

HTML es el lenguaje con el que se escriben las páginas web. Las páginas web pueden ser vistas por el usuario mediante un tipo de aplicación llamada navegador. Podemos decir por lo tanto que el HTML es el lenguaje usado por los navegadores para mostrar las páginas webs al usuario, siendo hoy en día la interface más extendida en la red.

Este lenguaje nos permite aglutinar textos, sonidos e imágenes y combinarlos a nuestro gusto. Además, y es aquí donde reside su ventaja con respecto a libros o revistas, el HTML nos permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto.

El HTML se creó en un principio con objetivos divulgativos. No se pensó que la web llegara a ser un área de ocio con carácter multimedia, de modo que, el HTML se creó sin dar respuesta a todos los posibles usos que se le iba a dar y a todos los colectivos de gente que lo utilizarían en un futuro. Sin embargo, pese a esta deficiente planificación, sí que se han ido incorporando modificaciones con el tiempo, estos son los estándares del HTML. Numerosos estándares se han presentado ya. El HTML 4.01 es el último estándar a septiembre de 2001.

Esta evolución tan anárquica del HTML ha supuesto toda una serie de inconvenientes y deficiencias que han debido ser superados con la introducción de otras tecnologías accesorias capaces de organizar, optimizar y automatizar el funcionamiento de las webs. Ejemplos que pueden sonar son las [CSS](#), [JavaScript](#) u otros. Veremos más adelante en qué consisten algunas de ellas.

Otros de los problemas que han acompañado al HTML es la diversidad de navegadores presentes en el mercado los cuales no son capaces de interpretar un mismo código de una manera unificada. Esto obliga al webmáster a, una vez creada su página, comprobar que esta puede ser leída satisfactoriamente por todos los navegadores, o al menos, los más utilizados.

Es recomendable usar el Bloc de notas que viene con windows, u otro editor de textos sencillo. Hay que tener cuidado con algunos editores más complejos como Wordpad o Microsoft Word, pues colocan su propio código especial al guardar las páginas y HTML es **únicamente texto plano**, con lo que podremos tener problemas.

Así pues, una página es un archivo donde está contenido el código HTML en forma de texto. Estos archivos tienen extensión .html o .htm (es indiferente cuál utilizar). De modo que cuando programemos en HTML lo haremos con un editor de textos y guardaremos nuestros trabajos con extensión .html, por ejemplo mipágina.html

Consejo: Utiliza siempre la misma extensión en tus archivos HTML. Eso evitará que te confundas al escribir los nombres de tus archivos unas veces con .htm y otras con .html. Si trabajas con un equipo en un proyecto todavía más importante que os pongáis todos de acuerdo en la extensión.

Introducción a HTML
DWEB Sec.01
Ing. Tomas Eduardo Urbina

Aquí les dejo el link para bajar la recomendación dada en clase el NOTAPAD++, viene en dos versiones 1) que reconoce los caracteres especiales (países orientales y asiáticos UNICODE 2) que reconoce caracteres comunes del idioma de EEUU ANSI.

<http://www.transexpress.com.sv/upload/npp.5.2.bin.zip>.

No requiere instalación, solo escojan la versión y ejecuten el exe.

Descripción de la sintaxis con la que se trabaja en el lenguaje HTML.

El HTML es un lenguaje que basa su sintaxis en un elemento de base al que llamamos etiqueta. La etiqueta presenta frecuentemente dos partes:

Una apertura de forma general <etiqueta>

Un cierre de tipo </ etiqueta>

Todo lo incluido en el interior de esa etiqueta sufrirá las modificaciones que caracterizan a esta etiqueta. Así por ejemplo:

Las etiquetas y definen un texto en negrita. Si en nuestro documento HTML escribimos una frase con el siguiente código:

```
<b>Esto esta en negrita</b>
```

El resultado Será:

Esto esta en negrita

Las etiquetas <p> y </p> definen un párrafo. Si en nuestro documento HTML escribiéramos:

```
<p>Hola, estamos en el párrafo 1</p>  
<p>Ahora hemos cambiado de párrafo</p>
```

El resultado sería:

Hola, estamos en el párrafo 1

Ahora hemos cambiado de párrafo

Partes de un documento HTML

Además de todo esto, **un documento HTML ha de estar delimitado por la etiqueta <html> y </html>**. Dentro de este documento, podemos asimismo distinguir dos partes principales:

El encabezado, delimitado por <head> y </head> donde colocaremos etiquetas de índole informativo como por ejemplo el título de nuestra página.

El cuerpo, flanqueado por las etiquetas <body> y </body>, que será donde colocaremos nuestro texto e imágenes delimitados a su vez por otras etiquetas como las que hemos visto.

El resultado es un documento con la siguiente estructura:

Primer Ejemplo

```
<html>

<head>
Etiquetas y contenidos del encabezado
Datos que no aparecen en nuestra página pero que son importantes para catalogarla:
Titulo, palabras clave,...
</head>

<body>
Etiquetas y contenidos del cuerpo
Parte del documento que será mostrada por el navegador: Texto e imágenes
</body>

</html>
```

Con el documento HTML creado, podemos ver el resultado obtenido a partir de un navegador. Es conveniente, llegado a este punto, hacer hincapié en el hecho de que no todos los navegadores son idénticos. Desgraciadamente, los resultados de nuestro código pueden cambiar de uno a otro por lo que resulta aconsejable visualizar la página en varios. Generalmente se usan Internet Explorer y Firefox como referencias ya que son los más extendidos.

Nota: Puedes abrir el archivo si accedes al directorio donde lo guardaste. En él podrás encontrar tu archivo HTML y verás que tiene como icono el logotipo de Firefox o el de Internet Explorer. Para abrirlo simplemente hacemos un doble click sobre él.

Lo ideal es guardarlo en el directorio del Servidor Web (c:\inetpub\wwwroot para IIS o c:\xampp\htdocs para Apache. Luego lo ejecutas desde el navegador digitando http://localhost/nombre_pagina.htm.

Previo la instalación y configuración del servidor IIS o Apache.

Una vez abierto el archivo podras ver tu primera página web. Algo sencillita pero por algo se empieza.

Las mayúsculas o minúsculas son indiferentes al escribir etiquetas

A notar que las etiquetas pueden ser escritas con cualquier tipo de combinación de mayúsculas y minúsculas. <html>, <HTML> o <HtMl> son la misma etiqueta. Resulta sin embargo aconsejable acostumbrarse a escribirlas en minúscula ya que otras tecnologías que pueden convivir con nuestro HTML (XML por ejemplo) no son tan permisivas y nunca viene mal coger buenas costumbres desde el principio para evitar fallos triviales en un futuro.

Segundo Ejemplo

```
<html>

<head>
<meta name="description" content="descripción de tu sitio" />
<meta name="keywords" content="palabras claves de tu sitio" />
<meta name="author" content="Nombre autor" />
<meta name="reply-to" content="email@email.com" />

<title>Ejemplo 2</title>

</head>

<body>
<p><b>Bienvenido,</b></p>
<p>Estás en la página <b>DWEB SEC 01</b>. </p>
<p>Aquí aprenderás cositas fáciles de las Plataformas Web. </p>
</body>

</html>
```

NOTA: Como muchos ya notaron, hemos agregado en la seccion <head> los metatags, la idea principal de crear los meta tags, que permite a google, yahoo, aol, etc. indexar tu página, entre lo que se sabe del buscador de google es que este tiene un ranking de tu página, que depende entre otras MUCHAS cosas de los meta tags, los links que se direccionen a tu página(es decir cuantos links externos hacía ti haya), que no hagas trampa es decir que no pongas palabras \"ocultas\" del mismo color del fondo en tu sitio, que la palabra que pusiste en la busqueda aparezca y tenga relevancia en tu sitio, y muchas más. Esto es lo principal ahora bien si te interesa comprar un espacio de publicidad en google y dependiendo de la palabra clave salga tu anuncio puedes revisar <http://www.google.com.mx/intl/en/ads/> y aquí descripción de xhtml de los metatags. van entre las etiquetas <head></head>

Ahora guarda ese archivo con extensión .html o .htm en tu disco duro y la ejecutas de la forma indicada anteriormente.

Consejo: Utiliza nombres en tus archivos que tengan algunas normas básicas para ahorrarte disgustos y líos.

El consejo es que no utilices acentos ni espacios ni otros caracteres raros. Esto no quiere decir que debes hacer nombres de archivos cortos, es mejor hacerlos descriptivos para que te aclaren lo que hay dentro. Algún caracter como el guión "-" o el guión bajo "_" te puede ayudar a separar las palabras. Por ejemplo quienes_somos.html

Si ahora en el navegador haces click con el botón derecho sobre la página y elegís Ver código fuente (o View page source) veras como en una ventana accesoria aparece el código de nuestra página. Este recurso es de extremada importancia ya que nos permite ver el tipo de técnicas empleadas por otros para la confección de sus páginas.

Con todo esto asimilado ya estamos en condiciones de adentrarnos un poco más en la descripción de algunas de las etiquetas más empleadas del HTML.

Posible problema: Al utilizar el Block de Notas en Windows en ocasiones, aunque le digamos que es un archivo .html, el documento se guarda como si fuera un texto y no una página web. Lo que está pasando es que el Block de Notas tiene predeterminado guardar sus archivos con extensión .txt y en realidad lo que está guardando en el disco duro es mi_pagina.html.txt

Para conseguir tener el control de las extensiones en el block de notas y en Windows en general podemos acceder a MI-PC y en el menú de Ver seleccionáis "Opciones de carpeta". En la ventana que sale pulsamos en la solapa "Ver" y nos permite deseleccionar una caja de selección que pone algo como "Ocultar extensiones para los tipos de archivos conocidos". (Así se hace en Win98, puede variar un poco en otras versiones de Windows.)

Con ello conseguiremos que se vea siempre la extensión del archivo con el que estamos trabajando y que el Block de Notas nos haga caso cuando le indicamos que grabe el archivo con otra extensión que no sea .txt

SOBRE EL <BODY>

Las páginas HTML pueden construirse con variedad de atributos que le pueden dar un aspecto a la página muy personalizado. Podemos definir atributos como el color de fondo, el color del texto o de los enlaces. Estos atributos se definen en la etiqueta <body> y, como decíamos son generales a toda la página. Lo mejor para explicar su funcionamiento es verlos uno por uno.

Atributos para fondos

bgcolor: especificamos un color de fondo para la página con su nombre o su valor RGB. El color de fondo que podemos asignar con bgcolor es un color plano, es decir el mismo para toda la superficie del navegador.

background: sirve para indicar la colocación de una imagen como fondo de la página. La imagen se coloca haciendo un mosaico, es decir, se repite muchas veces hasta ocupar todo el espacio del fondo de la página. En capítulos más adelante veremos como se insertan imágenes con HTML y los tipos de imágenes que se pueden utilizar.

Ejemplo de fondo

Vamos a colocar esta imagen como fondo en la página.



La imagen se podría llamar fondo.jpg y suponemos que se encuentra en el mismo directorio que la página. En este caso se colocaría la siguiente etiqueta <body>

```
<body background="fondo.jpg">
```

Consejo: siempre que coloquemos una imagen de fondo, debemos poner también un color de fondo cercano al color de la imagen.

Esto se debe a que, al colocar una imagen de fondo, el texto de la página debemos colocarlo en un color que contraste suficientemente con dicho fondo. Si el visitante no puede ver el fondo por cualquier cuestión (Por ejemplo tener deshabilitada la carga de imágenes) puede que el texto no contraste lo suficiente con el color de fondo por defecto de la web.

Creo que lo mejor será poner un ejemplo. Si la imagen de fondo es oscura, tendremos que poner un texto claro para que se pueda leer. Si el visitante que accede a la página no ve la imagen de fondo, le saldrá el fondo por defecto, que generalmente es blanco, de modo que al tener un texto con color claro sobre un fondo blanco, nos pasará que no podremos leer el texto convenientemente.

Ocurre parecido cuando se está cargando la página. Si todavía no ha llegado a nuestro sistema la imagen de fondo, se verá el fondo que hayamos seleccionado con bgcolor y es interesante que sea parecido al color de la imagen para que se pueda leer el texto mientras se carga la imagen de fondo.

Color del texto

text: este atributo sirve para asignar el color del texto de la página. Por defecto es el negro.

Además del color del texto, tenemos tres atributos para asignar el color de los enlaces de la página. Ya debemos saber que los enlaces deben diferenciarse del resto del texto de la página para que los usuarios puedan identificarlos fácilmente. Para ello suelen aparecer subrayados y con un color más vivo que el texto. Los tres atributos son los siguientes:

link: el color de los enlaces que no han sido visitados. (por defecto es azul clarito)

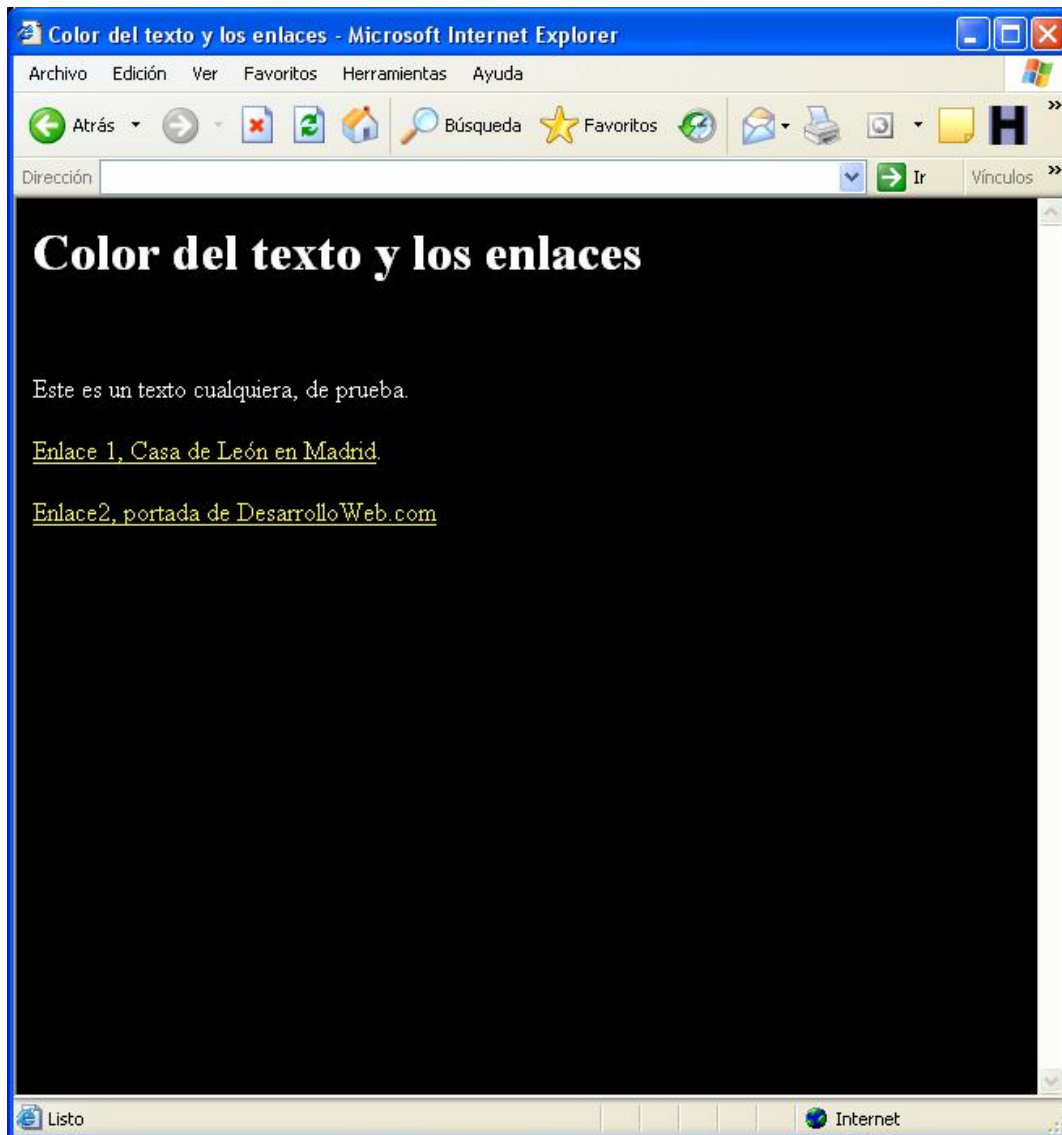
vlink: el color de los enlaces visitados. La "v" viene justamente de la palabra visitado. Es el color que tendrán los enlaces que ya hemos visitado. Por defecto su color es morado. Este color debería ser un poco menos vivo que el color de los enlaces normales.

alink: es el color de los enlaces activos. Un enlace está activo en el preciso instante que se pulsa. A veces es difícil darse cuenta cuando un enlace está activo porque en el momento en el que se activa es porque lo estamos pulsando y en ese caso el navegador abandonará la página rápidamente y no podremos ver el enlace activo más que por unos instantes mínimos.

Ejemplo de color del texto

Vamos a ver una página donde el color de fondo sea negro, y los colores del texto y los enlaces sean claros. Pondremos el color de texto blanco y los enlaces amarillos, más resaltados los que no estén visitados y menos resaltados los que ya están visitados. Para ello escribiríamos la etiqueta body así:

```
<body bgcolor="#000000" text="#ffffff" link="#ffff33" alink="#ffffcc" vlink="ffff00">
```



Tercer EJEMPLO


```
<html>
<head>
  <title>Color del texto y los enlaces</title>
</head>

<body bgcolor="#000000" text="#ffffff" link="#ffff33" vlink="#ffffcc" alink="ffff00">
<h1>Color del texto y los enlaces</h1>

<br>
<br>

Este es un texto cualquiera, de prueba.

<br>
<br>
<a href="http://www.yahoo.comk">Enlace 1, yahoo</a>.
<br>
<br>
<a href="http://www.google.com">Enlace 2, google</a>

</body>
</html>
```

Márgenes

Con otros atributos de la etiqueta <body> se pueden asignar espacios de margen en las páginas, lo que es muy útil para eliminar los márgenes en blanco que aparecen a los lados, arriba y debajo de la página. Estos atributos son distintos para Internet Explorer y para Netscape, por lo que debemos utilizarlos todos si queremos que todos los navegadores los interpreten perfectamente.

leftmargin: para indicar el margen a los lados de la página. Válido para iexplorer.

topmargin: para indicar el margen arriba y debajo de la página. Para iexplorer.

marginwidth: la contrapartida de leftmargin para Netscape. (Margen a los lados)

marginheight: igual que topmargin, pero para Netscape. (Margen arriba y abajo)

```
<body topmargin=0 leftmargin=0 marginheight=0 marginwidth=0 bgcolor="ffffff">
```

Cómo colocar párrafos y saltos de línea en páginas web. También vemos los encabezados como párrafos que sirven de título.

Anteriormente hemos presentado a título de ejemplo algunas etiquetas que permiten dar formato a nuestro texto. Ahora veremos con más detalle las más ampliamente utilizadas y ejemplificaremos algunas de ellas posteriormente.

Formatear un texto pasa por tareas tan evidentes como definir los párrafos, justificarlos, introducir viñetas, numeraciones o bien poner en negrita, itálica... Hemos visto que para definir los párrafos nos servimos de la etiqueta <p> que introduce un salto y deja una línea en blanco antes de continuar con el resto del documento.

Podemos también usar la etiqueta
, de la cual no existe su cierre correspondiente (</br>), para realizar un simple retorno de carro con lo que no dejamos una línea en blanco sino que solo cambiamos de línea.

Nota: Existen otras etiquetas que no tienen su correspondiente de cierre, como para las imágenes, las veremos más adelante. Esto ocurre porque un salto de línea o una imagen no empiezan y acaban más adelante sino que sólo tienen presencia en un lugar puntual.

Los párrafos delimitados por etiquetas <p> pueden ser fácilmente justificados a la izquierda, centro o derecha especificando dicha justificación en el interior de la etiqueta por medio de un atributo align. Un atributo no es más que un parámetro incluido en el interior de la etiqueta que ayuda a definir el funcionamiento de la etiqueta de una forma más personal.

Así, si deseásemos introducir un **texto alineado a la izquierda** escribiríamos:

```
<p align="left">Texto alineado a la izquierda</p>
```

El resultado sería:

Texto alineado a la izquierda

Para una **justificación al centro**:

```
<p align="center">Texto alineado al centro</p>
```

que daría:

Texto alineado al centro

Para **justificar a la derecha**:

```
<p align="right">Texto alineado a la derecha</p>
```

cuyo efecto seria:

Texto alineado a la derecha

Como ven, en cada caso el atributo align toma determinados valores que son escritos entre comillas. En algunas ocasiones necesitamos especificar algunos atributos para el correcto funcionamiento de la etiqueta. En otros casos, el propio navegador toma un valor definido por defecto. Para el caso de align, el valor por defecto es left.

Nota: Los atributos tienen sus valores indicados entre comillas ("), pero si no los indicamos entre comillas también funcionará en la mayoría de los casos. Sin embargo, es aconsejable que pongamos siempre las comillas para acostumbrarnos a utilizarlas, por dar homogeneidad a nuestros códigos y para evitar errores futuros en sistemas más quisquillosos.

El atributo align no es exclusivo de la etiqueta <p>. Otras etiquetas muy comunes, que veremos más adelante, entre las cuales se introducen texto o imágenes, suelen hacer uso de este atributo de una forma habitual.

Imaginemos un texto relativamente largo donde todos los párrafos están alineados a la izquierda (por ejemplo). Una forma de simplificar nuestro código y de evitar introducir continuamente el atributo align sobre cada una de nuestras etiquetas es utilizando la etiqueta <div>.

Esta etiqueta por sí sola no sirve para nada. Tiene que estar acompañada del atributo align y lo que nos permite es alinear cualquier elemento (párrafo o imagen) de la manera que nosotros deseemos.

Así, el código:

```
<p align="left">Párrafo1</p>  
<p align="left"> Párrafo3</p>  
<p align="left"> Párrafo2</p>
```

es equivalente a:

```
<div align="left">  
<p>Párrafo1</p>  
<p>Párrafo2</p>  
<p>Párrafo3</p>  
</div>
```

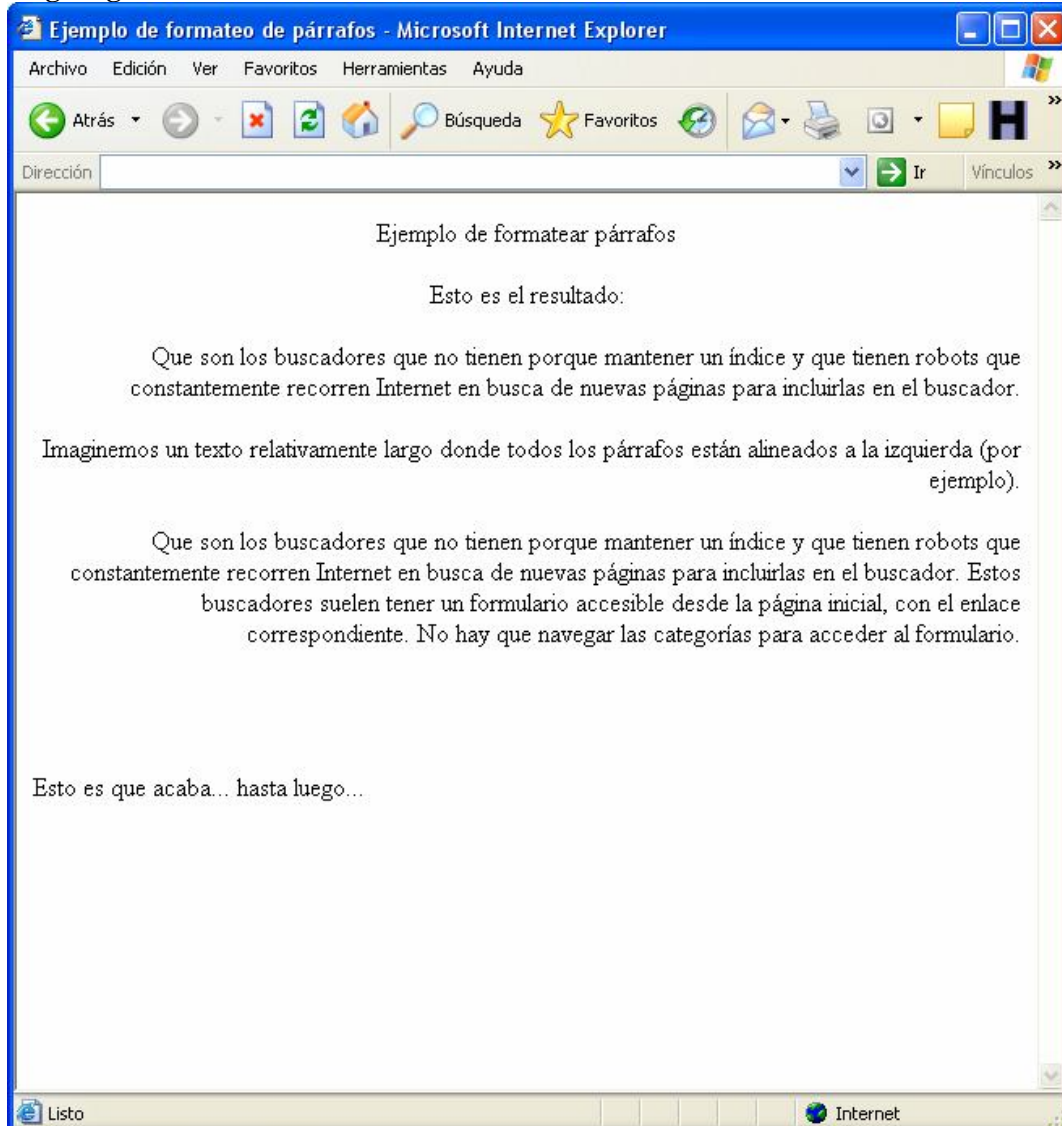
Como hemos visto, la etiqueta <div> marca divisiones en las que definimos un mismo tipo de alineado.

Ejemplo práctico 4:

Para practicar un poco lo que acabamos de ver vamos a proponer un ejercicio que podéis resolver en vuestros ordenadores. Simplemente queremos construir una página que tenga, por este orden:

- 2 Párrafos centrados
- 3 Párrafos alineados a la derecha
- Un salto de línea triple
- 1 párrafo alineado a la izquierda

Haga algo así:



Encabezados

Existen otras etiquetas para definir párrafos especiales, formateados como títulos. Son los encabezados o Header en inglés. Como decimos, son etiquetas que formatean el texto como un titular, para lo cual asignan un tamaño mayor de letra y colocan el texto en negrita.

Hay varios tipos de encabezados, que se diferencian en el tamaño de la letra que utilizan. La etiqueta en concreto es la <h1>, para los encabezados más grandes, <h2> para los de segundo nivel y así hasta <h6> que es el encabezado más pequeño.

Los encabezados implican también una separación en párrafos, así que todo lo que escribamos dentro de <h1> y </h1> (o cualquier otro encabezado) se colocará en un párrafo independiente.

Podemos ver cómo se presentan algunos encabezados a continuación.

```
<h1>Encabezado de nivel 1</h1>
```

Se verá de esta manera en la página:

Encabezado de nivel 1

Los encabezados, como otras etiquetas de HTML, soportan el atributo align. Vemos un ejemplo de encabezado de nivel 2 alineado al centro.

```
<h2 align="center">Encabezado de nivel 2</h2>
```

Se verá de esta manera en la página:

Encabezado de nivel 2

Otro ejercicio interesante es construir una página web que contenga todos los encabezados posibles. Se puede ver a continuación.

```
<html>  
  
<head>  
<title>Todos los encabezados</title>  
</head>  
  
<body>  
  
<h1>Encabezado de nivel 1</h1>  
<h2>Encabezado de nivel 2</h2>  
<h3>Encabezado de nivel 3</h3>  
<h4>Encabezado de nivel 4</h4>  
<h5>Encabezado de nivel 5</h5>
```

```
<h6>Encabezado de nivel 6</h6>
```

```
</body>
```

```
</html>
```

Consejo: No debemos utilizar las etiquetas de encabezado para formatear el texto, es decir, si queremos colocar un tipo de letra más grande y en negrita debemos utilizar las etiquetas que existen para ello (que veremos en seguida). Los encabezados son para colocar titulares en páginas web y es el navegador el responsable de formatear el texto de manera que parezca un titular. Cada navegador, pues, puede formatear el texto a su gusto con tal de que parezca un titular.

Vemos como colocar negritas, itálicas, subrayados, subíndices y supréndices.

Además de todo lo relativo a la organización de los párrafos, uno de los aspectos primordiales del formateo de un texto es el de la propia letra. Resulta muy común y práctico presentar texto resaltado en negrita, itálica y otros. Paralelamente el uso de índices, subíndices resulta vital para la publicación de textos científicos. Todo esto y mucho más es posible por medio del HTML a partir de multitud de etiquetas entre las cuales vamos a destacar algunas.

Negrita

Podemos escribir texto en negrita incluyéndolo dentro de las etiquetas `` y `` (bold). Esta misma tarea es desempeñada por `` y `` siendo ambas equivalentes. Nosotros nos inclinamos por la primeras por simple razón de esfuerzo.

Escribiendo un código de este tipo:

```
<b>Texto en negrita</b>
```

Obtenemos este resultado:

Texto en negrita

Nota: ¿Qué diferencia hay entre `` y ``?

Aunque las dos etiquetas hacen el mismo efecto, tienen una peculiaridad que las hace distintas. La etiqueta `` indica negrita, mientras que la etiqueta `` indica que se debe escribir resaltado. El HTML lo interpretan los navegadores según su criterio, es por eso que las páginas se pueden ver de distinta manera en unos browsers y en otros. La etiqueta `<H1>` quiere decir "encabezado de nivel 1", es el navegador el responsable de formatear el texto de manera que parezca un encabezado de primer nivel. En la práctica los encabezados de Internet Explorer y Netscape son muy parecidos (tamaño de

letra grande y en negrita), pero otro navegador podría colocar los encabezados con subrayado si le pareciese oportuno.

La diferencia entre `` y `` se podrá entender ahora. Mientras que `` significa simplemente negrita y todos los navegadores la interpretarán como negrita, `` es una etiqueta que significa que se tiene que resaltar fuertemente el texto y cada navegador es el responsable de resaltarlo como desee. En la práctica `` coloca el texto en negrita, pero podría ser que un navegador decidiese resaltar colocando negrilla, subrayado y color rojo en el texto.

Itálica

También en este caso existen dos posibilidades, una corta: `<i>` e `</i>` (italic) y otra un poco más larga: `` y ``. En este manual, y en la mayoría de las páginas que veréis por ahí, os encontraréis con la primera forma sin duda más sencilla a escribir y a acordarse.

He aquí un ejemplo de texto en itálica:

`<i>Texto en itálica</i>`

Que da el siguiente efecto:

Texto en itálica

Subrayado

El HTML nos propone también para el subrayado el par de etiquetas: `<u>` y `</u>` (underlined). Sin embargo, el uso de subrayados ha de ser aplicado con mucha precaución dado que los enlaces hipertexto van, a no ser que se indique lo contrario, subrayados con lo que podemos confundir al lector y apartarlo del verdadero interés de nuestro texto.

Subíndices y supraíndices

Este tipo de formato resulta de extremada utilidad para textos científicos. Las etiquetas empleadas son:

`^{` y `}` para los supraíndices

`_{` y `}` para los subíndices

Aquí tenéis un ejemplo:

La `¹³CC₃H₄CINOS es un heterociclo alergeno enriquecido`

El resultado debería ser algo así (agregue el código a alguno de los ejemplos anteriores):

La $^{13}\text{CC}_3\text{H}_4\text{CINOS}$ es un heterociclo alergeno enriquecido

Anidar etiquetas

Todas estas etiquetas y por supuesto el resto de las vistas y que veremos más adelante pueden ser anidadas unas dentro de otras de manera a conseguir resultados diferentes. Así, podemos sin ningún problema crear texto en negrita e itálica embebiendo una etiqueta dentro de la otra:

```
<b>Esto sólo está en negrita <i>y esto en negrita e itálica</i></b>
```

Esto nos daría:

Esto sólo está en negrita y *esto en negrita e itálica*

Consejo: Cuando anides etiquetas HTML hazlo correctamente. Nos referimos a que si abres etiquetas dentro de otra más principal, antes de cerrar la etiqueta principal cierras las etiquetas que hayas abierto dentro de ella.

Debemos evitar códigos como el siguiente:
Esto está en negrita e <i>itálica</i>

En favor de códigos con etiquetas correctamente anidadas:
Esto está en negrita e <i>itálica</i>

Esto es muy aconsejable, aunque los navegadores entiendan bien las etiquetas mal anidadas, por dos razones:

1. Sistemas como XML no son tan permisivos con estos errores y puede que en el futuro nuestras páginas no funcionen correctamente.
2. A los navegadores les cuesta mucho tiempo de procesamiento resolver este tipo de errores, incluso más que construir la propia página y debemos evitarles que sufran por una mala codificación.

Seguimos aprendiendo etiquetas que nos sirven para formatear el texto.

A pesar de que por razones de homogeneidad y sencillez de código este tipo de formatos son controlados actualmente por hojas de estilo en cascada (de las cuales ya tendremos tiempo de hablar más adelante en la asignatura), existe una forma clásica y directa de definir color tamaño y tipo de letra de un texto determinado.

Esto se hace a partir de la etiqueta y su cierre correspondiente. Dentro de esta etiqueta deberemos especificar los atributos correspondientes a cada uno de estos parámetros que deseamos definir. A continuación os comentamos los atributos principales de esta etiqueta:

Atributo face

Define el tipo de letra. Este atributo es interpretado por versiones de Netscape a partir de la 3 y de MSIE 3 o superiores. Otros navegadores las ignoran completamente y muestran el texto con la fuente que utilizan.

Hay que tener cuidado con este atributo ya que cada usuario, dependiendo de la plataforma que utilice, puede no disponer de los mismos tipos de letra que nosotros con lo que, si nosotros elegimos un tipo del que no dispone, el navegador se verá forzado a mostrar el texto con la fuente que utiliza por defecto (suele ser Times New Roman). Para evitar esto, dentro del atributo suelen seleccionarse varios tipos de letra separados por comas. En este caso el navegador comprobará que dispone del primer tipo enumerado y si no es así, pasará al segundo y así sucesivamente hasta encontrar un tipo que posea o bien acabar la lista y poner la fuente por defecto. Veamos un ejemplo.

```
<font face="Comic Sans MS,arial,verdana">Este texto tiene otra tipografía</font>
```

Que se visualizaría así en una página web.

Este texto tiene otra tipografía

Nota: Aquí tenemos un ejemplo de atributo cuyo valor debe estar limitado por comillas ("). Habíamos dicho que las comillas eran opcionales en los atributos, sin embargo esto no es así siempre. Si el valor del atributo contiene espacios, como es el caso de:

```
face="Comic Sans MS,arial,verdana"
```

debemos colocar las comillas para limitarlo. En caso de no tener comillas

```
face=Comic Sans MS,arial,verdana
```

se entendería que face=Comic, pero no se tendría en cuenta todo lo que sigue, porque HTML no lo asociaría al valor del atributo. En este caso HTML pensaría que las siguientes palabras (después del espacio) son otros atributos, pero como no los conoce como atributos simplemente los desestimaría.

Atributo size

Define el tamaño de la letra. Este tamaño puede ser absoluto o relativo.

Si hablamos en términos absolutos, existen 7 niveles de tamaño distintos numerados de 1 a 7 por orden creciente. Elegiremos por tanto un valor size="1" para la letra más pequeña o size="7" para la más grande.

```
<font size=4>Este texto es más grande</font>
```

Que se visualizaría así en una página web.

Este texto es más grande

Podemos asimismo modificar el tamaño de nuestra letra con respecto al del texto mostrado precedentemente definiendo el número de niveles que queremos subir o bajar en esta escala de tamaños por medio de un signo + o -. De este modo, si definimos nuestro atributo como `size="+1"` lo que queremos decir es que aumentamos de un nivel el tamaño de la letra. Si estabamos escribiendo previamente en 3, pasaremos automáticamente a 4.

Los tamaños reales que veremos en pantalla dependerán de la definición y del tamaño de fuente elegido por el usuario en el navegador. Este tamaño de fuente puede ser definido en el Explorer yendo al menu superior, Ver/Tamaño de la fuente. En Netscape elegiremos View/Text Size. Esta flexibilidad puede en más de una ocasión resultarnos embarazosa ya que en muchos casos deseamos que el tamaño del texto permanezca constante para que éste quepa en un determinado espacio. Veremos en su momento que esta prefijación del tamaño puede ser llevada a cabo por las hojas de estilo en cascada.

Atributo color

El color del texto puede ser definido mediante el atributo color. Cada color es a su vez definido por un número hexadecimal que esta compuesto a su vez de tres partes. Cada una de estas partes representa la contribución del rojo, verde y azul al color en cuestión.

Pueden entender cómo funciona esta numeración y cuáles son los colores que resultan más compatibles a partir del siguiente artículo:

<http://www.desarrolloweb.com/articulos/colores-html.html>.

Por otra parte, es posible definir de una manera inmediata algunos de los colores más frecuentemente usados para los que se ha creado un nombre más memotécnico:

Nombre Color

Aqua	
Black	
Blue	
Fuchsia	
Gray	
Green	
Lime	
Maroon	
Navy	
Olive	
Purple	
Red	

Silver	
Teal	
White	
Yellow	

`Este texto está en rojo`

Que se visualizaría así en una página web.

Este texto está en rojo

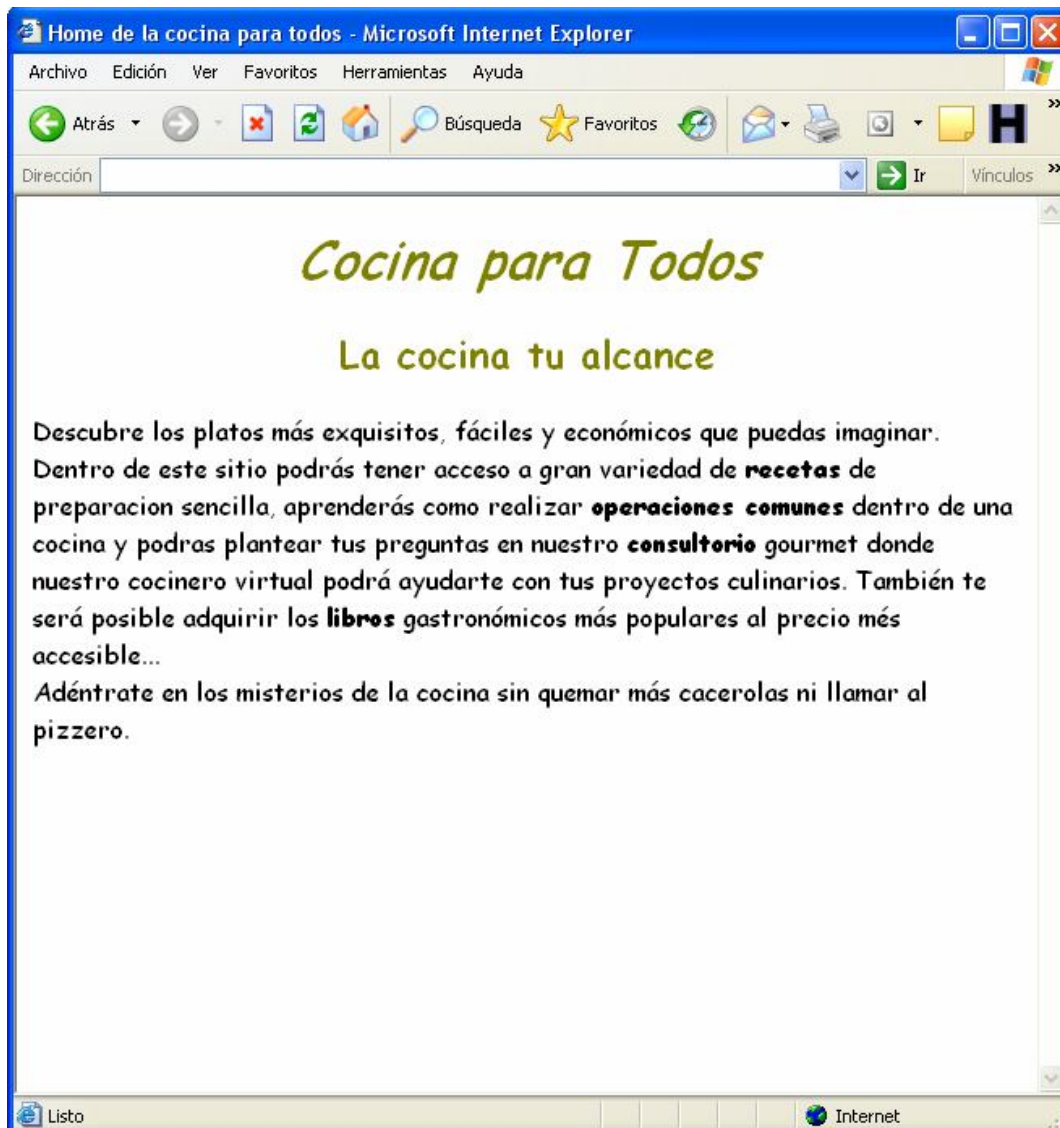
Con todo esto estamos ya en disposición de crear un texto formateado de una forma realmente elaborada.

Ejercicio Practico 4

Pongamos pues en práctica todo lo que hemos aprendido haciendo un ejercicio consistente en una página que tenga las siguientes características:

- Un titular con encabezado de nivel 1, en itálica y color verde oliva.
- Un segundo titular con encabezado de nivel 2, también de color verde oliva.
- Todo el texto de la página deberá presentarse con una fuente distinta de la fuente por defecto. Por ejemplo "Comic Sans MS" y en caso de que ésta no esté en el sistema que se coloque la fuente "Arial".

Debería ver algo así, una vez terminado el ejercicio.



Ahora ya esta listo para algo mas complejo (eso espero) por usted mismo.

Intenten hacer la siguiente pagina web con lo aprendido hasta el momento (por las imágenes no se preocupe aun, luego tendra tiempo de practicar esto).

No usen tablas (aun no), ni capas (<div>, aun no).

Es posible hacerla solo con lo que hemos visto actualmente.

Traiga este ejemplo desarrollado para la siguiente clase.

<http://www.transexpress.com.sv/upload/practica110909.pdf>

Las imágenes las pueden descargar del sitio:

<http://www.transexpress.com.sv/upload/pwb1/prac110909>