



UNIVERSIDAD
DE MURCIA



*Área de Tecnologías de la Información y
las Comunicaciones Aplicadas*

MANUAL BÁSICO DE CREACIÓN DE PÁGINAS WEB



INDICE

<u>WORLD WIDE WEB</u>	6
Utilización del servicio World Wide Web.	6
Servicio de Almacenamiento de páginas WWW de la UM	7
<u>Introducción</u>	9
<u>EI LENGUAJE HTML.</u>	9
<u>MI PRIMER DOCUMENTO HTML.</u>	10
<u>CARACTERÍSTICAS GENERALES DEL LENGUAJE HTML.</u>	11
Marcas y atributos.	11
Concatenación de marcas.	12
<u>CUATRO NORMAS FUNDAMENTALES</u>	12
HTML es simplemente texto	12
Igualdad de mayúsculas y minúsculas.	13
No importan los tabuladores, ni los saltos de línea	13
Caracteres especiales	13
<u>ESTRUCTURA DE UN DOCUMENTO HTML.</u>	14
<u>COMANDOS BÁSICOS DE HTML</u>	14
Definición de párrafos: P	15
Ruptura de líneas: BR.	15
Titulos de encabezamiento	15
Dando estilo al texto	16
Tipos de letras	16
Tamaño del texto.	16
Colores del texto.	17
Listas	18
Listas ordenadas: OL.	18
Listas desordenadas: UL.	19
Listas de definición: DL.	19
Texto preformateado: PRE.	20
Centrando texto.	20
Líneas horizontales.	20
Comentarios no visibles en la pantalla	21
<u>ENLACES: A.</u>	23
Enlaces a otras zonas de la misma página	24
Enlaces a otras páginas	27
Enlaces a una zona de otra página	29

<i>IMÁGENES</i>	29
Principales formatos gráficos en Internet.	30
Formato GIF.	30
Formato JPEG.	30
Inclusión de imágenes	31
Tamaño de las imágenes.	31
Escalado de imágenes.	31
Bordes de la imagen.	31
Alineación de texto con las imágenes.	32
Imágenes con texto alternativo.	32
Uso de imágenes como enlaces.	32
Imágenes como fondo de un documento.	32
Mapas sensibles.	33
<i>TABLAS</i>	36
Título de la tabla.	36
Bordes de la tabla.	37
Cabeceras de filas y de columnas.	37
Tamaño de la tabla.	37
Justificación de la tabla.	38
Espacio dentro de las celdas.	38
Forma de las celdas individuales.	38
Justificación del texto de las celdas.	38
Celdas de diferentes tamaños.	39
Celdas irregulares.	39
Color de cada celda.	39
Texto en una sola línea.	40
<i>FRAMES (MARCOS)</i>	40
<i>Formularios (FORMS)</i>	44
Campos de entrada	45
Campos de selección	47
Áreas de texto	48
<i>Hojas de estilo</i>	51
<i>EDITORES Y CONVERSORES</i>	52
Editores	52
Conversores	53
<i>PUBLICAR LAS PÁGINAS EN INTERNET.</i>	53
<i>DIRECCIONES INTERNET DE INTERÉS.</i>	54
<i>BIBLIOGRAFÍA.</i>	55



INTRODUCCIÓN AL WORLD WIDE WEB

WORLD WIDE WEB

El servicio **World Wide Web** (La Telaraña Mundial), también conocido como **WWW** o simplemente **Web**, es un Sistema de Información distribuido por Internet basado en la tecnología hipertexto/hipermedia, que proporciona una interface común a los distintos formatos de datos (texto, gráficos, vídeo, audio, etc.) y a los servicios de Internet existentes (FTP, news, telnet,...). Todo esto hace que el servicio Web sea el servicio de más utilizado en Internet.

Un **documento hipertexto**, es un texto en que cualquier palabra puede ser especificada como un enlace a otros documentos que contienen más información sobre dicha palabra, por lo que la lectura de un documento hipertexto no es secuencial o lineal, sino que se puede acceder a la información que nos interese desde otros conceptos relacionados (simplemente haciendo clic con el ratón en la palabra relacionada), y de esta forma avanzar de documento en documento hasta encontrar la información deseada. Estas palabras que poseen enlaces a otros documentos están marcadas de alguna manera para poder diferenciarlas.

Un **documento hipermedia** es un hipertexto, pero que no incluye sólo información textual sino también información multimedia, es decir, puede incluir gráficos, vídeo, y sonido.

A pesar de las diferencias de estos dos conceptos, a menudo se utiliza el término hipertexto para designar el significado de hipermedia.

Los **documentos Web** o también llamados **páginas Web** pueden estar localizados en diferentes sitios de Internet, estos sitios son llamados servidores Web. De manera que un documento WWW puede contener enlaces a otros documentos que se encuentran en el mismo servidor Web o en otros servidores Web, logrando así formar una telaraña mundial de información.

El lenguaje estandarizado para la creación de páginas Web es el **lenguaje HTML** (HyperText Markup Language, Lenguaje de Marcas Hipertexto). HTML es un lenguaje muy sencillo que permite describir documentos hipertexto. La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc) así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado).

Utilización del servicio World Wide Web.

Para utilizar el servicio Web se necesita una aplicación cliente capaz de entender o interpretar información HTML, a este tipo de aplicaciones se le conoce como **browsers o navegadores o visualizadores o hojeadores**. Mediante el browser el usuario puede acceder a los documentos HTML y moverse de un documento a otro a través de sus vínculos o enlaces, este hecho de moverse con el browser por las páginas WWW a través de sus enlaces se le conoce como Navegar por Internet.



El browser sabe cómo acceder a cada recurso de Internet, sabe cómo acceder a un servidor de FTP anónimo, a un servidor de News, ..., y por supuesto cómo conectarse a los servidores Web. El mecanismo que utiliza el browser para acceder a un recurso en cualquier lugar de Internet es el **URL** (Uniform Resource Locator, Localizador de Recursos Uniforme), comúnmente llamado dirección Internet.

Los URLs combinan el protocolo a utilizar para obtener el recurso: http (es el del Web), ftp, telnet, ...; junto con el nombre del host servidor, y el path completo del recurso (directorios y nombre de archivo). Los URLs constituyen en realidad los enlaces que permiten movernos de una página a otra, es decir Navegar por Internet, y que las podemos identificar dentro de una página WWW porque estas incluidos comúnmente como Texto en color subrayado, también puede ir incluidos dentro de una imagen, etc.

Servicio de Almacenamiento de páginas WWW de la UM

Con motivo de la gran demanda del Servicio de Información World Wide Web el Servicio de Informática ofrece a los Centros, Departamentos, Servicios, Grupos de trabajo e investigación universitarios la posibilidad de incorporar sus propias páginas de Web al servidor central de la Universidad de Murcia. De esta manera, la unidad podrá hacer pública información de interés de la misma (trabajos, personas del grupo, seminarios, programas, etc.) para que sea visualizada desde toda la Red Internet.

Este servicio se ofrece con carácter genérico a los distintas unidades, con el objeto de introducir información de interés del departamento o unidad, y en ningún caso se podrá utilizar con fines no académicos o de I+D.

Para poder hacer uso de este servicio es necesario solicitar el alta de una cuenta por centro, departamento, grupo en el servidor WWW del Área de Tecnologías de la Información y las Comunicaciones Aplicadas (ATICA) www.um.es, para ello hay que leer normas del servicio WWW y rellenar y enviar el formulario de WWW que se encuentran en <http://www.um.es/atica/web/>

En breve se le comunicara que ha sido de alta y que ya puede utilizarlo.

Esta cuenta permitirá la ubicación de las paginas WWW que deberán ser creadas con los medios propios de la sección correspondiente.

HTML



Introducción

Este manual está pensado para aprender los conceptos básicos necesarios para construir sus propias páginas Web. Para quien quiera ampliar sus conocimientos se suministran también unas referencias de consulta a otros sitios del Web.

En su confección se ha procurado seguir siempre un criterio eminentemente práctico. Para mejor asimilar los conceptos tratados, se propone al final de cada apartado un ejemplo práctico.

Primeras herramientas: Para comenzar sólo son necesarios dos elementos:

- *Un editor de texto.* Vale cualquiera, siempre que no formatee el texto. Por ello, los más adecuados son los más sencillos (como por ej., el block de notas de Windows 95). Si se utilizan procesadores como el Word, se deben guardar los ficheros como 'Sólo Texto', para que no introduzca órdenes de formateo, que pueden provocar errores al cargarlo en el navegador.
- *Un visualizador o navegador* del Web: Netscape, Microsoft Explorer, Mosaic, Opera, etc.

El editor de texto lo usaremos para ir escribiendo los documentos HTML, que será posteriormente interpretado por el navegador, con lo que iremos comprobando los cambios y añadidos que vayamos efectuando.

Existen ciertos programas que nos ayudan a automatizar este proceso, pero es muy conveniente comenzar a hacerlo de una manera manual, para comprender bien la estructura del lenguaje HTML

Método de trabajo: Con el editor de texto crearemos un fichero con el nombre que queramos (p. ej. ejemplo1), pero que debe tener necesariamente la extensión .html (o .htm si nuestro sistema operativo no soporta extensiones de más de tres letras).

Habrá un fichero distinto para cada apartado; conviene crear un directorio específico en su ordenador e irlos guardando en él, para poder repasar lo aprendido, aparte de que pueden ser necesarios para ejecutar otros ejemplos prácticos.

EI LENGUAJE HTML.

HTML es un lenguaje que se utiliza para la creación de páginas en la WWW. Por página entenderemos el documento que aparece en el visualizador o navegador.

HTML se compone de una serie de comandos, que son interpretados por el visualizador, o programa que utilizamos para navegar por el WWW. En última instancia es el visualizador el que ejecuta todas las órdenes contenidas en el código HTML, de forma que un visualizador puede estar capacitado para unas prestaciones, pero no para otras. Así, podremos especificar que una página tenga una imagen de fondo, o un texto parpadeando, pero si nuestro visualizador no está capacitado para esas funciones, no podremos verlas.

En esta guía se expondrán los comandos fundamentales de HTML.

MI PRIMER DOCUMENTO HTML.

Siga las siguientes instrucciones atentamente; ellas le permitirán crear un documento HTML en su ordenador.

Le recomiendo que cree un directorio en su ordenador para almacenar las páginas web que vaya haciendo al aprender; y que practique un tiempo antes de poner páginas en Internet.

1. Abra el editor de textos: en Windows 95 abra el Block de Notas
2. Teclee lo siguiente:

```
<HTML>

<HEAD>
<TITLE>Ejemplo 1 - Mi primera pagina Web</TITLE>
</HEAD>

<BODY>
<CENTER><H1>Mi primera pagina Web</H1></CENTER>
<HR>
Esta es mi primera pagina, aunque todavía es muy sencilla. <BR>Como
el lenguaje HTML no es difícil, pronto estaré en condiciones de hacer
cosas mas interesantes.
<P> Aquí va un segundo párrafo.</P>
</BODY>

</HTML>
```

3. Grabe este archivo con el nombre: *ejemplo1.html*
4. Abra el visualizador. No necesita conectarse a Internet para ver las páginas en su ordenador. Puede trabajar en local.
5. Elija "Archivo/Abrir página" en la barra de menú del navegador.
6. Seleccione el archivo ejemplo1.html que acaba de crear, y debe de ver la siguiente pantalla:



Mi primera pagina Web

Esta es mi primera pagina, aunque todavía es muy sencilla.

Como el lenguaje HTML no es difícil, pronto estaré en condiciones de hacer cosas mas interesantes.

Aquí va un segundo párrafo.

Las líneas en blanco y las indentaciones del texto se han puesto para mayor claridad, pero no son necesarias. De hecho, podría estar todo en una sola línea. Lo importante es el orden correcto de las etiquetas. Por cierto, una etiqueta puede estar anidada dentro de otra. Véase en el ejemplo cómo lo está la etiqueta <CENTER> dentro de la etiqueta <H1>. Es muy importante, en estos casos, que las etiquetas de inicio y de cierre vayan en el orden correcto, pues de lo contrario se producirían errores.

¡Usted ha creado su primer documento HTML!

CARACTERÍSTICAS GENERALES DEL LENGUAJE HTML.

Marcas y atributos.

El lenguaje HTML se estructura utilizando marcas o etiquetas o comandos (a partir de ahora utilizaremos indistintamente uno de 3 términos para denominar a los elementos que se estructura HTML). La forma general de una marca es la de un comando HTML encerrado entre dos signos de menor y mayor como a continuación se muestra:

<marca [atributos]>[</marca>]

El mecanismo de funcionamiento de estas marcas es muy sencillo. Cuando el visualizador encuentra el signo menor (<), examina todos los caracteres hasta que encuentra el final de la marca - el símbolo mayor (>). Entonces, interpreta el contenido de la marca, y aplica esa propiedad al texto que viene a continuación.

Hay marcas que se aplican a todo el documento HTML, o sólo desde el punto en que son insertadas hasta el final del documento. Otras se aplican exclusivamente a un fragmento del texto. En ese caso, el final de la aplicación se especifica con la misma marca precedida de la barra inclinada hacia atrás (/).

Las marcas pueden contener de forma opcional u obligatoria, lo que se denominan atributos o modificadores. Los atributos matizan el significado de la marca, y que se expresan de la siguiente forma:

<marca atrib1="valor1" atrib2="valor2">

El valor de los atributos se expresan encerrados entre comillas.

En la mayor parte de los visualizadores es posible omitir las comillas y colocar directamente el valor del atributo. Esta práctica, a pesar de estar extendida, no es muy aconsejable ya que no está normalizada, y no esta soportada por la totalidad de los visualizadores.

Algunos atributos sólo viene definidos por su nombre (no tienen valor); son los atributos llamados *compactos*.

Concatenación de marcas.

Las marcas se pueden anidar o encadenar una a continuación de otra, de forma que se pueden aplicar simultáneamente varias propiedades a una misma porción de documento. Así el texto encerrado en las marcas:

<MARCA1><MARCA2> Texto </MARCA2></MARCA1>

queda afectado tanto por las propiedades de la MARCA1 como por los de la MARCA2. Todas las marcas son independientes entre sí, por los las siguientes líneas de código HTML, tiene efectos idénticos, sean cuales sean las marcas concretas:

<MARCA1><MARCA2> Texto </MARCA2></MARCA1>
<MARCA1><MARCA2> Texto </MARCA1></MARCA2>
<MARCA2><MARCA1> Texto </MARCA1></MARCA2>
<MARCA2><MARCA1> Texto </MARCA2></MARCA1>

A pesar de que son expresiones completamente idénticas, es recomendable seguir el orden lógico de la concatenación, ya que esto facilita de forma sustancial la edición de documentos HTML. De las cuatro opciones anteriores, las correctas serían la primera y la tercera.

CUATRO NORMAS FUNDAMENTALES

HTML es simplemente texto

Lo primero es saber que un documento HTML es un archivo de texto simple, luego, se puede editar con cualquier editor de texto sencillo, como por ejemplo el Block de Notas de Windows 95/98/NT.



Igualdad de mayúsculas y minúsculas.

HTML no distingue entre mayúsculas y minúsculas en la especificación de marcas y sus atributos. Sin embargo, por legibilidad, es aconsejable codificar tanto marcas como atributos en mayúsculas.

No importan los tabuladores, ni los saltos de línea

Los visualizadores no toman en cuenta las tabulaciones, los saltos de línea ni los espacios en blanco extra. Esto tiene ventajas y desventajas. La principal ventaja es que permite obtener resultados uniformes y de buena presentación de manera bastante fácil.

La principal desventaja es que un documento HTML, por lo menos se debe usar las marcas `<P>...</P>` o `
` para evitar que quede todo el texto en una sola línea.

Caracteres especiales

En HTML existen algunos caracteres que son especiales porque juegan un papel dentro del mecanismo del funcionamiento de HTML, como sucede con los símbolos mayor que (`>`) y menor que (`<`), y otros porque en los primeros tiempos de HTML, no formaban parte del juego de caracteres internacionales del alfabeto, como sucede con los acentos.

Sea por los motivos que fuere, el caso es que existen ciertos símbolos que no pueden escribirse directamente, sino que deben sustituirse por una cadena de caracteres que el visualizador interpretará de forma correcta. Estas cadenas de caracteres comienzan siempre por el símbolo (`&`) seguido de una combinación de caracteres alfabéticos que tienen un significado especial en HTML. En la siguiente tabla se muestra cómo escribir algunos de estos caracteres:

Carácter especial	Transcripción HTML	Comentario
Los acentos	<code>&<vocal a acentuar> acute;</code>	En la actualidad la mayoría de los browsers los soportan
Símbolo menor que (<code><</code>)	<code>&lt;</code>	
Símbolo mayor que (<code>></code>)	<code>&gt;</code>	
La eñe	<code>&ntilde;</code>	En la actualidad la mayoría de los browsers la soportan
Espacios en blanco	<code>&nbsp;</code> (non breaking space)	HTML sólo reconoce un espacio en blanco entre palabras

De manera más general, para representar símbolos reservados o símbolos particulares, se puede utilizar el código ASCII del símbolo mediante la sintaxis siguiente:

&#c3b3digo_ASCII;

ESTRUCTURA DE UN DOCUMENTO HTML.

Un documento HTML, no es m3s que el texto definido entre las marcas:

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

Un documento HTML siempre se compone de las siguientes 2 partes:

```
<HTML>  
  
    <HEAD> Cabecera del documento  
    </HEAD>  
  
    <BODY> Contenido del documento  
    </BODY>  
  
</HTML>
```

- **Cabecera:** Se inicia mediante el comando <HEAD> y se termina con </HEAD>. Dentro de la cabecera hay informaci3n del documento, que no se ve en la pantalla principal, y que precisa las caracter3sticas del documento, principalmente el t3tulo del documento. El t3tulo del documento se declara entre las etiquetas <TITLE> y </TITLE>. El t3tulo debe ser breve y descriptivo de su contenido, pues ser3 lo que vean los dem3s cuando a3adan nuestra p3gina a su bookmark (o libro de direcciones favoritas).
- **Cuerpo:** se inicia mediante el comando <BODY> y se termina con el comando </BODY>. Este comando acepta numerosos modificadores. Dentro del cuerpo del documento se incluye cualquier car3cter imprimible.

En la pr3ctica algunos visualizadores no necesitan las etiquetas de comienzo y cierre de <HTML>, <HEAD>, y <BODY> para interpretar un documento HTML. Sin embargo, cuando dise3emos un p3gina Web debemos tener en cuenta a la mayor3a de usuarios posibles, por lo que es muy recomendable incluir estas marcas.

COMANDOS B3SICOS DE HTML

A continuaci3n se describen las marcas b3sicas de HTML que se pueden incluir en el cuerpo de un documento HTML.



Definición de párrafos: P

La marca <P> sirve para separar párrafos en HTML. Además de introducir un retorno de carro, fuerza un segundo retorno de carro para dejar una línea en blanco entre un párrafo y el siguiente. De esta manera los párrafos quedan más espaciados y el texto se vuelve más legible

En esta marca la etiqueta de fin es opcional.

Por defecto, el párrafo de un documento HTML está justificado a la izquierda. Pero podemos modificar la alineación horizontal de un párrafo mediante el atributo ALIGN de la marca <P>, que puede tomar los valores siguientes.

- LEFT: el párrafo es justificado a la izquierda. Valor por defecto.
- CENTER: el párrafo es centrado.
- RIGHT: el párrafo es justificado a la derecha.

Ruptura de líneas: BR.

La marca
 introduce un retorno de carro (o salto de línea) en el punto del documento en el que es colocada. Es equivalente al punto y aparte de un texto normal.

Esta marca es vacía de manera que no necesita de la etiqueta de fin de marca.

Titulos de encabezamiento

HTML dispone de seis cabeceras, o tipos de letra predefinidos de distinto tamaño, que se utilizan para marcar los títulos o resaltes en las páginas. Se activan con el comando <Hn> y se desactivan con </Hn>, donde n es un número de 1 a 6. siendo el número indicativo del tamaño. El tamaño mayor es el correspondiente al número 1.

```
<H1>Nivel de encabezado 1</H1>  
<H2>Nivel de encabezado 2</H2>  
<H3>Nivel de encabezado 3</H3>  
<H4>Nivel de encabezado 4</H4>  
<H5>Nivel de encabezado 5</H5>  
<H6>Nivel de encabezado 6</H6>
```

Puede experimentar en el ejemplo anterior, cambiando el número para comprobar el efecto que se logra.

Los títulos de encabezado requieren la marca de fin, y siempre provocan un salto de línea, aunque no se le indique.

Dando estilo al texto

Las marcas de estilo de texto especifican el tipo de letra (negrita, cursiva, ..), tamaño, color que se desea que aparezca el texto.

Tipos de letras

Algunas de las marcas que permiten especificar la tipografía de los caracteres son las siguientes:

```
<B>Negrita (Bold)</B>  
<I>Cursiva (Italic)</I>  
<BLINK>Parpadeante (Blink)</BLINK>  
<TT>Tamano fijo (TypeWriter)</TT>
```

```
<EM>énfasis (Emphasis)</EM>  
<STRONG>Gran énfasis (Strong)</STRONG>  
<U>Subrayado (Underline)</U>
```

Todas estas marcas requieren etiquetas de comienzo y fin, y se pueden anidar.

Tamaño del texto.

Para modificar el tamaño se utiliza el atributo SIZE de la marca , de la manera siguiente:

```
<FONT SIZE="tamaño">Texto</FONT>
```

El comando requiere la etiqueta de cierre.

HTML define siete tamaños de letra distintos, siendo el tamaño 3 el que se adopta por defecto. Sin embargo, la mayoría de los navegadores permiten establecer cuál es la correspondencia real del tamaño 3. Así, se puede hacer que el tamaño 3 corresponda a una fuente Arial de 16, o a una Times de 8.

Existen dos formas de establecer el tamaño de un trozo de texto.

- Tamaño de texto absoluto.

```
<FONT SIZE="argumento">Texto</FONT>  
donde argumento es un número entre 1 y 7
```




- Tamaño de letra relativos.

```
<FONT SIZE="argumento">Texto</FONT>
```

donde argumento es una cadena de caracteres que nos indica, con un signo + ó -, el número de veces que esa fuente va ser mayor o menor que el tamaño de la fuente por defecto

El uso de tamaños relativos nos permite olvidarnos de cual es el tamaño de letra que se está utilizando en el momento actual, por lo que, en general, será preferible a utilizar tamaños de letra absolutos.

Si queremos modificar el tamaño de todo el texto del documento, una forma de hacerlo sería englobar todo el cuerpo del documento en una etiqueta con el tamaño de letra que queremos utilizar. Así escribiremos algo parecido a lo que sigue:

```
<HTML>
<HEAD>
.....
</HEAD>
<BODY><FONT SIZE="4">
.....
(contenido del documento)
.....
</FONT></BODY>
</HTML>
```

Sin embargo, hay otra forma de hacerlo más elegante y legible, utilizando la siguiente etiqueta:

```
<BASEFONT SIZE="tamaño">
```

Esta etiqueta puede situarse en cualquier lugar del documento. Sin embargo, como se trata de una etiqueta que afecta a todo el documento, para mantener la legibilidad lo más lógico es colocarlo en la cabecera.

Si fijamos un tamaño de letra básico distinto del 3, debemos tener en cuenta que es posible que alguno de los tamaños relativos no pueda llevarse a cabo, puesto que se sale de los límites especificados.

Colores del texto.

Para dar color a un texto se utiliza el comando siguiente:

```
<FONT COLOR="color">Texto</FONT>
```

En un principio la forma de especificar un color en HTML, era mediante el formato hexadecimal. Consistía en tres números hexadecimales (números expresados en base 16) de

dos dígitos cada uno. Con un número hexadecimal de dos dígitos podemos expresar hasta 256 valores distintos. El primer número hexadecimal nos va a indicar la cantidad de rojo que tiene nuestro color, el segundo la cantidad de verde, y el tercero la cantidad de azul (el blanco será FFFFFFFF, el negro 000000). Esta forma de definir los colores como combinación de los colores rojo, verde, y azul se denomina RGB (red - green - blue).

Afortunadamente, la mayoría de los visualizadores, al menos en sus últimas versiones, admiten otra forma de insertar colores. En lugar de expresarlos, en formato hexadecimal, se expresan como cadena de caracteres. Esta cadena es el nombre del color en inglés.

Para dar color a todo el texto del documento podemos hacerlo con:

```
<BODY TEXT="blue">
```

Para especificar el color de fondo del documento deberemos utilizar:

```
<BODY BGCOLOR="blue">
```

Listas

Las listas en HTML proporcionan una forma de clasificar la información, y hacer que ésta sea más inteligible por parte del visitante de las páginas.

Listas ordenadas: OL.

También llamadas listas numeradas son aquellas en las que cada elemento tiene delante un número que indica el orden del elemento dentro del conjunto de la lista.

La marca para crear una lista ordenada es ` `. Son obligatorias las etiquetas de comienzo y fin de estas marcas.

Cada elemento de la lista ordenada se identifica escribiéndolo dentro de la marca ` `. La etiqueta de fin puede ser omitida.

Con la marca LI no es necesario introducir un retorno de carro detrás de cada elemento de la lista.

Es posible especificar el tipo de numeración que se dará (números arábigos, números romanos, letras, etc.), por defecto se sigue la numeración árabe.

Las listas numeradas pueden ser anidadas una dentro de otra.



Listas desordenadas: UL.

También llamadas listas no numeradas son listas en las que no es necesario numerar de alguna manera los elementos que la componen, sino que cada elemento tiene delante un símbolo (un punto, un cuadrado, etc..).

La forma de especificar una lista desordenada es mediante la marca . Las etiquetas de comienzo y fin de esta marca son obligatorias.

Cada elemento de la lista se identifica mediante la marca . La etiqueta de fin para la marca se puede omitir

También es posible en las listas desordenadas especificar el símbolo que va a aparecer delante de cada elemento de la lista.

Las listas desordenadas también pueden ser anidadas una dentro de otra.

Listas de definición: DL.

A diferencia de los dos tipos anteriores de listas, las listas de definición, se utilizan para presentar la información sin marcas, ni números, sino que utiliza los sangrados de los párrafos.

Las listas de definición están constituidas por términos y su subsiguiente definición.

La forma de implementar las listas de definición es con la marca <DL> de la siguiente forma:

```
<DL>  
  <DT> Nombre del termino  
  <DD> Definición del término  
</DL>
```

Cada término de la lista de definición se indica con la marca <DT> y no sangrará, mientras que la descripción del término se indica con la marca <DD> y sangrará hacia la derecha para resaltarla del término.

Antes y después de la descripción de cada palabra clave, las listas de definición incorporan un retorno de carro,

Las listas de definición también pueden anidarse.

Texto preformateado: PRE.

El hecho de que HTML ignore los retornos de carro a la hora de mostrar un texto no es un comportamiento arbitrario, sino que es la forma que tenemos de que, independientemente de la ventana del visualizador, nuestro párrafo quede convenientemente justificado

Sin embargo, es posible que en alguna ocasión queramos que el texto aparezca en el visualizador tal y como lo hemos introducido, respetando los retornos de carro e incluso los espacios que podamos haber dejado entre caracteres. Para ello podemos utilizar la marca `<PRE>` y englobar en ella todo el texto del documento.

La marca `<PRE>` respeta tanto los retornos de carro como los espacios en blanco del archivo original, y la letra aparece como tipo de letra de espacio no proporcional (El tipo de letra usado por el texto preformateado es el mismo que el de `<TT> ... </TT>`), y no con el tipo de letra habitual del visualizador. Esta marca permite mostrar el texto preformateado tal cual.

Además, en el texto preformateado no se interpretan los caracteres especiales HTML: `<`, `>` y `&`.

Esta marca requiere de las etiquetas de comienzo y fin.

Centrando texto.

Para centrar una parte del documento, se delimita lo que se desea centrar mediante el comando `<CENTER> ... </CENTER>`.

Líneas horizontales.

La marca `<HR>` permite trazar una líneas horizontales para separar distintas secciones de una página HTML

Por defecto, las líneas horizontales van de la parte izquierda de la ventana del visualizador a la derecha.. Por otro lado tienen la parte superior oscura y la inferior clara, por lo que dan una sensación de relieve.

Esta marca sólo requiere la etiqueta de comienzo.

Esta etiqueta tiene varios modificadores que permite cambiar el aspecto de la línea horizontal (ancho, longitud,...).



Comentarios no visibles en la pantalla

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

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

Ejemplo:

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

Ejemplo práctico nº2

En el editor de textos copiamos:

```
<HTML>
<HEAD>
<TITLE>Ejemplo 2 - Comandos básicos</TITLE>
</HEAD>

<BODY>
<CENTER><H1>Comandos básicos</H1></CENTER>
<!-- Este documento contiene todo lo que he aprendido hasta ahora sobre
HTML -->
<H2>Párrafos</H2>
<P>Este es mi primer párrafo del ejemplo 2</P>
<P>Este es mi segundo párrafo. Aquí hay una ruptura de línea<BR>de
texto</P>
<HR>
<H2>Listas</H2>
<H3>Ordenadas</H3>
<P>La <FONT SIZE="+1"><I>musica</I></FONT> que más me gusta es <I> (en
orden de preferencia): </I></P>
<OL>
  <LI>El rock</LI>
  <LI>El jazz <!-- la marca LI no necesita de la etiqueta de cierre -->
  <LI>La musica clasica
</OL>
<H3>Desordenadas</H3>
<P>Sin un orden particular, mis <FONT COLOR="green"><B> aficiones
</B></FONT> son las siguientes:</P>
<UL>
  <LI>El cine
  <LI>El deporte
  <LI>Natación
  <LI>Baloncesto
</UL>
<H2>De definición</H2>
<DL>
  <DT>Descripcion</DT>
  <DD>Es nombrar las propiedades de algo
  <DT>Lista descriptiva</DT>
  <DD>Es nombrar las propiedades de varios objetos
</DL>

<HR>

<CENTER>También sabemos centrar el texto</CENTER>

<PRE> y escribir          preformateado </PRE>

<BLINK>y a escribir texto parpadeante</BLINK>

<HR>

</BODY>
</HTML>
```



Guardamos el fichero de texto con el nombre *ejemplo2.html* y lo abrimos en el navegador, y nos deberá aparecer las siguiente pantalla:

Comandos básicos

Párrafos

Este es mi primer párrafo del ejemplo 2

Este es mi segundo párrafo. Aquí hay una ruptura de línea de texto

Listas

Ordenadas

La *musica* que más me gusta es (*en orden de preferencia*):

1. El rock
2. El jazz
3. La musica clasica

Desordenadas

Sin un orden particular, mis **aficiones** son las siguientes:

- El cine
- El deporte
- Natación
- Baloncesto

De definición

Descripcion

Es nombrar las propiedades de algo

Lista descriptiva

Es nombrar las propiedades de varios objetos

También sabemos centrar el texto

y escribir

preformateado

y a escribir texto parpadeante

ENLACES: A.

La clave de la facilidad de uso de los documentos HTML, reside fundamentalmente en los enlaces. Los enlaces (o links) proporciona al que visita las páginas un mecanismo muy sencillo para moverse entre distintos documentos, permitiéndole moverse en aquellos aspectos que considere más importantes, mientras que desde el punto de vista del que crea las páginas

permite distribuir de forma transparente al visitante el lugar o lugares en los que se van a almacenar los distintos documentos.

Para definir enlaces hipertexto se utiliza la marca <A> de la manera siguiente:

```
<A HREF="direccion">Texto del enlace</A>
```

donde:

- **direccion:** es el destino del enlace.
- **texto del enlace:** es el texto indicativo del enlace que aparece en la pantalla normalmente con color especial y subrayado.

<A> requiere de la marca de fin.

Vamos a distinguir tres tipos de enlaces:

1. Enlaces a otras zonas de la misma página
2. Enlaces a otras páginas.
3. Enlaces a otras zonas de otras páginas.

Enlaces a otras zonas de la misma página

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

Así es posible hacer enlaces que lleven al usuario de una parte del documento a otra dentro del mismo documento con solo hacer click. A este tipo de enlaces los llamaremos enlaces locales.

Estos enlaces constan de dos objetos:

- **El enlace:**

Corresponde a la zona en la cual el usuario hará click, para viajar a otra parte del documento.

Esta zona aparece subrayada en el documento. Un enlace local se crea delimitando la zona con el comando ... , en que nombre es la etiqueta que se está referenciando.

- **El nombre enlazado o referenciado**

Corresponde a la zona del documento a la cual el usuario llegará al hacer click en la referencia correspondiente. Un nombre local se crea delimitando la zona con el comando ... , en que nombre es la etiqueta que asignamos a esta zona.



Ejemplo práctico nº 3 de enlaces locales

```
<HTML>
<HEAD>
<TITLE>Ejemplo 3 - Enlaces locales</TITLE>
</HEAD>

<BODY>
<H2>Enlaces locales</H2>
<H3><A NAME="indice">Indice</A></H3>
<UL>
  <LI><A HREF="#uno">Sección uno</A>
  <LI><A HREF="#dos">Sección dos</A>
  <LI><A HREF="#tres">Sección tres</A>
  <LI><A HREF="#cuatro">Sección cuatro</A>
  <LI><A HREF="#cinco">Sección cinco</A>
  <LI><A HREF="#seis">Sección seis</A>
  <LI><A HREF="#siete">Sección siete</A>
  <LI><A HREF="#ocho">Sección ocho</A>
</UL>
<H3><A NAME="uno">Sección uno</A></H3>
<P>Esta es la sección 1 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="dos">Sección dos</A></H3>
<P>Esta es la sección 2 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="tres">Sección tres</A></H3>
<P>Esta es la sección 3 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="cuatro">Sección cuatro</A></H3>
<P>Esta es la sección 4 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="cinco">Sección cinco</A></H3>
<P>Esta es la sección 5 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="seis">Sección seis</A></H3>
<P>Esta es la sección 6 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="siete">Sección siete</A></H3>
<P>Esta es la sección 7 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

<H3><A NAME="ocho">Sección ocho</A></H3>
<P>Esta es la sección 8 de la página de enlaces locales</P>
<P>Click aqui para <A href="#indice">volver al índice</A>

</BODY>
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo3.html* y lo abrimos con el navegador, y debe aparecer la siguiente pantalla:

Enlaces locales

Indice

- [Sección uno](#)
- [Sección dos](#)
- [Sección tres](#)
- [Sección cuatro](#)
- [Sección cinco](#)
- [Sección seis](#)
- [Sección siete](#)
- [Sección ocho](#)

Sección uno

Esta es la sección 1 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección dos

Esta es la sección 2 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección tres

Esta es la sección 3 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección cuatro

Esta es la sección 4 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección cinco

Esta es la sección 5 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección seis

Esta es la sección 6 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección siete

Esta es la sección 7 de la página de enlaces locales

Click aqui para [volver al indice](#)

Sección ocho

Esta es la sección 8 de la página de enlaces locales

Click aqui para [volver al indice](#)



Como vemos, las referencias locales son muy útiles para crear índices al comienzo del documento.

Enlaces a otras páginas

Para enlazar a otro documento, es necesario conocer la ubicación exacta del documento que se referenciará. Una ubicación, puede ser referenciada en forma relativa o absoluta

- **Ubicación relativa:**

Se indica especificando la posición del documento en la estructura de subdirectorios a partir de la ubicación del documento actual. Sólo se puede usar para documentos ubicados en el mismo ordenador.

Una referencia relativa a otro documento se hace usando el comando:

```
<A HREF="ubicación"> ... </A>
```

- **Ubicación absoluta:**

Se indica especificando el URL (uniform resource locator) de la página que se está referenciando.

Los URL son una manera universal de especificar una dirección. La forma más básica de referenciar un hipertexto es usar el comando:

```
<A href="http://ordenador/directorio/archivo"> ...
```

El ordenador indica la máquina donde se encuentra el documento. El directorio y archivo indican su posición dentro de ese ordenador. Los URL tienen muchas formas distintas para referenciar distintos objetos.

Ejemplo práctico nº 4

```
<HTML>
<HEAD>
<TITLE>Ejemplo 4 - Enlaces a otras páginas</TITLE>
</HEAD>

<BODY>
<H1>Enlaces a otras páginas</H1>
<H2>Referencias relativas</H2>
<P>Se puede referenciar, por ejemplo, a un archivo localizado en el
mismo directorio, como por ejemplo, al <A
href="ejemplo1.html">ejemplo número 1</A>. También a un archivo
localizado en otro directorio, por ejemplo, esta
<A HREF="../guiahtml.html">guía HTML</a></P>

<H2>Referencias absolutas</H2>

<H3>Algunos servicios WWW en España</H3>
<P>La <A
HREF="http://www.um.es/~biblio/">biblioteca de la Universidad de Murcia</A>
que permite
acceso al catalogo de la biblioteca. También está el servidor WWW de <A
HREF="http://www.rediris.es/">RedIris</A> y el servidor
de <A HREF="http://www.boe.es/">BOE</A> para leer consultar de BOEs
</P>

<H3>Algunos servicios en el extranjero</H3>
<P>Para informarse de la actualidad mundial, se puede acceder al
servicio <A HREF="http://www.cnn.com/">CNN</A>. Para los amantes de
la música se encuentra el servidor oficial de la <A
HREF="http://mtv.com/">MTV</A> y para localizar distintos tipos de
información está <A HREF="http://www.yahoo.com/">Yahoo!</A>.

</BODY>
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo4.html* y lo abrimos en el navegador, y debe aparecer la siguiente pantalla:

Enlaces a otras páginas

Referencias relativas

Se puede referenciar, por ejemplo, a un archivo localizado en el mismo directorio, como por ejemplo, al [ejemplo número 1](#). También a un archivo localizado en otro directorio, por ejemplo, esta [guía HTML](#)

Referencias absolutas

Algunos servicios WWW en España

La [biblioteca de la Universidad de Murcia](#) que permite acceso al catalogo de la biblioteca. También está el servidor WWW de [RedIris](#) y el servidor de [BOE](#) para leer consultar de BOEs

Algunos servicios en el extranjero

Para informarse de la actualidad mundial, se puede acceder al servicio [CNN](#). Para los amantes de la música se encuentra el servidor oficial de la [MTV](#) y para localizar distintos tipos de información está [Yahoo!](#)



Enlaces a una zona de otra página

Para este tipo de enlaces se utiliza una mezcla de las dos formas anteriores. Se indica el URL de la página a referenciar, luego un signo # y luego el nombre de la zona a referenciar.

Ejemplo práctico nº5

```
<HTML>

<HEAD>

<TITLE>Ejemplo 5- Enlaces a zonas de otras páginas</TITLE>
</HEAD>

<BODY>
<H1>Enlaces a zonas de otras páginas</H1>

<P>Enlazamos a la
<A HREF="ejemplo3.html#uno">la
seccion 1 del ejemplo 3</A>.</P>

<P>Enlazamos a la
<A HREF="ejemplo3.html#cinco">la
secci&oacute;n 5 del ejemplo 3</A>.</P>

</BODY>

</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo5.html* y lo abrimos con en el navegador, y debe aparecer la siguiente pantalla:

Enlaces a zonas de otras páginas

Enlazamos a la [la seccion 1 del ejemplo 3.](#)

Enlazamos a la [la sección 5 del ejemplo 3.](#)

IMÁGENES

Las imágenes son un elemento esencial para diseñar páginas atractivas. No obstante, un excesivo número de imágenes en una página pueden hacer ralentizar el proceso de visualización (las imágenes son las partes de las páginas que más espacio ocupan, por lo que su transferencia requiere más tiempo que la del texto). Es importante conseguir un

compromiso entre una página atractiva (lo que lleva incluir gráficos), y una página que pueda cargarse en un tiempo razonable

Principales formatos gráficos en Internet.

Existen una gran cantidad de formatos gráficos en el mundo de los ordenadores, cada uno de ellos con sus ventajas e inconvenientes. Sin embargo, a la hora de publicar en Internet, solamente debemos tener en cuenta dos formatos muy específicos: el GIF y el JPEG. Su elección no es en absoluto un capricho, además de la alta resolución que consiguen, se trata como veremos, de dos formatos que comprimen el tamaño de las imágenes, por lo que éstas pueden transmitirse más rápidamente por Internet.

Para crear y manipular imágenes hay que utilizar un editor de imágenes, entre los más conocidos se encuentran: Paint Shop Pro, Gif Construction Set, y Graphic Workshop.

Formato GIF.

Las siglas GIF provienen del inglés Graphics Interchange Format o formato de intercambio de gráficos. Este formato fue desarrollado por CompuServe para proporcionar un mecanismo de transmisión de archivos de imágenes mediante líneas telefónicas. Este mecanismo consiste en comprimir las imágenes para una transmisión más rápida en un medio de baja capacidad como son los hilos telefónicos.

Las imágenes en formato GIF están limitadas a 256 colores simultáneos.

Una propiedad interesante del formato GIF es la de poder seleccionar un color “transparente”. Así, cuando la imagen en cuestión es mostrada en un visualizador, la zona de la imagen con el color transparente toma el color del fondo del documento.

Otra propiedad interesante del formato GIF, es la poder crear gráficos animados, el formato de este tipo de imágenes se le denomina GIF 89.

Los archivos de imágenes en este formato tienen asociados la extensión .gif.

Formato JPEG.

JPEG son las iniciales de Joint Photographic Expert Group (Grupo de Expertos Fotográficos Reunidos). Al igual que el formato GIF, comprime las imágenes para una transmisión más rápida, aunque difiere en el algoritmo de compresión.

Mientras que con el formato GIF la compresión de un archivo de imágenes viene determinada, JPEG nos permite especificar la relación de compresión deseada, por lo que podemos variar el tamaño del archivo. Naturalmente, a mayor relación de compresión, menor



tamaño y peor calidad en la imagen final. Disminuyendo la compresión obtenemos imágenes de mayor calidad pero también de mayor tamaño.

No debemos obsesionarnos con la calidad de las imágenes. Hay que tener presente que Internet es en ocasiones un medio de transmisión bastante lento, por lo que a veces es preferible perder algo de calidad en las imágenes antes de aburrir al visitante con nuestros documentos.

Los archivos de imágenes en este formato tienen asociado la extensión .jpg.

Inclusión de imágenes

La etiqueta que se utiliza para la inclusión de imágenes en una página Web es la siguiente:

```
<IMG SRC="imagen.jpg"></IMG>
```

donde imagen.jpg es un archivo de imagen

Tamaño de las imágenes.

Si no especificamos nada más, las imágenes aparecerán en el navegador con el tamaño que tienen originalmente.

Para modificar el tamaño de las imágenes se utilizan los siguientes modificadores en la etiqueta :

- **WIDTH:** especifica la anchura en pixels de la imagen.
- **HEIGHT:** especifica el número de pixels que tendrá de alto la imagen.

Escalado de imágenes.

Con el uso de los modificadores HEIGHT y WIDTH podemos ampliar, y reducir las imágenes sin deformarlas, es decir, escalarlas. Una manera cómoda de hacerlo es solamente especificar una de las dos dimensiones, se asume que la reducción o la ampliación es proporcional a la otra.

Bordes de la imagen.

Mediante el modificador BORDER se puede añadir bordes a nuestras imágenes.

Alineación de texto con las imágenes.

La alineación del texto con las imágenes se realiza mediante el modificador ALIGN. Los valores que este modificador puede tomar son: TOP, TEXTOP, CENTER, ABSCENTER, MIDDLE, ABSMIDDLE, BOTTOM, LEFT, y RIGTH.

Además podemos especificar la distancia que queremos establecer entre el gráfico y el texto. Si este parámetro no se establece, el texto es situado justo a continuación de la imagen. Para introducir espacios horizontales y verticales a continuación de la imagen se utilizan los modificadores VSPACE y HSPACE.

Imágenes con texto alternativo.

Los visualizadores tienen la opción de no mostrar las imágenes, en un intento de acelerar el acceso a las páginas. Muchas veces, las imágenes incluidas en una página son fundamentales para comprender la información que se ofrece. Por ejemplo, una página destinada a alumnos universitarios de asignaturas de estadística o matemáticas puede contener imágenes de fórmulas matemáticas. Para estos casos se puede añadir a la imagen un texto ALTerativo, que dé una idea del contenido de la imagen. Para añadir un texto alternativo, se utiliza la orden como se muestra a continuación:

```
<IMG SRC="nombre_imagen" ALT="Texto_alternativo">
```

Uso de imágenes como enlaces.

Para utilizar una imagen como enlace a otra parte de la página o a otra página, simplemente hay que introducirlas dentro de la etiqueta <A HREF> de la siguiente manera:

```
<A HREF="enlace"> <IMG SRC="imagen"> </A>
```

Imágenes como fondo de un documento.

Un efecto muy utilizado es el uso de imágenes como fondo de un documento HTML. La manera de hacerlo es con el atributo BACKGROUND de la marca <BODY>.

La imagen de fondo en general será más pequeña que el área de documento del navegador. Lo que hace entonces el navegador es repetir, partiendo de la esquina superior izquierda, la imagen que hemos elegido como fondo del documento.

A la hora de elegir la imagen de fondo, es preciso tener en cuenta que el texto del documento debe ser legible por encima de la imagen, por lo que hay que evitar colocar imágenes con colores demasiado fuertes.



Mapas sensibles.

Los mapas sensibles o mapas de imágenes es un mapa que permite realizar diferentes hiperenlaces en función de la “zona” de la imagen en que se pulse. Constituyen una herramienta de gran potencia, aunque son algo complicados de construir y mantener.

Los mapas sensibles están compuestos de dos elementos fundamentales: la imagen del mapa y los enlaces de cada zona del mapa.

Para crear un mapa debemos tener ficheros distintos. Uno será el de la imagen propiamente dicha y otro será un fichero de texto que tenga la extensión .map donde indicaremos las coordenadas de la imagen que nos enlazarán con cada página del hiperenlace. Por último, necesitamos un programa externo (CGI) que pueda procesar el fichero .map.

El contenido del fichero .map, que indica las coordenadas de nuestro mapa, debe tener la siguiente estructura:

- **default** URL por defecto si se pulsa en una zona sin hiperenlace.
- **figura** URL de hiperenlaces, coordenadas. La figura puede ser:
 - **circle** define una zona circular, el primer par de coordenadas indica el centro del círculo y el segundo la situación de un punto de la circunferencia.
 - **rect** define una zona rectangular, el primer par de coordenadas indica el vértice de la esquina superior izquierda y el segundo el vértice de la esquina inferior derecha.
 - **point** define un punto. El par de coordenadas indica la situación del punto.

Una vez creado el fichero .map, la instrucción a incluir en nuestra página Web para activar el mapa sería:

```
<A HREF="cgi-bin/imagemap/mipagina/mapa.map">  
<IMG SRC="imagen.gif" ISMAP>  
</A>
```

donde:

- **cgi-bin/imagemap**: Es el nombre del programa que procesa el mapa
- **mipagina/mapa.map**: Es el camino y nombre del fichero .map
- **imagen.gif**: Es el nombre del fichero de imagen
- **ISMAP**: Indica que la imagen es considerada un mapa

En los nuevos navegadores, existen unas etiquetas nuevas para crear mapas, que no necesitan de la ayuda auxiliar de un programa externo y son:

- **<MAP>...</MAP>**
Indica al mapa, su parámetro es:
name= nombre, damos el nombre de nuestro mapa con la extensión .map
- **<AREA>**
Define las áreas sensibles de la imagen con los siguientes parámetros:
 - **shape="tipo"** (Forma)
Indica el tipo de área a definir, hay 3 tipos:
 - **rect** (Área rectangular) se tienen que especificar las coordenadas de la esquina superior izquierda y de la esquina inferior derecha.

- **poly** (Polígono) se tienen que especificar las coordenadas de todos los vértices del polígono. El visualizador se encargará de cerrar la figura
- **circle** (Círculo) se tienen que especificar en primer lugar, las coordenadas del centro del círculo y, a continuación, el valor del radio (en puntos)
- **coords**="coordenadas" (Coordenadas)
Indica las coordenadas de la figura, es decir, dónde está cada una de esas áreas.
- **href**="URL" (Dirección)
Indica la dirección a la que se accede si se pulsa en la zona delimitada por el área indicada.

Si dos áreas se superponen, se ejecutará la que se encuentre en primer lugar en la definición del mapa. Es importante definir una última área que abarque la totalidad del gráfico para direccionar a una URL por defecto, con el objeto de contemplar el caso de que no se pulse sobre un área definida. Veamos un ejemplo:

```
<AREA name="casa">
<AREA shape="poly"coords="2,23,76,85,2" href="tejado.html">
<AREA shape="rect"coords="23,78,65,122" href="casa.html">
<AREA shape="circle"coords="36,75,95" href="arbol.html">
<AREA shape="rect"coords="2,23,76,85" href="dibujo.html">
</MAP>
```

Para activar el mapa debemos escribir el nombre de la imagen a mostrar y después indicar que dicha imagen está tratada como un mapa sensible. Esto se escribe así:

```
<IMG SRC="grafico.gif" USEMAP="#casa">
```



Ejemplo práctico nº6

```
<HTML>
<HEAD>
<TITLE>Ejemplo 6 - Imagenes </TITLE>
</HEAD>
<BODY BACKGROUND="nubes.jpg" TEXT="#AA0000">
<CENTER>
<H1>Imágenes</H1>
</CENTER>
<HR>
Esta es mi página del Web. No es muy extensa, pero tiene todos los
elementos básicos. Espero que os guste. Poco a poco le iré
añadiendo más cosas interesantes.
<P> <A HREF="ejemplo2.html"> <IMG SRC="hombre.gif"> </A> <FONT SIZE=+3>Mis
aficiones</FONT>
<P> <A HREF="ejemplo4.html"> <IMG SRC="casa.gif" BORDER="0"> </A> <FONT
SIZE=+3>Mis páginas favoritas</FONT>
<CENTER>
<H3> Un lugar ideal para mis vacaciones </H3>
<IMG SRC="isla.gif" ALT="isla">
</CENTER>
<HR>
<CENTER>
<H3>Expediente X - Mulder</H3>

<IMG SRC="3mulder.gif" WIDTH=119 HEIGHT=160></CENTER>
<HR>
</BODY>
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo6.html* y lo abrimos en el navegador, y aparecerá la siguiente pantalla:



TABLAS

Una tabla es una manera muy compacta y clara de mostrar la información. Una tabla en HTML se entiende como un conjunto de filas (fila=horizontal), apiladas una sobre otra. Cada fila contiene a su vez un conjunto de celdas, puestas una al lado de la otra.

Una tabla se declara usando el comando `<TABLE> ... </TABLE>`. Dentro de la tabla, se usa `<TR>.. </TR>` para indicar una fila y dentro de una fila, `<TD> ... </TD>` para delimitar el contenido de una celda. El elemento de cierre de fila y de celda es optativo.

El texto incluido dentro de cada celda puede ser formateado con cualquiera de las etiquetas vistas anteriormente

Todos estos comandos aceptan numerosos modificadores. Se pueden, entre otras cosas, unir celdas, especificar la alineación del contenido de la celda, y mucho más.

Para entender mejor el comando `<TABLE> ... </TABLE>`, está disponible una muy amplia gama de ejemplos.

Ejemplo de tabla

Aquí se muestra una tabla muy simple:

```
<TABLE>
<TR><TD>Celda A1</TD><TD>Celda B1</TD><TD>Celda C1</TD></TR>
<TR><TD>Celda A2</TD><TD>Celda B2</TD><TD>Celda C2</TD></TR>
<TR><TD>Celda A3</TD><TD>Celda B3</TD><TD>Celda C3</TD></TR>
</TABLE>
```

Título de la tabla.

Es conveniente que las tablas tengan un título que las identifique y aclare su contenido. Aunque ese título puede ser contenido colocando texto formateado antes de la tabla, HTML dispone de la etiqueta `<CAPTION>` para darle nombre a la tabla, de la siguiente forma:

```
<TABLE>
  <CAPTION> Nombre de la tabla </CAPTION>
  .....
</TABLE>
```



Esta etiqueta tiene la ventaja que el texto formateado se coloca automáticamente en el centro de la anchura de la tabla.

Bordes de la tabla.

Para colocar bordes a una tabla en HTML se utiliza el modificador BORDER dentro de la etiqueta <TABLE>. La sintaxis es:

```
<TABLE BORDER="anchura">  
.....  
</TABLE>
```

donde anchura indica la anchura en pixels del borde exterior de la tabla.

El modificador BORDER coloca bordes externos e internos en la tabla, pero la anchura que se especifica solo afecta a los bordes externos.

El efecto de sombreado del borde exterior se produce cuando el modificador BORDER toma un valor superior a uno. Si queremos que el borde exterior de la tabla sea igual que los bordes interiores, debemos hacer BORDER="1".

Cabeceras de filas y de columnas.

La inclusión de cabeceras en cada fila y columna ayuda a comprender el contenido de la tabla.

No hay ninguna dificultad para hacer esto con las etiquetas que hemos visto. Sin embargo, existe una forma alternativa de indicar la cabecera de una fila o columna mediante la etiqueta <TH>. La desventaja de usar esta etiqueta es que el texto que va dentro de ella no admite formateado como el de las otras celdas.

Tamaño de la tabla.

Por defecto el tamaño de la tabla viene dado por el texto o los gráficos que contiene. Sin embargo, es posible que en ocasiones queramos cambiar la anchura de nuestra tabla.

La forma de establecer una anchura fija en una tabla es utilizar el modificador WIDTH dentro de la etiqueta <TABLE>. La sintaxis es:

```
<TABLE WIDTH="anchura"> ..... </TABLE>
```

Aquí anchura puede ser un número que especifique en pixels la anchura absoluta de la tabla, o bien un porcentaje que indique la anchura de la tabla en relación a la anchura del área del documento del navegador.

A pesar de que podemos especificar la anchura de la tabla, si el texto de las celdas ocupa una anchura mayor, la tabla tomará la anchura mínima para que el texto de las celdas pueda verse íntegramente.

Justificación de la tabla.

Por defecto, nuestra tabla quedará justificada a la izquierda. La forma de centrarla es mediante la etiqueta <CENTER>.

Espacio dentro de las celdas.

Es posible que queramos que nuestro texto no quede apretado dentro de las celdas, sino que exista más espacio entre los textos de las celdas contiguas de forma que pueda leerse mejor. Esto puede hacerse mediante los dos siguientes modificadores:

- **CELLPADDING:** establece la distancia mínima en pixels entre los bordes de cada celda y el texto que va encerrado en ellas. Su valor por defecto es 1.
- **CELLSPACING:** establece la anchura en pixels de los bordes de cada celda.

Forma de las celdas individuales.

Justificación del texto de las celdas.

HTML permite justificar horizontal y verticalmente el texto dentro de cada celda. Para ello implementa los modificadores ALIGN y VALIGN dentro de la etiqueta <TD>. La forma de hacerlo es:

```
<TABLE>
  <TR>
    <TD ALIGN="horizontal" VALIGN="vertical">
      ....
    </TD>
  </TR>
  .....
</TABLE>
```



donde:

- **horizontal** puede tomar los valores CENTER, LEFT y RIGHT. Por defecto vale LEFT.
- **vertical**: puede tomar los valores TOP, MIDDLE, BOTTOM, y BASELINE. Por defecto vale CENTER.

Si queremos que una determinada justificación horizontal o vertical afecte a todas las celdas de una fila, podemos hacerlo incluyendo el modificador ALIGN o VALIGN dentro de la etiqueta <TR>.

Celdas de diferentes tamaños.

Por defecto, todas las celdas que constituyen una tabla en HTML tienen el mismo tamaño. Sin embargo, es posible establecer el tamaño de una determinada celda mediante dos modificadores de la etiqueta <TD>, que son WIDTH y HEIGHT:

- **WIDTH**: especifica la anchura del texto en una celda concreta. Se puede especificar como valor absoluto o bien como relativo en forma de porcentaje de la anchura de la tabla.
- **HEIGHT**: establece la altura de una celda individual.

No es posible usar los modificadores WIDTH y HEIGHT para aplicarlos a toda una fila. La única forma de hacerlo es poniendo en cada celda de la fila los valores correspondientes de WIDTH y HEIGHT.

Celdas irregulares.

Podemos desear que una de nuestras celdas ocupe varias filas y/o columnas. La forma de hacerlo es introducir los modificadores COLSPAN y ROWSPAN en la etiqueta <TD>:

- **COLSPAN**: especifica la altura de una celda en concreto en función de la altura de las celdas adyacentes. Se especifica en número de celdas adyacentes.
- **ROWSPAN**: especifica la anchura de una concreta en función de las celdas que están debajo. Se especifica en número de celdas de debajo.

Color de cada celda.

Para especificar el color de cada celda de la tabla se utiliza el modificador BGCOLOR en la etiqueta <TD>. Como formato de colores se acepta el hexadecimal o una cadena de caracteres.

Texto en una sola línea.

Si hemos indicado la anchura de una celda y ésta es menor que el texto incluido en ella, y queremos que el texto no ocupe más líneas, sino que se amplíe la anchura de la celda, debemos añadir el modificador NOWRAP dentro de la etiqueta <TD>.

Ejemplo práctico nº7

```
<HTML>
<HEAD>
<TITLE>Ejemplo 7 - Tablas</TITLE>
</HEAD>

<BODY>
<CENTER><H1>Tablas</H1></CENTER>
<H2>Demostracion de COLSPAN y Encabezados</H2>
<TABLE WIDTH="100%" BORDER="1">
<TR>
<TH COLSPAN="2">Head1</TH>
<TH COLSPAN="2">Head2</TH>
</TR>
<TR>
<TD>A</TD> <TD>B</TD> <TD>C</TD> <TD>D</TD>
</TR>
<TR>
<TD>E</TD> <TD>F</TD> <TD>G</TD> <TD>H</TD>
</TR>
</TABLE>

</BODY>
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo7.html* y lo abrimos con en el navegador, y debe aparecer la siguiente pantalla:.

Tablas

Demostracion de COLSPAN y Encabezados

Head1		Head2	
A	B	C	D
E	F	G	H

FRAMES (MARCOS)

Los frames (marcos) es una técnica que se utiliza para subdividir la pantalla del visualizador en diferentes ventanas, con la característica de que cada una de ellas se podrá manipular por separado, es como si cada una de ellas fuera una página Web. Esto es muy útil para, por



ejemplo, para mostrar permanentemente en una ventana los diferentes contenidos de nuestra página y, en la otra mostrar el contenido seleccionado.

Para definir las diferentes subventanas o frames se utilizan las marcas:

`<FRAMESET>` `</FRAMESET>` y `<FRAME>`.

Con `<FRAMESET>` indicamos cómo se va a dividir la pantalla principal, donde podrán ir varias directivas frameset anidadas con el fin de subdividir cualquiera de las subdivisiones. Sus atributos son:

- **rows**= "número o %" (Número de filas)
Se indica el número de filas en que subdividiremos la pantalla. El grupo de números que acompañan se indican en puntos o en tanto por cierto de cada una de las subventanas; en este caso se entiende que el primer valor que escribimos corresponde a la ventana superior, el segundo a la ventana inmediatamente inferior y así sucesivamente.

1ª Fila
2ª Fila
3ª Fila

Pongamos como ejemplo `<FRAMESET rows= "25%, 50%, 25%" >`, la pantalla quedaría así

25%
50%
25%

- **cols**= "número o %" (Número de columnas) Se indica el número de columnas en que subdividiremos la pantalla. Los números también se indican en puntos o en tanto por cierto. Aquí los valores se aplican de izquierda a derecha.

1ª Columna	2ª Columna	3ª Columna
------------	------------	------------

Pongamos como ejemplo `<FRAMESET cols= "120, *, 80" >`, la pantalla quedaría así:

Esta ventana tiene un ancho fijo: mide 120 puntos	Esta ventana ocupará el resto de la medida que no hayan ocupado las otras medidas fijas	Esta ventana tiene un ancho fijo: mide 80 puntos
---	---	--

Claro que también los podemos tener anidados, como por ejemplo:

`<FRAMESET cols= "250%,*" rows= "15%, *" >` la pantalla quedaría así:

Con la marca <FRAME> indicamos las propiedades de cada subventana y necesitamos una marca <frame> para cada subventana creada; los atributos son:

- **name= "nombre"** (Nombre)
Indica el nombre por el que nosotros nos vamos a referir a esa ventana.
- **src= "URL"** (Lugar de localización)
Indica un fichero .html cuyo contenido se mostrará en esa ventana.
- **marginwidth= número** (Ancho del margen)
Indica el margen izquierdo y derecho de la subventana, se da en puntos.
- **marginheight= número** (Alto del margen)
Indica el margen superior e inferior de la subventana, se da en puntos.
- **scrolling= "yes/no/auto"** (Desplazamiento)
Indica si se aplica una barra de desplazamiento (típica barra para moverse de arriba y abajo que aparece en Windows) a la subventana. en el caso en que la página que se cargue en ella no quepa en los límites de la subventana. El valor "yes" muestra siempre esta barra de desplazamiento, el valor "no" no la mostrará nunca (es decir, la parte del frame que no quepa en la pantalla no la veremos) y, por último, "auto" la mostrará sólo en caso de que sea necesario.
- **noresize**
Si se escribe este atributo, el usuario no podrá redimensionar las subventanas con el visor; ya que un usuario podría cambiar el tamaño de las subventanas con solo situar el cursor en un borde de la pantalla y desplazarlo.

Hay que tener en cuenta que todos aquellos navegadores que no soporten estas marcas de la versión HTML 3.0 no verán nada de los contenidos de nuestros frames. Es por ello que existe otra marca llamada <NOFRAMES> ... </NOFRAMES> y todo lo que se escriba aquí dentro será lo que vean esos visualizadores, mientras que no afectará a los visualizadores más potentes.



Ejemplo práctico nº8:

```
<HTML>
  <HEAD> <TITLE> Página con frames </TITLE> </HEAD>

  <FRAMESET rows= "35%, *" >
  <!--Creo dos subventanas horizontales, la de arriba ocupa un 35% de la
  pantalla y la de abajo el resto -->

  <FRAMESET cols= "20%, *" >
  <!--Creo dos subventanas verticales, la de la izquierda un 20% de la
  pantalla y la de la derecha el resto -->

  <FRAME name= "botones" src= "ejemplo6.html" scrolling="auto">
  <!--llamo a la subventana horizontal superior izquierda con el nombre de
  "botones" y muestro el documento "botonera.html" -->

  <FRAME name="logo" src="ejemplo1.html" scrolling="auto">
  <!-- llamo a la subventana horizontal superior derecha con el nombre de
  "logo" y muestro el documento "logomio.html" -->

  </FRAMESET>
  <!--Cierro la definición de las subventanas verticales que están en la 1ª
  subventana horizontal -->

  <FRAME name= "menu" src= "ejemplo2.html" scrolling= "auto">
  <!--Llamo a la subventana vertical inferior "menu" para que muestre los
  contenidos de mi pantalla "inicio.html" -->

  <NOFRAMES>
  <!--Indico las órdenes para aquellos navegadores que no soporten frames -->
  <BODY>
  Su navegador no muestra frames. Por favor pulse <A RREF=
  "aplicación.html">Aquí</A> para ir a la página sin frames.
  </BODY>
  </NOFRAMES>

  </FRAMESET>
  <!--Cierro la definición de las subventanas verticales -->
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo8.html* y lo abrimos con en el navegador.

Al usar frames nos encontramos con un problema, cuando queremos mostrar una página Web debemos indicarle al visualizador en qué subventana queremos que se muestre. Por defecto lo mostrará en la ventana donde se encuentre el enlace. Para poder elegir la subventana de destino del hiperenlace se añade nuevo atributo a la marca ` ... ` que se llama `target` que a su vez tiene estos valores:

- **target= "nombre-ventana"**
Muestra el hiperenlace en la ventana cuyo nombre se indica.
``
Muestra la página de IBM en la subventana llamada "home".
- **target= "_blank"**
Abre una nueva ventana del navegador y muestra el hiperenlace en esta (Si usamos Netscape esto provoca que tengamos funcionando dos copias del programa.)
``
Muestra la página de Microsoft en una nueva ventana del navegador.
- **target= "_self"**
Se muestra el hiperenlace en la subventana activa.
- **target= "_parent"**
El hiperenlace se muestra en el frameset definido anteriormente al actual. Si no hay ningún frameset anterior se muestra a pantalla completa suprimiendo todas las subventanas de la pantalla.
- **target= "_top"**
Suprime todas las subventanas de la pantalla y muestra el hiperenlace a pantalla completa.
``
Muestra la página del buscador de yahoo a pantalla completa.

Formularios (FORMS)

Los formularios o también llamados forms, se utilizan para la entrada y envío de datos (el procesamiento de los mismos es otro asunto) desde una página HTML. Podremos utilizar formularios para diseñar cuestionarios de todo tipo, por ejemplo, datos referentes al tipo de usuario que visita nuestra página, para hacer pedidos de algún producto, encuestas, suscripción a algún tipo de servicio, etc.

En un formulario podremos solicitar diferentes datos (campos) cada uno de los cuales quedará asociado a una variable. Una vez introducidos los valores en los campos, el contenido de éstos será enviado a la dirección (URL) donde esté el programa que pueda procesar las variables, a este programa externo se le suele denominar CGI (Common Gateway Interface, Interface de Compuerta Común).

La marca es `<FORM> ... </FORM>` cuyos atributos son:

- **action= "programa"** (Acción)
Se refiere a la acción a realizar cuando se pulsa el botón de envío del formulario. Esta



acción puede ser enviar por correo el contenido del formulario (action= mailto: <dirección e-mail> o activar algún otro URL, como por ejemplo un mensaje que nos indique que el envío se ha realizado con éxito.

- **method=** POST / GET

Que sirve para especificar el método según el cual se transferirán las variables. GET lo utiliza por defecto y no produce cambios en el documento destino, por ejemplo si consultásemos una base de datos. POST produce la modificación del documento de destino, por ejemplo, cuando se envían por correo las variables.

La información del formulario es devuelta en forma de una única lista o cadena compuesta por pares nombre/valor (NAME/VALUE), es decir, el nombre de cada campo seguido del valor que tiene. Cada campo se separa con el carácter &.

Campos de entrada

Los elementos de entrada de datos se denotan con la marca <INPUT> que tiene un atributo type con varias especificaciones:

- **type=** text (Campos de texto para una única línea)
Indica que el campo a introducir será un texto; se utilizan generalmente para introducir cadenas de texto cortas como nombres, direcciones, e-mail..
- **name=** campo
Define cada campo de un formulario y es necesario para identificar los valores asignados al mismo a la hora de procesar la información devuelta.
- **maxlength=** número
Número máximo de caracteres a introducir en el campo.
- **size=** número
La anchura visible del campo en caracteres.
- **value=** "texto"
Valor inicial del campo, normalmente será decir, vacío.

Ejemplo:

```
<INPUT type= "text" name="e-mail" maxlength=300 size= 60>
```

- **type=** password (Campo password)
Indica que el campo será una palabra de paso, por tanto los caracteres tecleados por el usuario en pantalla se verán como asteriscos. Sus atributos opcionales son los mismos que para type= text: name, maxlength, size y value.

Ejemplo:

<INPUT type="password" name="clave" size=12 >

- **type= checkbox** (Campos de chequeo de opción)

Indica que el campo se elegirá marcando una casilla, o más de una. El campo aparece como una cajita que puede estar marcada o no, según se active o desactive respectivamente. Sólo se devuelve el par nombre/valor correspondiente a un "checkbox" cuando está activada.

- value= "valor"
- name= campo
- checked: Sirve para iniciar un botón en su estado activado

Ejemplo:

<INPUT type= "checkbox" name=activado >

- **type= radio** (Botones Circulares)

Indica que el campo se elegirá marcando una casilla y sólo una, siendo los botones circulares. Como sólo se Puede tomar un valor, todos los elementos de este tipo que pertenezcan a un mismo grupo, deben tener el mismo valor para name. Sólo el botón seleccionado será el que genere un par nombre/valor en los datos devueltos del formulario.

- value="valor"
- name=campo

Ejemplo:

<INPUT type= "radio" name="edadf" value= "menos17" >

- **type= range** (Campo para Rangos)

Permite elegir un valor determinado entre los posibles de un rango delimitado por los tributos mín y máx. Se puede iniciar con un valor concreto utilizando.

- value= "valor"

Ejemplo:

<INPUT type= "range" name= nota min=1 max= 10 >

- **type= image** (Campo Imagen)

Indica que el campo contendrá el valor de las coordenadas del punto de la imagen pinchada, la imagen se indica con el atributo:

- src= "fichero de imagen"

Ejemplo:

<INTPUT type= "image" src= estadistica1.gif >



- **type= hidden** (Campo para Datos Ocultos)
No aparece ningún campo rellenable por el usuario, sino que se envía algún tipo de información de interés interno, como por ejemplo, la versión de un programa (si queremos saber cuál está utilizando el usuario), un identificador de cualquier tipo, información de estado, etc.
 - **value= "valor"**
 - **name= campo**

Ejemplo:

```
<INPUT type= "hidden" name= version value= "2.0a" >
```

- **type= submit** (Botones de envío)
Aparece en forma de botón que es el que debe pulsarse, una vez lo hemos completado, para enviar la información del formulario al programa indicado en <form>.
 - **value= "texto"**
Que indica el texto que aparecerá en el botón, por ejemplo, Pulse *aquí*.
- **src=imagen**
Imagen que sustituya al típico botón.
 - **name= campo**

Ejemplo:

```
<INPUT type= "submit" value= "Enviar" >
```

- **type= reset** (Botones de borrado)
Devuelve los campos del formulario a los valores de inicio especificados. De igual forma que para los botones de envío tiene los mismos atributos.
 - **value= "texto"**
Que indica el texto que aparecerá en el botón, por ejemplo, *Pulse aquí*.
 - **src= imagen**
Imagen que sustituya al típico botón.
 - **name= campo**

Ejemplo:

```
<INPUT type= "reset" value= "Borrar" >
```

Campos de selección

Este tipo de campos despliegan una lista de opciones, entre las que deben escoger una o

varias. Se utiliza para ellos la marca `<SELECT> ... </SELECT>` y sus atributos son:

- **name=** campo
Nombre del campo.
- **size=** número
Número de opciones visibles. Si se indica uno se presenta como un menú desplegable, si se indica más de uno se presenta como una lista con barra de desplazamiento.
- **múltiple**
Permite seleccionar más de un valor para el campo.

Las diferentes opciones de la lista se indican con la marca `<OPTION>` que puede incluir el atributo:

- **selected**
Para indicar cuál es la opción por defecto, si no se especifica por defecto sale la primera opción de la lista.

Ejemplo:

```
<SELECT name= Ciudades >  
<OPTION= Jaen >  
<OPTION= Palencia >  
<OPTION= Soria >  
<OPTION= Lugo >  
</SELECT>
```

Áreas de texto

Representan un campo de texto de múltiples líneas y aparecen como una caja. Normalmente se utiliza para que se incluyan en ellas comentarios. La marca es `<TEXTAREA> ... </TEXTAREA>` y sus atributos son:

- **name=** campo
Nombre del campo
- **cols=** número
Número de columnas de texto visibles
- **rows=** número
Número de filas de texto visibles
- **align=** valor
Puede ser top (arriba), middle (medio), bottom (abajo), left (izquierda) y right (derecha)



- **wrap= VIRTUAL / PHYSICAL**
Justifica el texto automáticamente en el interior de la caja. La opción physical envía las líneas de texto separadas en líneas físicas. La opción virtual envía todo el texto seguido.

Ejemplo

```
<TEXTAREA name= dirección rows= 60 cols=5>  
IBM Company Ine.  
1221 Beverly Hills. Massachusetts 90876ET56  
</TEXTAREA>
```

Ejemplo práctico nº9:

```
<HTML>  
<HEAD>  
<TITLE>Ejemplo 9 - Formularios </TITLE>  
</HEAD>  
<BODY>  
<FORM action="mailto:lml@um.es" method="post">  
Tu Nombre: <INPUT type=text name=nombre size=30>  
Tu Clave: <INPUT type=password name=clave size= 8>  
<P>  
Archivos a Enviar: <BR>  
<INPUT type=checkbox name=archivo value="Manual" > Manual de Html  
<INPUT type=checkbox name=archivo value="Mapthis">Programa Mapthis  
<INPUT type=checkbox name=archivo value="Help" > Archivo de Ayuda  
<P>  
Tu Edad: <BR>  
<INPUT type=radio name=edad value="-20" > Menos de 20 años  
<INPUT type=radio name=edad value="20-40" > Entre 20 y 40 años  
<INPUT type=radio name=edad value="+40" > Más de 40 años  
<P>  
<INPUT type=hidden name=lugar value="página personal" >  
¿Cómo encontraste mi página?  
<SELECT name=donde >  
<OPTION> De casualidad  
<OPTION> Por el buscador Ole  
<OPTION> Por el buscador Yahoo  
<OPTION> Me la recomendaron  
</SELECT>  
<P>  
Tus comentarios:  
<BR>  
<TEXTAREA name=comentario rows=5 cols=40 wrap=virtual>  
</TEXTAREA>  
<P>  
<INPUT type=submit value="Enviar" >  
<INPUT type=reset value="Borrar" >  
</FORM>  
</BODY>  
</HTML>
```

Guardamos el fichero de texto con el nombre *ejemplo9.html* y lo abrimos con en el navegador, y debe aparecer la siguiente pantalla:.

Tu Nombre: Tu Clave:

Archivos a Enviar:

Manual de Html Programa Mapthis Archivo de Ayuda

Tu Edad:

Menos de 20 años Entre 20 y 40 años Más de 40 años

¿Cómo encontraste mi página?

Tus comentarios:



Hojas de estilo

Este es un nuevo concepto que ha surgido gracias a la posibilidad que ofrecen algunos navegadores más avanzados, por tanto recordemos que no siempre es posible su visualización. Viene del término CSS (Cascading StyleSheet, Hojas de Estilo en Cascada) y está recogido en la W3C.

El CSS1 es un simple mecanismo que permite al diseñador y a los usuarios hacer una unión de estilo en el que se incluyen fuentes, tamaños, color, espacios, etc., (como si estuviésemos maquetando una página de una revista) y añadirlo a una página HTML. De hecho, necesitamos insertarlo dentro de las marcas y atributos que hemos aprendido del HTML. Existen 4 posibles formas de combinar:

```
<HTML>
  <HEAD>
    <TITLE>menu</TITLE>
    <LINK REL=STYLESHEET TYPE="text / css" HREF= "http://style.coni/cool"
    TITLE="cool">
    <STYLE TYPE= "text/css">
      @import url (http://style.coni/basic);
      H1 {color: blue}
    </STYLE>
  </HEAD>

  <BODY>
    <H1> La cabecera es azul </H1>
    <P STYLE= "color: green"> Mientras que el párrafo es verde
  </BODY>
</HTML>
```

- 1 Usando un LINK que llama a una CSS externa.
- 2 Una marca <STYLE> dentro de la <HEAD>.
- 3 Una hoja de estilo importada del CSS denominada "@import".
- 4 Un atributo de style usado dentro del <BODY>.

Su aprendizaje es sencillo pero lleno de propiedades (cerca de 50 que pueden determinar el aspecto de una página html), veremos con algunos ejemplos cómo funciona.

Si quisiéramos que todos los textos comprendidos dentro de <H1> fuesen azules escribiríamos esto:

<i>Marca html</i>	<i>propiedad</i>	<i>valor</i>
H1	{ color:	blue}

Se puede agrupar un número de marcas comunes para reducir las órdenes:

- H1, H2, H3 (font-family: helvetica)
- H1 { font-weight: bold;
Font-size: 12pt;
Line-height: 14pt.
Font-family: helvetica;
Font-variant: normal;
Font-style: normal; }

Y hay otras que se pueden agrupar así:

- H1 { font: bold 12pt/14pt helvetica } que es igual que el ejemplo anterior.

EDITORES Y CONVERSORES

En el presente apartado se provee de algunos comentarios y reflexiones acerca del uso de herramientas automatizadas para la generación de páginas Web: editores y conversores.

Editores

Seguramente usted ha visto o conoce programas que pueden asistir a una persona en la creación de páginas Web, y quizás se ha preguntado: ¿ uso un editor o aprendo HTML ?

La respuesta que yo le daría es la misma que si usted me preguntara si es mejor saber aritmética o ocupar una calculadora: ambas cosas no se contraponen entre sí.

Un editor HTML es una gran ayuda al diseñador Web: brinda la posibilidad de disponer y ver la página mientras se crea, y de observar inmediatamente cómo aparecerán distribuidos los colores y los elementos dentro del documento, es una ayuda para recordar las marcas y sus atributos, sin embargo; puede resultar inadecuado por lo siguiente:

- Poca flexibilidad, y en algunas ocasiones el código que generan en general es difícil de editar o mantener.
- El conjunto de comandos que incorpora puede ser limitado, en caso de que sea un editor producido por alguna empresa que haga visualizadores, o puede estar obsoleto.
- El HTML que generan está fuertemente orientado al formato físico, más que a la estructura, con lo cual se pierde universalidad en el documento.

Es recomendable comenzar por lo básico, procurando aprender HTML bien, creando documentos con editores de texto plano. Después si se desea se puede ocupar una combinación editor de texto plano más editor HTML para crear documentos, a fin de explotar al máximo las capacidades de ambos sistemas.



Los editores HTML se clasifican en dos tipos:

- **Wysiwyg** "what you see is what you get": en ellos se puede ir viendo en pantalla inmediatamente lo que se va creando. son muy útiles para apreciar los colores y la disposición en pantalla de los elementos.
- **No Wysiwyg**: ellos usualmente invocan una aplicación externa (un browser) para mostrar lo que se va creando. Son muy útiles para recordar los comandos HTML y sus atributos.

Existen editores mixtos: presentan la pantalla dividida en dos segmentos, uno con el código HTML y otro con el resultado final.

Por ejemplo, uno de los editores HTML Wysiwyg más completos es el FrontPage 98 de Microsoft (<http://www.eu.microsoft.com/frontpage/>), aunque es Wysiwyg, también permite trabajar en modo comando.

Conversores

Las herramientas de conversión permiten transformar un documento escrito en otro formato a un documento HTML.

El código HTML que generan es casi imposible de leer, debido a que hay muchas distinciones que existen en lenguajes de formateo de texto que no existen en HTML, y deben ser simuladas, con algún grado de truculencia.

¿Cuándo usarlos? cuando no hay otra alternativa, porque es inviable o ya se hizo un gran trabajo previo en otro sistema.

¿Cómo usarlos? la idea básica es simplificar al máximo el documento (eliminar las sangrías, uniformar los tamaños de letra, etc.) para luego pasar el conversor, generando un documento HTML.

Finalmente conviene revisar la salida usando un editor de texto, revisando que el código obtenido sea sintácticamente correcto, y dejando el documento lo suficientemente ordenado y legible como para que sea fácil de modificar al momento de realizar futuras actualizaciones.

Por ejemplo, el paquete ofimático Office 97 (Word, Excel, PowerPoint, ...) de Microsoft incorpora un conversor de documentos en formato Word a HTML. También el editor FrontPage 98, anteriormente mencionado, incorpora un conversor de distinto tipos de formato de documentos a HTML.

PUBLICAR LAS PÁGINAS EN INTERNET.

Hasta ahora sólo hemos realizado páginas que no son visibles desde Internet, sino que solamente se pueden visualizar en modo local en nuestro ordenador.

Para publicar nuestras páginas en Internet para que sean accesibles por cualquier otro usuario de Internet debemos transferirlas a un servidor Web, para ello hay que realizar los siguientes pasos:

1. Probar de forma local el correcto funcionamiento de las páginas creadas.
2. Solicitar la apertura de una cuenta en dicho servidor Web.
3. Transferirlas mediante FTP a la cuenta asignada en dicho servidor Web. Teniendo en cuenta que las páginas de deben transferir en modo ASCII, y los gráficos incluidos en las páginas en modo binario.
4. Probar desde Internet el acceso a dichas páginas.

Concretamente en la Universidad de Murcia, el Área de Tecnologías de la Información y las Comunicaciones Aplicadas (ATICA) ofrece a los Centros, Departamentos, Servicios, Grupos de trabajo e investigación universitarios la posibilidad de incorporar sus propias páginas de Web al servidor central de la Universidad de Murcia. Puede encontrar más información sobre este Servicio de Almacenamiento de páginas en Web en <http://www.um.es/atika/web/>

DIRECCIONES INTERNET DE INTERÉS.

Las siguientes direcciones Internet contienen guías y manuales donde puede encontrar más información sobre el lenguaje HTML:

Diversos manuales HTML - <http://www.dic.uchile.cl/manual.html>

Dpto. Informática y Comunicaciones
Universidad de Chile

WebMaestro – <http://www.wmaestro.com/webmaestro/>

Francisco Arocena

Tutorial HTML - <http://dns.uncor.edu/info/tutorial.htm>

Mercedes Doffi mdoffi@com.uncor.edu

Universidad Nacional de Córdoba
Argentina

Taller HTML - <http://www2.uca.es/taller-html/taller-html.html>

Universidad de Cádiz
España

Especificación del Consorcio W3 (W3C) ·del lenguaje HTML - <http://www.w3.org/MarkUp/>

Caracteres especiales en HTML –

<http://www.utoronto.ca/webdocs/HTMLdocs/NewHTML/entities.html>



BIBLIOGRAFÍA.

Tejedores del Web - <http://www.dic.uchile.cl/~manual/>

Carlos Castillo

Dpto. Informática y Comunicaciones

Universidad de Chile

Tutorial de HTML - <http://www.um.es/~psibm/tutorial/>

Juan José López García [jjloga@fcu.um.es]

Dpto. Psicología Básica y Metodología

Universidad de Murcia

WebMaestro – <http://www.wmaestro.com/webmaestro/>

Francisco Arocena



UNIVERSIDAD
DE MURCIA



*Área de Tecnologías de la Información y
las Comunicaciones Aplicadas*