacing=** número Indica el espacio en puntos que separa a las celdas que están dentro de la tabla.

**Cellpadding=** número Indica el espacio en puntos que separa el borde de cada celda y el contenido de ésta.

**Width=** número ó %

Indica el ancho de la tabla en puntos o en porcentaje en función del ancho de la ventana del navegador. Si no se indica este parámetro, el ancho se adecuará al tamaño de los contenidos de las celdas.

**Height=** número ó %

Indica la altura de la tabla en puntos o en porcentaje en función del alto de la ventana del navegador. Si no se indica este parámetro, la altura se adecuará a la altura de los contenidos de las celdas.

**Bgcolor=** código de color Especifica el color de fondo de toda la tabla.

**<TR>..... </TR>**

**Align=** LEFT / CENTER

/ RIGHT / JUSTIFY

Alinea el contenido de las celdas horizontalmente, hacia la izquierda (LEFT), centro (CENTER), derecha (RIGHT) o justificado (JUSTIFY).

**Valign=** TOP / MIDDLE / BOTTOM

Alinea el contenido de las celdas de la fila verticalmente hacia arriba (TOP), centro (MIDDLE) o abajo (BOTTOM).

**<TD>..... </TD>, <TH>..... </TH>**

**Align=** LEFT / CENTER / RIGHT / JUSTIFY

Indica cómo se debe alinear el contenido de la celda, a la izquierda (LEFT), a la derecha (RIGHT), centrado (CENTER) o justificado (JUSTIFY).

**Valign=** TOP / MIDDLE / BOTTOM

Indica la alineación vertical del contenido de la celda, en la parte superior (TOP), en el centro (MIDDLE), o en la inferior (BOTTOM).

**Rowspan=** número Indica el número de filas que ocupará la celda. Por omisión ocupa una sola fila.

**Colspan=** número Indica el número de columnas que ocupará la celda. Por omisión ocupa una sola columna.

**Width=** número ó %



Indica el ancho de la columna en puntos o en porcentaje en función del ancho de la ventana del navegador. Si no se indica este parámetro, el ancho se adecuará al tamaño de los contenidos. Este modificador sólo funciona en los navegadores modernos.

Bgcolor= código de color Especifica el color de fondo del elemento de la tabla.

Como se indicó anteriormente el contenido de las celdas puede ser cualquier elemento de HTML, un texto, una imagen, una hiperliga, una lista, etcétera.

Las tablas se definen de la siguiente manera. Primero, las características de la tabla, luego las de cada renglón y dentro de éste, cada celda. Así pues, una tabla con dos renglones y tres columnas se escribirá así:

Sintaxis:

```
<TABLE border=1> Etiqueta de inicio de la tabla
<TR> Etiqueta de inicio de renglón
<TD> Renglón 1, Celda 1 </TD>
<TD> Renglón 1, Celda 2 </TD> Conjunto de Celdas
<TD> Renglón 1, Celda 3 </TD>
</TR> Etiqueta de cierre de renglón
<TR> Etiqueta de inicio de renglón
<TD> Renglón 2, Celda 1 </TD>
<TD> Renglón 2, Celda 2 </TD> Conjunto de Celdas
<TD> Renglón 2, Celda 3 </TD>
</TR> Etiqueta de cierre de renglón
</TABLE> Etiqueta de cierre de la tabla
```

Ejemplo:

```
<TABLE border = 4 cellspacing = 4 cellpadding = 4 width =80%>.
<TH align = center>Buscadores
<TH align = center colspan = 2>Otras hiperligas
<TR>
<TD align = LEFT>Yahoo
<TD align = LEFT>Netscape
<TD align = LEFT>Lycos
<TR>
<TD align = LEFT>Infoseek
<TD align = LEFT>AltaVista
<TD align = LEFT>Goto
</TABLE>
```

## 10. HTML AVANZADO

En los capítulos anteriores se han presentado los elementos básicos para la creación de

páginas utilizando HTML. En este capítulo se introducen nuevas etiquetas con las que el desarrollador podrá ofrecer formularios de entrada de datos, la distribución de la información en toda la pantalla y la creación de páginas dinámicas.

## **10.1 Formularios**

Los formularios permiten a los usuarios interactuar con programas que se encuentran en los servidores remotos. En otras palabras los formularios proveen un medio por el cual se puede solicitar información, así como la manera de llevar a cabo el procesamiento de datos enviados por el usuario.

Existe una amplia variedad de etiquetas que facilitan la creación de los formularios. Por ejemplo se puede crear una sencilla caja de texto para solicitar el nombre, la dirección o cualquier otro dato del usuario que se requiera, o bien, se pueden proveer distintas opciones para que el usuario escoja sólo una de ellas

. A continuación se explican éstas y otras etiquetas para formularios.

### **10.1.1 La etiqueta <FORM>**

La etiqueta <FORM> es utilizada para delimitar un formulario para la entrada de datos. Puede haber varias formas en un documento, pero la etiqueta <FORM> no puede delimitar otra etiqueta <FORM>, es decir, un formulario no puede contener otro formulario.

La sintaxis general de la etiqueta <FORM> es la siguiente:

```
<FORM ACTION="URL" METHOD="GET| POST">
```

```
* * *
```

```
</FORM>
```

El atributo ACTION es un URL que especifica la ubicación a la cual se le suministrarán los contenidos de los campos de datos para que los procese. Esto puede ser una dirección de correo, pero generalmente se indica alguna clase de aplicación Cliente/Servidor el atributo ACTION no se define, el URL que se toma por omisión es el del documento por sí mismo.

El atributo METHOD indica la forma en que se envían los datos del formulario hacia el servidor.

Los valores válidos son **Get** y **Post**. El método GET es ideal para formularios que suministran información que no requieren de procesamiento externo, como por ejemplo para consulta de bases de datos. Sin embargo, para información que sirve para la actualización de bases de datos se debe utilizar el método POST, y en este caso ACTION señala a un programa CGI para procesar y actualizar dicha base de datos.

Hasta este punto sólo se ha especificado que en cierta parte del documento se presentará un formulario. En la siguiente sección se presentan las etiquetas para campos de datos.

### **10.1.2 Elementos de una forma**

Los elementos que pueden incorporarse a un formulario pueden dividirse en tres grandes campos:

- Campos de entrada <INPUT ... >
- Áreas de texto <TEXTAREA ... >

· Barras de opciones <SELECT ... >

### **Campos de entrada <INPUT ... >**

Todos estos elementos utilizan la etiqueta <INPUT> y se utilizan para presentar cajas de texto y botones de selección múltiple entre otros.

#### **Cajas de texto**

Para presentar cajas de texto la sintaxis es la siguiente:

```
<INPUT TYPE="TEXT" [SIZE="NN"] NAME="variable">
```

El atributo TYPE indica que la entrada será una caja de texto

El atributo SIZE indica el tamaño de la caja de texto. Por omisión SIZE tiene el valor de 20. El atributo NAME indica el nombre de la variable que almacenará la información que se introdujo en esta caja de texto.

Estos programas se denominan CGI, los cuales se presentarán más adelante.

Para otros tipos de campos de entrada este atributo cambia, por omisión TYPE tiene el valor de TEXT.

algunos ejemplos usando la etiqueta <INPUT> para poner cajas de texto, el código se muestra a continuación:

```
<H1> Ejemplo de un formulario </H1>
```

```
<HR>
```

```
<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">
```

```
Nombre: <INPUT TYPE="TEXT" NAME="nombre"><BR>
```

```
Edad: <INPUT TYPE="TEXT" NAME="edad" SIZE="3"><BR>
```

```
E-mail: <INPUT TYPE="TEXT" NAME="email" SIZE="30"><BR>
```

```
</FORM>
```

```
<HR>
```

#### **Password**

Una variante de este campo de texto es para el paso de claves secretas, la cual es conocida como **Password**. La sintaxis de esta variante es la siguiente:

```
<INPUT TYPE="PASSWORD" [SIZE="NN"] NAME="variable">
```

El efecto es que en lugar de aparecer la información tal cual la escribe el usuario aparecen varios asteriscos. El siguiente código muestra el uso de este tipo de elemento .

```
<H1> Ejemplo de un formulario </H1>
```

```
<HR>
```

```
<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">
```

```
Nombre: <INPUT TYPE="TEXT" NAME="nombre"><BR>
```

```
Edad: <INPUT TYPE="TEXT" NAME="edad" SIZE="3"><BR>
```

```
E-mail: <INPUT TYPE="TEXT" NAME="email" SIZE="30"><BR>
```

```
Clave de entrada: <INPUT TYPE="PASSWORD" NAME="clave" SIZE="10"><BR>
```

```
</FORM>.
```

<HR>

### **Submit y Reset**

Para enviar datos se requiere de un botón **Submit** en el formulario. La sintaxis de dicho botón es la siguiente:

```
<INPUT TYPE="SUBMIT" [VALUE="Texto sobre el botón"]>
```

El atributo VALUE fija el texto sobre el botón, por omisión el texto es "Submit".

Otro botón útil es **Reset**, el cual limpia el formulario por completo. La sintaxis de dicho botón es la siguiente:

```
<INPUT TYPE="RESET" [VALUE="Texto sobre el botón"]>
```

El atributo VALUE fija el texto sobre el botón, por omisión el texto es "Reset".

El siguiente código muestra el uso de los botones **Submit** y **Reset**.

```
<H1> Ejemplo de un formulario </H1>
```

```
<HR>
```

```
<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">
```

```
Nombre: <INPUT TYPE="TEXT" NAME="nombre"><BR>
```

```
Edad: <INPUT TYPE="TEXT" NAME="edad" SIZE="3"><BR>
```

```
E-mail: <INPUT TYPE="TEXT" NAME="email" SIZE="30"><BR>
```

```
Clave de entrada: <INPUT TYPE="PASSWORD" NAME="clave" SIZE="10"><BR>
```

```
<INPUT TYPE="SUBMIT" VALUE="Enviar datos">
```

```
<INPUT TYPE="RESET" VALUE="Limpiar formulario">.
```

```
</FORM>
```

```
<HR>
```

### **Checkbox y Radio**

Otro tipo de campo de entrada son los **Checkbox** y **Radio**. Los primeros son útiles cuando se ha definido una lista de opciones a escoger y el usuario puede seleccionar alguna de ellas. La sintaxis general es la siguiente:

```
<INPUT TYPE="CHECKBOX" NAME="nombre de la opción" [VALUE="valor de la opción"]  
[CHECKED] >
```

El atributo NAME es útil para los programadores de CGI, ya que con dicho nombre se podrá hacer referencia a los elementos seleccionados en dichos programas. El atributo VALUE indica el valor de la opción cuando el usuario active el **Checkbox**. El atributo CHECKED indica cuando los elementos están seleccionados por omisión.

Una forma especial de utilizar estos elementos es mediante los **Radio**. Los elementos **Radio** son muy parecidos a los **Checkbox** pero con la diferencia de que sólo se puede escoger uno a la vez.

Para lograr esto se tiene que dar el mismo valor al atributo NAME para formar grupos de **Radio**. En dichos grupos únicamente se puede seleccionar uno a la vez. El siguiente código muestra los elementos del código anterior pero en su versión de **Radio**.

<H3> Ejemplo de un formulario </H3>

<HR>

<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">

Indique el tipo de cursos que le gustar&iacute;a inscribirse<BR>

(seleccione uno):<BR>

<INPUT TYPE="RADIO" NAME="curso" >Espa&ntilde;ol<BR>

<INPUT TYPE="RADIO" NAME="curso" >Ingles<BR>

<INPUT TYPE="RADIO" NAME="curso" >Pintura<BR>

<INPUT TYPE="RADIO" NAME="curso" CHECKED>Matem&aacute;ticas<BR>

<INPUT TYPE="RADIO" NAME="curso" >Musica<BR>

<INPUT TYPE="RADIO" NAME="curso" >Literatura<BR>

<INPUT TYPE="SUBMIT" VALUE="Enviar datos">

<INPUT TYPE="RESET" VALUE="Limpiar formulario">

</FORM>

<HR>.Páginas para Internet

### **Áreas de texto <TEXTAREA ... >**

Las áreas permiten al usuario poner más de una línea de texto en un solo elemento. La sintaxis general de la etiqueta es la siguiente:

<TEXTAREA COLS="Número de columnas"

ROWS="Number de renglones"

NAME="Nombre del elemento">

TEXTO INICIAL

</TEXTAREA>

Los atributos COLS y ROWS indican el número de columnas y renglones del TEXTAREA respectivamente. EL atributo NAME es la referencia para los programadores de CGIs de este elemento. El texto entre las etiquetas <TEXTAREA> y </TEXTAREA> es el texto que contendrá el elemento al abrir la página por primera vez. A continuación se muestra el código fuente de un ejemplo de este elemento y su presentación final en un navegador.

<H3> Ejemplo de un formulario </H3>

<HR>

<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">

Llene sus observaciones en el siguiente espacio: <BR>

<TEXTAREA COLS="40" ROWS="6">

Texto inicial para el elemento

TEXTAREA.

</TEXTAREA><BR>

<INPUT TYPE="SUBMIT" VALUE="Enviar datos">

```
<INPUT TYPE="RESET" VALUE="Limpiar formulario">
</FORM>
<HR>
```

### ***Barras de opciones <SELECT ... >***

La etiqueta <SELECT> permite al usuario seleccionar una opción de un conjunto de alternativas descritas por etiquetas. Cada alternativa es representada por la etiqueta <OPTION>. La sintaxis general de esta etiqueta es la siguiente:

```
<SELECT NAME="Nombre del elemento"
SIZE="NN"
[MULTIPLE]>
<OPTION [SELECTED] [VALUE="valor1"]> Etiqueta1
<OPTION [SELECTED] [VALUE="valor1"]> Etiqueta2
<OPTION [SELECTED] [VALUE="valor1"]> Etiqueta3
...
<OPTION [SELECTED] [VALUE="valor1"]> EtiquetaN
</SELECT>
```

Para la etiqueta <SELECT> el atributo NAME indica el nombre del elemento SELECT, el valor que tiene este elemento es la etiqueta que al final se seleccionó. El atributo SIZE especifica el número de opciones visibles a la vez. El atributo MULTIPLE indica que pueden seleccionarse más de una opción a la vez.

Para la etiqueta <OPTION> el atributo SELECTED, si se indica, marca como elegida la etiqueta correspondiente. VALUE indica un valor especial para el programa CGI, si no se especifica algún valor se asigna a la etiqueta como tal valor. Las etiquetas vienen a ser las opciones que el usuario podrá escoger.

<H3> Ejemplo de un formulario </H3>

<HR>

```
<FORM ACTION="http://132.248.225.206/cgi-bin/procesa.pl" METHOD="GET">
```

Indique su país de origen:

```
<SELECTNAME="pais"
SIZE="1">
<OPTION VALUE="pais01"> Argentina
<OPTION VALUE="pais02"> Honduras
<OPTION SELECTED VALUE="pais03"> México
<OPTION VALUE="pais04"> Venezuela
</SELECT>
<BR>
<INPUT TYPE="SUBMIT" VALUE="Enviar datos">
<INPUT TYPE="RESET" VALUE="Limpiar formulario">
```

</FORM>

<HR>

### **10.1.3 CGI's**

El CGI es un conjunto de reglas para la conexión de aplicaciones externas con la información que tienen los servidores, tal como los servidores HTTP o Web. Un documento HTML proporcionado por un servidor Web es estático, lo cual significa que es un archivo de texto que no cambia. Un programa CGI, por otra parte, se ejecuta en tiempo real, por lo que puede proporcionar información dinámica.

Por ejemplo, si se desea "conectar" una base de datos de una plataforma UNIX al World Wide Web, para permitir su consulta a usuarios de todo el mundo, se necesita crear un programa CGI que el servidor Web ejecutará para transmitir información al controlador de la base de datos, recibir los resultados de la consulta y desplegarlos al navegador.

El ejemplo de la base de datos es una idea sencilla, pero se requiere de mucho tiempo para implementarla. Pero esto no es realmente una limitante para que se pueda realizar la conexión al Web, únicamente se necesita tener en mente que cualquier cosa que un programa CGI haga, no debe ser un proceso demasiado largo. De otra forma, el usuario que ha solicitado una consulta estará en su Navegador esperando a que algo ocurra.

Como un programa CGI es ejecutable, esto equivale básicamente a permitir que cualquiera corra el programa en el sistema que lo almacena, por lo cual existen medidas de seguridad que deben ser implementadas al usar programas CGI. El único efecto que el usuario probablemente notará, será el hecho de que los programas CGI necesitan residir en un directorio especial, para que el servidor Web lo reconozca antes de desplegarlo al Navegador. Este directorio está usualmente bajo el control del administrador del servidor Web, restringiendo al usuario común la creación de programas CGI. Hay otras formas para permitir el acceso a los scripts de CGI, pero esto depende de las políticas de administración establecidas.

Si se tiene la versión del servidor NCSA HTTPd, existe un directorio denominado `"/cgi-bin"`. Este es el directorio donde todos los programas CGI residen actualmente. Un programa CGI puede ser escrito en cualquier lenguaje que le permita ser ejecutado en el sistema, tal como:

- C/C++
- Fortran
- PERL
- TCL
- SHELL (para cualquier versión de UNIX)
- Visual Basic
- Delphi
- AppletScript

Si se cuenta con un lenguaje de programación como C o Fortran, se necesitan compilar los programas antes de ser ejecutados. En el caso del servidor de NCSA HTTPd el directorio `"/cgi-src"` contiene los códigos fuente de algunos programas CGI que están en el directorio `"/cgi-bin"`. Sin embargo, si se tiene en el sistema un lenguaje script, tal como PERL, TCL o Shell de UNIX, el script en sí mismo debe residir únicamente en el directorio `"/cgi-bin"`, ya que no existe un código fuente asociado. Muchos programadores prefieren

escribir CGI por lotes en vez de programas, ya que son más sencillos de depurar, modificar y mantener que los típicos programas compilados.

Para hacer llegar información a un programa CGI, es necesario que el usuario utilice una estructura de formulario definida por el lenguaje HTML que acabamos de ver. Los elementos de un formulario pueden variar dependiendo de la información que se desea obtener del usuario. A continuación se presenta el funcionamiento de un programa CGI con un servidor Web.

Luego que el usuario ha llenado un formulario de información en un Navegador, el usuario activa un botón (que es un vínculo hacia la aplicación CGI en el servidor).

1. El Navegador utiliza los parámetros de la definición del formulario (los atributos NAME de los elementos definidos), para enviar los datos al servidor.
2. Los datos viajan a través de Internet y llegan al servidor, pasando entonces hacia el programa CGI.
3. Se analizan los datos de la forma, según la petición del cliente, se procesan y el programa CGI "escribe" la respuesta en formato HTML que se enviará de regreso al cliente. La especificación del CGI establece que el servidor Web deberá "leer" el dispositivo definido como salida de datos del programa CGI.
4. El servidor agrega la información necesaria del encabezado HTTP y envía la salida del programa CGI a través de la Internet como un archivo HTML que el Navegador recibe.
5. El Navegador interpreta el código HTML y despliega los resultados en pantalla para el usuario. Este archivo generalmente deberá contener al menos, alguna notificación de que los datos fueron procesados por el servidor.

Como se puede apreciar, los CGI utilizan recursos del servidor que los proporciona. Es decir, el servidor debe contar con los compiladores adecuados, los controladores de bases de datos (si se utilizan bases de datos), etcétera.

Una de las principales características de los formularios es que permiten al usuario interactuar con los CGI, dando lugar a sitios "dinámicos". Los formularios permiten al visitante proporcionar datos, que serán luego procesados por los CGI. Con estos formularios se puede solicitar información para la consulta en una base de datos, solicitar inscripciones, anexar información, entre otros.

## **10.2 Frames**

Los frames nos permiten dividir una página en secciones. Una sección puede ser fija mientras que las demás se utilizan para desplegar información; esto hace que el sitio sea más fácil de navegar.

Supongamos que deseamos dividir en dos partes la ventana del navegador, la parte derecha y la parte izquierda. En este caso necesitamos de tres archivos html, el primero para la parte derecha, el segundo para la parte izquierda y el tercero que describe cómo se distribuyen los primeros frames.

Al tercer frame lo nombraremos *Frameset*. El efecto que se puede observar es que existe una página "padre" que hace referencia a otras denominadas "hijas" a las cuales se accede desde las primeras.

### **10.2.1 El archivo Frameset**

Este archivo describe la distribución de los frames y es un documento HTML



independiente que puede ser nombrado con cualquier nombre válido para el sistema donde se alojará. Existen algunas etiquetas que deben incluirse para este archivo:

<FRAMESET> y <FRAME>.

<FRAMESET [COLS={valor, porcentaje}] | [ROWS={valor, porcentaje}]> ...  
</FRAMESET>

Esta es la etiqueta principal de los frames. Todo frame será declarado entre dichas etiquetas. Es válido declarar nuevos <FRAMESET> dentro de otros. Algunas de las propiedades de esta etiqueta son las siguientes: **Rows** y **Cols**. **Rows** indica los frames que se desplegarán por renglones, por su parte **Cols** indican los frames que se desplegarán en forma de columnas. Estas propiedades son mutuamente excluyentes, es decir, para un <FRAMESET> sólo se puede utilizar **Rows** o **Cols**.

<FRAME [NAME="nombre\_del\_frame"] [SRC={URL}][SCROLLING="yes|no|auto"]  
[NORESIZE]>

Esta etiqueta sirve para la declaración de un simple frame. <FRAME> se coloca entre las etiquetas <FRAMESET> ... </FRAMESET>. Se debe utilizar un <FRAME> por cada frame en el que quiera dividir el área de despliegue del navegador. Algunas propiedades de la etiqueta son: NAME, SRC, SCROLLING y NORESIZE. NAME es utilizado para darle un nombre al frame para que pueda ser referido correctamente con base en las ligas de hipertexto. En nuestro ejemplo es importante que las ligas presenten información en el frame derecho y no en el izquierdo. SRC es parecido al de la etiqueta <IMG>, indica la ubicación del documento HTML que se desplegará en dicho frame. SCROLLING señala si se permite el despliegue de las barras de navegación en el frame. NORESIZE indica si el tamaño del frame puede cambiar automáticamente.

### 10.2.2 Uso de las etiquetas <FRAMESET> y <FRAME>

Los frames pueden colocarse de arriba hacia abajo (atributo COLS) o de izquierda a derecha (atributo ROWS). Cuando se definen los frames, se comienza con la esquina superior izquierda. El siguiente listado muestra un ejemplo del uso de las etiquetas <FRAMESET> y <FRAME>:

```
<FRAMESET COLS="150,*">  
<FRAME NAME="izquierda" URL="ligas.html">  
<FRAME NAME="derecha" URL="pagina_inicio.html">  
</FRAMESET>
```

En primer lugar tenemos que los frames se distribuirán en columnas (de arriba hacia abajo), además la primera columna tendrá un tamaño de 150 pixeles y el otro lo que reste del despliegue.

(representado por el '\*'), se puede utilizar en lugar de valores absolutos los porcentajes relativos al despliegue. El primer frame tendrá el nombre de "izquierda" y el segundo de "derecha". Por otra parte cada frame hace referencia a un documento inicial: ligas.html y pagina\_inicio.html respectivamente. Es importante recalcar que este documento no emplea la etiqueta <BODY>, ya que el primer <FRAMESET> toma su lugar. La Figura 11 muestra un ejemplo con los siguientes listados:

index.html:

```
<FRAMESET COLS="150,*">  
<FRAME NAME="izquierda" SRC="ligas.html">
```

```
<FRAME NAME="derecha" SRC="pagina_inicio.html">
</FRAMESET>
```

ligas.html:

```
<H1>Página de ligas</H1>
<A HREF="p1.html">Página 1</A>
```

página\_inicio.html:

```
<H1>Página de inicio</H1>
Esta página esta la del lado derecho.
```

p1.html:

```
<H1>Página No 1</H1>
```

*Figura 11. Pagina Web con frames.*

Pero existe aquí un problema: cuando se utiliza la liga del frame de la izquierda la página a la que se hace referencia aparece también a la izquierda.

Para lograr que el despliegue se efectúe en el frame de la derecha todas la ligas que se tengan en documento ligas.html deben utilizar el atributo TARGET y darle el valor del nombre del frame correspondiente.

```
<H1>Página de ligas</H1>
<A HREF="p1.html" TARGET="derecha">Página 1</A>.
```

## 11. PUBLICANDO

El planear la puesta en funcionamiento de un servicio de Internet (ya sea Telnet, correo electrónico, transferencia de archivos, etc.), involucra la consideración de muchos aspectos. Uno de los más importantes es el equipo y servidor que se utilizarán.

### 11.1 Tipos de servidores

Existen dos alternativas básicas: obtener un servidor propio o rentar espacio en otro servidor.

Tomar la decisión de levantar y configurar un servidor Web propio está lejos de ser fácil. Cuando se toma en cuenta la cantidad de tiempo y dinero que se puede gastar en este tipo de proyecto, no existe lugar para errores.

Hay una enorme diferencia entre poner su propio sitio Web y poner una página en algún servidor.

Esto último, se refiere a que cualquier persona o institución puede pagar una cantidad de dinero para levantar una página dentro de un sistema de acceso en Internet.

Si se tiene una compañía fuerte, lista para aprovechar el "ciber-mercado", se tiene una buena razón para levantar un servidor Web. Las razones que una compañía debiera considerar para tener su propio sitio Web son muchas: incluir imagen, mercado, proveedor de servicios, y posiblemente la venta directa.

Los institutos educacionales, el lugar de nacimiento de Internet, también tienden a tener sus propios sitios Web. Obviamente, estas instituciones tienen razones de peso para tener sus propios servidores Web: dar a conocer investigaciones, trabajos y

descubrimientos, y compartir información entre centros o instituciones.

Diversas organizaciones profesionales o de investigación también lanzan sus propios sitios Web, tales como el Área de Seguridad en Cómputo (<http://www.super.unam.mx/seguridad/>), el Instituto

Nacional de Investigaciones Nucleares (<http://www.inin.mx/>) y el Instituto Mexicano del Seguro Social (<http://www.imss.gob.mx/>). Numerosos políticos, organizaciones de beneficencia, activistas, religiosos, y otras organizaciones también tienen sus propios sitios.

### **11.2 Sitio propio vs Sitio rentado**

Existen muchas razones que se deben considerar para escoger entre administrar un servidor propio, o bien, pagar un servicio para albergar un sitio. A continuación se muestra un resumen de las ventajas y desventajas de cada opción:

#### **VENTAJAS DESVENTAJAS**

Construir un sitio propio

- Control absoluto sobre el contenido.
- Resulta "fácil" recoger información de visitantes vía formas.
- Capacidad para seleccionar herramientas de seguridad.
- Capacidad para rentar espacio en el servidor.
- La configuración del servidor puede tomar meses.
- Se debe administrar el servidor.
- Requiere un gran y dedicado equipo de administración.

*Ventajas y desventajas de construir un sitio propio*

#### **VENTAJAS DESVENTAJAS**

Pagar por un espacio

- No es muy caro.
- No requiere sistema de mantenimiento.
- Requerimientos mínimos de personal.
- Dependencia del proveedor del sitio sobre seguridad y mantenimiento.
- Depende del proveedor para cambios de contenidos.
- Puede no tener capacidad para implementar formas y programas

*Ventajas y desventajas de pagar por un sitio*

En general, la decisión dependerá de las necesidades y el presupuesto con que se cuente. Si el sitio es pequeño y la expectativa de tráfico es ligera, lo mejor sería rentar el espacio. Si el sitio es grande y activo —esto es, si mucha gente estará agregando datos y visitando el sitio— sería ventajoso tener un servidor propio.

### **11.3 Aspectos a evaluar**

Como se observa, para cada alternativa existen ventajas y desventajas. Cada una de estas razones debe ser analizada por el equipo de personas encargadas a fin de atender

las necesidades particulares del sitio que se desea crear.

Otro aspecto importante a considerar cuando se decide tener un servidor Web es el equipo donde será instalado. Dependiendo de la plataforma existen diferentes tipos de servidores, los cuales presentan características particulares en cuanto al desempeño, instalación, extensibilidad, seguridad que se puede implantar y mantenimiento. Cada uno de estos aspectos se describen a continuación.

#### · **Desempeño**

Se entiende por desempeño las características con que cuenta un equipo respecto a su rendimiento.

Esto es, la capacidad de atender las peticiones de los clientes (cuántos puede atender a la vez, la velocidad de respuesta, tipo de procesador, etc.). El sistema operativo que se utiliza en cada equipo es también un factor importante para el rendimiento de un servidor; un sistema operativo multi-usuario ofrece una base de desarrollo mejor que los sistemas mono-usuarios.

#### · **Instalación**

Es el proceso en el cual se establecen las características de funcionamiento del servidor. Va desde la adquisición del *software*, configuración (según requerimientos del sistema donde se monte, etc.), hasta su puesta en funcionamiento.

#### · **Extensibilidad**

En este punto se analizan las características de crecimiento de la plataforma. Es decir, si el equipo puede ser utilizado para más de un servicio, si pueden ser ampliados los recursos, tanto en *hardware* como en *software*, y la capacidad de utilizar herramientas nuevas, entre otros aspectos.

#### · **Seguridad**

Es uno de los aspectos más importantes que se toman en cuenta para el ofrecimiento de servicios en Internet. En los últimos años, la seguridad en las transacciones de información se ha vuelto vital, a tal grado, que se cuenta con organizaciones que buscan y publican errores o "huecos" de seguridad en los programas y equipos utilizados Internet. Estos huecos de seguridad pueden ser, por ejemplo, el paso de comandos de sistema a través de programas a fin de obtener información vital del sistema (números confidenciales, contraseñas, etc.), o bien, obtener privilegios sobre recursos del equipo.

#### · **Mantenimiento**

El mantenimiento tiene por objetivo que los servicios colocados en el servidor se mantengan trabajando en condiciones óptimas y se vayan adaptando a las necesidades que van surgiendo durante su uso. Dicho mantenimiento comprende actividades de monitoreo del sistema, detección y corrección de algún mal funcionamiento, actualización de datos, etcétera.

### **11.4 Servicios básicos a considerar**

Aunque los servicios que actualmente se pueden contratar para la colocación de sitios Web son extensos y varían de proveedor a proveedor, a continuación se lista los que son importantes para el establecimiento de un sitio Web.

a) *Conexión a Internet*. Este punto se refiere al tipo de conexión que se contrata con el proveedor.

Es importante considerar los tipos de conexión tales como módem, conexión directa, etc., a fin de determinar si la infraestructura de conexión es la adecuada para el sitio que se desarrolla.

b) *Diseño y mantenimiento del sitio*. Muchos proveedores de servicios añaden la elaboración y diseño del sitio Web como parte del contrato correspondiente. Los precios, sin embargo, varían por el número de imágenes, tipografía y otros elementos particulares que se desean incorporar a las páginas del sitio.

c) *Espacio de alojamiento*. Dentro del contrato es importante precisar el espacio total con el que se dispone para el desarrollo y puesta en funcionamiento del sitio Web. En la mayoría de los casos el espacio extra que llegara a requerirse repercute de forma directa en el costo total del sitio.

d) *Elaboración de programas CGI en el servidor*. Dependiendo de las capacidades del servidor que alojará el sitio, es posible generar programas y servicios agregados al sitio. Es importante recalcar que dependiendo de la infraestructura, sistema operativo y herramientas de programación adecuadas al servidor, es posible ofrecer la consulta a catálogos de información, conexión a bases de datos en línea, o bien, servicios de comercio electrónico.

e) *Soporte técnico*. Es necesario conocer las capacidades técnicas con las que el proveedor cuenta para la solución de problemas que se van generando con el desarrollo y mantenimiento del sitio.

En este sentido es necesario verificar las condiciones del contrato y su cabal aplicación.

f) *Otros servicios*. Cada proveedor ofrece una cartera de soluciones propias las cuales se ofrecen al usuario a fin de complementar los servicios básicos anteriormente listados. De entre ellos destacan los siguientes:

- Incorporación de las páginas del sitio a motores de búsqueda nacionales y/o internacionales.
- Motores de búsqueda locales.
- Desarrollo de la imagen corporativa.
- Acceso a bases de datos en línea.

## **12. HERRAMIENTAS DE PRODUCCIÓN**

Con el auge en el diseño, la elaboración y puesta en funcionamiento de sitios Web a nivel mundial, se han generado una amplia variedad de herramientas de producción y de referencia. Desde la aparición de editores de HTML hasta la generación y administración de sitios Web, todas estas herramientas permiten a los desarrolladores generar sitios cada vez más complejos con base en la creatividad y ritmo de desarrollo de los mismos. A continuación se presenta un breve resumen de las herramientas más importantes que se deben considerar para el desarrollo de sitios Web profesionales.

### **12.1 Editores**

Los editores de HTML trabajan normalmente desde el desarrollo general del sitio, es decir, la distribución de páginas y los niveles de las mismas, la verificación de estilos, entre otros.

## **12.2 Convertidores**

Otro tipo de herramientas útiles para el desarrollador son los convertidores. Es común encontrar documentos elaborados por herramientas de edición de texto ajenas al HTML, de los cuales se desea producir su versión para el Web. Algunas herramientas ya cuentan con alguna opción incorporada para tal efecto.

Otra opción que es común encontrar es la de un programa externo que realiza la conversión del formato original a un documento HTML. Esta última opción depende del formato y su flexibilidad para generar un documento HTML, por lo que se recomienda consultar al fabricante de su aplicación a fin de obtener uno de estos programas convertidores.

## **12.3 Editores de mapas**

Una tarea en exceso laboriosa, es la de generar un mapa a partir de una imagen. Un mapa es una imagen dividida en distintas áreas de toque, donde cada una de ellas nos envía a sitios distintos, según se dé clic en alguna de las mismas. Para generar un mapa se requiere de las coordenadas asociadas a las secciones en que se dividen. Las áreas de toque que se pueden generar son las siguientes: puntos, rectángulos, polígonos y círculos. Si la imagen resulta complicada esta tarea podría no resultar trivial.

## **13. CONSIDERACIONES DE DISEÑO**

El desarrollo actual de páginas para el Web involucra la combinación de distintos y muy variados elementos para conformar uno solo. Como se ha visto en los capítulos anteriores, es fácil incluir elementos que enriquecen y resaltan los formatos de la información que se desea ofrecer por Internet. En resumen, los elementos técnicos ofrecen un conjunto de herramientas, especificaciones y nuevos atributos que día con día extienden las capacidades de despliegue visual de los navegadores.

Pero todo esto tiene un costo en tiempo de análisis, de producción y especialmente del diseño. Estas consideraciones van desde una página personal hasta la producción de un sitio completo.

En la presente sección se abordan distintos puntos para la producción de sitios Web con base en consideraciones de diseño y producción.

### **13.1 Recomendaciones para la elaboración de sitios Web**

La capacidad del Web para publicar documentos a nivel mundial presenta varias problemáticas que se listan a continuación.

- Diferencias culturales.
- Contenido de una página.
- Distribución de la información en el sitio.
- Velocidad de transferencia.

#### **13.1.1 Diferencias culturales**

En el mundo existen una amplia variedad de personas de distintas sociedades, creencias y pensamiento que se encuentran en contacto por medio de Internet. Los documentos que se generan llegan hasta estos grupos y el mensaje del sitio o documento puede ser

tomado de forma distinta por cada uno de ellos. Es un buen consejo tener en mente que una página tiene un significado distinto para cada uno de nosotros y por ello lo que se considera de buen gusto para uno no lo es para los demás.

### **13.1.2 Contenido de una página**

Otro aspecto importante es el contenido y mensaje de un documento. No es igual elaborar el mensaje de una página para una empresa que vende computadoras, que para una institución de educación superior. Una vez que se ha definido el objetivo del sitio el proceso no concluye en este punto. Un documento Web siempre debe presentar alguna mejora o cambio en su contenido. Una buena referencia son las bitácoras de acceso al sitio, las cuales ofrecen información importante:

cuántas personas han visitado el sitio, desde dónde se ha visitado, qué páginas han visitado, entre otras.

### **13.1.3 Distribución de la información en el sitio**

Cuando se ha definido la información que se pondrá a disposición por el Web, ésta se debe organizar para su fácil consulta y referencia en el sitio. Existe más de una forma de lograr una buena distribución de la información, y esto también depende de la naturaleza de la misma. Un error común es el de querer conjuntar la información en una sola página, generando con esto documentos que se extienden por más de una pantalla para su despliegue en el navegador. Estudios recientes indican que el 90% de los usuarios busca hasta un equivalente de dos pantallas para el despliegue de información por página Web que consulta, después de lo cual sale del documento en busca de otro.

### **13.1.4 Velocidad de transferencia**

A pesar de las mejoras en el envío de información por Internet, muchos usuarios no cuentan con los elementos necesarios para ver el despliegue completo de un documento en un tiempo "razonable".

En general este tiempo varía en relación con varios factores: tránsito en la red, ancho de banda de transferencia, número de imágenes a transferir, etcétera.

## **13.2 El proceso de diseño**

A medida que se van conociendo el funcionamiento y los elementos propios del HTML podemos darnos cuenta de la gran variedad de diseños que podemos incluir en los documentos Web. Aunada a la discusión anterior, se debe señalar que el proceso de creación de un sitio no concluye con su puesta en funcionamiento en un servidor. Esto es solamente el principio. El proceso de diseño es uno más de los que integran la producción del sitio en su totalidad. Para lograr que un diseño sea adecuado a los objetivos establecidos se deben considerar los siguientes puntos:

- Capacidades del navegador.
- Plataforma de despliegue.
- Imágenes y composición de las páginas.
- Imagen corporativa.

### **13.2.1 Capacidades del navegador**

El lenguaje HTML se enriquece en la medida en que los navegadores progresan. Al final, si un documento se despliega, su apariencia se verá afectada por el navegador que se utilice.

Es importante señalar que en el desarrollo de un sitio se debe establecer desde el inicio los navegadores que se utilizarán para el diseño. Las soluciones para este problema están orientadas por la combinación de las siguientes características:

- Navegador que el usuario utiliza.
- Juego de etiquetas HTML para dichos navegadores.
- Equipo y sistema operativo que el usuario utiliza (Windows, Macintosh, UNIX, etcétera).

### **13.2.2 Plataforma de despliegue**

Al igual que el punto anterior es necesario considerar la plataforma desde la cual se accede el documento, ya que varía la calidad de un equipo a otro, por ejemplo Windows y Mac. Estas diferencias en resolución y calidad limitan el proceso de desarrollo y la calidad del mismo.

Aún dentro una misma plataforma, la calidad de las imágenes desplegadas, el tipo de letra que se utiliza y los efectos especiales que se utilizan pueden variar de una a otra.

Existen páginas cuyo nivel de despliegue óptimo se establecen desde el inicio, por ejemplo: 800 x 600 a 256 colores.

La solución es solo una: evaluar los documentos en estas plataformas y establecer el mínimo de calidad de todas ellas, o bien, crear un sitio para cada plataforma.

### **13.2.3 Imágenes y composición de las páginas**

Como se mencionó en el uso de la etiqueta <IMG>, cada vez que se le utiliza el navegador establece una petición para dicha imagen. En otras palabras, para cada imagen que se haga referencia desde una página se trae cada una al mismo tiempo, lo cual repercute en el tiempo de despliegue.

En este sentido se recomiendan varias alternativas para la integración de imágenes:

- Para fotografías e imágenes digitalizadas utilizar el formato JPEG tratando de establecer un promedio de 70 Kb para su tamaño.
- Los fondos, botones, imágenes animadas y cualquier otro elemento artístico producirlos en GIF, con un promedio de 40 Kb de tamaño.
- Los fondos no deben estorbar a la lectura del documento y si resaltarlos.

En cualquiera de los formatos de las imágenes, es importante definir una paleta de colores, ya que las resoluciones varían de un equipo a otro. Por lo general la paleta de colores utilizada es la de 256 colores, pero resulta en exceso pobre para los efectos artísticos la página, por lo que es frecuente encontrar paletas de colores de 16 bits o más, además de que se recomienda el uso de una paleta lo más ligera posible a fin de que el despliegue sea de la misma calidad en todos los equipos posibles.

### **13.2.4 Imagen corporativa**

Para sitios profesionales es importante mantener un aspecto corporativo en todos los



documentos que se desarrollan. Una técnica efectiva en este sentido es el establecimiento de plantillas para las distintas páginas del mismo. La base se toma en la página principal y de ahí se definen los diseños del segundo nivel y sucesivos. Las plantillas deben establecer con claridad los elementos que identificarán al sitio durante la navegación del mismo, procurando generar fórmulas y patrones que pueden ser reproducidos fácilmente.

Estas plantillas ayudan a organizar la forma de navegación y distribución de la información del sitio de forma gráfica y más accesible a los desarrolladores. En la Figura 3 se observa un ejemplo de plantilla con la correspondiente distribución de información en un sitio.

Una vez definida la imagen institucional, los documentos permiten en su conjunto establecer un sitio bajo el mismo diseño e ideas de desarrollo.

Se debe señalar que el uso de las mismas no es una regla rígida, sino más bien flexible, pero mantienen la identidad de un sitio en su totalidad.

Logotipo

Encabezado

Pie de página.

## **BIBLIOGRAFÍA**

Aprendiendo HTML 3.0 para WEB en una semana. Laura Lemay. Prentice-Hall Hispano Americana, SA. 2ª edición. 1996.

El gran libro de HTML. Joachim Shwarte. Marcombo, SA. 1996.

HTML 4. Guía de referencia y tutorial. José Luis Raya, José A. Moreno, Antonio López. ALFAOMEGA Grupo Editorial, SA de CV. 1999.

HTML. La guía completa. Chuck Musciano y Bill Kennedy. Colección O'REILLY. McGraw-Hill Interamericana Editores, SA de CV. 1999.

HTML. The Definitive Guide. Chuck Musciano & Bill Kennedy. Editorial O'Reilly. 3ª edición. Agosto, 1998.

Dynamic HTML. The Definitive Reference. Danny Goodman. Editorial O'Reilly. 1ª edición. Julio, 1998.

HTML 3.2. Referencia visual. Dean Scharf. Prentice-Hall Hispanoamericana. 2ª edición. 1997.

HTML avanzado. Bruce Morris. McGraw-Hill Interamericana Editores, SA de CV. 1997.

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

DR. JUAN RAMÓN DE LA FUENTE

Rector

LIC. ENRIQUE DEL VAL BLANCO

Secretario General

MTRO. DANIEL BARRERA PÉREZ

Secretario Administrativo

LIC. ALBERTO PÉREZ BLAS

Secretario de la Rectoría

DRA. ELVIA ARCELIA QUINTANA ADRIANO

Abogada General

DR. JOSÉ NARRO ROBLES

Coordinador General de Reforma Universitaria

DIRECCIÓN GENERAL DE SERVICIOS DE CÓMPUTO

CADÉMICO

DR. VÍCTOR GUERRA ORTIZ

Director General

MTRA. FERNANDA MERAZ RODRÍGUEZ

Subdirectora

MAT. CARMEN BRAVO CHAVESTE

Directora de Cómputo para la Docencia

LIC. ROSARIO ORTIZ OSORNIO

Subdirectora de Planeación Académica

ING. SERGIO DE ALVA ARGUINZONIZ

Coordinador del Centro Mascarones

LIC. ALEJANDRINA SAN JUAN REYES

Coordinadora del Centro de Extensión en Cómputo

y Telecomunicaciones Nuevo León

LIC. CECILIA MEDINA NAVARRO

Coordinadora del Centro Coapa de Extensión

en Cómputo y Telecomunicaciones

ACT. FABIÁN ROMO ZAMUDIO

Coordinador de Educación a Distancia

QUIM. LAURA MATA MONTIEL

Coordinadora de Infraestructura y Soporte Tecnológico

ACADEMIA DE PROFESORES DE TELECOMUNICACIONES

Revisión Técnica

LIC. SEMIRAMIS ZALDÍVAR GRANADA

Revisión Didáctica

DG TOÑA ZIMERMAN SONABEND

Jefa del Departamento de Diseño

LIC. MARTHA PATRICIA GARCÍA MORALES

Corrección de estilo

DG GABRIELA LILÍ MORALES NARANJO

Diseño editorial

DG CRISTINA GISPERT GALVÁN

Diseño de portada

Marzo de 2000