



DOCENTE
L.I. Luis Adrián Valdez Mendivil
vamen@live.com.mx

Antología Páginas Web

UNIDAD ACADÉMICA DE TRABAJO
JUAN DE DIOS BATIZ
CONALEP SINALOA

Modulo:
Diseño y Elaboración de Páginas Web

Nombre Alumno : _____
Grupo: _____ Fecha: _____

Índice

Antología Página Web

Introducción	3
Word Wide Web	4
Uso del servicio Word Wide Web	4
Conceptos básicos	5
HTML	8
Características generales del lenguaje HTML	10
Marcas y atributos	10
Comandos básicos	13
Definición de párrafos	16
Listas	17
Tablas	19
Insertar imágenes	21
Multimedia	26
Formularios	30
Frames	38
Icono en favoritos	40
Java script	41
HTML5 Y CSS	

Introducción

El módulo de Diseño y elaboración de páginas web se imparte en el sexto semestre y forma parte del núcleo de formación profesional de la carrera de Profesional Técnico y Profesional Técnico Bachiller en Informática. Su finalidad es que el alumno diseñe y desarrolle sitios web que combinen páginas con contenido estático y páginas con diferentes funcionalidades, dinámicas e interactivas facilitando la actualización de la información contenida en las bases de datos.

El módulo está conformado por dos unidades de aprendizaje. La primera unidad, le permite al alumno determinar la estructura del sitio web y el diseño de las páginas web estáticas siguiendo las especificaciones del W3C, haciendo uso de los lenguajes y herramientas correspondientes, lo cual le permitirá dar atención a los requerimientos del usuario. La segunda unidad, le ofrece los elementos para desarrollar páginas web dinámicas con acceso a bases de datos haciendo uso de lenguajes de programación del lado del usuario y del lado del servidor.

Los resultados del aprendizaje que plantea este módulo hacen una aportación importante para la construcción del perfil de egreso de la carrera de Profesional Técnico y Profesional Técnico-Bachiller en Informática, toda vez que le permiten estar en condiciones de estructurar soluciones de diseño y elaboración de páginas web cuya finalidad es la de satisfacer necesidades de información específicas de manera rápida y fácil a través de internet, y ofrecer productos y/o servicios de manera integral a través de un sitio web; promoviendo el desarrollo de competencias relativas al pensamiento reflexivo, la creatividad, la colaboración y, la comunicación.

El módulo de Diseño y elaboración de páginas web se relaciona con los módulos anteriores de Programación Básica, Construcción de bases de datos, y Programación con sistemas gestores de bases de datos, lo que le permite contribuir una formación integral en cuanto al manejo de una programación estructurada y de bases de datos.

Estas competencias se complementan con la incorporación de otras competencias básicas, profesionales y genéricas que refuerzan la formación tecnológica y científica, y fortalecen la formación integral de los educandos; que los prepara para comprender los procesos productivos en los que está involucrado para enriquecerlos, transformarlos, resolver problemas, ejercer la toma de decisiones y desempeñarse en diferentes contextos laborales, con una actitud creadora, crítica, responsable y propositiva; fomentando el trabajo en equipo, el desarrollo pleno de su potencial profesional y personal, y la convivencia de manera armónica con la sociedad y el medio ambiente.

Word Wide Web

El servicio **World Wide Web**, 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, correo electrónico, entre otros). 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.

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** (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 así como los diferentes efectos que se quieren dar.

Uso 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, navegadores o visualizadores**. 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** (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.

Conceptos básicos

Sitio Web

Un sitio Web es un conjunto de páginas web conectadas entre sí por medio de enlaces(link).

Hospedaje Web (Hosting)

El **hosting** es un espacio virtual en un servidor conectado permanente a internet, al colocar tu sitio en línea haces accesible a cualquier usuario conectado a internet sin importar la hora o lugar.

Nombre de Domino

Un dominio es tu dirección en internet.

Al adquirir un dominio personalizas tu sitio Web y tus cuentas de correo electrónico. Un dominio está compuesto por:

Nombre	Extensión
Nombre_del_sitio	.com
<ul style="list-style-type: none"> ▪ Tú lo eliges. ▪ Es exclusivo. ▪ De preferencia corto, que describa tu actividad y fácil de recordar. 	<ul style="list-style-type: none"> ▪ Tú lo eliges. ▪ Indica la actividad que realizas o procedencia territorial.

Página Web

Una **página web** es el nombre de un documento o información electrónica adaptada para la *World Wide Web* y que puede ser accedida mediante un navegador para mostrarse en un monitor de computadora o dispositivo móvil. Esta información se encuentra generalmente en formato HTML, y puede proporcionar navegación a otras páginas web mediante enlaces de hipertexto. Las páginas web frecuentemente incluyen otros recursos como hojas de estilo en cascada, guiones (*scripts*) e imágenes digitales, entre otros.

Realiza un mapa conceptual utilizando los cuatro conceptos básicos de la página 5.

Servicios en una página de internet

FTP (Protocolo de Transferencia de Archivos)

En informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.

Correo electrónico

Es un servicio de red que permite a los usuarios enviar y recibir mensajes y archivos rápidamente mediante sistemas de comunicación electrónicos. Principalmente se usa este nombre para denominar al sistema que provee este servicio en Internet, mediante el protocolo (SMTP).

Transferencia simple de correo electrónico (SMTP) Protocolo de red basado en texto, utilizados para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos (PDA, teléfonos móviles, etc.). Es un estándar oficial de Internet.

Introducción

Esta antología está pensada para aprender los conceptos básicos necesarios para construir sus propias páginas 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.** Puede ser 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). Si se utilizan procesadores como el Word, se deben guardar los ficheros como 'Sólo Texto', para que no introduzca órdenes de formato, que pueden provocar errores al cargarlo en el navegador.
- **Un visualizador o navegador** del Web: Netscape, Microsoft Explorer, Chrome, 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.

El 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 deHTML.

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 abra el Block de Notas
2. Teclee lo siguiente:

```
<HTML>
<HEAD>
<TITLE> Ejemplo 1 - Mi primera página Web </TITLE>
</HEAD>

<BODY>
<CENTER><H1>Mi primera página Web</H1></CENTER>
  Esta es mi primera página, aunque todavía es muy sencilla. <BR>
  Como el lenguaje HTML no es difícil, pronto estaré en condiciones de hacer cosas más
  interesantes.
</BODY>
</HTML>
```

3. Grabe este archivo con el nombre: index.html dentro de una carpeta de tu memoria USB llamada MI PRIMERA PAGINA
4. Abra el archivo index.html dando doble click sobre él.
6. 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.

Las líneas en blanco y saltos de línea 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.

¡Felicidades has creado tu 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 lo 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.

1.- 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.

2.- 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.

3.- No importa 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.

4.- 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	&<vocal a acentuar> acute;	En la actualidad la mayoría de los browsers los
Símbolo menor que (<)	<	
Símbolo mayor que (>)	>	
La eñe	ñ	En la actualidad la mayoría de los browsers la
Espacios en blanco	 (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: **&#código_ASCII;**

Estructura de un documento HTML

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

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

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

```
<HTML>
<TITLE>titulo de la página</TITLE>
<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 información del documento, que no se ve en la pantalla principal, y que precisa las características del documento, principalmente el título del documento.

Título del documento: se declara entre las etiquetas <TITLE> y </TITLE>. El título debe ser breve y descriptivo de su contenido, pues será lo que vean los demás cuando añadan nuestra página a su bookmark (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 carácter imprimible.

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

Comandos básicos.

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

Títulos 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:

Negrita (Bold)

<I>Cursiva (Italic)</I>

<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:

Texto

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.

Texto

Donde argumento es un número entre 1 y 7

- Tamaño de texto relativo.

Texto

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 cuál 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">
.....
(Contento 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 FFFFFF, 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">
```

Definición de párrafos.

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

Ruptura de línea.

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.

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.

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 término.
  <DT>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.

Centrado de 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 unas 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-->
```

Tablas

Las tablas son sin duda uno de los elementos más potentes del HTML. Con pocos elementos se pueden conseguir efectos espectaculares. En el interior de las celdas de una tabla, que pueden ser con borde visible o invisible, se puede poner cualquier cosa: texto de cualquier tamaño y color, imágenes, links... Por supuesto, además de permitir cualquier contenido, tienen sus propios atributos de alineación tanto horizontal como vertical, y atributos de dimensionado. Por defecto se autodimensionan, es decir, se adaptan en tamaño a su contenido, pero también es posible definirlo de forma fija. Las tablas han sido utilizadas largo tiempo para maquetar el contenido de las páginas, y

aunque las nuevas tendencias aconsejen hacer esto con bloques (capas XHTML), no es un elemento que vaya a desaparecer, y es mucho más sencillo de utilizar.

El elemento básico de definición de tabla es `<TABLE>` `</TABLE>` y en su interior se disponen los sub elementos `<TR>` para definir una fila (**Row**) `<TH>` para definir una cabecera (**Header**) `<TD>` para definir una celda de datos. Estos sub elementos también han de llevar sus correspondientes cierres: `</TR>` `</TH>` `</TD>`.

Una cabecera `<TH>` es lo mismo que una celda de datos `<TD>` pero de forma automática el texto de su contenido recibe los atributos de negrita y centrado. Sólo es posible definirlos al principio de las filas, de las columnas o de ambas a la vez.

He aquí una tabla-resumen de los elementos utilizados y los atributos que admite cada uno:

	TABLE	TR	TD	TH	CAPTION
BORDER	X	-	-	-	-
BORDERCOLOR	X	-	-	-	-
ROWSPAN	-	-	X	X	-
COLSPAN	-	-	X	X	-
ALIGN	-	X	X	X	X
VALIGN	-	-	X	-	-
WIDTH	X	-	X	-	-
HEIGHT	X	-	X	-	-
CELLPADDING	X	-	-	-	-
CELLSPACING	X	-	-	-	-
Resumen de tablas					

Veamos el significado de cada atributo:

- **BORDER** Indica el ancho de los bordes de la tabla. Se mide en píxeles. Si no se escribe este atributo, es equivalente a `BORDER=0` (por defecto).
- **BORDERCOLOR** Establece el color de los bordes de la tabla. No funciona igual en todos los navegadores.
- **CELLSPACING** Indica el número de píxeles que separan una celda de otra. Aunque pueda parecerlo, no hace lo mismo que `BORDER`. Su valor por defecto es 2.
- **CELLPADDING** Indica los píxeles de separación entre el borde de la celda y su contenido. Su valor por defecto es 1.
- **WIDTH** Según donde se escriba, sirve para controlar el ancho de toda la tabla o de sus columnas. Si se incluye en `<TABLE>` puede indicar el tamaño tanto en píxeles como en porcentaje respecto al ancho de la pantalla.
- **HEIGHT** Según donde se escriba, sirve para controlar el alto de toda la tabla o de sus columnas. Si se incluye en `<TABLE>` puede indicar el tamaño tanto en píxeles como en porcentaje respecto al alto de la pantalla.

- **ALIGN** Indica la alineación horizontal de los datos dentro de las celdas. Puede tener tres valores: **LEFT** (izquierda), **RIGHT** (derecha) y **CENTER** (centro).
- **VALIGN** Indica la alineación vertical de los datos dentro de las celdas. Puede tener tres valores: **TOP** (arriba), **BOTTOM** (abajo) y **MIDDLE** (centro).
- **ROWSPAN** Se utiliza en la definición de una celda (<TD>) o cabecera (<TH>) para indicar que su anchura o altura son equivalentes a un determinado número de filas.
- **COLSPAN** Se utiliza en la definición de una celda (<TD>) o cabecera (<TH>) para indicar que su anchura o altura son equivalentes a un determinado número de columnas.
- **NOWRAP** Para impedir que las líneas de texto dentro de una celda se trunquen en los espacios en blanco.

Tabla simple:

```
<table>
<tr>
<td>Celda 1</td>
<td>Celda 2</td>
<td>Celda 3</td>
</tr>

<tr>
<td>Celda 4</td>
<td>Celda 5</td>
<td>Celda 6</td>
</tr>
</table>
```

Vista:

Celda 1	Celda 2	Celda 3
Celda 4	Celda 5	Celda 6

Unificación de celdas

```
<table>
<tr>
<td>Campo 1</td>
<td>Campo 2</td>
<td>Campo 3</td>
</tr>
<tr>
<td colspan="2">Campos 4 y 5</td>
<td>Campo 6</td>
</tr>
<tr>
<td>Campo 7</td>
<td>Campo 8</td>
<td>Campo 9</td>
</tr>
</table>
```

Vista:

Campo 1	Campo 2	Campo 3
Campos 4 y 5		Campo 6
Campo 7	Campo 8	Campo 9

```

<table>
<tr>
<td rowspan="3">Campo unificado</td>
<td>Campo 2</td>
<td>Campo 3</td>
</tr>
<tr>
<td>Campo 5</td>
<td>Campo 6</td>
</tr>
<tr>
<td>Campo 8</td>
<td>Campo 9</td>
</tr>
</table>

```

Campo unificado	Campo 2	Campo 3
	Campo 5	Campo 6
	Campo 8	Campo 9

Celdas de encabezado:

```

<table>
<tr>
<th scope="col">Columna 1</th>
<th scope="col">Columna 2</th>
<th scope="col">Columna 3</th>
</tr>
<tr>
<td>Celda 1</td>
<td>Celda 2</td>
<td>Celda 3</td>
</tr>
<tr>
<td>Celda 4</td>
<td>Celda 5</td>
<td>Celda 6</td>
</tr>
</table>

```

Vista:

Columna 1	Columna 2	Columna 3
Celda 1	Celda 2	Celda 3
Celda 4	Celda 5	Celda 6

Insertar imágenes

Insertar imágenes en un documento permite crear páginas mucho más atractivas. Según el tipo de gráficos utilizado se pueden conseguir efectos realmente sorprendentes.

Para insertar una imagen en un documento HTML se utiliza el elemento ****. Este elemento puede ir acompañado de los atributos **SRC ALT ISMAP ALIGN WIDTH HEIGHT BORDER VSPACE HSPACE**.

Las imágenes utilizadas pueden estar en formato GIF o JPG. Estos formatos son interpretados directamente por el navegador. La diferencia entre estos dos formatos es su nivel de compresión. En efecto, el JPG tiene una compresión mucho mayor que el GIF, por lo que suele ser el formato más utilizado en el mundo web, donde siempre se debe perseguir que las páginas sean lo más ligeras posible. Por contra, cuanto mayor compresión tenga un gráfico menor calidad se consigue, pero para usos generales el JPG es perfectamente válido. Se puede usar cualquier otro formato como el MPG o el AVI (ambos de vídeo), pero entonces el visualizador llamará a un programa auxiliar, que previamente le habrás definido, para que sea éste el que visualice el fichero. La desventaja de las imágenes en formatos no tratados por el propio visualizador es que no verás el texto junto con la imagen, y por tanto no podrás imprimir la página compuesta. Para evitar esto, se utilizan los llamados "plug-in" que consisten en pequeños programas que permiten llamar a aplicaciones específicas ejecutándose como ventanas del navegador.

Imagen alineada a la izquierda (por defecto)

```
<IMG SRC="sugeren.gif">texto texto texto texto texto texto texto texto
```



texto texto texto texto texto texto texto texto

Imagen con un texto alternativo

```
<IMG SRC="sugeren.gif" ALT="AQUI VA UNA IMAGEN">
```

Sirve para que, si no es posible cargar la imagen, aparezca un texto en su lugar.

También se utiliza para que al colocar el puntero del ratón sobre la imagen, sin pulsar, aparezca el texto.



Imagen alineada a la izquierda. Texto alineado arriba

```
<IMG SRC="sugeren.gif" ALIGN=TOP> texto texto texto
```



texto texto texto

Imagen alineada a la izquierda. Texto alineado abajo

```
<IMG SRC="sugeren.gif" ALIGN=BOTTOM> texto texto texto
```



texto texto texto

Imagen alineada a la izquierda. Texto alineado al centro

```
<IMG SRC="sugeren.gif" ALIGN=MIDDLE> texto texto texto
```

**Imagen redimensionada a más. Texto alineado a la derecha de la imagen**

```
<IMG SRC="sugeren.gif" ALIGN=LEFT WIDTH=272 HEIGHT=92 > texto texto texto
```

***Imagen de fondo***

El fondo (background) de este documento es una imagen en formato GIF.

Es un atributo del elemento <BODY>

Se escribe así:

```
<BODY BACKGROUND="yellow_r.gif">
```

Y es compatible con todos los demás atributos de este elemento.

Enlaces (Links)

El siguiente es sin duda el elemento más importante del HTML, ya que *permite realmente "navegar" por uno o varios documentos*, que pueden residir en cualquier parte, pareciéndonos que siempre estamos en el mismo; a esto se le llama **hipertexto** ya que con esta posibilidad, en realidad, nuestro documento puede ser infinito en efecto, para la persona que está leyendo sobre un determinado tema, no hay diferencias evidentes que le hagan notar dónde está el documento que lee. Para el lector, todo parece un mismo documento, cuando en realidad, probablemente el conjunto de lo que lee está repartido por medio mundo, o en un plano más modesto, el documento leído puede estar compuesto en realidad por varios cientos de páginas que "saltan" de unas a otras sin notarlo.

Todo esto lo consigue el elemento <A> (por **A**nchor, en inglés = ancla o punto de anclaje). En lo sucesivo le llamaremos enlace o simplemente **link** (en inglés link=eslabón o enlace).

Pero seguramente te preguntarás qué es eso de un "link". Pues un link es un área de la pantalla, que puede contener una o varias palabras o una imagen, que es "sensible" al puntero del ratón, y al ponerlo sobre ella y pulsar el botón izquierdo el visualizador llamará a la página que tiene asignada el link. Habitualmente por defecto los visualizadores señalan un área linkeada subrayándola, si es texto, o poniéndole un borde si es una imagen, ambas cosas en color azul. Por supuesto, eso se puede cambiar en las configuraciones de visualizador. Si no se desea que aparezca el subrayado para mejorar la estética en algunos casos, se puede parametrizar con una instrucción de estilo.

Se escribirá:

```
<A HREF="http://www.miservidor.es/mifichero.htm"> Link de prueba</A>.
```

Y se vería así: [Link de prueba](http://www.miservidor.es/mifichero.htm)

Si lo que se desea es que aparezca sin subrayado o de otro color, o ambas cosas, se puede recurrir a una instrucción de estilo. Así:

```
<A HREF="http://www.miservidor.es/mifichero.htm" style="color:red;text-decoration:none";>
Link de prueba
</A>.
```

Y se vería así: [Link de prueba](http://www.miservidor.es/mifichero.htm) (Esto puede que no funcione en algunos navegadores antiguos)

El elemento **<A>** tiene dos atributos: **HREF** y **NAME**.

En el ejemplo de arriba al atributo HREF se le asigna el valor que aparece entre comillas: "<http://www.miservidor.es/mifichero.htm>". El texto sobre el que se monta el enlace es **Link de prueba** y realiza un link con el fichero **mifichero.htm** que está en el servidor **www.miservidor.es**.

Esta forma de escribir la ruta del link de forma completa, se utiliza cuando hay que saltar de una máquina a otra, si el enlace es para otra página de nuestro propio servidor es suficiente escribir la ruta virtual corta: **/mifichero.htm** o bien: **/otro_directorio/mifichero.htm** según proceda.

Como has visto, el punto de enlace se realiza en el texto escrito antes del cierre del elemento ****, pero también puede hacerse con una imagen cualquiera, que iría en lugar del texto, incluso puedes poner el texto junto con una imagen. Por ejemplo para hacer un link que nos lleve al índice desde una bola roja, se escribe:

```
Ir al índice <A HREF="indice.htm">  </A>.
```

Y este sería el resultado:

Ir al índice 

Al igual que el texto aparece subrayado en azul cuando forma parte de un enlace, los gráficos reciben un borde azul alrededor de toda la imagen, lo que a veces puede resultar poco estético. Esto también tiene solución, pero aquí se hace en la instrucción de la imagen:

```
Ir al índice <A HREF="indice.htm">  </A>.
```

Y este sería el resultado:

Ir al índice 

El valor de los atributos hay que escribirlos EXACTAMENTE como se deba. En el ejemplo anterior no funcionaría el link si escribiéramos **www.miservidor.es** en mayúsculas o con alguna otra diferencia, eso es un nombre de máquina y sería interpretado como OTRA máquina. Esto es aplicable a todos los parámetros: **servicio** (<http://>) **servidor** (www.miservidor.es) y **fichero** ([mifichero.htm](http://www.miservidor.es/mifichero.htm)).

Dentro de los parámetros pasados a HREF, podrás ver que al principio pone **http://**, esto es, el tipo de servicio al que el visualizador va a llamar, y hay varios :

Servicio	Descripción	Ejemplo	Efecto
http://	Servicios WWW	 WWW	WWW
ftp://	Servicios FTP	FTP	FTP
news://	Servicios NEWS	NEWS	NEWS
mailto://	Servicios E-mail	E-mail	E-mail
file:///C 	Fichero local	Fichero	Fichero

Además de HREF, el elemento <A> puede tener otro atributo: **TARGET**. Se utiliza para ordenar la apertura de otra ventana del visualizador con la página que se desee. Se escribirá:

```
<A HREF="indice.htm" TARGET="Ventana-2"> </A>
```

Esta instrucción mostrará la página **indice.htm** pero con [otra ventana](#) del visualizador, es decir tendremos lanzado el visualizador **2 veces**.

Del mismo modo, el atributo **TARGET** puede utilizarse para salir de una pantalla compuesta de frames. En efecto, si estamos en una pantalla con, por ejemplo, dos frames, cualquier link invocado desde cualquiera de ellos, hace que la página llamada aparezca en ese mismo frame. Para volver a una pantalla "normal" sin frames, se puede escribir un link con la sintaxis siguiente:

```
<A HREF="indice.htm" TARGET="_parent"> </A>
```

Existen otros valores para TARGET:

TARGET="_blank": Para que el enlace se muestre en una nueva ventana vacía. También puede darse un nombre cualquiera con el mismo efecto.

TARGET="_self": Para mostrar el enlace en la misma ventana o frame que lo referencia (valor por defecto).

TARGET="_top": El documento solicitado se cargará en la pantalla actual, ocupando toda la ventana y destruyendo la estructura de frames anterior, si existe.

Como se acaba de ver, el atributo **HREF** sirve para enlazar con otro documento, que puede estar en un servidor o ser un fichero local, pero que siempre se nos presentará desde la primera línea del mismo.

El atributo **NAME** se utiliza para definir algo así como "un punto de aterrizaje" en cualquier línea del documento de destino, de forma que nos aparecerá en pantalla desde la línea deseada y no desde el principio. Esto es muy útil cuando se trata de documentos largos divididos en secciones.

Se escribirá así:

En el documento activo:

```
<A HREF="http://www.miservidor.es/mifichero.htm#punto1"> Ir al punto 1</A>
```

En el documento destino:

```
<A NAME="punto1"></A>
```

NAME también puede utilizarse para saltar de una línea a otra dentro de un mismo documento. Se escribirá así:

Dentro del documento activo En la línea de partida:

```
<A HREF="#punto1">Ir al punto 1</A>
```

En la línea de destino

```
<A NAME="punto1"></A>
```

Por ejemplo, si haces "clik" [aquí](#) con el ratón, saltarás al principio de esta página.

Como habrás visto en el cuadro de arriba, es posible enviar un e-mail desde un link con la instrucción:

```
<A HREF="mailto:jac@uv.es">Enviar e-mail</A>.
```

Con esta sintaxis se invoca al cliente de correo predeterminado para que abra una ventana con la dirección deseada ya escrita. Si además quieres que dicha ventana se abra con el asunto (subject) y el texto (body) ya escritos se puede conseguir así:

```
<A HREF="mailto:jac@uv.es?&subject=Asunto de prueba&body=Texto de prueba">Enviar e-mail</A>
```

Multimedia

Pueden utilizarse indistintamente ficheros multimedia con las extensiones .wav, .aiff, .au, .mid, .mp3, .mpg, .mpeg, .avi, .wmv, .rm, etc., etc. Los navegadores no admiten directamente ningún tipo de formato multimedia por defecto, y es necesario cargar un plugin, que el cliente tendrá que buscar e instalar.

Un plugin es un pequeño programa que hará las veces de conector entre el navegador y el fichero multimedia, es decir, que dotará al navegador de una cierta capacidad que de serie no tiene. Los plugins no solamente son necesarios para reproducir los ficheros multimedia, sino también para poder cargar cualquier otro tipo de fichero que no sea HTML, que es el único formato que realmente entienden los navegadores.

Casi todos los navegadores precisarán de un plugin para reproducir cualquier tipo de formato (multimedia o no), excepto el Internet Explorer que echa mano de los ActiveX propios de los sistemas operativos Windows.

El primer elemento disponible en HTML para los ficheros multimedia fue **<EMBED>**. Este elemento, que en realidad nunca llegó a estar incluido oficialmente en el lenguaje por la w3c, todavía funciona en todos los navegadores modernos, pero presenta algunos problemas, que al no estar sujeto a estandarización alguna, ha provocado que cada navegador interprete a su manera las cosas. No está en la especificación del nuevo XHTML, que recomienda el uso del elemento **<OBJECT>** para estos menesteres.

<EMBED> cuenta con un buen número de atributos: **SRC, AUTOSTART, LOOP, STARTTIME, ENDTIME, VOLUME, WIDTH, HEIGHT, ALIGN, CONTROLS y HIDDEN.**

SRC Indica dónde está el fichero a reproducir, pero no puede utilizarse solo, necesita la compañía de otros atributos del elemento, concretamente los que definen el tamaño de la consola (que es un icono parecido al teclado que tiene un cassette pequeño): **WIDTH** (largo) y **HEIGHT** (alto).

Otro atributo de EMBED es **AUTOSTART** con dos posibles valores: **FALSE** (por defecto) y **TRUE**, y como ya habrás deducido, TRUE hace que comience la reproducción inmediatamente después de cargar la página, y sin necesidad de pulsar el botón "start" de la consola.

El siguiente atributo a analizar es **HIDDEN** que sólo tiene un valor: **TRUE**, y hace invisible la consola, pero evidentemente, si es invisible no puedes activar el sonido, y por tanto lo hace él mismo, siempre que no hayas puesto **AUTOSTART=FALSE**, claro.

Un atributo de repetición no podía faltar, y lo tenemos en **LOOP** y sus tres posibles valores: **FALSE** (por defecto), **TRUE** que fuerza a la repetición sin fin de la audición, e **INTEGER** donde **INTEGER** es el número de veces que deseas que se repita la audición.

No es obligado comenzar una audición por el principio del fichero. Imagínate que tienes una serie de instrucciones habladas, y quieres dar la oportuna en cada momento, no todas de golpe. Para esto tenemos los atributos **STARTTIME=mm:ss** y **ENDTIME=mm:ss** donde **mm** son los minutos y **ss** los segundos del punto de comienzo o fin de la audición. **Cuidado**: los valores hay que darlos con dos dígitos:

Por último tenemos el atributo **ALIGN** y sus valores **TOP, BOTTOM, CENTER, LEFT, RIGHT, MIDDLE y TEXTTOP** que funcionan igual que sus respectivos del elemento **IMG**, es decir, que no tiene efectos sobre el sonido o el vídeo, sino sobre la imagen de la consola y su alineación con el texto de la página.

He aquí un breve ejemplo con los atributos básicos:

```
<EMBED SRC="minueto.mid" WIDTH=160 HEIGHT=70 AUTOSTART="FALSE">
```

Introducción a formularios

Generalidades

Esta es una de las partes más técnicas y complejas del HTML. Los formularios o *forms* en inglés, son unas páginas especiales que se utilizan para realizar transacciones. En una página que contenga un formulario pueden además existir todos los elementos hasta ahora conocidos, incluso el propio formulario puede estar integrado dentro de otros elementos, como por ejemplo una tabla. Los formularios, en algunas impresoras, pueden dar problemas al imprimir las páginas que los contienen.

¿Cómo funcionan los formularios?

El formulario no es más que una página escrita en HTML como cualquier otra. Lo que la hace diferente es que permite capturar datos de varias maneras: directamente desde el teclado, escogiendo un valor de una lista desplegable o seleccionando una opción desde botones fijos o de todas ellas combinadas. Una vez terminada la captura de datos, estos forman un registro que es enviado a un servidor http (web) que lo procesará y devolverá una respuesta (o no). Pueden utilizarse también para enviar su contenido a una dirección de correo electrónico, o que simplemente abran otra página.

Código dentro del archivo para el envío:

```
<html>
<head>
<title> MENSAJE ENVIADO </title>
<meta http-equiv="Page-Enter" content="revealTrans(Duration=1,Transition=12)">
<meta http-equiv="Refresh" content="7;url=http://delasalle.tiase.com.mx/contacto.html">
</head>
<body>

<?php
$cuero = "Mensaje enviado desde TI@se.com.mx\n";
$cuero .= "Nombre: " . $_HTTP_POST_VARS["name"] . "\n";
$cuero .= "Email: " . $_HTTP_POST_VARS["email"] . "\n";
$cuero .= "Asunto: " . $_HTTP_POST_VARS["subject"] . "\n";
$cuero .= "Texto: " . $_HTTP_POST_VARS["text"] . "\n";
mail("saldinas@hotmail.com,rafael.medina@tiase.com.mx","Mensaje enviado desde
TI@se.com.mx",$cuero);
echo "
<font face=arial color=000000 size=2><br><br><br><br><center>su mensaje ha sido enviado
correctamente.<br>nos comunicaremos con usted en la mayor brevedad posible.";
?>
</body>
</html>
```

Código dentro de nuestra página:

```

<html>
<head>
<title> CONALEP COSTA RICA</title>
<script>
function Validar(form){
if (FORMULARIO.name.value == "")
{ alert("POR FAVOR INGRESE SU NOMBRE"); FORMULARIO.name.focus(); return;}
if (FORMULARIO.email.value == "")
{ alert("POR FAVOR INGRESE SU E-MAIL"); FORMULARIO.email.focus(); return;}
if (FORMULARIO.subject.value == "")
{ alert("POR FAVOR INGRESE EL ASUNTO"); FORMULARIO.subject.focus(); return;}
if (FORMULARIO.text.value == "")
{ alert("POR FAVOR INGRESE SUS COMENTARIOS"); FORMULARIO.text.focus(); return;}
FORMULARIO.submit();
}
</script>
</head>
<body>
<center>
<table border="0" width="950" cellspacing="0" cellpadding="0"><br><br>
<tr width="950" align="center">
<td>
<font face="lucida sans, verdana" size="2">
<form name=formulario action="contacto.php" method="post">
Nombre:
<INPUT TYPE="text" NAME="name" SIZE="41" MAXLENGTH="30"><BR><BR>
Correo:&nbsp;
<INPUT TYPE="text" NAME="email" SIZE="41" MAXLENGTH="30"><BR><BR>
Asunto:&nbsp;
<INPUT TYPE="text" NAME="subject" SIZE="41" MAXLENGTH="30"><BR><BR>
<label>
<textarea name="text" cols="39" rows="5"></textarea>
</label> <BR><BR>
<INPUT TYPE="button" VALUE="Envia" onClick="Validar(this.form)"
onmouseover="window.status='ENVIAR ';return true" onmouseout="window.status='';return true">
<input type="reset" value="Limpiar"
onmouseover="window.status='LIMPIAR ';return true" onmouseout="window.status='';return true">
</form>
</td>
</tr>
</table>
</body>
</html>

```

Frames

Los frames (en inglés *frame* = cuadro, bastidor o marco) es un elemento implementado por Netscape, que permite dividir la pantalla en varias áreas independientes unas de otras, y por tanto con contenidos distintos, aunque puedan estar relacionados. No hay límites para el contenido de cada una de estas áreas: tienen las mismas propiedades que la pantalla completa normal, tal y como la conocemos. No hay que confundir los Frames con las tablas. Su apariencia, a veces, puede ser similar, pero mientras el contenido de la celda de una tabla es fijo, en un área de pantalla creado por el elemento **FRAME** se dispone de todos los recursos del HTML. Es una zona viva.

Las páginas que contienen una definición de FRAME **no pueden** contener el elemento BODY ni ninguno de los elementos que habitualmente aparecen en el BODY **antes** del elemento **FRAMESET** que es el que define la creación del FRAME. Si esto no se cumple, el FRAME será ignorado.

Algunos visualizadores no soportan los Frames. Para que nuestra página con Frames no resulte opaca a ellos, se utilizará el elemento **NOFRAMES** que permite ofrecer un texto alternativo en entorno normal.

Los Frames permiten una flexibilidad de presentación extraordinaria, y para contenidos muy complejos, de difícil estructura por los medios convencionales del HTML, son casi insustituibles. De nuevo se presenta la duda de para qué visualizador es preferible programar nuestro WWW.

Por supuesto, los Frames son parametrizables en cuanto a tamaño y número de áreas, si éstas se pueden redimensionar por el usuario o son fijas, si tienen o no barras de scrolling, se pueden anidar, relacionar sus contenidos, etc. Veamos algunos ejemplos prácticos y sus sintaxis:

Un Frame básico. Sintaxis general

En general, todas las páginas que contengan definiciones de frames, se comportan como si fuesen llamadores o "lanzaderas", y deberán ser más o menos así:

```
<HTML>
<HEAD><TITLE> Mi título ></TITLE></HEAD>

<FRAMESET>
  <NOFRAMES>
    <BODY>
      Su visualizador no soporta frames. Pulse
      <A HREF="indice.htm">aquí </A> para volver.
    </BODY>
  </NOFRAMES>

  <FRAME SRC="pagina1.htm" >
  <FRAME SRC="pagina2.htm" >
</FRAMESET>
</HTML>
```

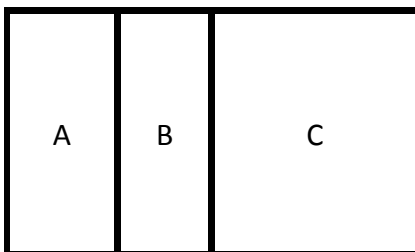
Fíjate en que no aparece el elemento <BODY> en su posición habitual, sino dentro de un elemento especial que se activa sólo cuando el visualizador no soporta los frames: **NOFRAMES**. Si no declaras el área NOFRAMES y el visualizador no soporta este efecto, no se verá nada. Obviamente, los visualizadores que sí soportan frames ignorarán el contenido del áreaNOFRAMES.

Hasta aquí ya podemos hacernos una idea de cómo funcionan los frames: Lo primero es crear una minipágina con la definición del frame, viene a ser algo así como una "lanzadera" y sólo contiene la definición del frame. Y por último, crear las páginas que constituirán el contenido de cada una de las áreas definidas en la "lanzadera". En el ejemplo anterior son **pagina1.htm** y **pagina2.htm**, y aquí es donde escribirás tus cosas; es decir que son páginas completamente normales, y que también pueden ser utilizadas de la forma habitual. A su vez, la "lanzadera" puede ser invocada con un link desde cualquier página normal. Como puedes ver, no es obligado crear todo tu documento con frames: puedes utilizarlos solamente allí donde sea necesario, si quieres.

Es posible definir cualquier combinación de áreas verticales y horizontales. La clave está en combinar adecuadamente los anidamientos del elemento <FRAMESET> con sus atributos **COLS** o **ROWS**según interese. Como puedes ver, el más importante es el primer <FRAMESET>, ya que es el que define cómo va a ser "troceada" la página inicialmente, si en porciones verticales u horizontales, y sobre esta base se deberán definir todos los demás anidamientos.

Un frame de 3 áreas verticales (COLS)

Se vería así:

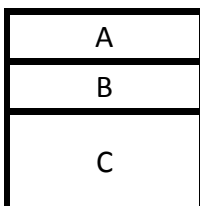


Y se escribiría; así:

```
<FRAMESET COLS=30%,20%,50%>
  <FRAME SRC="a.htm">
  <FRAME SRC="b.htm">
  <FRAME SRC="c.htm">
</FRAMESET>
```

Un frame de 3 áreas horizontales (ROWS)

Se vería así:

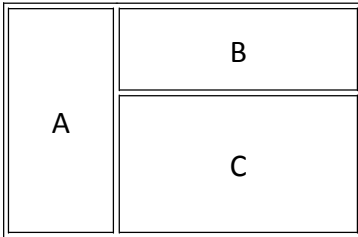


Y se escribiría así:

```
<FRAMESET ROWS=25%,25%,50%>
  <FRAME SRC="a.htm">
  <FRAME SRC="b.htm">
  <FRAME SRC="c.htm">
</FRAMESET>
```

Un frame combinado de un área vertical y dos horizontales

Se vería así:

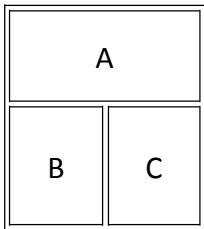


Y se escribiría así:

```
<FRAMESET COLS=20%,*>
  <FRAME SRC="a.htm">
    <FRAMESET ROWS=40%,*>
      <FRAME SRC="b.htm">
      <FRAME SRC="c.htm">
    </FRAMESET>
</FRAMESET>
```

Un frame combinado de dos áreas horizontales, la de abajo dividida en dos verticales

Se vería así:



Y se escribiría así:

```
<FRAMESET ROWS=50%,*>
  <FRAME SRC="a.htm">
  <FRAMESET COLS=50%,*>
    <FRAME SRC="b.htm">
    <FRAME SRC="c.htm">
  </FRAMESET>
</FRAMESET>
```

JavaScript

JavaScript es un lenguaje de programación interpretado, Se define como orientado a objetos, basado en prototipos, imperativo y dinámico.

Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador. Aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos [PDF](#), aplicaciones de escritorio (mayoritariamente [widgets](#)) es también significativo.

JavaScript se diseñó con una sintaxis similar al lenguaje C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

PASOS PARA REALIZAR EL REGISTRO EN HOSTING

1.- Ingresa al navegador web y teclea la siguiente dirección: www.geocities.ws

Te muestra la siguiente pantalla:



2.- A continuación haz clic en **Sign Up Now!** Para crear tu cuenta de usuario.

Una vez validado tu cuenta, anota tu usuario en el siguiente recuadro:

http://www.geocities.ws/

Anota tu contraseña en el siguiente recuadro

Contraseña:

3.- A continuación ingresa como usuario registrado haciendo clic en [Members](#) ubicado en la parte superior derecha de la pantalla.


4.- Una vez ingresado, dentro del panel haz clic en **File Manager** para empezar a construir tu sitio Web.








5.- Dentro del directorio te mostrara los archivos existentes dentro de este, donde deberás ubicar fácilmente el archivo index.html (este es tu archivo principal, el cual iniciara cuando ingreses tu dominio http://www.geocities.ws/TU_USUARIO)

Este archivo se genera automáticamente al momento de crear tu usuario, el cual deberás de eliminar y crear uno nuevo para tu proyecto final de semestre.

■■■■■■■■■■ 100% Script finished in 1.93 seconds






Language:

Directory Tree: root /

New dir New file Upload
Transform selected entries:

All	Name	Type	Size	Mod Time	Actions
	Up ..				
	index.html	HTML file	655	May 18 2012	View Edit Open

Directories: 0
 Files: 1 / 655 B
 Symlinks: 0
 Unrecognized FTP output: 0
 Data transferred from this IP address today: 0 B
 Data transferred to this FTP server today: 33.37 MB

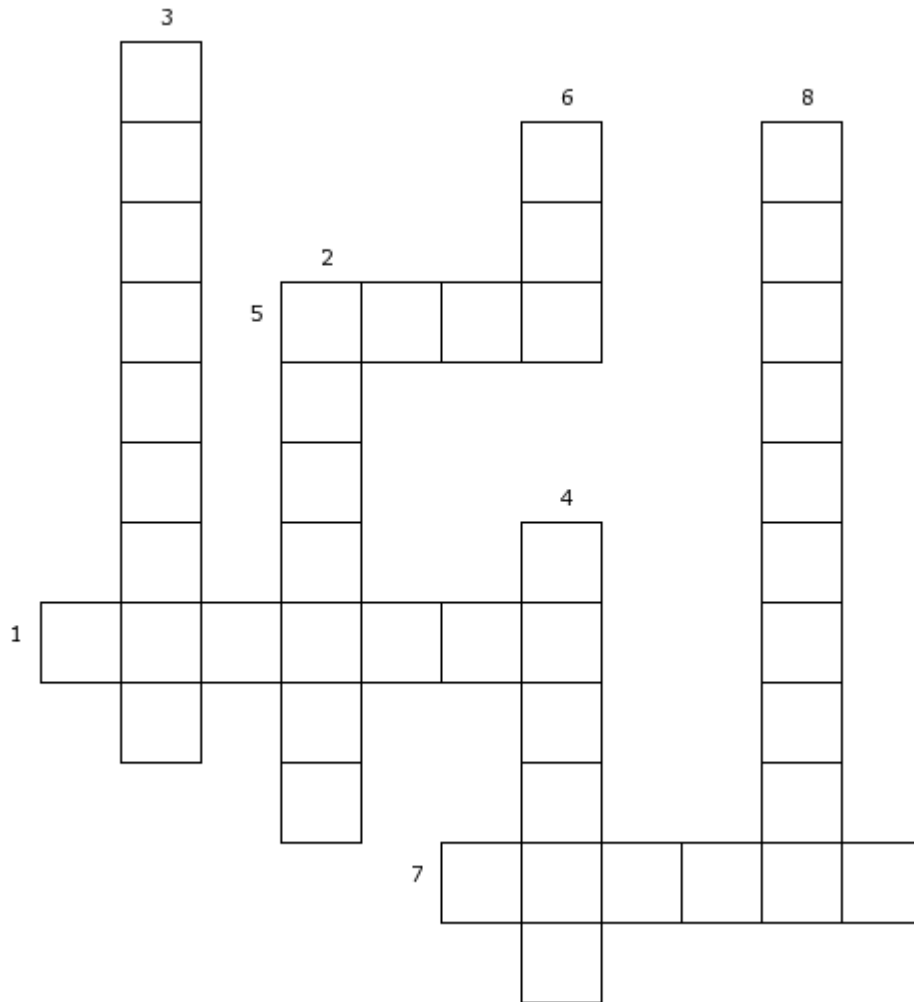
INSTRUCCIÓN I.- Llena la siguiente sopa de letras de acuerdo a las pistas que se te proporcionan a continuación (utiliza colores).

1. Es un Sistema de Información distribuido por Internet basado en la tecnología hipertexto/hipermedia
2. Es la interfaz entre el usuario y el internet que puede interpretar información HTML
3. Es el Lenguaje de Marcas Hipertexto
4. Pueden estar localizados en diferentes sitios de Internet, estos sitios son llamados servidores Web.
5. Es un texto en que cualquier palabra puede ser especificada como un enlace a otros documentos que contienen más información sobre dicha palabra.
6. Es un hipertexto que incluye gráficos, videos y sonido.
7. Es un Localizador de Recursos Uniforme.
8. Es un conjunto de páginas web conectadas entre sí por medio de enlaces.
9. Es un espacio virtual en un servidor conectado permanente a internet.
10. Es una dirección en internet.

J	B	Y	V	L	X	U	N	P	R	G	H	V	X	K	P	L	E	B	N	Y	B	V	R	A
O	D	O	M	I	N	I	O	O	T	J	J	J	J	C	S	F	N	R	P	Z	D	J	V	P
G	Ñ	W	E	D	O	Z	I	H	N	Y	R	J	J	R	T	W	G	W	L	X	K	J	E	A
L	W	A	P	A	O	V	O	W	Y	G	A	N	I	G	A	P	S	N	V	A	S	W	T	U
Q	O	F	T	U	E	A	E	L	Q	Ñ	D	H	A	F	N	X	M	Ñ	Ñ	R	X	P	D	T
R	Z	G	A	G	N	I	T	S	O	H	X	D	S	L	I	W	N	C	P	Z	A	B	G	Z
W	A	Q	U	C	C	B	M	K	S	H	W	P	K	D	H	V	X	Z	P	G	E	A	K	A
W	X	F	I	O	O	N	Q	H	I	P	S	H	V	D	F	S	Y	T	I	Q	X	P	T	J
K	J	Z	J	A	R	X	Ñ	P	Ñ	X	D	S	O	A	U	X	C	Q	S	R	Z	G	D	N
B	V	V	I	E	R	N	E	W	S	E	M	P	Y	L	E	Ñ	R	L	C	X	H	J	P	Q
S	S	W	A	E	J	R	G	N	P	W	A	P	M	Ñ	Ñ	L	E	Ñ	T	A	Ñ	F	D	I
Q	M	H	G	S	M	U	S	J	H	F	W	Y	Q	T	L	I	B	X	R	N	C	N	M	X
J	V	D	K	E	I	N	M	C	O	R	Q	W	H	J	S	D	Z	Z	Y	H	N	T	U	P
L	Q	J	D	R	E	T	I	V	W	O	T	X	E	T	R	E	P	I	H	I	M	W	J	O
E	H	I	U	W	D	A	I	J	W	D	C	A	Q	T	F	Y	I	V	Ñ	S	P	X	L	X
Q	A	R	I	W	R	D	T	O	L	A	Q	Ñ	Ñ	J	K	D	V	Y	J	Ñ	G	I	X	F
Z	Y	T	B	J	M	E	G	Y	W	G	R	C	X	R	N	X	G	Z	T	B	D	Q	N	U
Ñ	O	W	Q	J	U	J	W	A	Q	E	J	T	K	O	O	W	L	H	M	P	Ñ	E	M	Ñ
P	N	L	E	K	F	Ñ	K	Ñ	G	V	B	Z	E	F	S	F	J	T	W	F	I	P	A	V
N	I	A	O	V	H	H	G	Y	J	A	U	B	K	Ñ	M	I	U	M	M	V	A	J	U	B
Q	D	K	V	R	Y	W	E	B	L	N	C	Ñ	C	P	Z	L	X	L	W	L	L	R	U	B
L	I	C	T	V	L	O	W	B	K	A	U	G	V	L	I	F	D	T	U	I	O	G	G	Y
C	X	A	L	N	P	G	S	T	O	P	G	P	L	H	R	Y	Y	N	U	S	R	G	S	K
A	X	A	Ñ	S	Q	D	U	F	K	J	H	R	L	B	H	P	I	V	W	I	O	R	H	B
Z	E	R	Q	L	V	Y	U	C	B	Z	V	F	F	Z	U	L	L	X	K	U	K	K	E	Q

INSTRUCCIÓN II.- Llena el siguiente crucigrama de acuerdo a las pistas que se te proporcionan a continuación (con tinta azul). (8 Ac.)

- 1.- Es tu dirección en internet.
- 2.- Es un espacio virtual en un servidor conectado permanente a internet.
- 3.- Es la interfaz entre el usuario y el internet.
- 4.- HTML es simplemente texto, igualdad de mayúsculas y minúsculas, no importa tabuladores ni los saltos de línea, caracteres especiales son parte de cuatro_____
- 5.- Es un lenguaje que se utiliza para la creación de páginas en laWWW.
- 6.- Es el Localizador de Recursos Uniforme.
- 7.- Pueden contener de forma opcional u obligatoria, lo que se denominan atributos o Modificadores.
- 8.- Es un hipertexto, pero que no incluye sólo información textual sino también información multimedia.



INSTRUCCIÓN III. Describe brevemente la función de las siguientes marcas. (8Ac.)

<BODY> _____

<HTML> _____

</TITLE> _____

<HEAD> _____

<TITLE> _____

</HEAD> _____

</BODY> _____

</HTML> _____

INSTRUCCIÓN IV. Encuentra y encierra en un círculo los errores en el código de la siguiente página que se te presenta. (10 Ac.)

<HLMT>

<HEAD>

<TITLE> Salón de Belleza Mayra<TITLE>

</HE4D>

<BODY>

<CENTER> lista de accesorios de belleza

Sombras

Rímel

<IL>Rubor

Esmalte de uñas

</BODI>

</HTML

INSTRUCCIÓN V. Realiza en tu cuaderno el código HTML para generar el siguiente resultado.

CURRÍCULUM VITAE

← Negritas

← Negritas y centrado.

← Fuente arial, color azul y cursiva.

Nombre: *Apellido Paterno Materno y Nombre(s).*

Género: *Femenino o masculino*

Domicilio: *Av. Siempre viva # 1*

Correo electrónico: *usuario@correoweb.com*

Gustos personales: *Lectura, cine (drama, comedia)*

INSTRUCCIÓN VI. Contesta las siguientes cuestiones en tu cuaderno con el apoyo de tu antología. **(Son temas que ya vimos en clase)**

1. ¿Qué es un hosting?
2. ¿Qué es un navegador?
3. ¿Qué es un correo electrónico? menciona 4 servidores (empresas) que brinden este servicio y menciona un ejemplo de cada uno de ellos.
4. ¿Qué significa URL?
5. ¿Cuál es la función de la URL?
6. ¿Qué significa FTP y para qué sirve?
7. Menciona las cuatro normas del lenguaje HTML

INSTRUCCIÓN VII. Realiza en tu cuaderno el código HTML para generar el siguiente resultado.

NOMBRE DEL ALUMNO	PRIMER PARCIAL	SEGUNDO PARCIAL	FINAL
MIGUEL LAYUN	7	7	7
LUIS GABRIEL REY	8	9	9
RAUL ALONOSO JIMENEZ	10	10	10
MOISES MUÑOZ	8	8	8
RUBENS SAMBUEZA	9	9	9

Recuerda utilizar las siguientes etiquetas.

TABLE: Para declarar que es una tabla.

TR: Para indicar el inicio de una línea dentro de una tabla.

TD: Para crear una celda.

INSTRUCCIÓN VIII. Realiza en tu cuaderno el código HTML para generar el siguiente resultado.

CANCIONES	ARTITAS	CIENCIAS
<ul style="list-style-type: none"> • CANCION 1 • CANCION 2 • CANCION 3 • CANCION 4 • CANCION 5 	<ol style="list-style-type: none"> 1. ARTISTA 1 2. ARTISTA 2 3. ARTISTA 3 4. ARTISTA 4 5. ARTISTA 5 	<p>QUÍMICA ES LA CIENCIA QUE ESTUDIA...</p> <p>FÍSICA ES LA CIENCIA QUE ESTUDIA...</p> <p>MECATRÓNICA ES LA CIENCIA QUE ESTUDIA...</p>

INSTRUCCIÓN IX. Diseñar el código de una página web con las siguientes características:

1. Tabla que contenga una celda con una anchura de 900 y una altura de 200 pixeles (atributos: WIDTH y HEIGHT) que contenga como título el género musical de tu preferencia.
2. Tabla que contenga dos celdas con una anchura de 450 y una altura de 500 pixeles cada una (atributos: WIDTH y HEIGHT), en la celda 1 insertar un texto referente a tus gustos por la música mínimo 2 renglones y en la celda 2 inserta el código para colocar una imagen alineada al centro de la celda.
3. Tabla que contenga una celda con una anchura de 900 y una altura de 100 pixeles (atributos: WIDTH y HEIGHT) que contenga la palabra “Derechos reservados”.

MUSICA POP ENLACE 1 ENALCE 2 ENLACE3	
<p>La expresión música pop (del inglés <i>pop music</i>, contracción de <i>popular music</i>) hace referencia a una combinación de distintos géneros musicales^{1,2}altamente populares dentro de una sociedad. Este tipo de música es hecha para ser altamente comercializada. Muchos citan los comienzos de este género durante los años 50.</p>	
DERECHOS RESERVADOS	

INSTRUCCIÓN X. Realiza en tu cuaderno el código HTML para generar las siguientes banderas utilizando tablas.

a) CHILE



b) ALEMANIA



c) PANAMÁ



INSTRUCCIÓN X. Realiza en tu cuaderno el código HTML para generar las siguientes banderas utilizando Frames.

d) CHILE



e) ALEMANIA



f) PANAMÁ



Nota: Crea los códigos para TODOS los archivos a utilizar.

INSTRUCCIÓN XI. Describe los siguientes atributos de Frames:

a) Rows.- _____

b) Cols.- _____

c) Scrolling.- _____

d) Bordercolor.- _____

e) Marginwidth.- _____

INSTRUCCIÓN XII. DIBUJA la salida del siguiente código de Frames.

```
<FRAMESET COLS=20%,*>  
<FRAME SRC="pagina1.html">  
<FRAMESET ROWS=20%,*>  
<FRAME SRC=" pagina 2.html">  
<FRAME SRC=" pagina 3.html" name= principal>  
</FRAMESET>  
</FRAMESET>
```